CATHY - CATchment HYdrologic model

- User Guide -

*October 03, 2016*

# CATHY

## Introduction

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

……… ……… ……… ……………… ……… ……… ……… ……… ……………… ………………………

## How the model works

Three phases are required to implement a working CATHY model of a specific case study, namely pre-processing, processing, and post-processing, and the following steps are necessary to create the flow model:

1. Create the input files for the pre-processor **hap**
2. Run the pre-processor **hap**
3. Create the input files for the processor **CATHY**
4. Perform the simulation by running the processor **CATHY**
5. Run the post-processing subroutines (written in **Matlab**) to extract and view results

# Pre-processing

## Modules/subroutines for the pre-processing

The CATHY pre-processing requires the main program *cppp.f90* and all the subprograms (modules and subroutines) listed in Table 1. During the pre-processing, the main program and all involved modules and subroutines must be compiled in a specific order (as in the Table 1), as recalled in the main program *cppp*. For example the *mpar*and *mbbio*, being modules the first statements, must be compiled first.

Starting from DEM data (*dtm\_13.val*) and parameters file (*hap.in*), the main program *cppp.f90* and the related modules and subroutines produce a complete set of files describing physiographic features of a drainage system. Such files represent the input files for CATHY simulations and GIS visualization.

Table 1. Pre-processing modules and subroutines

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | | **Description** |
| **mpar** | .f90 | module | Module containing definitions of parameters assigned in the parameter file <parfile>. |
| **mbbio** | .f90 | module | The module manages the I/O between programs and binary files basin\_b/basin\_i. |
| **wbb\_sr** | .f90 | subroutine | This program defines the structure of the binary files basin\_b/basin\_i using the ascii DEM file <demfile>, which contains the catchment cell elevations. |
| **streamer** | .f90 | module | Module AB\_Normal called by dbase.f90 and shape.f90 |
| **shape** | .f90 | module | Module Shapefile |
| **qsort** | .f90 | subroutine | Quicksort. This subroutine is called by the subroutine csort. |
| **csort** | .f90 | subroutine | The subroutine CSORT (Cells Sort) sorts the cells of a catchment in the order of descending elevations. The binary file qoi contains the number of processed cells in the first record and the pointers i\_basin\_qo corresponding to the sorted cell elevations qo in the subsequent records. |
| **depit** | .f90 | subroutine | The subroutine DEPIT checks that each catchment cell has a lower cell among the neighbouring cells. When this is not verified the processed cell is raised of the quantity eps above le lowest neighbouring cells. The procedure is iterated until all the pits are removed. The number of modifications is recorded in the variable n\_modifiche. |
| **cca** | .f90 | subroutine | Contour Curvature Analysis (cca) evaluates the curvature of each cell and assigns the dmID. |
| **smean** | .f90 | subroutine | The subroutine SMEAN calculates the mean catchment slope based on maximum (theoretical) values obtained from facet analysis |
| **facet** | .f90 | subroutine | The subroutine FACET is called by dsf.f90. It calculates the aspect and the slope (positive downward) of the steepest direction within a given triangle (facet). |
| **dsf** | .f90 | subroutine | The subroutine DSF (Drainage System Features) determines two drainage directions for each DEM cell. Depending on parameter dmID, one or two drainage directions are actually employed. For each DEM cell, the following physiographic features are assigned:   1. weights associated to drainage directions: w\_1 and w\_2; 2. flows directions: p\_outflow\_1 and p\_outflow\_2; 3. elemental lengths of channels: epl\_1 and epl\_2 (it is assumed here that delta\_x=delta\_y); 4. local slopes (positive downward) of channel elements: local\_slope\_1 and local\_slope\_2; 5. the funcion ASk=AS\*\*k.   For at most the two neighbours spilling cells are also updated:   1. the upstream catchment area A\_inflow and 2. the upstream error made between selected and theoretical drainage directions sumdev\_num.   It refers to facet.f90. |
| **hg** | .f90 | subroutine | The subroutine HG (Hydraulic Geometry) determines the hillslope and channel cells within the catchment and assign the hydraulic geometry parameters. |
| **mrbb\_sr** | .f90 | subroutine | The program MRBB (Multiple RBB) reads the parameters contained in the binary files basin\_b/basin\_i and writes many DTM ascii files as output. |
| **dbase** | .f90 | module | Managing DBF files for both reading and writing. |
| **bb2shp\_sr** | .f90 | subroutine | The program reads the parameters contained in the binary files basin\_b/basin\_i and write a \*.SHP ArcGIS file.  It refers to DBF (dbase) and Shapefile (shape). |
| **cppp** | .f90 | main program | Main program for the pre-processing. |

## Input files

CATHY pre-processing requires two input files in ASCII format (editable with a text editor):

* dtm\_13.val
* hap.in

### The Digital Elevation Model

The *dtm\_13.val* is the main input file. It is a raster file containing the Digital Elevation Model (DEM) of the study area where the elevation of each cell of the basin is stored. This input file is an ASCII file exported from ArcGIS or GRASS and it is formed by a header and the matrix of data.

|  |  |
| --- | --- |
| **ArcGIS** | **GRASS** |
| ncols 5  nrow 4  xllcorner 1000.00000000  yllcorner 1000.00000000  cellsize 80.00  NODATA\_value ‐9999  20 18 16 15 8  20 19 18 17 10  20 18 16 15 8  ‐9999 15 14 13 2 | north: 1320.00000000  south: 1000.00000000  east: 1400.00000000  west: 1000.00000000  rows: 4  cols: 5  20 18 16 15 8  20 19 18 17 10  20 18 16 15 8  ‐9999 15 14 13 2 |

Figure 1. Example of ASCII files exported from ArcGIS and GRASS

The information contained in the ASCII files, exported from ArcGIS or GRASS, must be reorganized in the *dtm\_13.val* and *hap.in* files. The matrix of elevation without the header is named *dtm\_13.val*, while the header information are stored in the section of ‘structural parameters’ of *hap.in*.

### The parameter input file

The *hap.in* is a parameter file containing all the structural and terrain analysis parameters to perform the terrain analysis. The section “*structural parameters*” contains information on the discretization of the catchment surface.

|  |
| --- |
| ------------------------------------------------------------------------------------  STRUCTURAL PARAMETERS  ------------------------------------------------------------------------------------  Grid spacing along the x-direction = 20.00 (1)  Grid spacing along the y-direction = 20.00 (2)  DEM rectangle size along the x-direction = 45 (3)  DEM rectangle size along the y-direction = 43 (4)  Number of cells within the catchment = 1210 (5)  X low left corner coordinate = 2268651.15126480 (6)  Y low left corner coordinate = 2452773.22533040 (7)  ------------------------------------------------------------------------------------ |

|  |  |
| --- | --- |
| (1), (2) | Distance between each center cell in x and y directions |
| (3), (4) | Number of square cells in x direction (number of matrix columns) and in y direction (number of matrix rows) |
| (5) | Number of cells that make up the grid of square cells (number of elements of the matrix) |
| (6), (7) | Information related to the x and y low left coordinates are obtained from the ASCII files exported from ArcGIS or GRASS. |

The section “*terrain analysis parameters*” provides information useful for extracting a conceptual drainage network from the grid-based DEM. Much of the following descriptions are mainly extrapolated from:

* Orlandini et al. (2003) “Path-based methods for the determination of nondispersive drainage directions in grid-based digital elevation models”
* Orlandini and Moretti (2009) “Determination of surface flow paths from gridded elevation data”
* Camporese et al. (2010) “Surface-subsurface flow modeling with path-based runoff routing, boundary condition-based coupling, and assimilation of multisource observation data”

|  |
| --- |
| ------------------------------------------------------------------------------------  TERRAIN ANALYSIS PARAMETERS  ------------------------------------------------------------------------------------  Depit threshold slope = 0.100E-03 (8)  Drainage directions method (LAD:1,LTD:2) = 2 (9)  Upstream deviation memory factor (CBM:0,PBM:1) = 0.100E+01 (10)  Threshold on the contour curvature (NDM:-1E10;DM:+1E10) = -0.100E+11 (11)  Nondispersive channel flow (0:not-required;1:required) = 1 (12)  Channel initiation method (A:1,AS\*\*k:2,ND:3) = 1 (13)  Threshold on the support area (A) = 0.400000000E+06 (14)  Threshold on the AS\*\*k function = 16000.00 (15)  Exponent k of the AS\*\*k function = 2.00 (16)  Threshold on the normalized divergence (ND) = -0.100E-01 (17)  Path threshold slope = 0.100E-03 (18)  Drainage direction of the outlet cell (if necessary...) = 8 (19)  Boundary channel construction (No:0,Yes:1) = 0 (20)  Coefficient for boundary channel elevation definition = 0.50 (21)  Coefficient for outlet cell elevation definition = 0.50 (22)  ------------------------------------------------------------------------------------ |

|  |  |
| --- | --- |
| (9) | The definition of the **drainage directions** is done through either the D8-LAD (eight drainage directions, least angular deviation) scheme or the D8-LTD (eight drainage directions, least transverse deviation) scheme. The classical D8 (eight drainage directions) scheme (O’Callaghan and Mark, 1984; Marks et al., 1984), the earliest and simplest method for specifying flow direction, assigns flow from each pixel to one of its eight neighbors, either adjacent or diagonal, in the direction with steepest downward slope. D8 …......…......…….. (Tarboton, 1997).  In the D8-LAD and D8-LTD methods ….…....…......…....... |
| (10) | In both methods a dampening factor (λ) (i.e**. deviation memory factor**) must be considered. For λ=0 the selection of the drainage directions is based only on the local angular/transversal direction (i.e. cell based method, CBM); for λ=1, full memory of the upstream angular/transversal deviations is retained (i.e. path based method, PBM). The D8-LAD method with λ=0 reproduces the classical D8 method. As shown in Orlandini et al., 2003 along the valleys, the D8-LTD method with λ=1 appears to provide significantly better results than the D8-LAD method with λ=0. |

|  |  |
| --- | --- |
| (11) | The surface flow paths can be described by using single flow directions (convergent terrains) and multiple flow directions (divergent terrains). The contour curvature, defined as the curvature of contour lines, provides a measure of the morphological convergence of the terrain, and can therefore be applied to determine whether to use single or multiple flow directions for each grid cell. The selection of the **plan curvature threshold** allows for a morphologically meaningful use of single (nondispersive, NDM:-1E10) and multiple (dispersive, DM:+1E10) flow directions across a terrain. Combining this information with the method for the definition of the drainage directions and the dampening factor (λ), the following relevant cases can be highlighted:   * D8-LAD with λ=0 and NDM: D8 classical method proposed by O’Callaghan and Mark (1984) and Marks et al. (1984); * D8-LAD with λ=0 and DM: D∞ method proposed by Tarboton (1997); * D8-LTD with λ=1 and NDM: D8-LTD proposed by Orlandini et al. (2003); * D8-LTD with λ=1 and DM: a variant of D8-∞ (D∞-LTD). |
| (12) | A further specification must be done in the section “**Nondispersive channel flow**”: if equal to 0, channel cells where flow will potentially be dispersive are considered actually dispersive; if equal to 1, channel cells where flow will potentially be dispersive are forced to be nondispersive. |

The algorithm that incorporates the D8-LAD and D8-LTD methods for identifying the drainage network and calculating the drainage areas within a catchment requires three preliminary operations, in which (1) DEM cells of the catchment are sorted into descending elevation order (through the subroutine *csort.f90*), (2) a recursive procedure is used to raise (by filling them) the elevations of the cells located in flat or depressed areas (i.e. pits) so as to ensure a drainage direction with a small positive slope (downward) for all the cells of the catchment (through the subroutine *depit.f90*), and (3) DEM cells are sorted again into descending elevation order (through the subroutine *csort.f90* again).

|  |  |
| --- | --- |
| (8) | The quantity used by the subroutine *depit* to raise depressed cell is equal to the depit threshold slope times the grid cell size. The value of the **depit threshold slope** (i.e. the ratio between that quantity used to raise depressed cells and the grid size) is often set equal to 10-4.The D8-LAD or the D8-LTD method is then applied to form the drainage network and to calculate the drainage areas (through the subroutine *dsf.f90*). DEM cells are processed in the order of descending elevation. Upstream drainage areas are summed up over all the drained cells. |
| (13) | The distinction between hillslope and channel flow can be based on three criteria (channel initiation method) (13) by posing a threshold condition on:   * the upstream drainage area A (14) (e.g., Montgomery and Foufoula-Georgiou, 1993) * the function ASk (15,16), being S the local terrain slope and k an exponent (Montgomery and Dietrich,1988, 1989) * the gradient divergence normalized by mean gradient (17) (land surface curvature divided by the mean terrain slope) (Howard, 1994).   Using for instance the first criterion, **rill flow is assumed to occur for all those cells for which the upstream drainage area A [L2] does not exceed the threshold value At [L2], while channel flow is assumed to occur for all those cells for which A equals or exceeds At.** The option to explicitly define channel head locations is also included in the model. |
| (18) | **Path threshold slope** is not important for the extraction of the river network. (It is used only in the tool program *odx.f90* that calculates the optimal DELTA\_X for the simulation according to the physiographic characteristics obtained from the application of the hydraulic geometry theory). |
| (19) | If necessary, the **drainage direction of the outlet cell** must be specified. If this parameter is set equal to 0, the drainage direction of the outlet cell is defined automatically and set equal to the drainage direction of the upslope cell with the biggest upslope area. Drainage directions can be identified by pointers to the downslope cells. The rule employed in this set of programs is shown in the following sketch:   |  |  |  | | --- | --- | --- | | 3 | 6 | 9 | | 2 |  | 8 | | 1 | 4 | 7 |   The other possible value of this parameter are those reported in the sketch, where the center cell is the current cell. |
| (20) | The option “**Boundary channel construction (No:0,Yes:1)**” is useful to exceed the limitation of the program that perform correctly basins with only a single outlet cell. If, known the outlet cell, is necessary to define automatically the boundary of an unknown catchment, set this option equal to one maybe helpful. With this parameter equal to one, independently of the shape of the DEM, that obviously must exceed the extension the study basin, the elevation of the cells belonging to the boundary is pulled down and an outlet cell is defined. Is underlined here that the shape of the DEM is free but a sufficiently regular shape is suggested. Exceptional parameter value:   * Boundary channel construction = 10; the boundary channel construction is performed but the default procedure for depicting the boundary channel constructed described below is skipped. * Boundary channel construction >10; the boundary channel construction and the other standard procedure are performed but all the DEM is shifted upward by the value of the boundary channel construction parameter. |
| (21) | If the “Boundary channel construction” parameter is set equal to 1, the “**Coefficient for boundary channel elevation definition**” is the value of the coefficient that, multiplied for the elevation of the lowest cell of the basin, allows to define the elevation of the boundary channel cells. Value between 0.05 and 0.95 are suggested. |
| (22) | If the “Boundary channel construction” parameter is set equal to 1, the “**Coefficient for outlet cell elevation definition**” is the value of the coefficient that, multiplied for the elevation of the boundary channel cells, allows to define the elevation of the outlet cell. To be sure that the outlet cell is lower than all boundary channel cells value between 0.05 and 0.95 are suggested. |

The Boundary channel construction is performed in wbb\_sr subroutine. After the definition of the boundary channel and the outlet cell another operation is performed: all cells of the boundary channel starting from a cell near the outlet cell are depitted with the slope defined at the parameter “Depit threshold slope”. As reported in the description of the Boundary channel construction parameter, this operation can be skipped if the Boundary channel construction parameter is set equal to 10.

