ISBA-TOP HYDRO-METEOROLOGICAL COUPLING : USERS AND DEVELOPERS GUIDE

This document describes how to activate the coupling between ISBA and TOPODYN hydrological model for flash-flood simulations. A first part dedicated to users presents how to activated the options and lists the system features. The second part is dedicated to developers : the routines that were added to SURFEX code as well as the modified ones are described.

SURFEX version : V 8.1
Documentation date : December 2017

Outline :

1 Aims of introducing a TOPODYN approach in SURFEX

A « TOPMODEL » (Beven et Kirby, 1979) approach has been introduced into ISBA in order to model the lateral distribution of soil water among watersheds. Soil water can thus move from a soil column to the neighbouring ones, what is crucial to better simulate fast responding rivers dynamics. A transfer module of water along the hillslopes and into the river permits then to compute a discharge at the river outlets. Those options can be activated in the SURFEX platform. The TOPMODEL version used is called TOPODYN (Pellarin et al., 2002), it is adapted to fast responding watersheds such as those of the Mediterranean area.

A full description of the coupled system can be obtained with the following papers :

Bouilloud et al, 2010 *Vincendon et al., 2010* *Vincendon, 2010* (thèse en français)	Original version of the coupled ISBA-TOP system (with ISBA-3L) and calibration of the parameters.
Vincendon et al., 2016	Version based on ISBA-DF with an option to discretise the main catchments in smaller sub-catchments.
Edouard, 2016 (thèse en français) *Edouard et al, 2017*	Version based on ISBA-DF with an option to perturb the hydrodynamical parameters and the initial soil moisture conditions.
Vincendon et al., 2011	On the perturbation of input precipitation.
Artinyan et al., 2016	On how to vary river water speed following the discharge.

Users Part

2 How to activate ISBA-TOP coupling ?

The coupling with TOPODYN is possible in the OFFLINE version of SURFEX. The ISBA-TOP simulation is configured using namelists in the *OPTIONS.nam* file. See SURFEX documentation to begin with.
The domain and catchments and sub-catchments definition is performed in the first step called PGD. The simulation starting dates and soil conditions (moisture and temperature) are defined at the PREP step. The forcing files in netcdf format, FORCING.nc, has to be created previously to the simulation with a new driver called CREATE_FORC. The simulation duration is defined in this step through a number of time steps. Finally, the proper simulation is performed at the OFFLIN step.

The ISBA namelists have been complemented or new namelists have been added to :

- allow the lateral distribution of water among the watershed following the TOPODYN principles, and/or
- transfer sub-surface runoff and deep drainage to the watersheds outlet to compute total discharges and/or
- create forcing files for SURFEX in netcdf (file FORCING.nc) from several files in several formats.

2.1 NAMELISTS propoer to ISBA-TOP

The following namelist allows to activate the lateral soil water distribution and to define the watersheds at the PGD step.

- **NAM_PGD_TOPD**

CCAT(*bv*)	Name of the river outlet numbered *bv* (*bv* from 1 to 1000) Maximal length = 15 characters
LCOUPL_TOPD	if T, the discharge at the outlets defined previously will be computed thanks to the routing module *Warning : the TOPODYN approach is used for runoff computation only if CRUNOFF = 'TOPD' in the namelist NAM_SGH_ISBAn; else, la parameterisation given in CRUNOFF is applied and runoff et drainage are routed*
LSUBCAT	if T, sub-catchments are either defined automatically or, if LDUMMY_SUBCAT=T ,read in the present namelist
LDUMMY_SUBCAT	if T, l sub-catchments are defined by the user
NSUBCAT(*bv*)	Only if LDUMMY_SUBCAT=T, number of sub-catchments pour the main catchment *bv*.
XLX(*bv*,*sbv*)	Only if LDUMMY_SUBCAT=T, X-coordinate (lambx for instance) of the outlet of the sub-catchment *sbv* of the main catchment *bv*. *sbv* is between 1 and NSUBCAT(*bv*)
XLY(*bv*,*sbv*)	Only if LDUMMY_SUBCAT=T, Y-coordinate (lamby for instance) of the outlet of the sub-catchment *sbv* of the main catchment *bv*. *sbv* is between 1 and NSUBCAT(*bv*)
CFILE_SUBCAT(*bv*)	File with information about the sub catchments of the main catchment *bv*
CSUBCAT(*bv*,*sbv*)	Name of the sub catchment *sbv* of the main catchment *bv*.
XF_PARAM_BV(*bv*)	Value of *f* parameter for the exponential profile of Ksat for the catchment *bv* (only for ISBA-3L).
XC_DEPTH_RATIO_BV(*bv*)__	Value of *alpha* parameter for the exponential profile of Ksat for the catchment *bv* (*dc*=*alpha***d2*) (only for ISBA-3L).
LWRITE_SEVERITY_MAPS	To produce some « severity maps » (description section 3.4)

Ex :

```
&NAM_PGD_TOPD
        CCAT(1) = "boucoiran" ,
        CCAT(2) = "bagnols" ,
        LCOUPL_TOPD=T,
        XF_PARAM_BV(1) = 2.0,
        XC_DEPTH_RATIO_BV(1)=1.50,
        XF_PARAM_BV(2) = 3.0,
        XC_DEPTH_RATIO_BV(2)=1.50
/
```

```
&NAM_PGD_TOPD
        LCOUPL_TOPD=T,
        LSUBCAT= T,
        LDUMMY_SUBCAT = T,
        CCAT(1) = "ners",
        NSUBCAT(1)=2,
        XLX(1,1)=731805,
        XLY(1,1)=1897330,
        XLX(1,2)=739697,
        XLY(1,2)=1905471
        CCAT(2) = "lodeve",
        NSUBCAT(2)=0
/
```

The following namelist is used to activate TOPODYN and the routing module, it must be present at the OFFLIN step.

- **NAM_TOPD**

LBUDGET_TOPD	if T, all the budget components will be computed and written in special output files
LSTOCK_TOPD	if T, output files from a previous simulation (called "stock" files) will be read to start the current one.
NNB_TOPD	ratio between ISBA time step and frequency of calling TOPODYN coupling (and also routing)
NFREQ_MAPS_WG	Frequency of writing soil water contents in files at specific format (.map)
NFREQ_MAPS_ASAT	Frequency of writing saturated areas in files at specific format (.map)
NFREQ_MAPS_RUNOFF	Frequency of writing runoff maps in files at specific format (.map)
NNB_STP_STOCK	Number of time steps to read in the "stocks" files
NNB_STP_RESTART	Number of time steps to write in the "stocks" files if the restart mode is chosen (LRESTART=T in NAM_IO_OFFLINE)
XSPEEDR(*bv*)	Speed of water in the river for the catchment *bv* (Default value = 1 m/s).
XSPEEDG(*bv*)	Speed of water in the ground for the catchment *bv* (Default value = 0,1 m/s)
XSPEEDH(*bv*)	Speed of water over the slopes for the catchment *bv* (Default value = 0,1* XSPEEDR(*bv*))
LSPEEDR_VAR	if T, the speed of water in the river is defined following the discharge value, see *Artinyan et al., 2016*.
XQINIT(*bv*)	Initial discharge for the catchment *bv*. (Default value = 0 m^3/s).
XRTOP_D2(*bv*)	Ratio between ISBA second layer (*d2*) and the soil depth where lateral distribution is activated (Default value = 1).
LPERT_PARAM	if T, the hydrodynamical parameters will be perturbed following the method of *Edouard et al. 2016*.
LPERT_INIT	if T, the initial conditions will be perturbed following the method of *Edouard et al. 2016*.

Ex :

```
&NAM_TOPD    LBUDGET_TOPD=T,
             LSTOCK_TOPD=F,
             NNB_TOPD=4,
             NFREQ_MAPS_WG=0,
             NFREQ_MAPS_ASAT=0,
             NNB_STP_RESTART=21
             XSPEEDR(1) = 3.0,
             XSPEEDG(1) = 0.3,
             XSPEEDR(2) = 1.0,
             XSPEEDG(2) = 0.1,
/
```

The following namelist is used to create forcing files at the netcdf format (this use is independent from the ISBA-TOP coupling, a new driver called CREATE_FORC has been developed).

- NAM_FORC_OFFLINE_NC

NNB_FORC_STP	Number of time steps in the forcing file, same as number of time steps of the simulation.
NNB_FORC_SEQUENCES	Number of sequences that use different data sources (max=5)
NSTP_BEG(s)	Time step at the beginning of sequence s.
NSTP_END(s)	Time step at the end of sequence s.
CTYPE_SEQUENCES(s)	Type of forcing (data sources) used in sequence s on 6 characters. Possible values are : 'IDEA ': idealised values chosen by the user in the same namelist (see variables *_IDEA) 'SAFRAN' : SAFRAN data (default value) , in a single file called *datafile* in grib format. 'NORAIN' : same as 'SAFRAN' but rainfall is set to 0 'RADAR ' : rainfall data are taken in radar QPE in a lat,lon,value format (unit is 1/10 of mm): as many files as time steps are needed. The other required parameters come from *datafile* file if present else default values are used. 'MODEL ' : data come from grib files containing meteorological models forecasts: as many files as time steps are needed. 'PERTRR' : same as the previous option but precipitations are perturbated and NNB_MEMBERS_ENS (see below) FORCING files are created
NNB_MEMBERS_ENS	Number of members of the ensemble in case of perturbation of rainfall in a given sequence..
CGRIB_BASE_NAME(s)	Base name for forcing files in grib format
CRAD_BASE_NAME(s)	Base name for forcing files in txt format (lat lon val), used for radar QPE
CGRIB_TYPE(s)	If grib files are used, source of forcing (model)
XTA_IDEA(s)	For idealized cases, value of 2m- temperature
XQA_IDEA(s)	For idealized cases, value of 2m-specific humidity
XDIRSW_IDEA(s)	For idealized cases, value of short wave direct radiation
XSCASW_IDEA(s)	For idealized cases, value of short wave scattered radiation
XLW_IDEA(s)	For idealized cases, value of long wave radiation
XPS_IDEA(s)	For idealized cases, value of surface pressure
XRAIN_IDEA(s)	For idealized cases, value of liquid precipitations
XSNOW_IDEA(s)	For idealized cases, value of solid precipitations
XWINDSPEED_IDEA(s)	For idealized cases, value of 10m-wind speed
XWINDDIR_IDEA(s)	For idealized cases, value 10m-wind direction

!