The sections “*rivulet network parameters*” and “*channel network parameters*” provide information useful for …....…......…......…......…......…......…......…......…......…......…......…......…......…......…........

|  |
| --- |
| ------------------------------------------------------------------------------------  RIVULET NETWORK PARAMETERS (HYDRAULIC GEOMETRY OF THE SINGLE RIVULET)  ------------------------------------------------------------------------------------  Rivulet spacing = 20.000 (23)  Reference drainage area (As\_rf) = 0.484000000000E+06 (24)  Flow discharge (Qsf\_rf,w\_rf) = 0.010 1.000 (25)  Water-surface width (Wsf\_rf,b1\_rf,b2\_rf) = 13.761 0.260 0.500 (26)  Resistance coefficient (kSsf\_rf,y1\_rf,y2\_rf) = 2.989 0.200 0.300 (27)  Initial flow discharge (Qsi\_rf) = 0.000  ------------------------------------------------------------------------------------  CHANNEL NETWORK PARAMETERS  ------------------------------------------------------------------------------------  Reference drainage area (As\_cf) = 0.484000000000E+06 (28)  Flow discharge (Qsf\_cf,w\_cf) = 0.010 1.000 (29)  Water-surface width (Wsf\_cf,b1\_cf,b2\_cf) = 2.752 0.260 0.500 (30)  Resistance coefficient (kSsf\_cf,y1\_cf,y2\_cf) = 29.891 0.200 0.300 (31)  Initial flow discharge (Qsi\_cf) = 0.000 (32)  ------------------------------------------------------------------------------------ |

|  |  |
| --- | --- |
| (23) | Rivulet spacing (for parameters 23 to 31 see readme.txt) |
| (24) |  |
| (25) |  |
| (26) |  |
| (27) |  |
| (28) |  |
| (29) |  |
| (30) |  |
| (31) |  |
| (32) |  |

## Running the pre-processor

Running the pre-processor for a given case study (i.e. specific DEM and input parameters) requires the creation of an executable file (*cppp.exe*).

The executable file is generated through the compilation ofthe main program *cppp.f90* and all the subprograms (in Teminal or Command Prompt) as follows:

|  |
| --- |
| *gfortran -O -o* ***cppp*** *mpar.f90 mbbio.f90 wbb\_sr.f90 csort.f90 qsort.f90 depit.f90 cca.f90 smean.f90 dsf.f90 facet.f90 hg.f90 mrbb\_sr.f90 bb2shp\_sr.f90 shape.f90 dbase.f90 streamer.f90 cppp.f90* |

(equal to the last line of the *compgfor* file)

Or alternatively:

|  |
| --- |
| *gfortran -c mpar.f90 mbbio.f90 wbb\_sr.f90 streamer.f90 shape.f90 qsort.f90 csort.f90 depit.f90 cca.f90 smean.f90 facet.f90 dsf.f90 hg.f90 mrbb\_sr.f90 dbase.f90 bb2shp\_sr.f90*  *gfortran –c cppp.f90*  *gfortran cppp.o mpar.o mbbio.o wbb\_sr.o streamer.o shape.o qsort.o csort.o depit.o cca.o smean.o facet.o dsf.o hg.o mrbb\_sr.o dbase.o bb2shp\_sr.o –o* ***cppp*** |

Once the executable file is generated, it is necessary to run it following the steps:

* Copy the executable cppp of the pre-processor (i.e. cppp.exe) in the same directory where the input files (dtm\_13.val and hap.in) are already saved;
* Open the Command Prompt (or Terminal for Mac OS X) window;
* Change the path to this directory, write the name of the executable (i.e. cppp) and enter.

Once the running of the executable file has started, it is necessary to follow the instructions below:

* Select the header type depending on the type of format of the output files that you want: CATHY will only accept the option = 2 (the header in the format of GRASS).
* Select the nodata value 🡪 This must be 0.0.
* Select the pointer system 🡪 HAP system (option 1)
* Final compiling

Running the executable file has allowed to generate a complete set of files describing physiographic features of the drainage system, as shown in Table 2.

Table 2. Pre-processing outputs – Processing inputs

|  |  |  |
| --- | --- | --- |
| IFN |  | I/O file names (see subroutine OPENIO for unit IFN input) |
| IIN10 | dem | DEM information (dem.dat) |
| IIN20 | lakes\_map | Map of lake cells (to be excluded by the domain)(lakes\_map.dat) |
| IIN21 | zone | Map of zone cells |
| IIN23 | qoi\_a | Vector containing the index of the cells ordered into descending elevation value |
| IIN25 | dtm\_w\_1 | Raster of the weights ‐ cardinal |
| IIN26 | dtm\_w\_2 | Raster of the weights ‐ diagonal direction |
| IIN27 | dtm\_p\_outflow\_1 | Raster of the drainage directions ‐ cardinal direction |
| IIN28 | dtm\_p\_outflow\_2 | Raster of the drainage directions ‐ diagonal direction |
| IIN29 | dtm\_local\_slope\_1 | Raster of the local slopes ‐ cardinal direction |
| IIN30 | dtm\_local\_slope\_2 | Raster of the local slopes ‐ diagonal direction |
| IIN31 | dtm\_epl\_1 | Raster of the elemental path length ‐ cardinal direction |
| IIN32 | dtm\_epl\_2 | Raster of the elemental path length ‐ diagonal direction |
| IIN33 | dtm\_kSs1\_sf\_1 | Raster of surface roughness coefficient (ks) ‐ cardinal direction |
| IIN34 | dtm\_kSs1\_sf\_2 | Raster of surface roughness coefficient (ks) ‐ diagonal direction |
| IIN35 | dtm\_Ws1\_sf\_1 | Raster of surface water width ‐ diagonal direction (Each value is the surface water width of the cells scaled in space and assigned to the cardinal drainage direction.) |
| IIN36 | dtm\_Ws1\_sf\_2 | Raster of surface water width ‐ cardinal direction (Each value is the surface water width of the cells scaled in space and assigned to the diagonal drainage direction.) |
| IIN37 | dtm\_b1\_sf | Raster of at‐a‐station scaling coefficients for the surface water width (i.e., surface water width is scaled in time with a power-law function with that exponent). |
| IIN38 | dtm\_y1\_sf | Raster of at‐a‐station scaling coefficients for the surface roughness (ks) coefficient |
| IIN39 | dtm\_nrc | Raster containing the number of rivulets per hillslope cell. |
|  | basin\_b |  |
|  | basin\_i |  |
|  | dtm\_A\_inflow |  |
|  | dtm\_hcID |  |
|  | dtm\_q\_output |  |
|  | qoi |  |

Since these files represent the input files for CATHY simulation, they must be in the same directory where other files required by the CATHY are.

It could be a good practise to create a “**dataset**” folder, which contains all the input/output files divided in preprocessing (for the output files of the pre-processing), input (for the other input files), and output (at the beginning it is an empty folder where the output files of CATHY simulation will be saved). Of course, this is just an idea; how to organize the input/output files is completely user's discretion. The important thing is that there is consistency between the dataset folder organization and the *cathy.fnames* file.

# Processor CATHY

## Input files for the processor CATHY

The input and output files required and generated by CATHY are listed in *cathy.fnames* file, as shown in Figure 2. In the first line of this file it is necessary to specify the position of these input/output files.

Open the *cathy.fnames* file, verify the correct position of all the files listed and, use the backslash (\) in quotes in Windows, the forward slash (/) are used in Mac OS X.

The unit number associated to each input and output file (e.g. unit IIN1) are used by CATHY in reading and writing statements, respectively (ref. Table 33).

|  |
| --- |
| '……….………... Directory ………..……….. /dataset'  'parm' unit IIN1  'grid' unit IIN2  'root\_map' unit IIN3  'soil' unit IIN4  'ic' unit IIN5  'atmbc' unit IIN6  'sfbc' unit IIN7  'nansfdirbc' unit IIN8  'nansfneubc' unit IIN9  ……….………...……….………...……….……….….….….….  ……….………...……….………...……….……….….….….….  'output/enpsia' unit IOUT54  'output/ensubvol' unit IOUT55  'output/enpsiz' unit IOUT56  'output/wtdepth' unit IOUT57  'output/peatdef' unit IOUTPT  'output/term' unit ITERM  (if ITERM=6 output is to terminal and this file is ignored) |

Figure 2. Part of the *cathy.fnames* file

Some of these inputs are automatically generated during the pre-processing (Table 2), while others are new input files that should be updated with appropriate parameters for the specific case study (Table 3). All these files must be located, even empty, in the directory where you want to run the simulation, so the program can find them.

Table 3. Input files for the processor CATHY

|  |  |  |
| --- | --- | --- |
| IIN1 | parm | Control parameters for the simulations FLOW3D and SURF\_ROUTE |
| IIN2 | grid | Grid info when they are different from the input DEM |
| IIN3 | root\_map | Raster map of the root zone depth |
| IIN4 | soil | Soil parameter |
| IIN5 | ic | Initial conditions |
| IIN6 | atmbc | Atmospheric BC's (rainfall/evaporation rates) |
| IIN7 | sfbc | Seepage face BC's |
| IIN8 | nansfdirbc | Non‐atmospheric, non‐seepage face Dirichlet BC's (see subroutines BCONE and BCNXT for unit IIN8 input) |
| IIN9 | nansfneubc | Non‐atmospheric, non‐seepage face Neumann BC's(see subroutines BCONE and BCNXT for unit IIN9 input) |
| IIN11 | dem\_parameters | DEM parameters and other parameters (if GRID=TRUE they can be found in the grid file) |
| IIN16 | retctab | Soil characteristics in tabular data |
| IIN17 | posizione\_serb | Position of reservoirs and buffer cells (posizione\_serb.dat) |
| IIN18 | livelli\_iniz\_s | Initial levels in reservoirs (livelli\_iniz\_serb.dat) |
| IIN19 |  | Depitting parameter epsilon (depit.dat) |
| IIN22 | effraininp | Effective rainfall input file (this file is read only in the case of surface simulation) |
| IIN40 | enkf | EnKF input |
| IIN50 | nudging | Nudging input (see subroutines NUDONE and NUDNXT for unit IIN50 input) |
| IIN51 | mesh | Mesh input file |
| IIN60 | base\_map | Raster map of the catchment impermeable basement. |

### Parameters of the simulation (*parm* - IIN1)

The parameters of the simulation are contained in *parm* (Figure 3).

|  |
| --- |
| 2 0 IPRT1 DAFLAG  2 3.1860e-06 ISIMGR PONDH\_MIN  0 0.01 KSLOPE TOLKSL  -3.0 -1.0 -3.0 -1.0 PKRL PKRR PSEL PSER  -3.0 -2.5 -1.5 -1.0 PDSE1L PDSE1R PDSE2L PDSE2R  0 0 0 ISFONE ISFCVG DUPUIT  1.0 1 1 TETAF LUMP IOPT  0 0.01 NLRELX OMEGA  0 1.0e-4 1.0e30 1.0e+30 L2NORM TOLUNS TOLSWI ERNLMX  12 4 8 ITUNS ITUNS1 ITUNS2  0 500 1.0e-06 ISOLV ITMXCG TOLCG  1.0e+02 1.0e-02 4.32e+04 3.155328e+08 DELTAT DTMIN DTMAX TMAX (10 y)  0.0 1.05 0.0 0.80 DTMAGA DTMAGM DTREDS DTREDM  1 36  2592000.0 5184000.0 7776000.0 10368000.0  12960000.0 15552000.0 18144000.0 20736000.0  23328000.0 25920000.0 28512000.0 31104000.0  33696000.0 36288000.0 38880000.0 41472000.0  44064000.0 46656000.0 49248000.0 51840000.0  54432000.0 57024000.0 59616000.0 62208000.0  64800000.0 67392000.0 69984000.0 72576000.0  75168000.0 77760000.0 80352000.0 82944000.0  85536000.0 88128000.0 90720000.0 93312000.0 IPRT NPRT TIMPRT(I),I=1,NPRT    30  839 881 882  935 978 979  686 687 727  527 566 567  372 373 410  190 224 225  65 66 93  612 651 652  539 540 579  1268 1269 1283 NUMVP NODVP(I),I=1,NUMVP    1 NR (# of output nodes)  882 935 727 567 410 225 93 612 540 1268 CONTR(I),I=1,NR    0 NUM\_QOUT, IDQOUT(I) I=1,NUM\_QOUT  Units: meters, seconds |

Figure 3. *parm* input file

The first two parameters in the first line are IPRT1 and DAFLAG. Generally IPRT1=3 during the first running of cathy.exe is set equal to 3, then it can set equal to 2. DAFLAG must be set in function of the type of the data assimilation scheme used; if data assimilation is not used, this parameter can be set equal to 0.

|  |  |
| --- | --- |
| IPRT1: | Flag for output of input and coordinate data in subroutines DATIN and GEN3D  = ‐1 reads in coordinates from file IIN51 (mesh input file) just after grid generation (in subroutine grdsys)  = 0 prints parameters only (default)  = 1 prints parameters + b.c. + geom. char.  = 2 prints parameters + b.c. + geom. char. + grid info  = 3 prints parameters + b.c. + geom. char. + grid info, X, Y, Z coordinate values in subroutine GEN3D, and then terminates program execution |
| DAFLAG: | Flag for the choice of the data assimilation scheme:  = 0 nudging (if NUDN=0, no data assimilation)  = 1 EnKF with Evensen's algorithm (Ocean Dynamics, 2004)  = 2 EnKF with parameters update  = 3 Particle filtering (SIR algorithm)  = 4 Particle filtering (SIR algorithm) with parameters update |

In this file it can be specify the type of simulation and the type of surface grid through the parameter ISIMGR. Although there are several options, it is better to set this parameter equal to 2 (coupled FLOW3D and SURF\_ROUTE); otherwise (e.g. ISIMGR=1, the excess water that could be runoff is eliminated, thus pressure =0 (atmospheric pressure) and the minimum ponding head is not considered).

|  |  |
| --- | --- |
| ISIMGR: | Flag for type of simulation and type of surface grid  =0 subsurface flow only (FLOW3D) with general triangular grid input  =1 subsurface flow only (FLOW3D) with DEM input and triangular grid generated from this DEM  =2 coupled subsurface flow (FLOW3D) and surface routing (SURF\_ROUTE) with DEM input and triangular grid generated from this DEM  =3 for surface flow (SURF\_ROUTE) simulation with DEM input. The local contribution rate to surface runoff is directly read from effraininp input file by the subroutines EFFONE.f and EFFNXT.f. |
| PONDH\_MIN: | Minimum ponding head: if PNEW > PONDH\_MIN, then at that node there is ponding; otherwise there is no ponding. |