Ex : for a single sequence of 121 hours of radar QPE

&NAM_FORC_OFFLINE_NC NNB_FORC_STP = 121,
NNB_FORC_SEQUENCES=1,
NSTP_BEG(1)=1,
NSTP_END(1) = 121,

```
            CTYPE_SEQUENCES(1) ='RADAR ',
            CRAD_BASE_NAME(1) ='RADAR'
/
```

2.2 Options to choose in SURFEX to use ISBA-TOP

Moreover, some classical namelists of SURFEX should be filled as follows :

- To manage inputs/outputs and time steps :

```
&NAM_IO_OFFLINE     CSURF_FILETYPE  = 'ASCII '  ,
             CFORCING_FILETYPE =   'NETCDF'  ,
             CTIMESERIES_FILETYPE = 'TEXTE '  ,
             LPRINT  = T              ,
             LRESTART = T               ,
             XTSTEP_SURF=900.              ,
             XTSTEP_OUTPUT = 3600./
```

LRESTART should be T only if the user wishes to run another simulation starting from the current one.

```
&NAM_SGH_ISBAn   CRUNOFF='TOPD',
                 CKSAT='EXP'/
```

CRUNOFF='TOPD' activates the lateral transfers of soil water. It is possible to choose another option for runoff (e.g. DT92) and to activate only the routing module to route the runoff and drainage along the hillslopes and river as described in the topographical files.
To do so, the user chooses CRUNOFF='DT92' in the namelist NAM_SGH_ISBAn and LTOPD='T' in the namelist NAM_COUPL_TOPD.
CKSAT='EXP' permits to read *f* and *dc*, that are the two parameters describing the exponential decrease of Ksat. F and dc are specified in the namelist NAM_TOPD (1 value for each watershed). A file called *carte_f_dc.txt* will be created at the step PGD. It gives for each ISBA mesh a value of both parameters.

Initially, it was possible to activate the lateral distribution of water only with ISBA-3L and with a single patch for each grid mash. Since version V8 of SURFEX, it is possible to use several patches and the multi-layer version of ISBA (ISBA-DF) even if the lateral distribution still concerns the root zone of the soil.

To use ISBA-DF, the following choices have to be made in OPTIONS.nam :

```
  &NAM_DIAG_ISBAn     LPGD          = T ,
                      LSURF_EVAP_BUDGET = T ,
                      LSURF_MISC_BUDGET = T ,
                      LSURF_MISC_DIF   = T ,
                      LWATER_BUDGET    = T
  /
  &NAM_SGH_ISBAn      CRUNOFF   = "TOPD",
                      CKSAT     = 'DEF',
  /
```

3 Options/Use cases :

3.1 *f and dc* parameters with 3-L version of ISBA :

When ISBA-3L is used, the user can choose a profile for the decrease of the saturated hydraulic conductivity (*Ksat*) with depth defining a value for the decrease parameter *f* and the compactation depth *dc* (see Bouilloud et al., 2010 or Vincendon et al., 2010). To do so, in namelist **&NAM_SGH_ISBAn,** one must choose **CKSAT='EXP'** and precise, for each catchment *bv*, the values of **XF_PARAM_BV(*bv*)** and **XC_DEPTH_RATIO_BV(*bv*)** (the last one is in fact the ratio between the depth of the root layer of ISBA-3L annotated d_2 and *dc*) in the namelist **NAM_PGD_TOPD**. Default values are XF_PARAM_BV(*bv*)=2,5 and XC_DEPTH_RATIO_BV(*bv*)=1.

3.2 How to discretise the catchments in sub catchments :

In the original version of ISBA-TOP, the catchments had to be disjoint and the 5 topographic files (cf. annexe 3) had to be prepared previously to the simulation.
With version V8.1 of SURFEX, it is possible to start from the same main disjoint catchments and to ask for their discretization in sub catchments. To do so, the logical **LSUBCAT** in namelist has to be set true **(LSUBCAT='T').** The sub catchments acn be either (i) determined automatically or (ii) defined by the user.

In case (i) of an automatic determination, the computation time might be quite long. First, at the PGD step, the system determines the sub catchments such as :
- the area of sub catchments is greater than 500 pixels,
- the defined sub catchments are disjoint,
- the distance between the outlets of sub catchments is greater than 10 pixels,
- this distance is positive.

Then at the simulation step (OFFLINE), the computation is even longer when the number of catchments is high. This is the default use case (in NAM_PGD_TOPD, LDUMMY_SUBCAT='F' and XLX et XLY are not defined).

The use case (ii) in which the user prescribe the sub catchments is recommended. It allows to choose outlets that correspond to measurements stations for instance, what is useful for discharge simulation assessment. The coordinates of the outlet of each sub catchment according to the used MNT have to be known. The user must choose **LDUMMY_SUBCAT='T'** in namelist **NAM_PGD_TOPD** and fill in, for each subcatchment sbv of the main catchment bv, the variables **XLX(*bv,sbv*), XLY(*bv,sbv*) and CSUBCAT(*bv,sbv*)**. This information can also be indicated in the file called **CFILE_SUBCAT(*bv*).** Each line of this ascii file corresponds to a sub catchment and contains the following information :
CSUBCAT(*bv,sbv*), XLX(*bv,sbv*), XLY(*bv,sbv*), XQ2(*bv,sbv*), XQ10(*bv,sbv*), XQ50(*bv,sbv*).
The three last variables are respectively the return periods at 2 years, 10 years et 50 years. They are used to produce the severity maps (see section 3.4).

In both cases, an output file for each main catchment *BV* is produced at PGD step, it is called *BV_outletXY* and has a line for each sub catchment with its number sbv in the main catchment and the coordinates of the corresponding outlet as in the following example:
«
```
Sub catchment number           2 lambX=   935625.00000000000      lamby=   1832325.0000000000
 Sub catchment number           3 lambX=   935075.00000000000      lamby=   1831175.0000000000
```

Sub catchment number	4 lambX=	933975.00000000000	lamby=	1830875.0000000000
Sub catchment number	5 lambX=	933825.00000000000	lamby=	1830025.0000000000
Sub catchment number	6 lambX=	933075.00000000000	lamby=	1828475.0000000000
Sub catchment number	7 lambX=	932025.00000000000	lamby=	1827775.0000000000

»

It is possible to identify a point of the river (outlet of a sub catchment) as a reservoir (**LRESERVOIR(*bv,sbv*)='T'**) or the point of derivation of the river (**LDERIV(*bv,sbv)*='T').**This implies that the user have available the rules of management of the reservoir or derivation. Be carefully, at the time being, those rules are directly in the code of routine « routing_sub.F90 ». This might be corrected so that the rules are read in an external file.

3.3 Perturbation of the parameters and/or initial soil moisture conditions :

Edouard et al., 2016 developed a method to consider the uncertainty that affect initial soil moisture conditions in the model parameters. For initial soil moisture, a map of random smoothed coefficients is multiplied to the initial map. To do so, the user must choose **LPERT_INIT='T'** in namelist **NAM_TOPD.** Previously, it is necessary to indicate the seed that will be used for the random perturbation as well as the mean for the perturbation in a file called « randomMapGeneration.txt ». The first line is a integer that will be the seed and the second line is a real that is the averaged value of the random multiplicative coefficients.
To introduce a perturbation in the value of the hydrodynamical parameters, the user must choose **LPERT_PARAM='T'** in namelist **NAM_TOPD.**
A file called « randomNumbers.txt » gives the value of the random coefficients for each parameter. The coefficients are organised as follows :
- 1ère line : coefficient perturbing WSAT
- 2nde line : coefficient perturbing WWILT
- 3ème line : coefficient perturbing WFC (be careful, for ISBA-DF, WFC is W33)
- 4ème line : coefficient perturbing BCOEF
- 5ème line : coefficient perturbing MATPOTSAT
- 6ème line : coefficient perturbing HYDCONDSAT
- 7ème line : coefficient perturbing M (parameter of TOPODYN)

A sensitivity study has showed that the parameters that have the highest impact of flash-flood simulations with ISBA-TOP are Wsat, Ksat (HYDCONDSAT in code) and M. Thus it is recommended to perturb only those 3 parameters.
The script shell « generateRandomFiles.sh » (see appendix 5) generates « randomMapGeneration.txt » and « randomNumbers.txt » many times so as to produce many simulations that allow to better sample the uncertainty and produce an ensemble.

3.4 Severity maps

It is possible to produce some « severity maps ». In those maps, the river sections are coloured according to its discharge at a given time step:
- in blue, if the discharge remains below the 2-year return period (variable **XQ2** that is read in file **CFILE_SUBCAT** of namelist **NAM_PGD_TOPD**),
- in yellow, if the discharge exceeds the 2-year return period but remains below the 10-year return period (variable **XQ10** that is read in file **CFILE_SUBCAT**),
- in orange, if the discharge exceeds the 10-year return period but remains below the 50-year return period (variable **XQ50** that is read in file **CFILE_SUBCAT**),
- in red, if the discharge exceed the 50-year return period.

To do so, the user has to choose **LWRITE_SEVERITY_MAPS=.TRUE.** in namelist **NAM_PGD_TOPD**

3.5 Variation of the water speed in the river according to the discharge.

Artinyan et al., 2016 developed a parameterisation of the water speed in the river according to the discharge. Be careful, these equations are coded directly in the routine *routing.F90*. To use this option, the user has to set **LSPEEDR_VAR=.TRUE.** in namelist **NAM_TOPD.** The value of XSPEEDR(*bv*) is then the maximum possible value for the speed of water un the river.

3.6 How to create forcing files and how to perturb input rainfall :

A driver called CREATE_FORC allows to read files that contain the values of the meteorological parameters at several formats and to produce a FORCING.nc file used at OFFLINE step. It is integrated in SURFEX. The PGD.txt file must be elaborated previously and be present in the simulation directory. The code can be compiled as the others drivers of SURFEX (PGD, PREP et OFFLINE), see SURFEX website : http://www.umr-cnrm.fr/surfex/

CREATE_FORC allows to use several forcing “sequences”. These sequences can manage several types of input data, defined in namelist **NAM_FORC_OFFLINE_NC** on 6 characters in the variable **CTYPE_SEQUENCES(s).** If the name of the type is shorter, it has to be followed by a space that will be annotated as ␣ in this document.

- Idealized sequence: Constant values in a single sequence can be chosen with the option **CTYPE_SEQUENCES(s)=’IDEA␣␣’ .** The values have to be filled in the variables **X*_IDEA** of the same namelist (cf. section 2.1)

- SAFRAN sequence: data in a grib file called ‘datafile’ and containing forecasts or analyses of the SAFRAN model can be used **CTYPE_SEQUENCES(s)=’SAFRAN’.**

- « No more rain » sequence : It is possible to use SAFRAN data for all the meteorological variables except for rainfall that can be set to 0 filling **CTYPE_SEQUENCES(s)=’NORAIN’.** If ‘datafile’ is present in the simulation directory, it is read. If not default values are used.

- « lat,lon,value » sequence : Rainfall from ascii files containing on each line « lat,lon,value » can be read using **CTYPE_SEQUENCES(s)=’RADAR␣’**. The unit of rainfall in this case is 1/10 of mm. The other meteorological variables come from ‘datafile’ if present or default values if not. For rainfall values, on file is needed for each time step of simulation. The name of those files is defined by the variable **CRAD_BASE_NAME(s)**. For instance, if CRAD_BASE_NAME(1)=’CLP’, the first sequence will use files called ‘CLP_01.txt’ for the first time step, ‘CLP_02.txt’ for the second time step,…

- Sequence with meteorological models outputs in GRIB : The needed variables can be read in GRIB files coming from meteorological models forecasts with **CTYPE_SEQUENCES(s)=’MODEL␣ ’** . Once more, one file is needed for each time step and its name is defined from the variable **CGRIB_BASE_NAME(s)**. If CGRIB_BASE_NAME(1)=’GRIBARO’, the file used for the first sequence and time step is called ‘GRIBARO_01.grb’, for the second time step GRIBARO_02.grb’,…

The codes for each variables in the GRIB files differ according to the meteorological model. The type of model can thus be specified in the variable CGRIB_TYPE(s) on 6 characters also. The possible types are : ‘SAFRAN’, ’AROME␣‘, ‘MESONH’, ‘AROMAN’, ‘PEAROM’, ‘PEAROP’,’ARPEGE’ (see appendix 5.)

- Mixed sequence model/radar : It is possible to use meteorological forecast for all the variables except for the rainfall that can be read in a « lat,lon,value » file choosing **CTYPE_SEQUENCES(s)='MODRAD'.** For each time step, both a GRIB file and a ascii file (lat,lon,value) is needed.

- Sequence in which rainfall are perturbed : **CTYPE_SEQUENCES(s)='PERTRR'** lit allows to read grib files as 'MODEL␣ ' but a perturbation of the location, amplitude and structure of the rainfall field is introduced following the method of *Vincendon et al., 2011.* In this case, the number of forcing file required is indicated in variable **NNB_MEMBERS_ENS**.

The probability density functions used for this perturbation of rain objects are coded directly in routines « change_loc_objects.F90 » (for location) and « change_ampli_rain.F90 » (for amplitude).

3.5 Input/output files

All the input files needed for a « classical » SURFEX simulation are of course needed but additional ones are required to activate ISBA-TOP coupling. It is recommended to perform the different steps (PGD, PREP, CREATE_FORC et OFFLIN) of a simulation in the same directory.

Step	Input files	Output files
PGD	- *OPTIONS.nam* - files that describe the orography on SURFEX grid (Ex : *gtopo30.hdr et gtopo30.dir*), - files that describe soil texture (Ex : *CLAY_HWSD_MOY.dir, CLAY_HWSD_MOY.hdr, SAND_HWSD_MOY.dir, SAND_HWSD_MOY.hdr*), - files that describe land use (Ex : Ecoclimap). - topographical files that describe the catchments at the TOPODYN resolution (typically 50m) from MNT. There are 5 files to describe a catchment(see appendix 3). - optional file to describe sub catchments	- PGD.txt - two files of mask for each catchment : *BV.mask_topd* et *BV.mask surf* - *carte_f_dc.txt* if CKSAT='EXP' - if the main catchments are divided into sub catchments : *BVOutlet.map, BVSub-Cat.map, BVSubPix.map* etand if the catchment definition is automatic *BV_outletsXY* et *BV_overlap*
PREP	- *PGD.txt* - *OPTIONS.nam* - files describing the initial soil conditions for moisture and temperature for the 3 layers (surface, root and deep) in ascii format (hug1,dat, hug2,dat, hug3,dat, tg1,dat, tg2,dat).	*PREP.txt*
CREATE_FORC	- *PGD.txt* - *OPTIONS.nam* - files with original values of rainfall and other meteorological variables at each time stepfor the whole SURFEX domain (cf. section 3.5).	*FORCING.nc* or ensemble of files *FORCING_mb.nc*
OFFLINE	- *PGD.txt* - *PREP.txt* - *OPTIONS.nam* - *FORCING.nc* - *BV.mask_topd et BV.mask surf* - *carte_f_dc.txt* if CKSAT='EXP', - optional file to describe sub catchments - files to perturb hydrodynamical parameters and initial soil moisture conditions *randomMapGeneration.txt* and *randomNumbers.txt*	3 files by catchment with the discharges at the outlet of the main catchments and optionnaly sub catchments : + *BV_q_runoff.txt* is the discharge due to runoff + *BV_q_drainage.txt* is the discharge due to deep drainage + *BV_q_total.txt is* the total discharge. - if LBUDGET_TOPD =T : + 2 files by catchment : *bilan_bv_BV.txt* and *bilan_nobv_BV.txt* are the water budget among the watershed BV or out of it. + *bilan_q.txt* : water budget of all the watersheds + *bilan_total.txt* : water budget of all the domain - maps of R0, Asat, Wg at regular frequencies - severity maps if LWRITE_SEVERITY_MAPS=T

NB : Special case of RESTART :
If LRESTART=T, more out files are produced. They all have an index _sav.

- stock_sav.txt contains the values of drainage and runoff to route among the catchments. It contains as many steps as specified in the variable NNB_STP_STOCK.
- surfcont_sav.map contains the contributive area for each mesh of the whole domain at the end of the simulation.
- 1 file for each catchment CAT_xwtop_sav.map contains for each pixel of the catchment CAT the water content at the end of the simulation.

To start a new simulation (LSTOCK_TOPD=T), all those files must be renamed changing _sav in _init. One has to be careful for stock_init.txt file : the number of time steps read in the file is the one specified in NNB_STP_RESTART variable.

Developers part

4 Functions of routines.

Note : in the input/output data of this table, the « TYPES » are not indicated. (cf. technical SURFEX documentation http://www.umr-cnrm.fr/surfex//spip.php?rubrique150).

4.1 Routines developed specifically for ISBA-TOP coupling

Routine name	*Function*	*Input data*	*Output data*
AVG_PATCH_WG	Averages the values of Wg(soil liquid water content), Wgi (soil solid water content) and Dg (soil depth) of all the patches on a mesh.		PWG, PWGI, PDG
BUDGET_COUPL_ROUT	Computes the components of water budget.	KNI: number of ISBA meshes KFORC_STEP : current time step	(global variables declared in MODD_BUDGET_COUPL_ROUT)
CONTROL_WATER_BUDGET	Controls et corrects the water budget before and after lateral soil water distribution by Topodyn.	PWGM : water content of each ISBA mesh before lateral soil water distribution PWG :water content of each ISBA mesh before lateral soil water distribution PDG :soil depth PMESH_SIZE : size of each ISBA mesh PAVG_MESH_SIZE : averaged size of ISBA meshes PWSAT : Saturated water content for each mesh PWOVSAT_IBV : Water volume above saturation by mesh and catchment	PWG : water content of each ISBA mesh PWOVSAT_IBV : Water volume above saturation by mesh and catchment
COUPLING_SURF_TOPD	Activates either the coupling between ISBA and TOPODYN or the routing of ISBA runoff and drainage on the specified watersheds	HPROGRAM : calling program KI : number of ISBA meshes	
COUPL_TOPD	Initialisations. Computation of TOPODYN variables from ISBA variables Soil water lateral distribution.	HPROGRAM : calling program HSTEP Atmopsheric loop index. KI :number of ISBA meshes	

Routine name	*Function*	*Input data*	*Output data*
	Update of ISBA variables from new TOPODYN variables.	KSTEP : current time step	
COUPL_TOPD_SUB	Same COUPL_TOPD with sub catchments treatments		
DESC_CATCHMENTS	Defines sub catchments as required by the user (LDUMMY_SUBCAT='T').	JCAT : number of the main catchment	
DG_DFTO3L	Calculates the soil depth of 3 layers (surface, root, deep) like in ISBA-3L from the depth of the several layers of ISBA-DF		PDG : soil depth of 3 layers
DIAG_ISBA_TO_ROUT	Computes hourly values of runoff and drainage and change their unit from $kg.m^{-2}.s^{-1}$ to m^3.	PVARC : Cumulated variable at current time step PVARCP: Cumulated variable at former time step	PVARROUT : Hourly variable
DISPATCH_WG	Disagregates the soil liquid and solid water contents of a given mesh of several patches.	PWG : Liquid water content on the whole mesh (all patches) PWGI : Solid water content on the whole mesh PDG : soil depth for the given mesh	(Modification of global variables XWG)
FIND_SUB_CATCH	Defines sub catchments of a main catchment.	JCAT : number of the main catchment	
FLOWDOWN	Affects to all pixels i connected to a single pixel p (neighbour upstream) the same value as pixel p : Vi (i=1..nbConnectedPixel)=Vp The connected pixels are defined thanks to a mask.	KNMC : catchment grid points number PVAR: variable to propagate PCONN : connections of a pixel KLINE : number of each pixel, sorted by their altitude in the catchment	
FLOWDOWN_SUB	Same as FLOWDOWN with sub catchments treatments		
GET_UPSLOPE	Get upslope pixels and computes the area of the corresponding sub catchment	PCONN : connections of a pixel KFILE : index used to sort the connected pixels KNMC : catchment grid points number KOUTLET : index of a considered river pixel PDX : pixel size in the used DTM KNXC, KNYC : coordinates of the outlet of the main catchment. PTOPD : altitude of the pixels of the main catchment	KLIST_PIX : sub catchment mask KN_SUBPIX : number of pixels in the sub catchment

Routine name	*Function*	*Input data*	*Output data*
INIT_BUDGET_COUPL_ROUT	Allocates and initialises the water budget variables when the coupling or routing is activated.	KNI : number of ISBA meshes	
INIT_COUPL_TOPD	Initialises variables useful for coupling	HPROGRAM : calling program	(global variables of MODD_COUPLING_TOPD)
INIT_SURF_TOPD	Driver of INIT_TOPD_OL, INIT_COUPL_TOPD and INIT_BUDGET_COUPL_ROUT	HPROGRAM : calling program KI : number of ISBA meshes	
INIT_TOPD	Initialises variables specific to Topodyn	HPROGRAM : calling program	
INIT_TOPD_OL	Same as INIT_TOPD but for variables useful only for OFFLINE stop.	HPROGRAM : calling program	
INIT_TOPD_PGD	Same as INIT_TOPD but for variables useful only for PGD stop.	HPROGRAM : calling program	
ISBA_TO_TOPD	Computes a Topodyn variable from an ISBA variable using NMASKT	PVARI(:) : Variable on ISBA meshes	PVART(:,:) : Variables on a given pixel of a given catchment
ISBA_TO_TOPDSAT	Computes runoff to rout on the Topodyn pixels from ISBA runoff accounting from the saturation level of each pixel.	PKAPPA: hydrodynamical index of each pixel PKAPPAC : saturated hydrodynamical index KI : number of ISBA meshes PRO_I : runoff on ISBA meshes	PRO_T: runoff on the pixels of the catchments
MAKE_MASK_ISBA_TO_TOPD	Computes NMASKI such as: NMASKI(num ISBA mesh, num CATchment, num of pixel in ISBA mesh and CATchment)= num pixel in CATchment	KI : number of ISBA meshes	
MAKE_MASK_TOPD_TO_ISBA	Computes NMASKT such as : NMASKT(num CATchment, num of pixel sorted by altitude in the CATchment)=num corresponding ISBA mesh	KI : number of ISBA meshes	
MODD_BUDGET_COUPL_ROUT	Declares water budget variables when coupling or routing is activated.		
MODD_COUPLING_TOPD	Declares variables to couple ISBA and Topodyn.		
MODD_DUMMY_EXP_PROFILE	Declares f et dc if the user has chosen their values.		
MODD_TOPD_PAR	Declares some parameters specific to Topodyn		
MODD_TOPODYN	Declares variables specific to Topodyn		
MODE_RANDOM_PERT	CREATE_RANDOM_MAP : generates a map of random coefficients to perturb initila soil moisture conditions from a seed read in an ascii file.	ZRANDOM_MAP : smoothed map of random coefficients	ZRANDOM_MAP
	READ_RANDOM_NUMBER : read in a file random coefficients	ZRANDOM : random number	ZRANDOM
MODE_SOIL_PERT	Same as MODE_SOIL of SURFEX but with perturbation of hydrodynamic parameters.		
PGD_TOPD	Computes masks to match ISBA and TOPODYN grids	HPROGRAM : calling program	