Other parameters are:

|  |  |
| --- | --- |
| KSLOPE: | = 0 for analytical differentiation of moisture curves  = 1 for "chord slope" and analytical differentiation  = 2 for "chord slope" and centered difference formulas  = 3 for localized "chord slope" and analytical differentiation  = 4 for localized "tangent slope" differentiation  (the "chord slope" formula is the tangent approximation suggested by Huyakorn et al (WRR 20(8) 1984), wherein derivatives are approximated using pressure heads at the current and previous nonlinear iterations; "tangent slope" differentiation is a different tangent approximation wherein derivatives are approximated using pressure heads at the endpoints of a given range (eg: endpoints PKRL, PKRR for the derivative of relative hydraulic conductivity). For KSLOPE=1, 2 the chord slope formula is used at every iteration and at all nodes (with some exceptions as dictated by TOLKSL). For KSLOPE=3 or 4 the chord or tangent slope formulas are used only at those nodes whose pressure heads fall within given ranges (see PKRL, PKRR, etc), hence 'localized'; for nodes whose pressure heads fall outside these ranges, analytical differentiation is used.) |
| TOLKSL: | Tolerance for chord slope formula. Whenever the chord slope formula is to be applied (for KSLOPE=1 or 2 at every iteration and at all nodes; for KSLOPE=3 at those nodes whose pressure heads fall within given ranges), it is applied only if the absolute pressure head difference (between the current and previous nonlinear iterations) is larger than TOLKSL. If the difference is smaller than TOLKSL, then differentiation is done either analytically (KSLOPE=1, 3) or with a centered difference formula (KSLOPE=2). |
| PKRL, PKRR: | Left and right endpoints of the pressure head range within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the derivative of relative hydraulic conductivity |
| PSEL, PSER: | Left and right endpoints of the pressure head range within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the derivative of effective saturation (moisture content for the case of extended van Genuchten curves, IVGHU=1) |
| PDSE1L, PDSE1R, PDSE2L, PDSE2R: | Left and right endpoints of the two pressure head ranges within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the second derivative of effective saturation (moisture content for the case of extended van Genuchten curves, IVGHU=1). (Two ranges are specified since in general d(Se)/dP is non‐monotonic.) |

In addition some parameters must be set only if the problem includes the presence of a seepage face (i.e. a boundary between the saturated flow field and the atmosphere along which ground water discharges: h = z; pressure = atmospheric at the ground surface); otherwise the parameters will not be considered.

|  |  |
| --- | --- |
| ISFONE: | =0 seepage face exit point updating performed by checking all nodes on a seepage face  =1 seepage face exit point updating performed by checking only the one node above and one node below the current exit point position |
| ISFCVG: | =0 convergence of seepage face exit points is not a condition for convergence of the nonlinear iterative procedure  =1 convergence of seepage face exit points is a condition for convergence of the nonlinear iterative procedure |
| DUPUIT: | =0 all the nodes below the seepage face exit point are at atmospheric pressure  =1 all the nodes below the seepage face exit point are at hydrostatic pressure. Caution: to use only if ISFONE=1! |

Other parameters are:

|  |  |
| --- | --- |
| TETAF: | Weighting parameter for FLOW3D time stepping scheme (1.0 backward Euler; 0.5 Crank-Nicolson; TETAF is set to 1.0 for steady state problem). |
| LUMP: | =0 for distributed mass matrix; otherwise matrix is lumped |
| IOPT: | Iteration scheme to implement:  =1 for Picard iteration scheme  =2 for Newton iteration scheme |

|  |  |
| --- | --- |
| NLRELX: | Flag for nonlinear relaxation  =0 no relaxation  =1 relaxation with constant relaxation parameter OMEGA  =2 relaxation with iteration-dependent relaxation parameter OMEGA, calculated using Huyakorn et al's adaptation (WRR 1986 22(13), pg 1795) of Cooley's empirical scheme (WRR 1983 19(5), pg 1274) |
| OMEGA: | Non-linear relaxation parameter: OMEGA > 1, over-relaxation; OMEGA < 1, under-relaxation. Input value of OMEGA is used only for the case NLRELX=1 (constant relaxation parameter). Input value of OMEGA is ignored otherwise: for NLRELX=0 relaxation is not applied; for NLRELX=2 OMEGA is calculated at each nonlinear FLOW3D iteration |
| L2NORM: | =0 to use the infinity norm in the test for convergence of the nonlinear FLOW3D and coupled FLOW3D/SURF\_ROUTE iterations; otherwise the L2 norm is used. |
| TOLUNS: | Tolerance for convergence of nonlinear FLOW3D iterations |
| TOLSWI: | Tolerance for boundary condition switching check in FLOW3D iterations (switching check is only performed when PINF or PL2 are smaller than TOLSWI; so for e.g. if TOLSWI = TOLUNS, switching check is only performed after convergence and not after each iteration) |
| ERNLMX: | Maximum allowable convergence or residual error in the nonlinear FLOW3D solution. If the convergence or residual errors become larger than ERNLMX, ERRGMX is set to TRUE and the code back‐steps. This avoids occurrences of overflow or underflow when nonlinear iterations diverge. |

In *parm* it must be specified the parameters related to the adaptive time-stepping approach implemented in the model. In particular the iteration bounds (ITUNS, ITUNS1, and ITUNS2), the initial DELTAT, the minimum DTMIN and the maximum DTMAX time step allowed for the computations, and the total time of the simulation TMAX.

|  |  |
| --- | --- |
| ITUNS: | Maximum nonlinear FLOW3D iterations per time step |
| ITUNS1: | If ITER < ITUNS1, time step size is increased |
| ITUNS2: | If ITUNS1 <= ITER < ITUNS2, time step size is not altered if ITUNS2 <= ITER < ITUNS, time step size is decreased if ITER = ITUNS (i.e., convergence not achieved in ITUNS iterations), we back‐step unless time step size cannot be reduced any further (DELTAT = DTMIN). Back‐stepping is also triggered if the linear solver failed (LSFAIL = TRUE) or if the convergence or residual errors become larger than ERNLMX (ERRGMX = TRUE). |
| DELTAT: | Initial and current FLOW3D time step size (DELTAT >=1.0e+15 on input indicates steady state problem) |
| DTMIN: | Minimum FLOW3D time step size allowed |
| DTMAX: | Maximum FLOW3D time step size allowed |
| TMAX: | Time at end of simulation (TMAX is set to 0.0 for steady state problem) |
| DTMAGA: | Magnification factor for FLOW3D time step size (additive) |
| DTMAGM: | Magnification factor for FLOW3D time step size (multiplicative) |
| DTREDS: | Reduction factor for FLOW3D time step size (subtractive) |
| DTREDM: | Reduction factor for FLOW3D time step size (multiplicative) |

Other parameters are:

|  |  |
| --- | --- |
| ISOLV: | Flag for non-symmetric linear solver  = -5 BiCGSTAB (preconditioned with D^-1)  = -4 BiCGSTAB (not preconditioned)  = -3 TFQMR (preconditioned with D^-1)  = -2 TFQMR (not preconditioned)  = -1 TFQMR (preconditioned with K^-1)  = 0 BiCGSTAB (preconditioned with K^-1)  = 1 GRAMRB (minimum residual)  = 2 GCRK(5) (ORTHOMIN)  = 3 IBM's NONSYM (direct solver) |
| ITMXCG: | Maximum number of iterations for conjugate gradient linear system solvers |
| TOLCG: | Tolerance for convergence of conjugate gradient linear system solvers |

Finally, you can specify:

* the number (NPRT) and the related time values (TIMPRT) for detailed nodal output (red box in Figure 3)
* the number of surface nodes for vertical profile output (NUMVP) and related node ID’s (NODVP) (green box in Figure 3)
* the number of nodes selected for partial output (NR) and corresponding node ID’s (CONTR) (blue in Figure 3)
* the number of selected cells for hydrograph output (NUM\_QOUT) and corresponding cell ID’s (IDQ\_OUT).

|  |  |  |
| --- | --- | --- |
| IPRT: | Flag for detailed output at all nodes and velocity and water saturation output at all elements (velocity and water saturation output in the case IPRT=4 can be used as input to TRAN3D and DUAL3D codes)  =0 don't print nodal pressure, velocity, water saturation, or relative conductivity values  =1 print only nodal pressure head values  =2 print nodal pressure head and velocity values  =3 print nodal pressure, velocity, and relative conductivity values  =4 print nodal pressure, velocity, relative conductivity, and overall storage coefficient values, and print element velocity and nodal water saturation values | |
| NPRT: | Number of time values for detailed nodal output and element velocity output (see description of IPRT, TIMPRT) | |
| TIMPR (NPRT): | Time values for detailed output. Detailed output is produced at initial conditions (TIME=0), at time values indicated in TIMPRT, and at the end of the simulation (TIME=TMAX). Detailed output consists of: values of pressure head, velocity, water saturation, and relative conductivity (depending on setting of IPRT) at all nodes; velocity, and water saturation (depending on setting of IPRT) at all elements; vertical profiles of pressure head, water saturation, and relative conductivity for the NODVP surface nodes; pressure head, water saturation, and SATSUR values at the surface nodes. | |
| NUMVP: | | Number of surface nodes for vertical profile output |
| NODVP (NUMVP): | | Node number's for surface nodes selected for vertical profile output |
| NR: | | Number of nodes selected for partial output |
| CONTR (NR): | | Node number's for partial output |
| NUM\_QOUT: | | Number of selected cells for hydrograph output |
| ID\_QOUT: | | I\_BASIN index of selected cells for hydrograph output |

In selecting nodes and cells IDs it is important to remind the different numbering systems adopted between the subsurface and surface module and that these systems are node-based for the subsurface and cell-based for the surface:

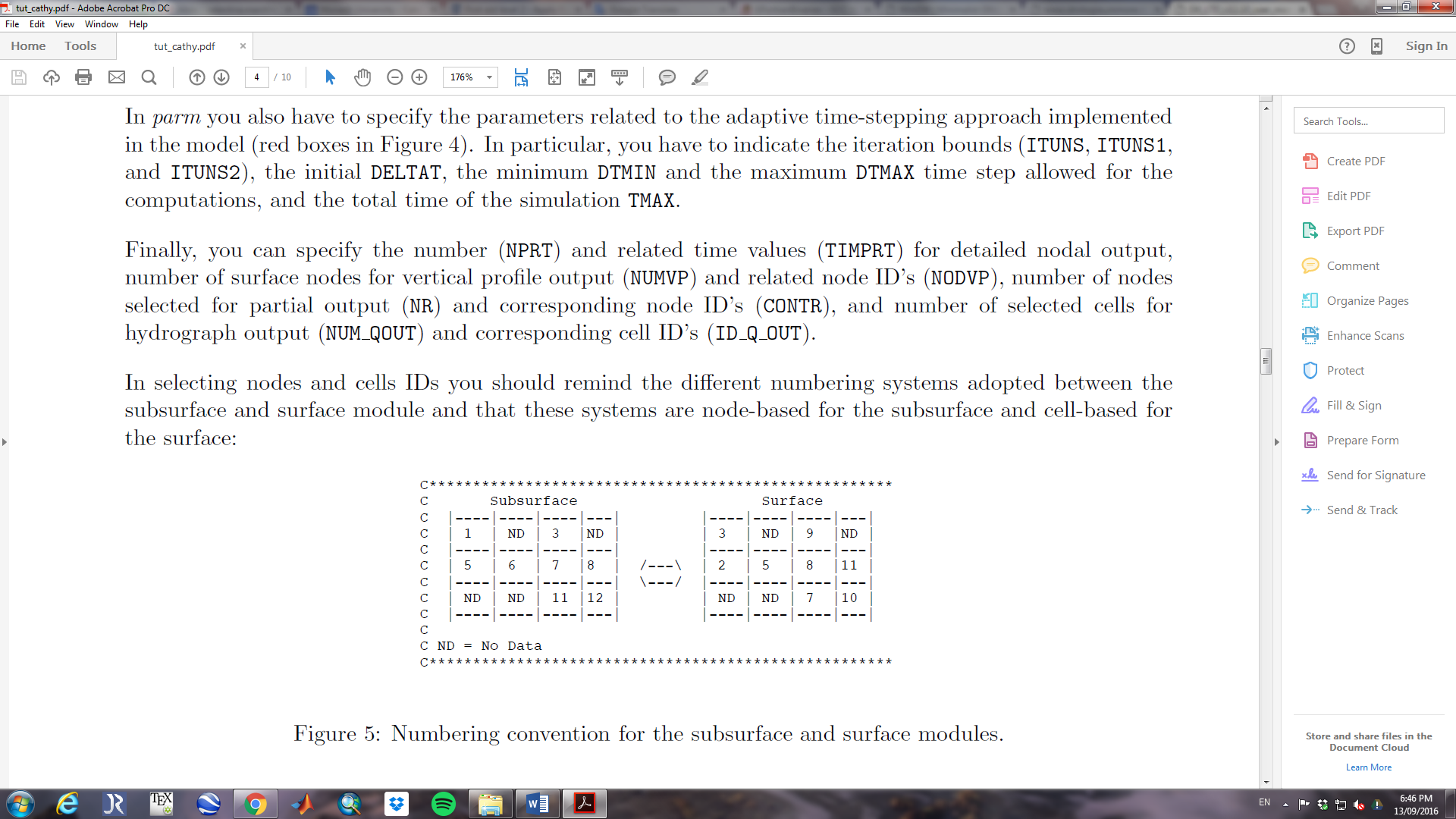


Figure 4. Numbering convention for the subsurface and surface modules

### Model grid (*dem\_parameters* - IIN11)

There are two options to construct the resolution grid for the CATHY processor whether you want to use a regular mesh or not. In the case of a coupled simulation you have to use a DEM (regular mesh) and the associated input file to be considered is *dem\_parameters* (Figure 5).

In this file the resolution of the DEM must be specified through the parameters DELTA\_X and DELTA\_Y (this variables must appear without any comment between them). In addition it is necessary to specify the numbers of material types in the porous media (NZONE), the number of vertical layers (NSTR), and their relative thickness (ZRATIO).

|  |
| --- |
| 10. 10. delta\_x delta\_y  1.0e+0 factor  1 dostep  1 20 20 nzone nstr n1  0 1 5 ivert isp base  0.05 0.05 0.05 0.05 0.05  0.05 0.05 0.05 0.05 0.05  0.05 0.05 0.05 0.05 0.05  0.05 0.05 0.05 0.05 0.05 zratio(i),i=1,nstr |

Figure 5**.** *dem\_parameters* input file

In this file several parameters have to be specified, specifically:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DELTA\_X, DELTA\_Y: | | Cell dimensions for the resolution of the DEM | | |
| FACTOR: | | Multiplicative factor for DEM values (e.g. to change the units of the elevation) | | |
| DOSTEP: | | Step adopted in coarsening the mesh | | |
| NZONE: | | The number of material types in the porous medium | | |
| NSTR: | | The number of vertical layers | | |
| N1: | | The maximum number of element connections to a node | | |
| ZRATIO (NSTR): | | The thickness of vertical layers or the fraction of total grid height that each layer is to occupy (ZRATIO (1) is for the surface‐most layer. ZRATIO values must sum to 1.) |

It is also necessary to select the options for constructing the mesh (IVERT and ISP) and the thickness of the subsurface mesh (BASE).

|  |  |
| --- | --- |
| IVERT: | =0 each layer will be parallel to the surface, including the base of the 3‐d grid. ZRATIO is applied to each vertical cross section.  =1 base of the 3‐d grid will be flat, and ZRATIO is applied to each vertical cross section  =2 base of the 3‐d grid will be flat, as will the NSTR‐1 horizontal cross sections above it. ZRATIO is applied only to the vertical cross section having the lowest elevation.  =3 for each cell of the dem a single depth value is read in file input IIN60 (basement). ZRATIO is applied to each vertical cross section.  =4 the first NSTR‐1 layers from the surface will be parallel to the surface and the base of the 3‐d grid will be flat. ZRATIO is applied only to the vertical cross section having the lowest elevation. |
| ISP: | =0 for flat surface layer (only one Z value is read in, and is replicated to all surface nodes);  otherwise surface layer is not flat (Z values read in for each surface node);  (for ISP=0, IVERT=0, 1, and 2 yield the same 3‐d mesh, given the same values of BASE and ZRATIO). |
| BASE: | Value which defines the thickness or base of the 3‐d mesh.  For IVERT=0, BASE is subtracted from each surface elevation value, so that each vertical cross section will be of thickness BASE, and the base of the 3‐d mesh will be parallel to the surface. For IVERT=1 or 2, BASE is subtracted from the lowest surface elevation value, say ZMIN, so that each vertical cross section will be of thickness (Z ‐ ZMIN) + BASE, where Z is the surface elevation for that cross section. The base of the 3‐d mesh will thus be flat. |
|  |  |

### Atmospheric forcing term (*atmbc* - IIN6)

In this file the following parameters must be set up:

|  |  |
| --- | --- |
| HSPATM: | = 0 for spatially variable atmospheric boundary condition inputs;  = blank or =9999 if unit IIN6 input is to be ignored;  = otherwise atmospheric BC's are homogeneous in space. |
| IETO: | = 0 for linear interpolation of the atmospheric boundary condition inputs between different ATMTIM;  = otherwise the inputs are assigned as a piecewise constant function (ietograph). |
| TIME: | Time at current time level (seconds) |
| VALUE: | (meters per seconds) |

The example in Figure 6 shows a case with the atmospheric forcing term homogeneous in space (HSPATM=1) and where the inputs are assigned as a piecewise constant function (IETO=1). The values are those of a 200-min rainfall event at a uniform intensity of 3.3·10-4 m/min, followed by 100 min of drainage. To be consistent with the units used in the other input files, the value must be converted in meters per seconds and the time in seconds.

|  |
| --- |
| 1 1 HSPATM,IETO  0.0000000e+00 TIME  5.5e-06 VALUE  12.000000e+03 TIME  0.00 VALUE  18.000000e+03 TIME  0.00 VALUE |

Figure 6. ***atmbc*** input file

In case of evaporation and precipitation it is necessary to provide the model with the net value among the two on the selected time step.