Routine name	*Function*	*Input data*	*Output data*
PREP_RESTART_COUPL_TOPD	Writes the files needed for a new simulation starting from the final state of the current simulation.	HPROGRAM : calling program KI : number of ISBA meshes	
READ_CONNEX_FILES	Reads "connection" files	HPROGRAM : calling program HFILE : name of file to read HFORM : format of file to read KNMC : catchment grid points number	PCONN : connections of a pixel KLINE : num of pixel sorted by altitude in the catchment
READ_FILE_ISBAMAP	Reads file at specific format .map with the ISBA meshes	KUNIT: file unit number KI : number of ISBA meshes	PVAR: variable to read
READ_FILE_MASKTOPD	Reads mask files produced at PGD step	KI : number of ISBA meshes	
READ_NAMELISTS_TOPD	Driver of routines to read namelists specific to the ISBA-TOP coupling	HPROGRAM : calling program	
READ_NAM_PGD_TOPD	Reads namelist NAM_PGD_TOPD	HPROGRAM : calling program	OCOUPL_TOPD : T for ISBA-TOP coupling HCAT : Catchments name PF_PARAM_BV: f parameter for each catchment PC_DEPTH_RATIO_BV: dc parameter for each catchment LWRITE_SEVERITY_MAPS : T for writting severity maps
READ_NAM_TOPD	Reads namelist NAM_TOPD_n	HPROGRAM : calling program	OBUDGET_TOPD : T for computing the water budget KNB_TOPD : number of ISBA time step within a Topodyn time step OSTOCK_TOPD : T for reading output files of a former simulation KNB_STOCK :

Routine name	*Function*	*Input data*	*Output data*
			number of time step to be considered from the former simulation KNB_RESTART : number of time step to write for a following simulation (restart) PSPEEDR: speed of water in the river PSPEEDG: speed of water in the ground PQINIT : initial discharge PRTOP_D2 : ratio of D2 where lateral distribution is allowed
READ_SLOPE_FILE	Reads "slope" files	HPROGRAM : calling program HFILE : name of the file to read HFORM : format of the file to read KNMC : catchment grid points number	PTANB : slope tan B PSLOP : slope/length of pixel PDAREA : drained area for each pixel PLAMBDA : pure topographic index
READ_SUBCAT_FILE	Reads files describing the sub catchments	HPROGRAM : calling program HFILE : name of the file to read HFORM : format of the file to read KNMC : catchment grid points number KCAT : number of the catchment	
READ_TOPD_FILE	Reads topographic files (.map)	HPROGRAM : calling program HFILE : name of the file to read HFORM : format of the file to read KNPT : number of the pixel in the catchment	PTOPO_READ : variable to read in the file
READ_TOPD_HEADER_CONNEX	Reads header of "connection" files	HPROGRAM : calling program HFILE : name of the file to read HFORM : format of the file to read	KNMC : catchment grid points number

Routine name	*Function*	*Input data*	*Output data*
READ_TOPD_HEADER_DTM	Reads header of « FilledDTM » files	HPROGRAM : calling program HFILE : name of the file to read HFORM : format of the file to read	PX0 : abscissa of bottom-left pixel PY0 : ordinate of bottom-left pixel KNXC : number of topographic grid points along abscissa axis KNYC : number of topographic grid points along ordinate axis PNUL : undefined value in topographic files PDXT : catchment rid mesh size
RECHARGE_SURF_TOPD	Computes the water recharge and maximum deficit for Topodyn_lat routine.	KI : number of ISBA meshes PHI : variation of the water level on ISBA meshes since the last coupling time step (m)	PHT : water level added to a pixel before lateral distribution (m)
RESTART_COUPL_TOPD_N	Reads all files needed in case of « restart »	HPROGRAM : calling program KI : number of ISBA meshes	
ROUT_DATA_ISBA	Transfer runoff and drainage computed by ISBA.	HSTEP atmopsheric loop index. KI :number of ISBA meshes PSURF_STEP : time step ISBA (s) KSTEP : current time step	
ROUTING	Transfer runoff and drainage up to the outlet of the catchments.	PRO: Runoff to rout PDR: Drainage to rout KSTEP: current time step	
ROUTING_SUB	Same as ROUTING with sub catchments treatment.		
SAT_AREA_FRAC	Computes the contributive area on a mesh from the number of saturated pixels (i.e. with null definit).	PDEF deficits to saturation on pixels GTOPD : T if lateral distribution is activated	GTOPD PAS saturated areas of an ISBA mesh PAS_MESHBV contributive area fraction in Isba meshes

Routine name	***Function***	***Input data***	***Output data***
			by catchment
TOPD_TO_DF	Updates the soil water content of ISBA DIF after lateral soil water distribution by TOPODYN.	PWG : soil water content after lateral soil water distribution	Update of the global variable XWG
TOPD_TO_ISBA	Computes the averaged water content of an ISBA mesh from the water content of pixels in the mesh and ISBA water content for the area without pixels. Also computes un XWOVSAT to close the water budget.	KI :number of ISBA meshes KSTEP : current time step of ISBA-TOP coupling GTOPD : T if lateral distribution is activated	
TOPD_TO_ISBA_SLOPE	Computes slope from the MNT	KI :number of ISBA meshes	
TOPODYN_LAT	Uses the deficit to saturation for each pixel and a maximum value for deficits to laterally transfer soil water according to topography.	PRW : water recharge	PDEF : deficit to saturation of a pixel PKAPPA : hydrodynamical index PKAPPAC: saturated hydrodynamical index GTOPD : T if lateral distribution is activated
TOPODYN_LAT_SUB	Same as TOPODYN_LAT with sub catchments treatment		
WRITE_ALERT_MAPS	Writes severity maps	KCAT :catchment number KSTEP : time step of coupling HSTEP : index de time step du couplage en caractère	
WRITE_BUDGET_COUPL_ROUT	Write results of budget computation at each time step.		
WRITE_DISCHARGE_FILE	Write discharge at each time step.	HPROGRAM : calling program HFILE : name of the file to write HFORM : format of the file to write KYEAR: current year KMONTH : current month KDAY: current day KH: current hour KM: curent minutes PQTOT : total discharge	KM
WRITE_DISCHARGE_FILE_SUB	Same as WRITE_DISCHARGE_FILE with sub catchments treatment		
WRITE_FILE_1MAP	Writes a variable in a .map file	PVAR : variable to write HVAR : name of the variable KCAT : number of the catchment	
WRITE_FILE_ISBAMAP	Writes an ISBA variable in a .map file	KUNIT: unit of output file	

Routine name	***Function***	***Input data***	***Output data***
		PVAR : variable to write KI : number of ISBA meshes	
WRITE_FILE_MAP	Writes a Topodyn variable in a .map file	PVAR : variable to write HVAR : name of the variable	
WRITE_FILE_MASK_TOPD	Writes mask files at PGD step.	KI : number of ISBA meshes	
WRITE_FILE_VECMAP	Writes a variable in a .vec file	PVAR : variable to write HVAR : name of the variable KCAT : number of the catchment	

4.2 Modified SURFEX functions :

Routine name	*Functions*	*Modifications*
COMPUTE_ISBA_PARAMETERS	Computes ISBA parameters	CALL INIT_SURF_TOPD
COUPLING_ISBA_n	Couples ISBA and atmosphere	CALL COUPLING_SURF_TOPD
EXP_DECAY_SOIL_FR	Manages the exponential profile of Ksat	One argument more : PC_DEPTH_RATIO (ratio between D2 and DC).
HYDRO_SGH	Determines Horton runoff, the saturated area , Dunne runoff , infiltration rate , interception rate and infiltration in flooded plains.	Treatment of runoff when ISBA-TOP coupling is activated (LCOUPL_TOPD=T and HRUNOFF='TOPD')
INIT_VEG_PGD_n	Initialises soil properties according to the vegetation	Perturbation of ISBA initial soil moisture if LPERT_SOIL=T
MODE_READ_EXTERN	Reads ISBA parameters	Perturbation of ISBA hydrodynamical parameters if LPERT_PARAM=T
OFFLINE	Driver	Call routines for ISBA-TOP coupling, budget computation and writing results.
PGD_ISBA	Produces the file describing the study domain,	CALL PGD_TOPD (masks isba/topd and carte_f_dc.txt)
READ_ASCLLV	Reads ascii files « Lat Lon Value »	Bug on MAC : modification of parameter ILONG
READ_ALL_NAMELISTS		CALL READ_NAMELISTS_TOPD
READ_ISBA_CONF_n	Reads ISBA namelists	One possibility added 'TOPD' for CRUNOFF and 'EXP' for 'CKSAT'

4.3 Forcing files creation :

Routine name	*Functions*	*Input data*	*Output data*
CHANGE_AMPLI_RAIN	Modifies amplitude of rainy objects in the input rainfall field	PRR : champ de pluie initial KI : abscisse du point considéré KJ : ordonnée du point considéré	PRR_NEW : champ de pluie dont la localisation a été modifiée.
CHANGE_LOC_OBJECTS	Modifies location of rainy objects in the input rainfall field	PRR : champ de pluie initial KI : abscisse du point considéré KJ : ordonnée du point considéré KSTEP : time step PRR_NEW : champ de pluie dont la localisation a été modifiée.	PRR_NEW : champ de pluie dont la localisation a été modifiée.
CREATE_FORC	Driver to create FORCING.nc		
CUMUL_TO_HOURLY_RR	Computes rainfall at hourly time step from cumulated rainfall et change unit (from mm to kg/m2/s).	PRRC : pluies cumulées PRR1 : pluie horaire KNB_STEPS : nb. Total de time step	PRR1 : pluie horaire
FIND_POINTS	Isolates areas where rainfall exceeds a given threshold.	PRR1 : pluie horaire PTHRESHOLD : seuil KI : nombre maximum de points en X KJ : nombre maximum de points en Y	OINAREA : variable logique Vraie si le point est tel que la pluie excède le seuil considéré.
GAUSS_1REAL	Computes the probability to obtain a given value in a pdf following a Gaussian law with a given mean and standard deviation.	PAVG : moyenne PSTDDEV : écart-type PRAIN : valeur testée	PDENSITY_FCT : probabilité d'obtenir la velaue d'entrée dans le fonction de gauss définie par les moyenne et écart-type d'entrée.
GET_GRIB_CODE	Initialises code value for GRIB files reading	CGRIB_TYPE : type de fichier GRIB to read	CREC_NAME : nom du param to read KNUM_GRIB : code du param to read KTYPE_GRIB : code

			du type de niveau KLEV1: niveau
INIT_FORC_GRIB	Initialises variables for forcing interpolation after routine INIT_PGD_SURF_ATM	HPROGRAM : calling program	
MODD_FORC_OFFLINE_NC	Declares variables useful to read forcing files.		
MODD_OBJ	Declares variables to characterise objects		
MODD_PERT_RAIN	Declares variables to perturb rainfall		
PGD_GRIDTYPE_INIT	Initialises the parameters if the grid on which rainfall are projected	HGRID :grid type KGRID_PAR : number of parameters to define the grid PGRID_PAR : parameters that define the grid	
READ_FORC_GRIB	Reads forcing variables in GRIB files	HFILE: name of the GRIB file to read CGRIB_TYPE : type of GRIB to read OINTERP : true if interpolation	PT2M : tempé. à 2m PQ2M : hum. spécifique à 2m PRADSW : rayt courte longueur d'ondes PRADLW : rayt longue longueur d'ondes PU10M,PV10M : composantes du vent à 10m PPS :pression de surface PRR1 :précipitations liquides horaires PRS1 : précipitations solides horaires
READ_FORC_SAF	Reads forcing variables for SAFRAN GRIB files	KDATE_DEB : beginning date KNB_HOUR_SAF : number of hours to read	PT2M,PQ2M,PRADSW,PRADLW,PFF10M,PRR1
READ_GRIB_LARGE_RAIN	Reads rainfall on all the domain covered by the grib file	CGRIB_TYPE : type of the GRIB file to read	PLARGE_RAIN : pluie horaire sur la grande grille
READ_GRIB_SAF	Reads a SAFRAN grib file.	Same as READ_GRIB	Idem READ_GRIB