### Initial conditions (*ic* - IIN5)

The initial conditions file contains the pressure heads distribution for the study area (INDP). For example, to simulate a uniform water table depth or 0.5 m or 1.0 m from the ground surface, INDP=3 and WTHEIGHT=4.5 are selected in the file *ic* in Figure 7.

|  |
| --- |
| 3 0 INDP IPOND  4.5 WTHEIGHT |

Figure 7. *ic* input file

|  |  |
| --- | --- |
| INDP: | Flag for pressure head initial conditions (all nodes)  =0 for input of uniform initial conditions (one value read in)  =1 for input of non-uniform IC's (one value read in for each node)  =2 for calculation of fully saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVHE). In the case of IPOND>0, the fully saturated hydrostatic IC is calculated (in subroutine ICVHEPOND) starting from the ponding head values at the surface nodes, rather than surface pressure heads of 0.  =3 for calculation of partially saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVHWT) with the water table height (relative to the base of the 3‐d grid) given by parameter WTHEIGHT  =4 for calculation of partially saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVDWT) with the water table depth (relative to the surface of the 3‐d grid) given by parameter WTHEIGHT |
| WTHEIGHT: | For the case INDP=3, specifies the initial water table height relative to the base of the 3‐d grid |
| IPOND: | Flag for ponding head initial conditions (surface nodes)  =0 no input of ponding head initial conditions; otherwise (IPOND = 1 or 2) ponding head initial conditions are read into PONDNOD, and, where PONDNOD > 0, these values are used to update the surface node values in PTIMEP read in according to the previous INDP flag  =1 uniform ponding head initial conditions (one value read in)  =2 non-uniform ponding head initial conditions (one value read in for each node) |

### Boundary conditions (*nansfdirbc* -IIN8, *nansfneubc* - IIN9, *sfbc* - IIN7)

The boundary conditions are defined in the *nansfdirbc* (Dirichlet), *nansfneubc* (Neumann), and *sfbc* (seepage face) files. The structure of these files is the same as you can see from the Figure 8. To simulate the no-flow boundaries conditions for the bottom and vertical sides of the domain it is necessary to set NDIR and NDIRC equal to zero (Figure 8).

|  |
| --- |
| 0.0000000000000000E+000 TIME  0 0 NDIR, NDIRC  4.3200E+05 TIME  0 0 NDIR, NDIRC |
| (a) |
| 0.00000 TIME  0 0 ZERO NQ (Important, the first value must be 0!)  4.32e+05 TIME  0 0 ZERO NQ |
| (b) |
| 0.0000E+00  0  4.320E+05  0 |
| (c) |

Figure 8. (a) *nansfdirbc*, (b) *nansfneubc*, and (c) *sfbc* input files

|  |  |
| --- | --- |
| NDIR | Number of non-atmospheric, non‐seepage face Dirichlet nodes in 2-d mesh. The BC's assigned to these surface nodes are replicated vertically (compare NDIRC) |
| NDIRC | Number of 'fixed' non-atmospheric, non-seepage face Dirichlet nodes in 3‐d mesh ('fixed' in the sense that these BC's are not replicated to other nodes ‐ compare NDIR) |
| NQ(3) | Number of non-atmospheric, non‐seepage face Neumann nodes in 3‐d mesh |

To simulate different boundary conditions, it is necessary to indicate the number of selected nodes through NDIR or NDIRC, then to specify the node ID’s that you want to consider and eventually the value of pressure head or flux that you want to assign.

### Soil parameters (*soil* - IIN4)

The porous media properties are defined in the *soil* file. The first thing that must be decides is the type of relationship to describe the hydraulic characteristics of the unsaturated soil (i.e. retention curves). This can be done through the choice of the parameter IVGHU amongst the several options.

Then it is necessary to assign the values of the parameters VGN, VGRMC, and VGPSAT which describe the shape of the curves. Finally, you have to define the saturated hydraulic conductivity along the x, y, and z direction, the specific storage, and the porosity (blue box in Figure 11). This has to be done for each material type delimited in zone and repeated for the NSTR vertical layers indicated in dem parameters. In addition, in the soil file you have to set the value of min, the parameter used in the model to simulate the advent of stage two evaporation, and you have also to switch on the IPEAT flag if you want simulate the peat soil deformation.

|  |
| --- |
| -6.40e+20 PMIN  0 1.0 IPEAT,SCF  0.4000 0.2250 DELTA0 LAMBDA (from the sloping plane example)  0.4000 0.16 0.0558 4.0252 1.0 SMCANA,SMCREF,SMCWLT,PZ,OMGC  0 IVGHU  1.51689 0.031 -0.53945 VGN,VGRMC,VGPSAT (SANDY CLAY LOAM)  0.015 2.0 3.0 -10.0 0.01 HUALFA,HUBETA,HUGAMA,HUPSIA,HUSWR  2.0 HUN  2.0 3.5 HUA,HUB  3.3 0.0 -0.12 BCBETA,BCRMC,BCPSAT (CLAY-LOAM)  2.693E-05 2.693E-05 2.693E-05 1.000E-03 0.40  2.012E-05 2.012E-05 2.012E-05 1.000E-03 0.40  1.721E-05 1.721E-05 1.721E-05 1.000E-03 0.40  1.502E-05 1.502E-05 1.502E-05 1.000E-03 0.40  1.352E-05 1.352E-05 1.352E-05 1.000E-03 0.40  1.254E-05 1.254E-05 1.254E-05 1.000E-03 0.40  1.158E-05 1.158E-05 1.158E-05 1.000E-03 0.40  1.073E-05 1.073E-05 1.073E-05 1.000E-03 0.40  9.857E-06 9.857E-06 9.857E-06 1.000E-03 0.40  8.924E-06 8.924E-06 8.924E-06 1.000E-03 0.40  8.104E-06 8.104E-06 8.104E-06 1.000E-03 0.40  7.423E-06 7.423E-06 7.423E-06 1.000E-03 0.40  6.863E-06 6.863E-06 6.863E-06 1.000E-03 0.40  6.398E-06 6.398E-06 6.398E-06 1.000E-03 0.40  5.996E-06 5.996E-06 5.996E-06 1.000E-03 0.40  5.640E-06 5.640E-06 5.640E-06 1.000E-03 0.40  3 ZONE\_VERT  6 15 16 STR\_ZON(I),I=1,ZONE\_VERT  PERMX PERMY PERMZ ELSTOR POROS  Units: [m] [s] |

Figure 9. *soil* input file

|  |  |  |  |
| --- | --- | --- | --- |
| PMIN units:  [m sec] | 'air dry' pressure head value (for switching control of atmospheric boundary conditions during evaporation) | | |
| IPEAT: | Flag for peat soil deformation  =0 constant porosity (in unsaturated soil)  =1 consider porosity variations with water saturation | | |
| SCF: | soil cover fraction (fraction of soil covered in vegetation) | | |
|  | |  | |
| SMCREF,SMCWLT, SMCANA: | | | Parameters for root water uptake module. SMCREF is field capacity, SMCWLT is wilting point, SMCANA is anaerobiosis point.  SMCANA 🡪 Read only if IVGHU=0 |
| IVGHU: | | = -1 table look up for moisture curves  = 0 for van Genuchten moisture curves  = 1 for extended van Genuchten moisture curves  = 2 for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) with Kr=Se\*\*n conductivity relationship  = 3 for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) with conductivity relationship from Table 3 of 1984 paper (log\_10 Kr(Se) curve)  = 4 for Brooks‐Corey moisture curves | |
| CBETA0, THETA0, CANG | | Parameters for Camporese adaptation of Pyatt and John relation for peat soil deformation | |
| VGN, (VGM), VGRMC, VGPSAT | | Parameters for van Genuchten and extended van Genuchten moisture curves (other 'VG' parameters ‐ specific storage, porosity, and VGPNOT ‐ are assigned nodally). VGM is derived from VGN. VGRMC is residual moisture content. For IVGHU=0, VGPNOT is (porosity ‐ VGRMC)/porosity, or (1 ‐ residual water saturation). For IVGHU=1, VGPNOT is a continuity parameter, derived by imposing a continuity requirement on the derivative of moisture content with respect to pressure head. | |
| HUN, HUA, HUB, HUALFA, HUBETA, HUGAMA, HUPSIA, HUSWR | | Parameters for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) (other 'HU' parameters ‐ specific storage and porosity ‐ are assigned nodally). HUN is only used for IVGHU=2; HUA and HUB are only used HUGAMA, for IVGHU=3. HUSWR is residual water saturation, which is equivalent to residual moisture content/porosity. | |
| BCBETA, BCRMC, BCPSAT | | Parameters for Brooks-Corey moisture curves (other 'BC' parameters ‐ specific storage and porosity ‐ are assigned nodally). BCRMC is residual moisture content | |
| ZONE\_VERT | | Number of vertical zones for the soil parameters (used if DAFLAG > 1) | |
| STR\_ZON(ZONE\_VERT) | | For each vertical zone contain the last layer (str) of the zone. Examples: if ZONE\_VERT=1, then STR\_ZON (1)=NSTR; if ZONE\_VERT=NSTR, then STR\_ZON=[2 3 4 ... NSTR]. | |

### Other input files

In addition to the previously described input files, the processor CATHY needs of other input files that, even empty, must be located in the directory where you want to run the simulation, so the program can find them.

* **base\_map (IIN60):** Raster map of the catchment impermeable basement (bedrock). Each value represents the thickness of the corresponding cell. It is used to define the catchment impermeable basement independently from the DEM; this happens when specific data are known.
* **effraininp (IIN22):** effective rainfall input file (it is used by the subroutines effone.f and effnxt.f for the case of surface routing only).
* **enkf (IIN40), nudging (IIN50):** Both of these input files contain data and are involved in the modeling when data assimilation (DA) schemes are used; otherwise they are empty files. The two DA schemes implemented in CATHY are dynamical relaxation (Newtonian nudging) and the ensamble Kalman filter (EnKF).
* **grid (IIN2):** this file contains information on the grid info when they are different from the ones from the input DEM used in the preprocessing.
* **root\_map (IIN3):** contains the raster map of the root zone depth as shown in the figure below

|  |
| --- |
| north: 100.00000000  south: 0.00000000  east: 110.00000000  west: 0.00000000  rows: 10  cols: 11  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 |

* **mesh (IIN51):** Mesh input file
* **retctab (IIN16):** is used only if you want to use capillarity curves that are variable in space in the form of tabular data (soil characteristics in tabular data).
* **posizione\_serb (IIN17), livelli\_iniz\_s (IIN18):** the IIN17 input file represents the position of reservoirs and buffer cells and the IIN18 input file represents the initial levels in reservoirs. Both of them are employed in the “lake boundary-following” procedure ……..

## Running the processor CATHY

Once all the input files are updated for the specific case study, it is necessary to move to the directory where the source FORTRAN files are contained (*cathy\_main.f*). Then it is necessary to edit the *openio.f* file by replacing the forward slash (/) with back slash (\) for Windows and the *CATHY.H* file to properly define the dimensions of the specific problem (dimension of the arrays).

It is very important to do not exceed too much in the values actually needed for the specific problem for an accurate dimensioning of the problem. This allows you to allocate during the simulation the memory space that you actually need.