READ_INTERP_FORC_GRIB	Reads a forcing variable from a GRIB file and interpolates on ISBA grid.	HFILE: name of the GRIB file to read KNUM_GRIB : code of the variable to read KLTYPE_GRIB : code of the type of level KLEV1: level OINTERP : true if interpolation	PFIELD_OUT : valeur du paramètre de forçage sur la grille SURFEX
READ_INTERP_FORC_SAF	Reads a forcing variable from a GRIB file and interpolates on ISBA grid.	HFILE: name of the GRIB file to read KNUM_GRIB : code of the variable to read KLTYPE_GRIB : code of the type of level KLEV1: level OINTERP : true if interpolation	PFIELD_OUT : value of the forcing variable on ISBA grid
READ_INTERP_QPE_PLUVIOS	Reads quantitative estimates of rain coming from rain gauges and interpolates on ISBA grid.	HFILE: name of the GRIB file to read	PFIELD_OUT : rainfall on ISBA grid
READ_INTERP_QPE_RADAR	Reads quantitative estimates of rain coming from weather radars and interpolates on ISBA grid.	HFILE: name of the GRIB file to read	PFIELD_OUT : rainfall on ISBA grid
REGIONS_2D	Determines the areas where meshes are connected and with rainfall higher than a threshold (OINAREA = true)	OINAREA : T if mesh where rainfall exceeds a given thresholds KNUMREG : Number of the area that encloses the considered mesh KNBTREG : total number of regions	KNUMREG , KNBTREG
SELECT_MEMBER	Select some members in a random pool according to a probability law.		
WRITE_TIME	Convert format to write time	ITIME ISPACE HSTEP	HTDATE YPAS

Routines MésoNH, SURFEX ou OFFLINE récupérées pour la lecture des GRIB :

WRITE_NETCDF	Variables DOUBLE become variables REAL

5. Routines integration in offline :

Routines developed specifically for ISBA-TOP coupling are written in bold.

Routines to create forcing files :

CREATE_FORC
- => INIT_FORC_GRIB
- => PREP_OUTPUT_GRID
- **=> READ_FORC_GRIB**
  - **=> GET_GRIB_CODE**
  - **=> READ_INTERP_FORC_GRIB**
    - => PREP_GRIB_GRID
    - => READ_GRIB
    - => HOR_INTERPOL
- **=> READ_GRIB_LARGE_RAIN**
- **=> CUMUL_TO_HOURLY_RR**
- **=> READ_FORC_SAF**
- **=> READ_INTERP_QPE_RADAR**
- **=> READ_INTERP_QPE_PLUVIOS**
- **=> CHANGE_LOC_OBJECTS**
- => HOR_INTERPOL
- **=> SELECT_MEMBER**
- **=> CHANGE_AMPLI_RAIN**
- =>*WRITE_NETCDF*

Domain et catchments :

PGD
- => PGD_SURF_ATM
  - => PGD_NATURE
    - => *PGD_ISBA*
      - **=> PGD_TOPD**
        - **=> READ_NAM_PGD_TOPD**
        - **=> INIT_TOPD_PGD**
          - **=> INIT_TOPD**
          - **=> DESC_CATCHMENTS**
            - **=> GET_UPSLOPE**
            - **=> WRILE_FILE_1MAP**
          - **=> FIND_SUBCATS**
            - **=> GET_UPSLOPE**
            - **=> WRILE_FILE_1MAP**
        - => GET_GRIDTYPE_CONF_PROJ
        - => LATLON_CONF_PROJ
        - => GET_GRIDTYPE_LONLAT_REG
        - => XY_IGN
        - **=> MAKE_MASK_TOPD_TO_ISBA**
        - **=> MAKE_MASK_ISBA_TO_TOPD**
        - **=> WRITE_FILE_MASKTOPD**

Simulation :

OFFLINE

=> *READ_ALL_NAMELIST*
=> READ_NAMELISTS_TOPD
=> READ_NAM_PGD_TOPD
=> READ_NAM_TOPD
=> READ_NAMELISTS_ISBA_n
=> *READ_ISBA_CONF_n*
=> READ_NAMELISTS_GARDEN_n
=> *READ_TEB_GARDEN_CONF_n*

=> INIT_SURF_ATMN
=> INIT_NATUREn
=> INIT_ISBAn
=> *COMPUTE_ISBA_PARAMETERS*
=> INIT_SURF_TOPD
=> INIT_TOPD_OL
=> INIT_TOPD
=> READ_TOPD_HEADER_DTM
=> READ_TOPD_FILE
=> READ_TOPD_HEADER_CONNEX
=> READ_CONNEX_FILE
=> READ_SLOPE_FILE
=> READ_TOPD_FILE
=> INIT_COUPL_TOPD
=> DG_DFTO3L
=> AVG_PATCH_WG
=> READ_FILE_MASKTOPD
=> READ_TOPD_FILE
=> ISBA_TO_TOPD
=> RESTART_COUPL_TOPD
=> READ_TOPD_FILE
=> READ_FILE_ISBAMAP
=> *READ_ISBA_CONF_n*

=> COUPLING_SURF_ATMN
=> COUPLING_NATUREn
=> COUPLING_ISBA_SVAT_n
=> COUPLING_ISBA_OROGRAPHY_n
=> COUPLING_ISBA_CANOPY_n
=> *COUPLING_ISBA_n*
=> COUPLING_SURF_TOPD
=> BUDGET_COUPL_ROUT
=> COUPL_TOPD
=> DG_DFTO3L
=> AVG_PATCH_WG
=> INIT_BUDGET_COUPL_ROUT
=> ISBA_TO_TOPD

=> **CREATE_RANDOM_MAP**
=> **READ_RANDOM_NUMBER**
=> **RECHARGE_SURF_TOPD**
=> **TOPODYN_LAT**
=> **SAT_AREA_FRAC**
=> **TOPD_TO_ISBA**
=> **TOPD_TO_DF**
=> **DISPATCH_WG**
=> **CONTROL_WATER_BUDGET**
=> **ISBA_TO_TOPDSAT**
=> **DIAG_ISBA_TO_ROUT**
=> **ROUTING**
=> **WRITE_FILE_ISBAMAP**
=> **COUPL_TOPD_SUB**
idem COUPL_TOPD sauf
=> **TOPODYN_LAT_SUB** à la place de **TOPODYN_LAT**
=> **ROUTING_SUB** en plus de **ROUTING**
=> **ROUT_DATA_ISBA**
=> **DIAG_ISBA_TO_ROUT**
=> **ISBA_TO_TOPD**
=> **DIAG_ISBA_TO_ROUT**
=> **ROUTING**

=> ISBA
=> HYDRO
=> HYDRO_SOIL
=> *HYDRO_SGH*
=> **WRITE_DISCHARGE_FILE**
=> **WRITE_BUDGET_COUPL_ROUT**
=> **WRITE_DISCHARGE_FILE_SUB**

=> **PREP_RESTART_COUPL_TOPD**
=> **WRITE_FILE_MAP**
=> **WRITE_FILE_ISBAMAP**

References :

Artinyan, E., Vincendon, B., Kroumova, K., Nedkov, N., Tsarev, P., Balabanova, S. and Koshinchanov, G., 2016: Flodd forecasting and alert system for Arda River basin, Journal of Hydrology, 541, 457-470.

Bouilloud, L., Chancibault, K., Vincendon, B., Ducrocq, V., Habets, F., Saulnier, G.-M., Anquetin, S., Martin, E. and Noilhan, J., 2010: Coupling the ISBA Land Surface Model and the TOPMODEL Hydrological Model for Mediterranean Flash-Flood Forecasting : Description, Calibration, and Validation Journal of Hydrometeorology, 11, 315-333.

Edouard S., 2016 : Prévision d'ensemble des crues rapides méditerranéennes, PhD thesis, Université de Toulouse, Université Toulouse III - Paul Sabatier.

Edouard S., Vincendon B. and Ducrocq V., 2017 : Ensemble-based flash-flood modelling : Taking into account hydrodynamic parameters and initial soil moisture uncertainties, accepted for publication in Journal of Hydrology.

Vincendon B, 2010 : Apport des modèles météorologiques de résolution kilométrique pour la prévision des crues rapides méditerranéennes : vers une prévision d'ensemble des débits en région Cévennes-Vivarais, PhD thesis, Université de Toulouse, Université Toulouse III - Paul Sabatier.

Vincendon B., Ducrocq V., Saulnier G.M.; Bouilloud L., Chancibault K., Habets F., Noilhan J., 2010 : Benefit of coupling the ISBA land surface model with a TOPMODEL hydrological model dedicated to Mediterranean flash floods, Journal of Hydrology, 394, 256-266.

Vincendon B., Ducrocq V., Nuissier O. et Vié B. : Perturbation of convection-permitting NWP forecasts for flash-flood ensemble forecasting, *Nat. Hazards Earth Syst. Sci.,* 11, 1529-1544, doi:10.5194/nhess-11-1529-2011, 2011

Vincendon B., Dewaele H., Edouard S., Ducrocq V., Lespinas F., Delrieu G., Anquetin S., 2016 : Modeling flash floods for any Mediterranean sub-catchments for road management purposes, *Journal of Hydrology*, available online 30 May 2016, http://dx.doi.org/10.1016/j.jhydrol.2016.05.054.

Appendix 1 :
Details on some routines

I. TOPODYN_LAT :

Aim

topodyn_lat.F90 aims at managing the lateral transfer of the water of each pixel of a catchment according to the hydrological similarity index of this pixel. It needs the topographic files that describe the catchments (see appendix 3).

Input data consists of a water level to be distributed on each pixel of each catchment = HTp
Output data are :

- Water deficit for each pixel of each catchment DEFp,
- the hydrological similarity index of each pixel of each catchment λp,
- the critical index for which saturation is reached for each catchment λsatbv.

How does it works ?

Each catchment is treated independently from the other.