Figure 10 shows part of the file CATHY.H; the most important elements to set are highlighted in red.

|  |
| --- |
| INTEGER ROWMAX,COLMAX  INTEGER MAXCEL,MAXRES  INTEGER NODMAX,NTRMAX  INTEGER NP2MAX,MAXSTR  INTEGER NMAX,NTEMAX  INTEGER NPMAX,NQMAX,NSFMAX,NNSFMX  INTEGER MAXDIR  INTEGER MAXNUDN,MAXNUDT,MAXNUDC,MAXNENS  INTEGER MAXZON,MAXTRM,MAXIT  INTEGER NRMAX,MAXPRT,MAXVP  INTEGER N1MAX,NTPMAX  INTEGER MAXBOT,INTBOT  INTEGER NTAB,COARSE  INTEGER MAXQOUT,MAXENNOUT  PARAMETER (ROWMAX=53**,**COLMAX=49**,**COARSE=1)  PARAMETER (MAXCEL=ROWMAX\*COLMAX,MAXRES=1)  PARAMETER (NODMAX=(ROWMAX/COARSE+1)\*(COLMAX/COARSE+1))  PARAMETER (NTRMAX=2\*MAXCEL/(COARSE\*COARSE))  PARAMETER (NP2MAX=1,MAXSTR=18)  PARAMETER (NMAX=NODMAX\*(MAXSTR + 1),NTEMAX=3\*NTRMAX\*MAXSTR)  PARAMETER (NPMAX=180,NQMAX=1,NSFMAX=10,NNSFMX=18)  PARAMETER (MAXDIR=NODMAX + NPMAX + NSFMAX\*NNSFMX)  PARAMETER (MAXNUDN=6,MAXNUDT=480,MAXNUDC=1,MAXNENS=100)  PARAMETER (MAXZON=1,MAXTRM=513000,MAXIT=100)  PARAMETER (NRMAX=10,MAXPRT=2800,MAXVP=65)  PARAMETER (N1MAX=30,NTPMAX=N1MAX\*NMAX)  PARAMETER (MAXBOT=1,INTBOT=1,MAXQOUT=1)  PARAMETER (NTAB=100)  PARAMETER (MAXENNOUT=52560) |

Figure 10. Snapshot of the file *CATHY.H*

The dimensioning parameters are:

|  |  |
| --- | --- |
| ROWMAX | maximum NROW, with NROW = number of rows in the DEM |
| COLMAX | maximum NCOL, with NCOL = number of columns in the DEM |
| COARSE | coarsening factor for grid generation from DEM |
|  |  |
| MAXCEL | ROWMAX\*COLMAX (maximum NCELL)  NCELL = number of cells in the DEM of the catchment, including "lake" cells |
| MAXRES | maximum NUMRES, with NUMRES = number of 'reservoirs' defined in the DEM |
| NODMAX | maximum NNOD  NNOD = number of nodes in 2-d mesh  = number of surface nodes in 3-d mesh |
| NTRMAX | maximum NTRI  NTRI = 2\*NCELL when SURF\_ROUTE is active, otherwise must be assigned explicitly = number of triangles in 2-d mesh |
| NP2MAX | maximum NDIR  NDIR = number of non-atmospheric, non-seepage face Dirichlet nodes in 2-d mesh |
| MAXSTR | maximum NSTR, with NSTR = number of vertical layers |
| NMAX | NODMAX\*(MAXSTR + 1) (maximum N)  N = NNOD\*(NSTR + 1)  = number of nodes in 3-d mesh |

|  |  |  |
| --- | --- | --- |
| NTEMAX | 3\*NTRMAX\*MAXSTR (maximum NT)  NT = 3\*NTRI\*NSTR  = number of tetrahedra in 3-d mesh | |
| NPMAX | maximum NP, NP = NDIR\*(NSTR + 1) + NDIRC  NP = total number of non-atmospheric, non-seepage face Dirichlet nodes in 3-d mesh  NDIRC = number of 'fixed' non-atmospheric, non-seepage face Dirichlet nodes in 3-d mesh | |
| NQMAX | maximum NQ  NQ = number of non-atmospheric, non-seepage face Neumann nodes in 3-d mesh | |
| NSFMAX | maximum NSF, with NSF = number of seepage faces | |
| NNSFMX | maximum number of nodes on a seepage face + 1 | |
| MAXDIR | NODMAX + NPMAX + NSFMAX\*NNSFMX (maximum NUMDIR)  NUMDIR= total number of Dirichlet nodes in 3-d mesh | |
| MAXNUDN | maximum NUDN  NUDN = number of observation points for nudging or EnKF (NUDN=0 for no nudging) | |
| MAXNUDT | maximum NUDT, with NUDT = number of observation times for nudging or EnKF | |
| MAXNUDC | maximum NUDC  NUDC = number of concurrent observation datasets for nudging at any given time | |
| MAXNENS | maximum NENS, with NENS = number of realizations for the EnKF data assimilation | |
| MAXZON | maximum NZONE, with NZONE = number of material types in the porous medium | |
| MAXTRM | maximum NTERM (value of MAXTRM should be at least 10  NTERM = number of nonzero elements in system matrices  (it is good practice to consider MAXTRM = rowmax\*colmax\*maxstr\*10) | |
| MAXIT | maximum ITUNS, with ITUNS = maximum nonlinear FLOW3D iterations per time | |
| NRMAX | maximum NR (ref. parm file), with NR = number of nodes selected for partial output | |
| MAXPRT | maximum NPRT (ref. parm file), with NPRT = number of time values for detailed output | |
| MAXVP | maximum NUMVP (ref. parm input file)  NUMVP = number of surface nodes for vertical profile output | |
| N1MAX | maximum N1 (it is good to have N1 ≤ 20)  N1 = maximum number of element connections to a node | |
| NTPMAX | N1MAX\*NMAX | |
| MAXBOT | maximum IBOT (defined real working storage dimension NONSYM solver)  IBOT = size of real working storage for NONSYM solver | |
| INTBOT | MAXBOT + 6\*NMAX + 1 (defined integer working storage dimension for NONSYM solver) - Note: the values of MAXBOT and INTBOT should be set to 1 when NONSYM is not used | |
| NTAB | maximum NPOINT | |
| MAXQOUT | maximum NUM\_QOUT, with NUM\_QOUT = # of surface cells for discharge output | |
| MAXENNOUT | | Is related to data assimilation (DA) (to do not considered if you do not have DA) |

Once saved *CATHY.H*, it is time to compile the CATHY processor in the Terminal/Command window; the executable file (*cathy.exe*) is generated through the compilation ofthe main program *cathy.main.f* and all the subprograms *\*.f* (Teminal/Command Prompt) as follows:

|  |
| --- |
| *gfortran -c \*.f*  *gfortran \*.o -L\MinGW\lib -llapack -lblas -o* ***cathy*** |

At this point, you have to copy the generated executable file (i.e. *cathy.exe*) in the folder ‘**dataset**’ (where the input files are saved: *cathy.fnames* and *CATHY.H*) and to start the numerical simulation (in command/terminal window run *cathy.exe*) to obtain the output files.

## Output files from the processor CATHY

Table 3. Output files

|  |  |  |
| --- | --- | --- |
| IOUT1 | debug | Debugging |
| IOUT2 | risul | output of the simulation |
| IOUT3 | xyz | X, Y, Z coordinate values |
| IOUT4 | iter | Convergence behaviour and errors norms for each iteration of every time step |
| IOUT5 | mbeconv | Mass balance and convergence behaviour at each time step  (REL. MBE (%) should be as small as possible) |
| IOUT6 | vp | Vertical profile output in fixed nodes |
| IOUT7 | hgatmsf | Atmospheric and seepage face hydrograph output |
| IOUT8 | hgnansf | Non‐atmospheric, non‐seepage face hydrograph output |
| IOUT9 | hgflag | Detailed HGFLAG output (counter for anomalous, implausible, or erroneous atmospheric inflow, outflow, and runoff occurrences) |
| IOUT10 | sfflag | Detailed SFFLAG output (counter for anomalous, implausible, or erroneous occurrences along seepage faces) |
| IOUT11 | psi | Pressure head output at all nodes |
| IOUT12 | velnod | Velocity output at all nodes (Transport) |
| IOUT13 | sw | Water saturation output at all nodes (SW) for input to TRAN3D and DUAL3D groundwater contaminant transport |
| IOUT14 | ckrw | Relative hydraulic conductivity output at all nodes |
| IOUT15 | velelt | Velocity output at all elements, for input to TRAN3D and DUAL3D groundwater contaminant transport codes (Transport) |
| IOUT16 | psisurf | Pressure head output at surface nodes |
| IOUT17 | satsurf | SATSUR output at surface nodes |
| IOUT18 | swsurf | Water saturatiom output at surface nodes |
| IOUT19 | nansfdir | Non‐atmospheric, non‐seepage face Dirichlet BCs at each time step |
| IOUT20 | nansfneu | Non‐atmospheric, non‐seepage face Neumann BCs at each time step |
| IOUT30 | hgsfdet | Detailed seepage face hydrograph output (Incoming and outgoing flows at the seepage face) |
| IOUT31 | hgnansfdirdet | Detailed non‐atmospheric, non‐seepage face Dirichlet hydrograph output |
| IOUT32 | hgnansfneudet | Detailed non‐atmospheric, non‐seepage face Neumann hydrograph output |
| IOUT36 | cumflowvol | Output of cumulative flow volumes VSFTOT, VNDTOT, VNNTOT, VNUDTOT, and VTOT |
| IOUT40 | net.ris | SURF\_ROUTE module input data (dem data, geometry data, etc.) |
| IOUT41 | hgraph | Surface runoff hydrograph: plot the computed discharge at the outlet (streamflow) |
| IOUT42 | pondhead | SURF\_ROUTE ponding head output (Pond head superficial nodes) |
| IOUT43 | dtcoupling | CPU, time stepping, iteration and other diagnostics of the surface and subsurface modules at each time step |
| IOUT44 | recharge | Detailed recharge output (spatial map of recharge flux) |
| IOUT50 | hgnudging | Detailed nudging "hydrograph" output |
| IOUT51 | tsnudging | Detailed time series output of model results at the nudging observation points. For intercomparison with a model simulation without nudging, run the same simulation but with NUDG=0.0 or NUDEPS=0.0 (don't set NUDN=0 since NUDSMC cannot be calculated without the coordinates of the nudging observation points!), and plot the results in the IOUT51 output file from both runs, together with the NUDTIM and NUDVAL data from the nudging input file. (This output file is designed for NUDN <= 10; for NUDN > 10 the output will need re‐structuring.) |
| IOUT52 | enpsif | Detailed time series output of the ensemble of PNEW realizations before the update. |
| IOUT53 | enqoutlet | Detailed time series output of the ensemble of outlet Q\_OUT realizations after the update. |
| IOUT54 | enpsia | Detailed time series output of the ensemble of PNEW realizations after the update. |
| IOUT55 | ensubvol | Detailed time series output of the ensemble of subsurface water volume. |
| IOUT56 | enpsiz | Ensemble parameters, initial conditions, weights and SIR updates |
| IOUT57 | wtdepth | Water table depth  (Time, WTDEPTH(NODVP(I)), I=1,2,...,NUMVP) |
| IOUTPT | peatdef | Void ratio output at all nodes in case of deformable peat |
| ITERM | term | Set ITERM to 6 in BLOCK DATA subprogram for terminal output; otherwise unit ITERM output is to a file |
|  | grid2d.exp | Numbering of the grid (open with Argus1) |
|  | grid3d | shows the 3d grid |

# Post-processing

Once the simulation has been successfully completed, it is necessary to run the post-processing subroutines written in Matlab code to extract and view the results of the simulation. Visualizing the results is an important part of the analysis as it allows to assess the quality of the model and to analyze the parameters involved in the simulation. Specifically, several Matlab scripts have been developed (). Running these different subroutines allows to visualize the performance of the variables contained in the output files that have been produced by the compiling of the CATHY processor.

Table 5. Post-processing subroutines written in Matlab (.m)

|  |  |  |  |
| --- | --- | --- | --- |
| ***.m* file** | **Input file** | **Variables** | **Description** |
| atmdiagn | dtcoupling | ? | Compares potential and actual atmospheric forcings |
| atmforc | dtcoupling | ? | Compares potential and actual atmospheric forcings |
| COCumflowvol | cumflowvol | (3) Time  (8) net (VTOT) | Processes cumflowvol (cumulative flow volumes) and creates graph:  Time (h) - Net Flow Volume (m3) |
| COeta | dtcoupling | ? | Processes actual evapotranspiration and creates graphs |
| COHGATMSF | hgatmsf | (3) Time  (6) OVL. FLUX  (7) RET. FLUX | Processes HGATMSF and creates graph:  Time (day) - Overland flow (m3/day)  Time (day) - Return Flow (m3/day) |
| COHGNANSFDIRDET | hgnansfdirdet | (3) Time  (5) NANSF DIR FLX | Processes HGNANSFDIRDET and creates graph:  Time (day) - NANSFDir Flux (m3/day) |
| COHGRAPH | dtcoupling  hgraph | dtcoupling:  (3) Time  (11) Atmpot-r | Processes HGRAPH and creates graph:  Time (days) – Potential atmospheric forcing rate (m/day)  Time (days) – Streamflow (m3/day) |
| COHGSFDET | hgsfdet | (3) Time  (5) NET SEEPFACE FLX | Processes HGSFDET and creates graph:  Time (days) – Net seepage face flux (m3/d) |
| COMBECONV | mbeconv | (3) Time  (17) REL.MBE (%) | Processes MBECONV and creates graph:  Time (hours) - Relative mass balance error (%) |
| contourimg  contouring |  |  | Find contours for an image (square contours, not as usual around point, but around square elements), and draw these (that is: draw contour lines around regions, consisting of square elements) |
| COPONDHEAD | dtcoupling xyz  pondhead |  | Computes ponding head from input files xyz and pondhead |
| COPsisurf | dtcoupling psisurf | dtcoupling:  (3) Time  (11) Atmpot-r  psisurf | Processes Psisurf and creates graphs ……. |
| COrecharge | dtcoupling recharge | dtcoupling:  (3) Time  (11) Atmpot-r | Processes recharge and creates graphs |
| COSatsurf | satsurf |  | Processes Satsurf and creates graphs |
|  |  |  |  |
| COSection |  |  |  |
| COSWSurf |  |  |  |
| covp |  |  |  |
| covpextr |  |  |  |
| coxyz |  |  |  |
| dem\_analysis |  |  |  |
| dem3d |  |  |  |
| dtcoupling |  |  |  |
| efficiency |  |  |  |
| mesh2d |  |  |  |
| mesh3d | grid3d |  | Creates a 3D representation of the 3D grid |
| munique |  |  |  |
| Qefficiency |  |  |  |
| transect |  |  |  |
| vangen |  |  |  |
| Velocity | xyz  velnod |  | Computes velocity at all nodes from input files xyz and velnod, and creates quiver plots |
| vp2wtd |  |  |  |
| WatertableD |  |  |  |
| WatertableZ |  |  |  |
| wtdepth |  |  |  |
| WTefficiency |  |  |  |

To open and read the output files the following code is used:

|  |
| --- |
| **fopen** is used to open the text file. This command returns an integer file identifier equal to or greater than 3; MATLAB reserves file identifiers 0 for standard input, 1 for standard output (the screen), and 2 for standard error. If fopen cannot open the file, then is -1. **'r'** means: open file for reading.  dtcoupling = fopen('output file directory', 'r')  Define **a** and **nstep** and at the beginning set them equal to zero. Define the empty matrix **ATMDIAGN**.  a=0;  nstep=0;  ATMDIAGN=[];  The *While* loop is used to evaluate the expression (a~=(-1); ~= is NOT EQUAL) and repeat the execution of the two statements in the loop while the expression is true. **a** is a text string, but if the line contains the end-of-file marker a is equal to -1.  while (a~=(-1))  a = fgetl(dtcoupling); % Reads the next line of the specified file  nstep = nstep+1; % Find out number of lines and timesteps (nstep)  end  a=0;  nstep = nstep-31; % Eliminate the description lines in the file  frewind(dtcoupling); % Set the file position indicator to the beginning    for i = 1:28 % In this file the first 28 lines are descriptions  fgets(dtcoupling); % Read line from file, keeping newline characters  end  ATMDIAGN = fscanf(dtcoupling,'%g', [22,nstep]);  Reads file data into an array, ATMDIAGN, with dimensions [22,nstep] and positions the file pointer after the last value read. fscanf populates ATMDIAGN in column order. '%g' means 'Floating-point fields' and can contain any of the following: Inf, -Inf, NaN, or -NaN  ATMDIAGN = ATMDIAGN';  close ('all');  Define the variable 'time' that will be used for the graph.  time(1)=ATMDIAGN(1,3); % Extract TIME value (column 3) from the first row  for i = 2:nstep  time(i)=time(i-1)+ATMDIAGN(i-1,2);  end |

This code can change depending on the number of description lines at the beginning of the file (i.e. nstep = nstep - #) and the number of columns of the matrix ATMDIAGN (i.e. [#, nstep]).

Table 6. Dimensioning Parameters (defined in parameter include file CATHY.H)

|  |  |
| --- | --- |
| **ID** | **Description** |
| ROWMAX | Maximum NROW |
| COLMAX | Maximum NCOL |
| MAXCEL | ROWMAX\*COLMAX (maximum NCELL) |
| MAXRES | Maximum NUMRES |
| NODMAX | Maximum NNOD |
| NTRMAX | Maximum NTRI |
| NP2MAX | Maximum NDIR |
| MAXSTR | Maximum NSTR |
| NMAX | NODMAX\*(MAXSTR + 1) (maximum N) |
| NTEMAX | 3\*NTRMAX\*MAXSTR (maximum NT) |
| NPMAX | Maximum NP |
| NQMAX | Maximum NQ |
| NSFMAX | Maximum NSF |
| NNSFMX | Maximum # of nodes on a seepage face + 1 |
| MAXDIR | NODMAX + NPMAX + NSFMAX\*NNSFMX (maximum NUMDIR) |
| MAXNUDN | Maximum NUDN |
| MAXNUDT | Maximum NUDT |
| MAXNUDC | Maximum NUDC |
| MAXNENS | Maximum NENS |
| MAXZON | Maximum NZONE |
| NTAB | Maximum NPOINT |
| MAXTRM | Maximum NTERM (value of MAXTRM should be at least 10\*N) |
| MAXIT | Maximum ITUNS |
| NRMAX | Maximum NR |
| MAXPRT | Maximum NPRT |
| MAXVP | Maximum NUMVP |
| N1MAX | Maximum N1 |
| NTPMAX | N1MAX\*NMAX |
| MAXBOT | Maximum IBOT (defined real working storage dimension for NONSYM solver) |
| INTBOT | MAXBOT + 6\*NMAX + 1 (defined integer working storage dimension for NONSYM solver) Note: the values of MAXBOT and INTBOT should be set to 1 when NONSYM is not used |
| NTAB | Maximum NPOINT |
| DEMRES | Resolution factor for grid generation from DEM |
| MAXQOUT | Maximum NUM\_QOUT |

Table 7. Actual or Minimum Dimensions

|  |  |
| --- | --- |
| **ID** | **Description** |
| NROW | # of rows in the DEM |
| NCOL | # of columns in the DEM |
| NCELL | # of cells in the DEM of the catchment, including "lake" cells; set to 0 for the case DEM = FALSE |
| NUMRES | # of 'reservoirs' defined in the DEM |
| NNOD | # of nodes in 2-d mesh. These are the surface nodes for the 3-d mesh -- they are all designated as atmospheric boundary condition nodes (rainfall and evaporation inputs), except for those surface nodes which are specifically designated as non atmospheric BC's (see description of NDIR, NDIRC, NQ, NSF, and IFATM). |
| NTRI | 2\*NCELL (when SURF\_ROUTE is active, otherwise must be assigned explicitly) = # of triangles in 2-d mesh |
| NDIR | # of non-atmospheric, non‐seepage face Dirichlet nodes in 2-d mesh. The BC's assigned to these surface nodes are replicated vertically (compare NDIRC) |
| NSTR | # of vertical layers |
| NPOINT | # of points of the table moisture curves input |
| N | NNOD\*(NSTR + 1) = # of nodes in 3-d mesh |
| NT | 3\*NTRI\*NSTR = # of tetrahedra in 3-d mesh |

(Continuing from Table 7)

|  |  |  |
| --- | --- | --- |
| **ID** | **Description** | |
| NDIRC | # of 'fixed' non-atmospheric, non-seepage face Dirichlet nodes in 3‐d mesh ('fixed' in the sense that these BC's are not replicated to other nodes ‐ compare NDIR) |
| NP(3) | NDIR\*(NSTR + 1) + NDIRC = total # of non-atmospheric, non‐seepage face Dirichlet nodes in 3‐d mesh |
| NQ(3) | # of non-atmospheric, non‐seepage face Neumann nodes in 3‐d mesh |
| ANP,  ANQ | Actual values of NP and NQ that are passed to the FLOW3D module |
| NSF | # of seepage faces (see description of NSFNOD) |
| NUMDIR | Total # of Dirichlet nodes in 3-d mesh |
| NUDN | # of observation points for nudging (NUDN=0 for no nudging) |
| NUDT | # of observation times for nudging |
| ENKFT | # of observation times for ensemble kalman filter (EnKF) |
| NUDC | # of concurrent observation datasets for nudging at any given time. NUDC is updated at the end of every time step. If NUDC=0, nudging is inactive at the current time. |
| NZONE | # of material types in the porous medium |
| NTERM | # of nonzero elements in system matrices (symmetric storage used for Picard scheme; non-symmetric storage for Newton) |
| ITUNS | Maximum nonlinear FLOW3D iterations per time step |
| NR | # of nodes selected for partial output |
| NPRT | # of time values for detailed nodal output and element velocity output (see description of IPRT, TIMPRT) |
| NROUT | # of time values for detailed output in ENKF case |
| NUMVP | # of surface nodes for vertical profile output |
| N1 | Maximum # of element connections to a node |
| IBOT | Size of real working storage for NONSYM solver (# of nonzero elements in the LU decomposition of system matrix) |
| NENS | Number of realizations in the ensemble kalman filter (EnKF) and the particle filter (SIR). If a realization does not converge, NENS is decreased by one. |
| NENS0 | Initial number of realizations in EnKF/SIR |
| NENSMIN | Minimum number of realizations in EnKF/SIR: if NENS<NENSMIN the simulation ends. |
| NEFF | Effective number of the realizations: estimation of the number of realizations that would be copied in the resample step of the SIR algorithm. | |
| NEFFMIN | Minimum value of NEFF to not require the resampling step after the update of the weights WSIR in the SIR algorithm: the resampling step of the SIR algorithm is performed (RESAMP=TRUE) only if NEFF < NENSMIN. | |
| NOBS | Number of observations (measurements) in the EnKF | |
| NERT | Number of ERT observations (measurements) in the EnKF (NERT < NOBS ) | |
| ISEED | Seed number for the generation of random numbers | |
| CONTDS | Number of time that the coefficient of variation DSMEAS in duplicated in UPDATESIR. It is used for the parameter update using SIR algorithm. | |
| ZONE\_VERT | Number of vertical zones for the soil parameters (used if DAFLAG > 1) | |

Table 8. General and FLOW3D Integer Parameters, Flags, and Indices

|  |  |
| --- | --- |
| **ID** | **Description** |
| KPRT | Index to current time value for detailed output |
| IPEAT | Flag for peat soil deformation  =0 constant porosity (in unsaturated soil)  =1 consider porosity variations with water saturation |
| IPRT | Flag for detailed output at all nodes and velocity and water saturation output at all elements (velocity and water saturation output in the case IPRT=4 can be used as input to TRAN3D and DUAL3D codes)  =0 don't print nodal pressure, velocity, water saturation, or relative conductivity values  =1 print only nodal pressure head values  =2 print nodal pressure head and velocity values  =3 print nodal pressure, velocity, and relative conductivity values  =4 print nodal pressure, velocity, relative conductivity, and overall storage coefficient values, and print element velocity and nodal water saturation values |

(Continuing from Table 8)

|  |  |
| --- | --- |
| **ID** | **Description** |
| IPRT1 | Flag for output of input and coordinate data in subroutines DATIN and GEN3D  = ‐1 reads in coordinates from file IIN51 just after grid generation (in subroutine grdsys)  = 0 prints parameters only (default)  = 1 prints parameters + b.c. + geom. char.  = 2 prints parameters + b.c. + geom. char. + grid info  = 3 prints parameters + b.c. + geom. char. + grid info, X, Y, Z coordinate values in subroutine GEN3D, and then terminates program execution |
| ISIMGR | Flag for type of simulation and type of surface grid  =0 subsurface flow only (FLOW3D) with general triangular grid input  =1 subsurface flow only (FLOW3D) with DEM input and triangular grid generated from this DEM  =2 coupled subsurface flow (FLOW3D) and surface routing (SURF\_ROUTE) with DEM input and triangular grid generated from this DEM |
| IVERT | =0 each layer will be parallel to the surface, including the base of the 3‐d grid. ZRATIO is applied to each vertical cross section.  =1 base of the 3‐d grid will be flat, and ZRATIO is applied to each vertical cross section  =2 base of the 3‐d grid will be flat, as will the NSTR‐1 horizontal cross sections above it. ZRATIO is applied only to the vertical cross section having the lowest elevation.  =3 for each cell of the dem a single depth value is read in file input IIN60 (basement). ZRATIO is applied to each vertical cross section.  =4 the first NSTR‐1 layers from the surface will be parallel to the surface and the base of the 3‐d grid will be flat. ZRATIO is applied only to the vertical cross section having the lowest elevation. |
| ISP | =0 for flat surface layer (only one Z value is read in, and is replicated to all surface nodes); otherwise surface layer is not flat (Z values read in for each surface node) (for ISP=0, IVERT=0, 1, and 2 yield the same 3‐d mesh, given the same values of BASE and ZRATIO) |
| INDP | Flag for pressure head initial conditions (all nodes)  =0 for input of uniform initial conditions (one value read in)  =1 for input of non-uniform IC's (one value read in for each node)  =2 for calculation of fully saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVHE). In the case of IPOND>0, the fully saturated hydrostatic IC is calculated (in subroutine ICVHEPOND) starting from the ponding head values at the surface nodes, rather than surface pressure heads of 0.  =3 for calculation of partially saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVHWT) with the water table height (relative to the base of the 3‐d grid) given by parameter WTHEIGHT  =4 for calculation of partially saturated vertical hydrostatic equilibrium IC's (calculated in subroutine ICVDWT) with the water table depth (relative to the surface of the 3‐d grid) given by parameter WTHEIGHT |
| HTIDIR | =0 for temporally variable non‐atmospheric, non‐seepage face Dirichlet boundary conditions inputs; otherwise non‐atmospheric, non‐seepage face Dirichlet boundary conditions inputs are homogeneous in time (see also notes following description of QPOLD) |
| HTINEU | =0 for temporally variable non‐atmospheric, non‐seepage face Neumann boundary conditions inputs; otherwise non‐atmospheric, non‐seepage face Neumann boundary conditions inputs are homogeneous in time (see also notes following description of QPOLD) |
| HSPATM | =0 for spatially variable atmospheric boundary condition inputs; blank or =9999 if unit IIN6 input is to be ignored; otherwise atmospheric BC's are homogeneous in space |
| HTIATM | =0 for temporally variable atmospheric boundary condition inputs; otherwise atmospheric BC's are homogeneous in time (see also notes following description of ATMINP) |
| IETO | =0 for linear interpolation of the atmospheric boundary condition inputs between different ATMTIM; otherwise the inputs are assigned as a piecewise constant function (ietograph) |
| LUMP | =0 for distributed mass matrix; otherwise matrix is lumped |
| IOPT | =1 for Picard iteration scheme  =2 for Newton iteration scheme |
| NLRELX | Flag for nonlinear relaxation  =0 no relaxation  =1 relaxation with constant relaxation parameter OMEGA  =2 relaxation with iteration-dependent relaxation parameter OMEGA, calculated using Huyakorn et al's adaptation (WRR 1986 22(13), pg 1795) of Cooley's empirical scheme (WRR 1983 19(5), pg 1274) |
| L2NORM | =0 to use the infinity norm in the test for convergence of the nonlinear FLOW3D and coupled FLOW3D/SURF\_ROUTE iterations; otherwise the L2 norm is used. |
| ISOLV | Flag for non-symmetric linear solver  =-5 BiCGSTAB (preconditioned with D^-1)  =-4 BiCGSTAB (not preconditioned)  =-3 TFQMR (preconditioned with D^-1)  =-2 TFQMR (not preconditioned)  =-1 TFQMR (preconditioned with K^-1)  =0 BiCGSTAB (preconditioned with K^-1)  =1 GRAMRB (minimum residual)  =2 GCRK(5) (ORTHOMIN)  =3 IBM's NONSYM (direct solver) |
| IVGHU | = -1 table look up for moisture curves  = 0 for van Genuchten moisture curves  = 1 for extended van Genuchten moisture curves  = 2 for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) with Kr=Se\*\*n conductivity relationship  = 3 for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) with conductivity relationship from Table 3 of 1984 paper (log\_10 Kr(Se) curve)  = 4 for Brooks‐Corey moisture curves |
| IRETC | =0 table retention curves variable with layers  =1 table retention curves constant with layers |

(Continuing from Table 8)

|  |  |
| --- | --- |
| **ID** | **Description** |
| KSLOPE | = 0 for analytical differentiation of moisture curves  = 1 for "chord slope" and analytical differentiation  = 2 for "chord slope" and centered difference formulas  = 3 for localized "chord slope" and analytical differentiation  = 4 for localized "tangent slope" differentiation  *(the "chord slope" formula is the tangent approximation suggested by Huyakorn et al (WRR 20(8) 1984), wherein derivatives are approximated using pressure heads at the current and previous nonlinear iterations; "tangent slope" differentiation is a different tangent approximation wherein derivatives are approximated using pressure heads at the endpoints of a given range (eg: endpoints PKRL, PKRR for the derivative of relative hydraulic conductivity). For KSLOPE=1, 2 the chord slope formula is used at every iteration and at all nodes (with some exceptions as dictated by TOLKSL). For KSLOPE=3 or 4 the chord or tangent slope formulas are used only at those nodes whose pressure heads fall within given ranges (see PKRL, PKRR, etc), hence 'localized'; for nodes whose pressure heads fall outside these ranges, analytical differentiation is used.)* |
| ISFONE | =0 seepage face exit point updating performed by checking all nodes on a seepage face  =1 seepage face exit point updating performed by checking only the one node above and one node below the current exit point position |
| ISFCVG | =0 convergence of seepage face exit points is not a condition for convergence of the nonlinear iterative procedure  =1 convergence of seepage face exit points is a condition for convergence of the nonlinear iterative procedure |
| DUPUIT | =0 all the nodes below the seepage face exit point are at atmospheric pressure  =1 all the nodes below the seepage face exit point are at hydrostatic pressure. CAUTION: TO USE ONLY IF ISFONE=1! |
| KSFCV | Total number of seepage face exit point convergence failure occurrences (over all nonlinear iterations and all time steps) |
| KSFCVT | Total number of seepage face exit point convergence failures (over all seepage faces, all nonlinear iterations, and all time steps) |
| ITER | Iteration index for nonlinear FLOW3D iterations for each time step |
| MAXITER | Max iteration index for nonlinear FLOW3D iterations for each time step for the ensemble of realizations of EnKF |
| ITUNS1 | If ITER < ITUNS1, time step size is increased |
| ITUNS2 | If ITUNS1 <= ITER < ITUNS2, time step size is not altered if ITUNS2 <= ITER < ITUNS, time step size is decreased if ITER = ITUNS (ie, convergence not achieved in ITUNS iterations), we back‐step unless time step size cannot be reduced any further (DELTAT = DTMIN). Back‐stepping is also triggered if the linear solver failed (LSFAIL = TRUE) or if the convergence or residual errors become larger than ERNLMX (ERRGMX = TRUE). |
| NSTEP | Time step index |
| KBACKT | Number of back‐stepping occurrences at current time level |
| KBACK | Total number of back‐stepping occurrences over all time steps |
| ITRTOT | Total number of nonlinear FLOW3D iterations over all time steps |
| ITMXCG | Maximum # of iterations for conjugate gradient linear system solvers |
| NITER | Number of iterations for the linear solver at each nonlinear FLOW3D iteration |
| NITERT | Number of iterations for the linear solver at each time step |
| ITLIN | Total number of iterations for the linear solver over all nonlinear FLOW3D iterations and all time steps |
| KLSFAI | Total number of linear solver failures |
| IMAX | Largest integer number (machine dependent) |
| MINBOT | Minimum IBOT (value returned from the solver) |
| NDZ | # of zero elements on the diagonal of the system matrices (signals an error condition) |
| DAFLAG | Flag for the choice of the data assimilation scheme:  = 0 nudging (if NUDN=0, no data assimilation)  = 1 EnKF with Evensen's algorithm (Ocean Dynamics, 2004)  = 2 EnKF with parameters update  = 3 Particle filtering (SIR algorithm)  = 4 Particle filtering (SIR algorithm) with parameters update |
| ERT\_FLAG | Flag for assimilation of ERT measures  = 0 no ERT measures assimilated  = 1 ERT measures assimilated  = 2 ERT measures as output in the detailed output times |