A mask is created with pixels where water is added (HTp>=0) : the value is 1 for them , XUNDEF for the other pixels. This value is propagated to the connected pixels (i.e. neighbour upstream) : if the mask of pixel p is 1, the mask of all the connected pixels is 1. (routine *flowdown.F90*).
The following only concerns those pixels.

Water deficit before lateral redistribution are : DEFpBefore=DMAXp – HTp. (DMAXp and are computed in routine *recharge_surf_topd.F90*). The catchment averaged deficit is deduced : DEF_AVGbv_Before = SUM(DEFpBefore)/Number of connected pixels.

The hydrological similarity indexes before lateral distribution are computed:
λp =ln(HTp*pixel area/tan(beta)) .
max(λp) and min(λp) are kept.

To compute the saturation index :
Classes of indexes between the max(λp) and min(λp) are determined (pixels within the same class will have the same hydrological behaviour). The class that permit to be close to the water state before redistribution is searched.
For a given class C,
the saturation index is λsatC=min(λp) + pas *(C-1)
the index of dry pixel is λsecC=λsatC-DMAXp/M.

The class of each pixel is then determined :

- if λp>=λsatC, the pixel is saturated
- if λp<=λsecC, the pixel is dry => moy(DMAX)C=Somme(DMAX)p/Nb pixels secs

If all the pixels are saturated or if all the pixels are dry or if all the pixels are either saturated or dry, moy(λp)C=0.
Else, moy(λp)C=SUM(λp) on unsaturated and wet pixels/ Number of those unsaturated and wet pixels.

The catchment-averaged deficit before water redistribution (water lacking in average to reach saturation) is compared to the catchment-averaged deficit for class C : a difference is computed.

The class Cmin that lead to the minimum difference is searched. And the parameters of this class are selected for this catchment :
λsatbv=λsatCmin
nb saturated pixels =nb saturated pixels for class Cmin
nb dry pixels =nb dry pixels for class Cmin
moy(DMAXp)=moy(DMAX)Cmin
moy(λp)=moy(λ)Cmin.

Finally, local deficits are computed :
For pixel p :

- if λsat-DMAXp/M < λp < λseuil,

DEFp = M *(DEF_AVGbv_Before-λp)*((M*DEF_AVGbv_Before-DMAXp*fraction dry pixels)/(1-fraction saturated pixels -fraction dry pixels) or 0 if the denominator is <=0.

- if λp >= λsatbv, DEFp=0
- if λp<= λsatbv-DMAXp/M, DEFp=DMAXp.

II. ROUTING :

Aim

routing.F90 aims at transferring both contributions to discharge , runoff (fast contribution) and deep drainage (slow contribution) up to the outlet of a catchment.
Input data are : the volume of runoff and drainage for the given time step, current time step (ISBA and Topodyn), final time step of the simulation.
Output data is the total discharge at the outlet at the given time step.

How does it works ?

Runoff and drainage are routed with a geomorphological method.

For each pixel of a catchment, the distance along the hillslope and along the river up to the outlet can be read in the topographical files (see appendix 3). From the speed of water along the slopes, in the ground and in the river, the time of transfer of water from any pixel p of a catchment BV up to the outlet of BV is computed.

At each time step, the time when the produced runoff an drainage will arrive at the outlet is thus computed. The discharge at a given time step is the sum of the volume of water that has arrived at the outlet at the considered time step.

Both components (runoff and discharge) are added to obtain the total discharge.

Appendix 2 :
Water recharge from ISBA to TOPMODEL

Aim

ISBA mesh which surface is S, the water level is HI (in meters). The aim is to dispatch this water on the corresponding pixels (at the MNT resolution).
N pixels which surface is s are enclosed in the ISBA mesh..
S=N.s when a mesh is totally included into a catchment.
The volume of water to dispatch on pixels is :

(1) WaterVolToDispatchInitially=HI.S=HI.N.s

The Topodyn approach needs for each pixel the topographical index, the « water deficit » (water lacking to reach saturation), the maximum « authorized » deficit (annotated DMAXp, introduced by Saulnier et Datin (2000) to allow the budget closure). The water deficit for a pixel p can be expressed as a water level in m, annotated HTp.

In Topodyn, the water added at a given time step (and that should be distributed among the catchment) is called « water recharge ». This « recharge » notion is kept here even if it can be negative.

Recharge computation

At time step t-dt, the water content of pixel p is WGp(t-dt).
The water content at time step t is changed according to the variation computed by ISBA (ΔWGi(t)) between t-dt and t on ISBA mesh i. This variation is applied to the pixels after lateral distribution at the previous time step (Wgp(t-dt)).

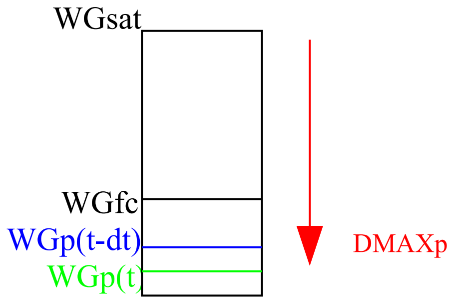
The new water content of a pixel before lateral distribution is :
WGp(t)=WGp(t-dt) + ΔWGi(t)/Hp with Hp being the depth of the sol layer.

At time step t, several scenarios can happen regarding the evolution of water content of pixel p. According to those secanrii the computation of recharge and maximal deficit will be different.

1- If the water content is reduced : ΔWGi(t) < 0

a If WGp(t-dt) < WGfc :

Then at time step t, the water content is still below field capacity :



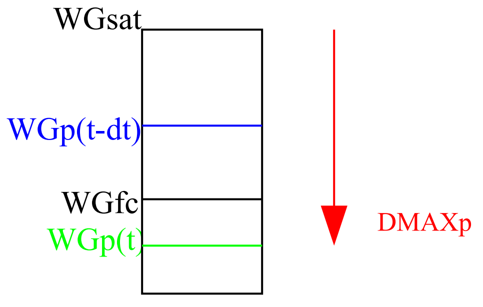
The water content is still below field capacity, so no water is available for lateral distribution by Topodyn but the threshold for deficits is given to topodyn as the level at time t (lowest level) :

HTp = 0

DMAXp = (WGsat – WGp(t))* Hp

b If WGfc < WGp(t-dt) < WGsat :

i. If WGp(t) < WGfc :

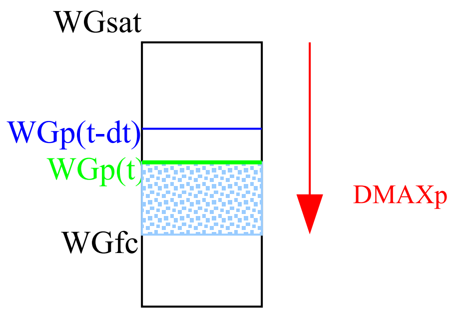


Same as previous case :

HTp = 0

DMAXp = (WGsat – WGp(t))* Hp

ii. If WGfc < WGp(t) < WGsat :



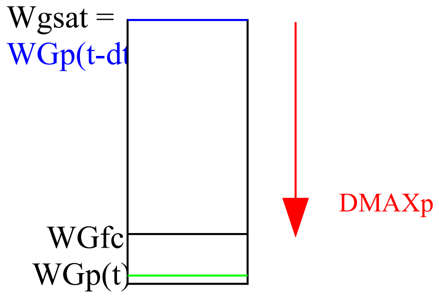
An hypothesis is performed : the water above field capacity is available for lateral transfers. The threshold for deficits is given to Topodyn as the level of field capacity (in this case Topodyn can distribute water that has been distributed already)

HTp = (WGp(t) - WGfc)* Hp

DMAXp = (WGsat – WGfc)* Hp

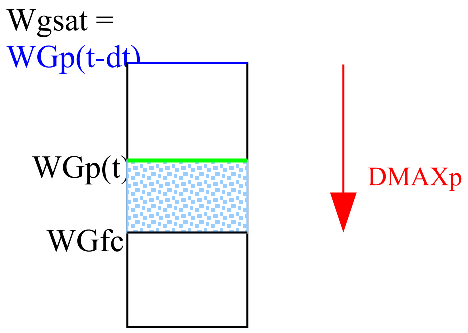
c If WGp(t-dt) = Wgsat :

i. if WGp(t) < WGfc :



Same as b) i.
HTp = 0
DMAXp = (WGsat – WGp(t))* Hp

ii. If WGfc < WGp(t) < WGsat :

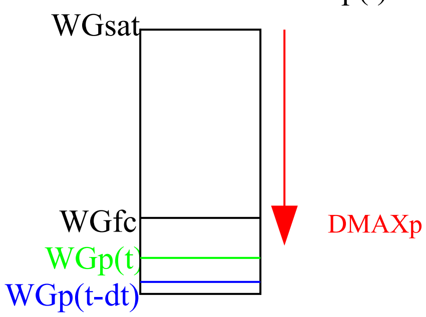


The water above field capacity is available for lateral transfers. The threshold for deficits is given to topodyn as the level of field capacity :
HTp = (WGp(t) - WGfc)* Hp
DMAXp = (WGsat – WGfc)* Hp

2- If the water level is increased : ΔWGi(t) >0.

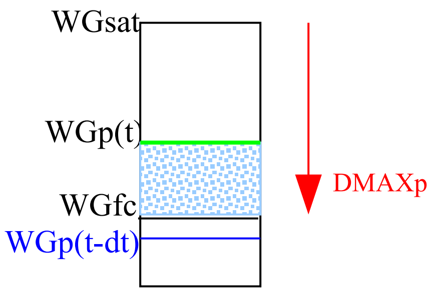
a If WGp(t-dt) < WGfc :

i. If WGp(t) < WGfc :



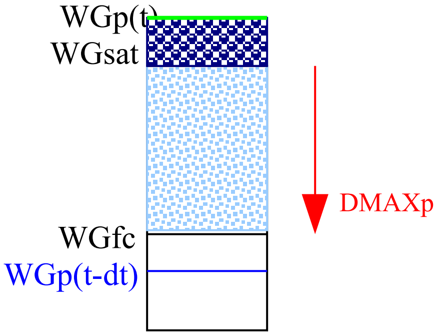
The water content remain below field capacity. The threshold for deficits is given to Topodyn as the level at time t :
HTp = 0
DMAXp = (WGsat – WGp(t))* Hp

ii. Si WGfc < WGp(t) < WGsat :



The water above field capacity is available for lateral transfers. The threshold for deficits is given to topodyn as the level of field capacity :
HTp = (WGp(t) - WGfc)* Hp
DMAXp = (WGsat – WGfc)* Hp

iii. If water content at time t exceeds saturation level :



The water between WGfc and Wgsat is available for lateral transfers. The threshold for deficits is given to topodyn as the level of field capacity :

HTp = (WGsat - WGfc)* Hp

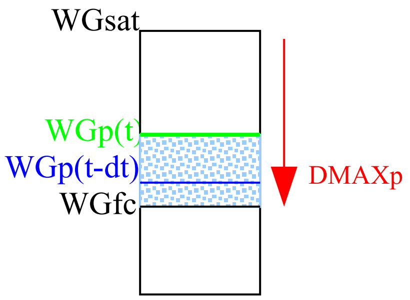
DMAXp = (WGsat – WGfc)* Hp

An excess of water has to be dispatched on others unsaturated pixels. Let HOSp be this excedant water for pixel p.

HOSp=(WGp(t) – WGsat) * Hp

b If WGfc < WGp(t-dt) < WGsat :

i. If WGfc < WGp(t) < WGsat :

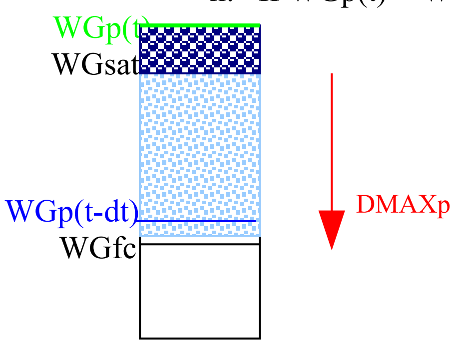


The water above field capacity is available for lateral transfers. The threshold for deficits is given to topodyn as the level of field capacity :

HTp = (WGp(t) - WGfc)* Hp

DMAXp = (WGsat – WGfc)* Hp

ii. If WGp(t) > WGsat :



The water between WGfc and Wgsat is available for lateral transfers. The threshold for deficits is given to topodyn as the level of field capacity :

HTp = (WGsat – WGfc) *Hp

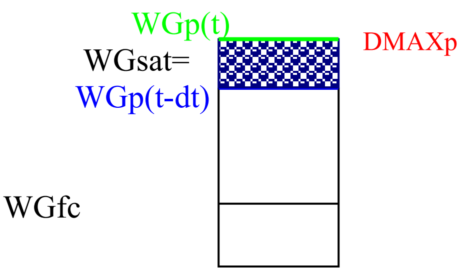
DMAXp = (WGsat – WGfc) * Hp

An excess of water has to be dispatched on others unsaturated pixels.

HOSp=(WGp(t) – WGsat) * Hp

c If WGp(t-dt)=WGsat :

At time step t, the water content exceeds saturation.



The pixel being already saturated, the exeedant water is not available for lateral distribution but dispatched on unsaturated pix els.

HTp = 0

DMAXp = 0

HOSp=(WGi(t) – WGsat) * Hp

3- Finally

With the hypothesis that the water recharge is always above field capacity (that implies that sometimes, TOPODYN transfers laterally water that has been transferred at the previous time step) only 3 cases concern a pixel p, whether we are in recession or in flood :

New water content	*Recharge to be provided to Topodyn (HTp)*	*Maximum deficit to be provided to Topodyn (DMAXp)*	*Water to dispatch on the unsaturated pixels of the mesh*
WGp(t) < WGfc	0	(WGsat – WGp(t))* Hp	
WGfc<= WGp(t) < WGsat	(WGp(t) - WGfc)* Hp	(WGsat – WGfc)*Hp	
WGp(t) >= WGsat	(WGsat - WGfc)*Hp	(WGsat – WGfc)*Hp	(WGp(t) – WGsat) * Hp

II- <u>Case of saturated pixels.</u>

If Π pixels become saturated with the water supplied by ISBA, a part of this supplied water will be « consumed ».

(2) Water Volume used to reach saturation = Σ_Π(dp. s)= s . Σ_Π(dp)
with dp =deficit of pixel p reaching saturation.

The rest of the water is dispatched in a second step on the unsaturated pixels.

(3) Water Volume above saturation = Σ_p(hp. s) = s . Σ_p(hp)
with hp =level above saturation for pixel p.

So the new volume of water to be provided to Topodyn for lateral transfer on unsaturated pixels is :

(4) Water Volume for Topodyn distribution on N-Π pixels = Initial Water Volume or Topodyn distribution – Water Volume used to reach saturation
= HI.N.s - s . Σ_p(dp)
= s . (HI.N - Σ_p(HTp))

For water level, the considered surface is the one of unsaturated pixels :

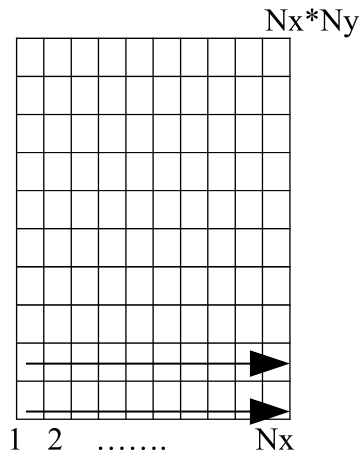
(5) Water Level for Topodyn distribution on N-Π pixels = s/(S-p.s) . (HI.N - Σ_p(HTp))
= 1/(N-p) . (HI.N - Σ_p(HTp))
= (HI.N - Σ_p (HTp)) /(N-p)

Appendix 3 :
« Topographic » files for TOPODYN

Five specific files are deduced from a digital elevation model (DEM). TOPODYN works on pixels defined in those files.

I. DTM files

In the DTM files, the geographical domain is split in pixels of 50-m resolution. The elevation is given for each pixel. The header of the file gives the bottom-left point coordinates in extended LAMBERT II. Then the elevation of the pixels classified from the bottom-left point to the up-right point.

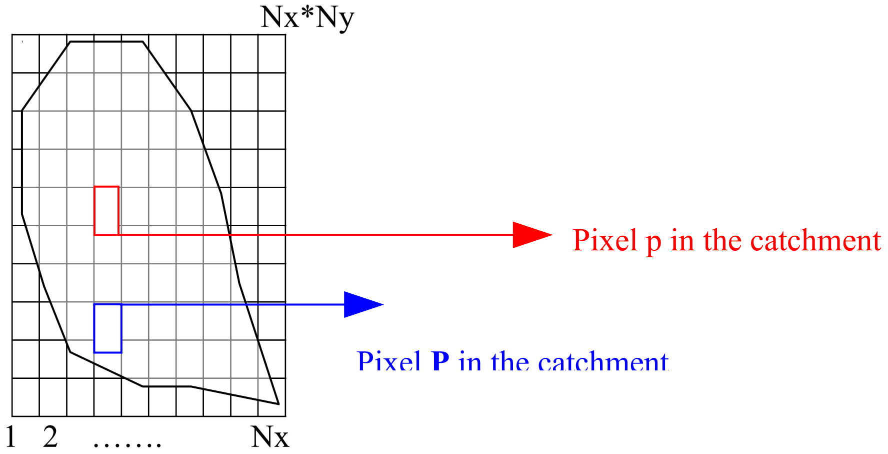


Here is an example of such a file :

«

709000.000 Abscissa bottom-left pixel
1913000.000 Ordinate bottom-left pixel
1521 Number of Abscissa
1441 Number of Ordinates
0.000 Outside value
50.000 Mesh size (m)
35.000 Minimal value
1696.000 Maximal value
672.
696.
720.
741.
760.
778.
793.
803.
…. »

Those files can contain pixels with an elevation very low compared to its 8 neighbours, this creates a kind of sink in the DEM.
Those sinks can be filled (with a program called « fillsink ») and the corrected files are the ones used. For a catchment called CAT, they are called ***CAT_FilledDTM.map***.
Their structure is the same than the one described formally.
In those files, the domain is a rectangle that surrounds the catchment. The pixels of this rectangle are either a default value (0 generally) if the pixel is out of the catchment or the elevation if the pixel is within the catchment.



Those elevation are arranged in the variable **XTOPD**(**P**) of the model.

II. « Connection » file

From the file CAT_FilledDTM.map, a « connection » file can be computed. For each pixel, the upstream pixels (the neighbours with an higher elevation) are listed as well as the fraction of water that the given pixel can receive from its upstream neighbours. Those upstream contributing neighbours are called « connected ».

For a pixel numbered p1(numbering from the bottom-left corner to the up-right one), the file gives :

- the pixel elevation,
- the number of upstream neighbours
- for each of the upstream neighbours,
  - number of the neighbour
  - percentage of drained area for the pixel p1/ drained area for the neighbour pixel.

« Pixel_Ref Z_(m) Pixel_Type Nb_Upslopes_Pixels (UpPixel_Ref,UpPixel_%Giving)
2191761 Number of pixels defining the catchment
37.000 Minimal elevation (m)
1696.000 Maximal elevation (m)
1521 Number of Abscissa
1441 Number of Ordinates
50.000 Space Resolution (m)
734698 1696.000 0 0 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000
736219 1695.000 0 0 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000
734699 1695.000 0 0 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000 0. 0.000000
…. »

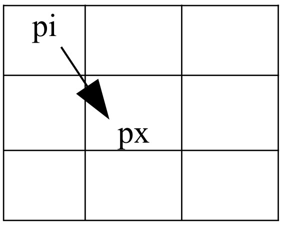
Two kinds of « connections » files can be obtained :
- in the mono-directional approach (« Monod » files), a pixel can only give water to the neighbour with the higher slope.
- in the multidirectional approach (« Multid » files) : a pixel can distribute water to all the downstream pixels proportionally to the slope.

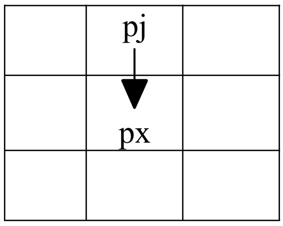
The hydrological network is defined in two steps. First, the « water sources » are searched thanks to two thresholds :

- a0 : the smallest drainage area : this determine the length of the network
- lamba0 : the smallest topographic index : this determine the of level of branching of the network.

From those sources, the connected pixels permit to obtain the river.

__How is computed the percentage of transferred water with the MONOD method :__

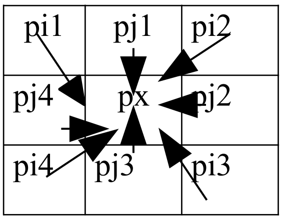




If Zi>Zx and Zi=Zmax
Slope (i to x)=$(Zi-Zx)/(\Delta x\ .2^{1/2})$

IF Zj>Zx AND Zj=Zmax
Slope (j to x)=$(Zj-Zx)/(\Delta x\ )$

__How is computed the percentage of transferred water with the MULTID method :__



TotalWaterRecieved(px) =
$Sum_i[(Zi-Zx)/(\Delta x.2^{1/2})]+$
$Sum_j[(Zj-Zx)/(\Delta x)]$

WaterFlux(pi1 vers px) =
Pente(i1 to x)/
[Sum_i(Slope(i))+Sum_j(Slope(j))]

WaterFlux(pi1vers px) =
(Zi1-Zx) /
[$Sum_i(Zi-Zx)+Sum_j(Zj-Zx).2^{1/2}$)]

WaterFlux(pj1vers px) =
$(Zj1-Zx).2^{1/2}$ /
[$Sum_i(Zi-Zx)+Sum_j(Zj-Zx).2^{1/2}$)]

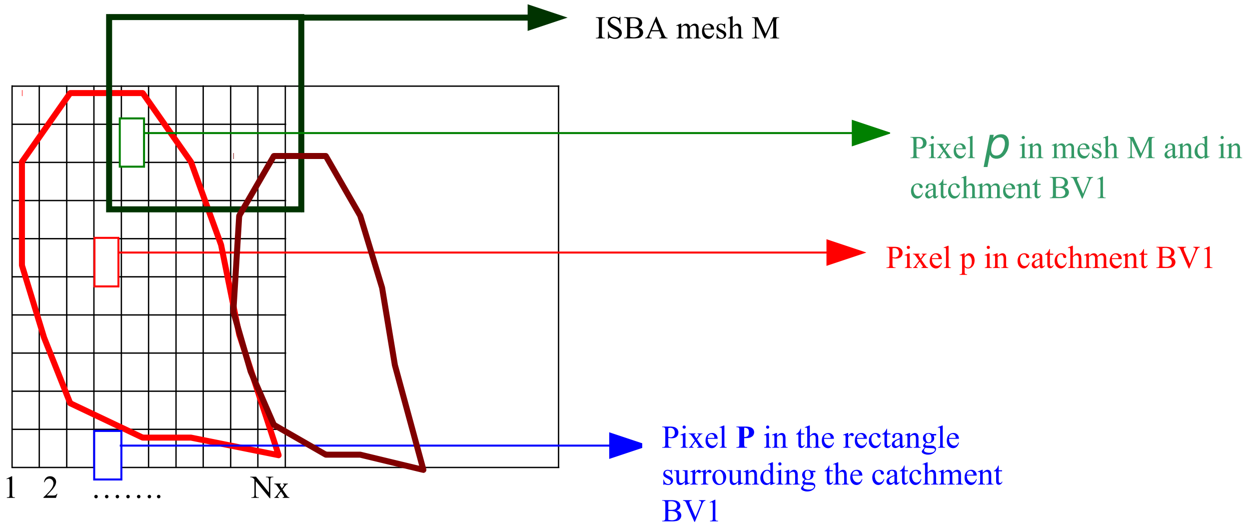
The produced files for a catchment CAT is called ***CAT_River_MonoD.vec*** or ***CAT_River_MultiD.vec***.