(Continuing from Table 8)

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDFLAG | Flag for the choice of the variable to be assimilated by nudging:  = 0 soil moisture;  = 1 pressure head (in this case NUDG must be properly scaled. |
| WFLAG | Flag for the choice of the weighting functions for nudging:  = 0 Cressman‐type functions [Paniconi et al., AWR, 2003];  = 1 Exponential and Gaussian correlation functions for time and spatial influence, respectively. |

Table 9. Integer Parameters for SURF\_ROUTE

|  |  |
| --- | --- |
| **ID** | **Description** |
| NSURF | # of time steps for the surface routing module (estimated to satisfy the Courant criterion) at the current time level of the subsurface module |
| NSURFT | # of time steps for the surface routing module at the current subsurface time level including those from any back‐steps at the current time level |
| NSURFT\_T | Total # of time steps for the surface routing module over all subsurface time steps, excluding back‐steps |
| NSURFT\_TB | Total # of time steps for the surface routing module over all subsurface time steps, including back‐steps |
| NCELNL | # of cells in the DEM of the catchment excluding the "lake" cells |
| IPOND | Flag for ponding head initial conditions (surface nodes)  =0 no input of ponding head initial conditions; otherwise (IPOND = 1 or 2) ponding head initial conditions are read into PONDNOD, and, where PONDNOD > 0, these values are used to update the surface node values in PTIMEP read in according to the previous INDP flag  =1 uniform ponding head initial conditions (one value read in)  =2 non-uniform ponding head initial conditions (one value read in for each node) |
| DOSTEP | step adopted in coarsening the mesh |
| NCELL\_COARSE | # of cells in the coarse mesh |
| HSPEFF | =0 for spatially variable effective rainfall inputs, otherwise effective rainfall is homogeneous in space. |
| HTIEFF | =0 for temporally variable effective rainfall inputs; otherwise effective rainfall is homogeneous in time (see also notes following description of ATMINP) |
| NUM\_QOUT | # of selected cells for hydrograph output |
| ID\_QOUT | I\_BASIN index of selected cells for hydrograph output |
| I\_BASIN | DEM cells index such as calculated in the pre‐processor, and stored in the QOI\_SN vector. The index is calculated in this way: |

Table 10. Integer Parameters for Nudging and EnKF

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDCTR | Counter for most recent nudging observation time. NUDCTR is incremented by one whenever the next value of TIME, which is updated at the end of the current time level, falls  C within the temporal influence window of the next observation time in NUDTIM. |
| ENKFCTR | Counter for most recent EnKF observation time. ENKFCTR is incremented by one whenever the next value of TIME, which is updated at the end of the current time level, is equal to the next observation time in ENKFTIM. |
| MAXENNOUT | Maximum number of output for the EnKF = int(TMAX/DTOUT) |

Table 11. Logical Flags (True and False)

|  |  |
| --- | --- |
| **ID** | **Description** |
| FL3D | Flag indicating whether the simulation involves the subsurface component or not (FLOW3D)  FALSE if no FLOW3D call (SURF\_ROUTE simulation only)  TRUE if FLOW3D is to be called (coupled or not) |
| SURF | Flag indicating whether the simulation involves the surface routing component or not (SURF\_ROUTE)  FALSE if no SURF\_ROUTE call (FLOW3D simulation only)  TRUE if SURF\_ROUTE is to be called (coupled or not) |
| DEM | Flag indicating whether a DEM is input or not  FALSE if no DEM input  TRUE if DEM input |
| GRID | Flag indicating whether the triangular surface grid is input or generated  FALSE if the grid is generated from DEM input  TRUE if the grid is read in as input |
| PONDING | Flag indicating whether there is ponding at the surface and SURF\_ROUTE has to be called  FALSE no ponding  TRUE ponding ‐> call SURF\_ROUTE |
| PONDP | PONDING value at previous time level |
| DTGMIN | Flag indicating whether the current time step size is greater than the minimum allowed  FALSE if not greater  TRUE if greater |
| LSFAIL | flag for linear solver  FALSE if linear solver did not fail  TRUE if linear solver failed |
| ERRGMX | Flag indicating whether the FLOW3D convergence or residual errors have become greater than the allowed maximum  FALSE if not greater  TRUE if greater |
| NORMCV | Flag for convergence of the norm of pressure head differences in the nonlinear FLOW3D iterative procedure  FALSE if the norm has not converged  TRUE if the norm has converged |
| ITAGEN | Flag indicating whether we can iterate again in the nonlinear FLOW3D iterative procedure  FALSE if we cannot iterate again  TRUE if we can iterate again |
| SFCHEK | Flag indicating whether it is necessary to check for seepage face exit point convergence as a condition for convergence of the nonlinear FLOW3D iterative procedure  FALSE if it is not necessary to check  TRUE if it is necessary to check |
| KSFZER | Flag for number of seepage face exit points which did not converge at each nonlinear FLOW3D iteration  FALSE if one or more exit points did not converge  TRUE if all exit points converged |
| NOBACK | flag for back‐stepping  FALSE if back‐stepping still possible  TRUE if no back‐stepping possible |
| BCKSTP | Flag indicating whether we are back‐stepping  FALSE we are not in the back‐stepping case  TRUE we are doing back‐step |
| ENKF | Flag indicating whether we are using EnKF or SIR schemes.  FALSE ‐> no EnKF/SIR  TRUE ‐> EnKF/SIR |
| RESAMP | If we are using SIR scheme (DAFLAG=3 or 4), RESAMP flag indicats when it necessary to make a RESAMPLE after SIR weights update |
| UPD | Flag indicating whether an update has been made at the last time step |

Table 12. Real Scalars for Initial Conditions and Vertical Discretization

|  |  |
| --- | --- |
| **ID** | **Description** |
| WTHEIGHT | For the case INDP=3, specifies the initial water table height relative to the base of the 3‐d grid |
| BASE | Value which defines the thickness or base of the 3‐d mesh. For IVERT=0, BASE is subtracted from each surface elevation value, so that each vertical cross section will be of thickness BASE, and the base of the 3‐d mesh will be parallel to the surface. For IVERT=1 or 2, BASE is subtracted from the lowest surface elevation value, say ZMIN, so that each vertical cross section will be of thickness (Z ‐ ZMIN) + BASE, where Z is the surface elevation for that cross section. The base of the 3‐d mesh will thus be flat. |

Table 13. Real Scalars for Time Stepping and Linear and Nonlinear Iterations

|  |  |
| --- | --- |
| **ID** | **Description** |
| TETAF | Weighting parameter for FLOW3D time stepping scheme (1.0 backward Euler; 0.5 Crank-Nicolson; TETAF is set to 1.0 for steady state problem). |
| DELTAT | Initial and current FLOW3D time step size (DELTAT >=1.0e+15 on input indicates steady state problem) |
| DELTAT0 | Initial FLOW3D time step size, stored in case of EnKF for reinitializing DELTAT when an update occurs |
| DELTATS | Initial and current SURF\_ROUTE time step size (this time step is kept constant during the simulation, and the value equals to the DELTAT assigned in the parm input file, IIN1) |
| DTMIN | Minimum FLOW3D time step size allowed |
| DTMAX | Maximum FLOW3D time step size allowed |
| DTAVG | Average FLOW3D time step size used for the simulation |
| DTSMAL | Smallest FLOW3D time step size used during the simulation |
| DTBIG | Largest FLOW3D time step size used during the simulation |
| TSMAL | First time at which DTSMAL is used |
| TBIG | First time at which DTBIG is used |
| DTMAGA | Magnification factor for FLOW3D time step size (additive) |
| DTMAGM | Magnification factor for FLOW3D time step size (multiplicative) |
| DTREDS | Reduction factor for FLOW3D time step size (subtractive) |
| DTREDM | Reduction factor for FLOW3D time step size (multiplicative) |
| TMAX | Time at end of simulation (TMAX is set to 0.0 for steady state problem) |
| TIME | Time at current time level |
| TIMEP | Time at previous time level |
| OMEGA | Non-linear relaxation parameter: OMEGA > 1, over-relaxation; OMEGA < 1, under-relaxation. Input value of OMEGA is used only for the case NLRELX=1 (constant relaxation parameter). Input value of OMEGA is ignored otherwise: for NLRELX=0 relaxation is not applied; for NLRELX=2 OMEGA is calculated at each nonlinear FLOW3D iteration |
| OMEGAP | OMEGA value at previous nonlinear FLOW3D iteration |
| TOLUNS | Tolerance for convergence of nonlinear FLOW3D iterations |
| TOLSWI | Tolerance for boundary condition switching check in FLOW3D iterations (switching check is only performed when PINF or PL2 are smaller than TOLSWI; so for e.g. if TOLSWI = TOLUNS, switching check is only performed after convergence and not after each iteration) |
| ERNLMX | Maximum allowable convergence or residual error in the nonlinear FLOW3D solution. If the convergence or residual errors become larger than ERNLMX, ERRGMX is set to TRUE and the code back‐steps. This avoids occurrences of overflow or underflow when nonlinear iterations diverge. |
| TOLCG | Tolerance for convergence of conjugate gradient linear system solvers |

Table 14. Real Scalars for output times for ENKF/SIR methods

|  |  |
| --- | --- |
| **ID** | **Description** |
| DTOUT | Time step for not‐detailed output. Not‐detailed output is produced at initial conditions (TIME=0), at DTOUT‐spaced time values, and at the end of the simulation (TIME=TMAX). DTOUT is read in the file "parm". Not‐detailed output consists of:hgraph,mbeconv. |
| TIMEPOUT | Previous output time |
| TIMESTOP | Next time in which time iterations stop for produce the output or to make an update |
| TIMESTART | Time of the last output/update |

Table 15. Real Scalars for Ponding

|  |  |
| --- | --- |
| **ID** | **Description** |
| PONDH\_MIN | Minimum ponding head: if PNEW > PONDH\_MIN, then at that node there is ponding; otherwise there is no ponding |

Table 16. Real Scalars for Nudging and EnKF

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDG | Nudging factor "G" which determines the relative strength of the nudging (dynamical relaxation) term with respect to the physical forcing term |
| DSRETC, DSKS, DSSTOR, DSPOROS, DSSURF, DSKSZ | Initial coefficient of variation, in the EnKF/SITR scheme, for the retention curve parameters, the saturated hydraulic conductivity (x,y), the elastic storage coefficient, the porosity, the surface routing parameters, and the measurements, the saturated hydraulica conductivity (z), respectively |
| ENDSSURF\_KS, ENDSSURF\_WS | Ensemble coefficient of variation of surface routing parameters |
| DSMEASMAX | Maximum value for the coefficient of variation of measurements in the SIR updates. |
| DSIC | Standard deviation, in the EnKF data assimilation scheme, for the initial conditions (normally distributed) |
| DSATM | Standard deviation, in the EnKF data assimilation scheme, for the atmospheric boundary conditions (lognormally distributed), expressed in percentage respect to the nominal values. |
| ATMTAU | Time correlation scale of the observation errors for the atmospheric boundary condition input. CAUTION: ATMTAU must be set to a minimum of DTMAX, in order to keep the parameter ALPHA (see atmnxt, atmbak) greater than 0.0. |
| QTIMEP\_1 | QTIMEP for the current realization |
| QNEW\_1 | QNEW for the current realization |
| WSUM | Sum of the weights WSIR |
| STORE\_SAT | Saturation volume calculated with the porosity in input |
| QEN\_SPREAD | Standard deviation of the ensemble streamflow at the outlet |

Table 17. Miscellaneous Real Scalars

|  |  |
| --- | --- |
| **ID** | **Description** |
| AREATOT | Total area of the catchment surface |
| VOLTOT | Total volume of the discretized catchment |
| FHORT | Fraction of surface nodes that are Horton saturated |
| FDUNN | Fraction of surface nodes that are Dunne saturated |
| FPOND | Fraction of surface nodes that are ponded |
| FSAT | Fraction of surface nodes that are saturated or ponded |
| VOLUME\_OUT | …… |
| VOLUME\_SUP | …… |
| VOLSUPNEW | …… |
| EVAP\_EFF | …… |
| INF\_TOT | …… |
| RMAX | Largest double precision # (machine dependent) |
| RMIN | Smallest double precision # (machine dependent) |

Table 18. Real\*4 Scalars for Cpu Timing

|  |  |
| --- | --- |
| **ID** | **Description** |
| CPUSUB | Cpu time for subsurface flow module |
| CPUSUB\_T | Total cpu time for subsurface flow module |
| CPUSURF | Cpu time for surface routing module |
| CPUSURF\_T | Total cpu time for surface routing module |
| CPUNL | Total cpu time for nonlinear scheme |
| CPUOVH | Total cpu time for overhead:   * data input, initialization, and output of initial conditions (once) * construction of tetrahedral elements (once) * volume calculations (once) * set up of storage indices and pointers (once) * velocity calculations (every time step for the case IPRT > 1) * hydrograph calculation (every time step) * input, interpolation, and switching control of atmospheric boundary conditions for the next time level (every time step) * update of pressure heads for the next time level(every time step) * back‐stepping procedure (when needed) * final output (once) |
| CPUMN | Total cpu time for the simulation |
| CPUUPD | Cpu time for all the updates |
| CPUUPD1 | Cpu time for a single update |
| CPUUPD2 | Cpu time for all the updates over the number of realization |
| CPUNLT | …… |
| AVGNL | …… |
| AVGLIN | …… |
| AVGLNL | …… |
| ATCTS | …… |
| ATCNL | …… |
| ANCTS | …… |
| ANCNL | …… |
| PCMN | …… |
| PCNL | …… |
| PCOVH | …… |
| PCNLT | …… |
| PCVEC1‐9 | …… |

Table 19. Real\*4 Array for Cpu Timing

|  |  |
| --- | --- |
| **ID** | **Description** |
| CPUVEC(11) | Cpu times for different sections of nonlinear schemes:   1. unsat characteristics 2. initialization of system matrices 3. assembly of local system components into global matrices 4. calculation of RHS without boundary conditions 5. construction of global LHS system matrix 6. calculation of BC contributions to RHS 7. linear solver and calculation of residual 8. extraction of pressure head solution from the difference solution and re‐setting of solution and of COEF1 for Dirichlet nodes 9. back‐calculation of fluxes at all Dirichlet nodes, mass balance calculation, application of nonlinear relaxation scheme (if required), calculation of nonlinear convergence and residual error norms, switching control of atmospheric BCs, and calculation of new position of the exit point along each seepage face 10. …timer for SURF\_ROUTE... 11. …timer for nudging... |