For ISBA-TOP, only the MULTID file is kept : it is renamed ***CAT_ connections.vec***

In those files, only the pixels of the catchment are listed (not all those of the surrounding rectangle.).

The data of « connection » files are arranged in the variable **XCONN** of the model :
XCONN(BV,p,1)= **P**
XCONN(BV,p,2)= XTOPD(**P**)
XCONN(BV,p,3)= 0
XCONN(BV,p,4)= Total number of pixels connected to the pixel p (from 0 to 8)
XCONN(BV,p,5)= n° (**P**1) of the pixel connected to pixel p
XCONN(BV,p,6)= percentage of water coming to p from **P**1
XCONN(BV,p,7)= n° (**P**2) of the pixel connected to pixel p
XCONN(BV,p,8)= percentage of water coming to p from **P**2
XCONN(BV,p,9)= n° (**P**3) of the pixel connected to pixel p
XCONN(BV,p,10)= percentage of water coming to p from **P**3
XCONN(BV,p,11)= n° (**P**4) of the pixel connected to pixel p
XCONN(BV,p,12)= percentage of water coming to p from **P**4
XCONN(BV,p,13)= n° (**P**5) of the pixel connected to pixel p
XCONN(BV,p,14)= percentage of water coming to p from **P**5
XCONN(BV,p,15)= n° (**P**6) of the pixel connected to pixel p
XCONN(BV,p,16)= percentage of water coming to p from **P**6
XCONN(BV,p,17)= n° (**P**7) of the pixel connected to pixel p
XCONN(BV,p,18)= percentage of water coming to p from **P**7
XCONN(BV,p,19)= n° (**P**8) of the pixel connected to pixel p
XCONN(BV,p,20)= percentage of water coming to p from **P**8

III. ISBA-TOP variables used to manage the masks (to go from a grid to another) :



NMASKT (created by the routine *make_mask_topd_to_isba.F90*)
NMASKT(BV,p)=M

NMASKI (created by the routine *make_mask_isba_to_topd.F90*)
NMASKI(M,BV, *p*)= p

NLINE (read in connection files and initialised in routine i*nit_topd.F90*)
NLINE(BV, **P**)= p

NNPT(BV)=NX(BV)*NY(BV) = number of **P**

NNMC(BV)=number of pixels within BV=number of p

IV. « slope » files

For each pixel, le drainage area as well as the topographic indexes (a/tg(beta)) can be computed and arranged in the files ***CAT_A_ ….*** and ***CAT_ATB_….*** respectively.
For ISBA-TOPODYN, only the file ***CAT_ATB__River_MultiD.vec*** is used : it is called ***CAT_slope.vec*** .
In those files, only the pixels of the catchment are listed (not all those of the surrounding rectangle.).

V. « distance » files

From the file ***CAT_FilledDTM.map*** and from the location of the river (files _River_…), the distance that water has to run over the hillslopes to reach the river and the distance that water has to run along the river to reach the outlet can be determined.
For a catchment CAT, the files are called ***CAT_Hillslope_Distance.map*** and ***CAT_River_Distance.map.*** They have the same structure as the file ***CAT_FilledDTM.map***
For ISBA-TOP simulations, the files are called respectively, ***CAT_HillDist.map*** et ***CAT_RiverDist.map.***

VI. Useful files for ISBA-TOPODYN

Finally, 5 files for each catchment are used :

CAT_FilledDTM.map
CAT_connections.vec
CAT_slope.vec
CAT_RiverDist.map
CAT_HillDist.map

Appendix 4 :
Grib codes for each CGRIB_TYPE

Those codes are managed in routine GET_GRIB_CODE. f90

	DEFAULT/ PEAROM	**SAFRAN**	**AROME/ AROMAN**	**PEAROP**	**ARPEGE**	**MESONH**
T2M	KNUM_GRIB=11 KTYPE_GRIB=105 KLEV1=2	cf. DEFAULT	cf. DEFAULT	KNUM_GRIB=167 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=11 KTYPE_GRIB=111 KLEV1=2	KNUM_GRIB=11 KTYPE_GRIB=105 KLEV1=2
Q2M	KNUM_GRIB=51 KTYPE_GRIB=105 KLEV1=2	cf. DEFAULT		KNUM_GRIB=51 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=51 KTYPE_GRIB=111 KLEV1=2	KNUM_GRIB=51 KTYPE_GRIB=105 KLEV1=2
U2M	/				/	/
RADSW	KNUM_GRIB=111 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=116 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=105 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=176 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=111 KTYPE_GRIB=111 KLEV1=0	KNUM_GRIB=111 KTYPE_GRIB=105 KLEV1=0
RADLW	KNUM_GRIB=112 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=115 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=104 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=177 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=112 KTYPE_GRIB=111 KLEV1=0	KNUM_GRIB=112 KTYPE_GRIB=105 KLEV1=0
U10M	KNUM_GRIB=33 KTYPE_GRIB=105 KLEV1=10	/	cf. DEFAULT	KNUM_GRIB=165 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=33 KTYPE_GRIB=111 KLEV1=10	KNUM_GRIB=33 KTYPE_GRIB=105 KLEV1=10
V10M	KNUM_GRIB=34 KTYPE_GRIB=105 KLEV1=10	/	cf. DEFAULT	KNUM_GRIB=166 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=34 KTYPE_GRIB=111 KLEV1=10	KNUM_GRIB=34 KTYPE_GRIB=105 KLEV1=10
FF10M	/	KNUM_GRIB=32 KTYPE_GRIB=105 KLEV1=10				/
PS	KNUM_GRIB=1 KTYPE_GRIB=1 KLEV1=0	cf. DEFAULT	cf. DEFAULT	KNUM_GRIB=152 KTYPE_GRIB=109 KLEV1=1		KNUM_GRIB=1 KTYPE_GRIB=105 KLEV1=1
RR1	KNUM_GRIB=150 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=169 KTYPE_GRIB=1 KLEV1=0	cf. DEFAULT	KNUM_GRIB=84 KTYPE_GRIB=1 KLEV1=0	KNUM_GRIB=62 KTYPE_GRIB=111 KLEV1=0	KNUM_GRIB=62 KTYPE_GRIB=1 KLEV1=0
RS1	KNUM_GRIB=99 KTYPE_GRIB=1 KLEV1=0	cf. DEFAULT	cf. DEFAULT			KNUM_GRIB=79 KTYPE_GRIB=1 KLEV1=0
RG1	KNUM_GRIB=29 KTYPE_GRIB=1 KLEV1=0	/	cf. DEFAULT			KNUM_GRIB=78 KTYPE_GRIB=1 KLEV1=0

Appendix 5 :
Example of script generateRandomFiles.sh

```
#/bin/sh
FIC_RANDOM1='randomMapGeneration.txt'
FIC_RANDOM2='randomNumbers.txt'

#######################################################################
#cettte première partie permet de générer des nombres aléatoires pour
#faire varier les paramètres hydrodynamiques
#pour cela on les fait varier entre leur valeurs extrêmes.
#######################################################################
for i in `seq 21 1 100`
do
        mkdir simu$i
        #a=1
        a=`echo $[(RANDOM % ($[60]+1))+70] | awk '{print $1/100}'`
        #b=`echo $[(RANDOM % ($[40000]+1))+80000] | awk '{print $1/100000}'`
        b=1
        #c=`echo $[(RANDOM % ($[20000]+1))+90000] | awk '{print $1/100000}'`
        c=1
        #d=`echo $[(RANDOM % ($[60000]+1))+70000] | awk '{print $1/100000}'`
        d=1
        #e=`echo $[(RANDOM % ($[50000]+1))+75000] | awk '{print $1/100000}'`
        e=1
        #f=1
        f=`echo $[(RANDOM % ($[100])+1)] | awk ' {if($1%2==0){ print 0.2+$1*0.008}else {print 1+$1*0.036}}'`
        #g=1
        g=`echo $[(RANDOM % ($[100])+1)] | awk ' {if($1%2==0){ print 0.5+$1*0.005}else {print 1+$1*0.015}}'`

        echo "alea Wsat" $a > simu$i\/nbresalea
        echo "alea WWilt " $b >> simu$i\/nbresalea
        echo "alea WFC " $c >> simu$i\/nbresalea
        echo "alea BCOEF" $d >> simu$i\/nbresalea
        echo "alea MATPOTSAT   " $e >> simu$i\/nbresalea
        echo "alea Ksat   " $f  >> simu$i\/nbresalea
        echo "alea M   " $g >> simu$i\/nbresalea

#######################################################################
#on génère ici les nombre aléatoires permetant de créer la carte de
#perturbation de l'humidité initiale de +/-20%
#######################################################################
h=`echo $[($RANDOM % ($[10000000] + 1)) ]`
j=`echo $[($RANDOM % ($[20] + 1)) + 90]| awk '{print $1/100}'`
#j=`echo $[($RANDOM % ($[40] + 1)) + 90]| awk '{print $1/100}'`
        echo "alea hug " $h >> simu$i\/nbresalea
        echo "alea hug " $j >> simu$i\/nbresalea

#######################################################################
#on écrit ensuite ces différents nombre dans les fichiers qui seront
#lus par le modèle.
#######################################################################
echo $h > ${FIC_RANDOM1}
echo $j >> ${FIC_RANDOM1}
        echo $a > ${FIC_RANDOM2}
        echo $b >> ${FIC_RANDOM2}
        echo $c >> ${FIC_RANDOM2}
        echo $d >> ${FIC_RANDOM2}
        echo $e >> ${FIC_RANDOM2}
        echo $f >> ${FIC_RANDOM2}
        echo $g >> ${FIC_RANDOM2}

exit 0
```