Table 20. Integer Arrays for Grid, BCs, Outputs, and System Matrices

|  |  |
| --- | --- |
| **ID** | **Description** |
| TP (N) | # of elements connecting to each node |
| TP2D (NNOD) | … |
| TRIANG(4,NTRI) | Element connectivities in 2‐d mesh (TRIANG(4,I) indicates material type for 2‐d element I) |
| CELL (5,NCELL) | Cell connectivities in DEM to 2‐d mesh (CELL(5,I) indicates material type for cell I) |
| TETRA (5,NT) | Element connectivities in 3‐d mesh (TETRA(5,I) indicates material type for 3‐d element I) |
| IVOL (NT) | Sign of the volume of each element |
| CONTP2(NDIR) | Non‐atmospheric, non‐seepage face Dirichlet node #'s in 2‐d mesh |
| CONTP (3,NP) | Non‐atmospheric, non‐seepage face Dirichlet node #'s in 3‐d mesh |
| CONTQ (3,NQ) | Non‐atmospheric, non‐seepage face Neumann node #'s in 3‐d mesh |
| ACONTP(ANP) ACONTQ(ANQ) | Actual values of CONTP and CONTQ that are passed to the FLOW3D module |
| IFATM (NNOD) | IFATM(I)=0 if surface node I is a Neumann atmospheric boundary condition node  IFATM(I)=1 if surface node I is a Dirichlet atmospheric boundary condition node  IFATM(I)=2 if surface node I is "ponded", treated as a Dirichlet atmospheric boundary condition node  IFATM(I)=‐1 if surface node I is not an atmospheric boundary condition node  Note: surface nodes are numbered 1,...,NNOD in the 3‐d mesh, so there is no need for a pointer array giving the node #'s for the surface nodes |
| IFATMP(NNOD) | IFATM values at previous time level |
| SATSUR(NNOD) | SATSUR(I)=1 if surface node I is unsaturated  SATSUR(I)= 2 if surface node I is Horton saturated (infiltration excess mechanism)  SATSUR(I)=3 if surface node I is Dunne saturated (saturation excess mechanism) |
| NSFNUM(NSF) | # of nodes on each seepage face |
| NSFNOD(NSF,NNSFMX) | Node #'s on each seepage face. The node #'s for each seepage face must be input in descending order by elevation. That is, along seepage face I, Z (NSFNOD (I,J)) .GE. Z (NSFNOD (I,J+1)) must hold for J=1,...,NSFNUM(I)‐1.  Seepage faces can be defined, for instance, above a well, along a stream bank, or along a combination of stream bank and surface nodes. For a configuration of seepage face and stream, the stream nodes should be designated as non‐atmospheric, non‐seepage face nodes, for instance as Dirichlet nodes with a pressure head distribution in hydrostatic equilibrium, the node at the surface of the stream being assigned a pressure head value of zero.  For output purposes, we set NSFNOD (I,NSFNUM(I)+1)=‐9999. |
| SFEX (NSF) | The exit point on each seepage face. The seepage face nodes above the exit point are 'potential' seepage face nodes, are treated as zero flux Neumann BC's, and the pressure heads here should be negative (unsaturated). The seepage face nodes below the exit point (and including the exit point) are 'actual' seepage face nodes, are treated as zero pressure head Dirichlet BC's (saturated), and the back‐calculated fluxes here should be negative (outflow).The position of the exit point for the first time step is calculated from the initial conditions. The new position of the exit point is calculated after every nonlinear iteration of every time step, and the boundary conditions for the seepage face nodes are adjusted to reflect changes in the position of the exit point. For the case where seepage face I is completely saturated (all seepage face nodes are 'actual'), SFEX(I)=1. For the case where seepage face I is completely unsaturated (all seepage face nodes are 'potential' and there is no exit point), SFEX(I)=NSFNUM(I)+1.This convention simplifies the handling of seepage face nodes (relying on the fact that FORTRAN 77 does not execute a DO loop if the iteration count is zero or negative). |
| SFEXIT(NSF) | SFEX values at previous nonlinear iteration |
| SFEXP (NSF) | SFEX values at previous time level |
| NODDIR(NUMDIR) | Node #'s for all Dirichlet nodes in 3‐d mesh |
| CONTR (NR) | Node #'s for partial output |
| NODVP (NUMVP) | Node #'s for surface nodes selected for vertical profile output |
| IA (NTERM) | Row indices in storage of system matrices for non-symmetric case |
| JA (N1\*N) | Column indices (in ascending order) in storage of system matrices |
| TOPOL (N+1) | Pointer to first nonzero element of each row which is stored in the system matrices (the diagonal entry in symmetric storage case) |

(Continuing from Table 20)

|  |  |
| --- | --- |
| **ID** | **Description** |
| TETJA (4,4,NT) | Gives the index within JA (global position) of each component of the 4 x 4 local system matrices (upper triangle of 4 x 4 arrays only for symmetric case). For the symmetric case, construction of TETJA requires the reordering of the nodes of the tetrahedra, so that assembly of the system matrices can be done by indexing directly the upper triangular part of the matrix. |
| IP3 (3,3) | 3 x 3 permutation matrix |
| IP4 (4,4) | 4 x 4 permutation matrix |
| SFFLAG(5) | ‐ counter for anomalous, implausible, or erroneous occurrences along seepage faces(see output statements 2100,2200 in subroutine SFINIT, 2100,2200,2300,2400 in subroutines EXTONE, EXTALL, and 2500 in subroutine FLUXMB) |
| HGFLAG(9) | Counter for anomalous, implausible, or erroneous atmospheric inflow, outflow, and runoff occurrences (see subroutine HGRAPH) |
| IER (7) | Error flags |
| INSYM (IBOT+6N+1) | Integer scratch vector for NONSYM solver |

Table 21. Integer Arrays for SURF\_ROUTE

|  |  |
| --- | --- |
| **ID** | **Description** |
| ZONE (NROW,NCOL) | Map of different material zones |
| LAKES\_MAP (NROW,NCOL) | Map of lakes in the DEM.  =0 if no lake cell; otherwise lake cell |
| BASE\_MAP (NROW,NCOL) | Raster map of the catchment impermeable basement. Each value represents the thickness of the corresponding cell. |
| DTM\_P\_OUTFLOW\_1 (NCOL,NROW) | Raster map of cardinal flow directions. |
| DTM\_P\_OUTFLOW\_2 (NCOL,NROW) | Raster map of diagonal flow directions. |
| NODI (NROW+1,NCOL+1) | Raster containing the numbering of surface nodes. |
| INDCEL (NROW,NCOL) | Cell # for each active cell in the DEM, excluding lake cells |
| INDCELWL (NROW,NCOL) | Cell # for each active cell in the DEM, including lakes cells |
| QOI\_SN (NCELL) | Vector containing the I\_BASIN index.  The cells are ordered in a descending elevation order. |
| CELTYPE (NCELL) | Cell type. =0 if channel cell; otherwise reservoir cell of type CELTYPE. |
| CELLCOL (NCELL) | Column # for each cell |
| CELLROW (NCELL) | Row # for each cell |
| CELLCOL\_WL (NCELL) | Column # for each cell counting lake cells |
| CELLROW\_WL (NCELL) | Row # for each cell counting lake cells |
| CELLS\_R (NUMRES) | …… |
| MODIF (NCELL) | …… |
| RESERVR (NCELL) | Reservoir #. =0 if not a reservoir cell. |
| C N\_HA (NUMRES) | …… |

Table 22. Integer Arrays for Nudging and EnKF

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDTET(NUDN) | Pointer to the element within which the observation points for nudging are located |
| ENKFNOD(NOBS,2) | ENKFNOD(I,1) is the pointer to the nodes or cells where the observations for EnKF are located. ENKFNOD(I,2) is a flag indicating the assimilated quantity:  0 ‐ soil moisture (node‐based).  1 ‐ pressure head (node‐based);  2 ‐ output discharge (cell‐based);  3 ‐ electrical potential  ATTENTION!  Observation cells MUST be ordered by descending elevation in input file enkf! Moreover, if DAFLAG=1 (Evensen) soil moisture observation nodes must be put AFTER discharge observation cells; (OLD: vice versa for DAFLAG=2 (Keppenne)). |
| ENIFATM(NNOD,NENS) ENIFATMP(NNOD,NENS) | Store values of IFATM and IFATMP (see above) for realization in the EnKF. |
| ENNSTEP(NENS) | Store the number of time steps for each realization. |
| ENNOUT(NENS) | Number of output for each realization of EnKF |
| ENITRTOT(NENS) | Total number of nonlinear iterations over all time steps for each realization |
| MAXNSTEP(2) MINSTEP(2) | Store the realization which has done more (or less) time steps and the corresponding number of steps |
| ENITLIN(NENS) | Store the total number of linear iteration or each realization |
| ENFLAG(NENS) | Initialized to 1s; we put 0 if the realization does not converge |
| ENPT(NENS) | Pointer to the realizations in the ensemble; when the i‐th realization does not converge, ENPT(i) points to the last realization and the ensemble size NENS is decreased by one. |
| STR\_ZON(ZONE\_VERT) | For each vertical zone contain the last layer (str) of the zone. Examples: if ZONE\_VERT=1, then STR\_ZON (1)=NSTR; if ZONE\_VERT=NSTR, then STR\_ZON=[2 3 4 ... NSTR]. |

Table 23. Real Arrays for Mesh Configuration and Boundary Conditions

|  |  |
| --- | --- |
| **ID** | **Description** |
| X (N) | x‐coordinates (for 2‐d mesh on input) |
| Y (N) | y‐coordinates (for 2‐d mesh on input) |
| Z (N) | z‐coordinates (surface elevation values on input ‐ see description of ISP) |
| DEPTH (NNOD) | Depth value for the 2‐d mesh on input, read only if IVERT=3 |
| XC (NT) | x‐coord at the centroid of each tetrahedra |
| YC (NT) | y‐coord at the centroid of each tetrahedra |
| ZC (NT) | z‐coord at the centroid of each tetrahedra |
| VOLNOD(N) | Absolute value of volume assigned to each node |
| VOLU (NT) | Absolute value of the volume of each element |
| VOLUR (NT) | reciprocal of VOLU |
| ZRATIO(NSTR) | Fraction of total grid height that each layer is to occupy (see also description of IVERT). ZRATIO (1) is for the surface‐most layer. ZRATIO values must sum to 1. |
| PRESC (NP) | Non‐atmospheric, non‐seepage face Dirichlet values at current time level |
| PTIM (3) | Most current input time values for non‐atmospheric, non‐seepage face Dirichlet BC's, with PTIM(1) < PTIM(2) < PTIM(3) and PTIM(2) < TIME <= PTIM(3) |
| PINP (3,NP) | Non‐atmospheric, non‐seepage face Dirichlet values corresponding to PTIM times. |
| PRESC(I) | is obtained from PINP(2,I) and PINP(3,I) by linear interpolation (not any more, now by piecewise constant function). PINP(1,I) values are needed in the event that, after back‐stepping, we have PTIM(1) < TIME <= PTIM(2) |
| Q (NQ) | Non‐atmospheric, non‐seepage face Neumann values at current time level QTIM (3) ‐ most current input time values for non‐atmospheric, non‐seepage face Neumann BC's, with QTIM(1) < QTIM(2) < QTIM(3) and QTIM(2) < TIME <= QTIM(3) |
| QINP (3,NP) | Non‐atmospheric, non‐seepage face Neumann values corresponding to QTIM times. Q (I) is obtained from QINP(2,I) and QINP(3,I) by linear interpolation. QINP(1,I) values are needed in the event that, after back‐stepping, we have QTIM(1) < TIME <= QTIM(2) |
| QPNEW (NP) | Back‐calculated flux values at non‐atmospheric, non‐seepage face Dirichlet nodes at current time level |

(Continuing from Table 23)

|  |  |
| --- | --- |
| **ID** | **Description** |
| QPOLD (NP) | QPNEW values at previous time level |
| QTRANIE(N) | Root‐zone water uptake at current time level always positive, changes sign in BCPIC |
| ZROOT(NNOD) | depth of root zone at every surface node |
| SCF | soil cover fraction (fraction of soil covered in vegetation) |
| Notes: | 1. For a simulation using temporally homogeneous non‐atmospheric, non‐seepage face Dirichlet (Neumann) BC's, input data on unit IIN8 (IIN9) should contain a single value of PTIM (QTIM) (0.0) and a single set of PINP (QINP) data. Alternatively, to properly handle the case where the datasets for different simulations are kept in the same file (separated by blank lines), the input data on unitIIN8 (IIN9) for temporally homogeneous non‐atmospheric, non‐seepage face Dirichlet (Neumann) BC's should contain, as above, a value of PTIM(QTIM) of 0.0 followed by the PINP(QINP) values, and then a value of PTIM(QINP) equal to or larger than TMAX(1.0e+15, say) followed by the same PINP (QINP)values specified at time 0.0. 2. If the first input time value is greater than 0.0, we set the initial (time 0.0) non‐atmospheric, non‐seepage face Dirichlet (Neumann) BC inputs to 0.0 3. If TIME is larger than the last PTIM (QTIM) value on unit IIN8 (IIN9), HTIDIR (HTINEU) is set to 1 and the last input values are used for the rest of the simulation. To properly handle the case where the datasets for different simulations are kept in the same file (separated by blank lines), follow the procedure described in (a). |
| SFQ (NSF,NNSFMX) | Back‐calculated flux values at actual seepage face nodes at current time level |
| SFQP(NSF,NNSFMX) | SFQ values at previous time level |
| ARENOD(NNOD) | Area assigned to each surface node (needed for conversion of atmospheric rainfall/evaporation rates to volumetric fluxes) |
| ATMPOT(NNOD) | Precipitation (+ve) / evaporation (‐ve) fluxes at current time level for each surface node. These are potential infiltration/exfiltration values. |
| ATMACT(NNOD) | Actual fluxes (infiltration/exfiltration values) for atmospheric boundary condition nodes at current time level.  For IFATM (I) =1, ATMACT (I) = back‐calculated flux value;  For IFATM (I) =‐1, ATMACT (I) is disregarded. |
| ATMOLD(NNOD) | ATMACT values at previous time level |
| ATMTIM(3) | Most current input time values for atmospheric BC's, with ATMTIM(1) < ATMTIM(2) < ATMTIM(3) and ATMTIM(2) < TIME <= ATMTIM(3) |
| ATMINP(3,NNOD) | Iinput atmospheric rainfall/evaporation rates corresponding to ATMTIM times. ATMPOT (I) is obtained from ATMINP (2,I) and ATMINP(3,I) by linear interpolation and conversion of rate to volumetric flux. ATMINP(1,I) values are needed in the event that, after back‐stepping, we have ATMTIM(1) < TIME <= ATMTIM(2) |
| EFFTIM(2) | Most current input time values for effective rainfall inputs, with EFFTIM (1) < EFFTIM(2) and EFFTIM(1) < TIME <= EFFTIM(2). |
| Notes: | 1. For a simulation using temporally homogeneous atmospheric rates, input data on unit IIN6 should contain a single value of ATMTIM (0.0) and a single set of ATMINP data. Alternatively, to properly handle the case where the datasets for different simulations are kept in the same file (separated by blank lines), the input data on unit IIN6 for temporally homogeneous rates should contain, as above, a value of ATMTIM of 0.0 followed by the ATMINP rates, and then a value of ATMTIM equal to or larger than TMAX (1.0e+15, say) followed by the same ATMINP rates specified at time 0.0. 2. If there is no ATMTIM, ATMINP input, HTIATM is set to 1 (homogeneous in time) and atmospheric input rates are set to 0.0. 3. If the first input time value is greater than 0.0, we see the initial (time 0.0) atmospheric input rates to 0.0. 4. If TIME is larger than the last ATMTIM value on unitIIN6, HTIATM is set to 1 and the last input atmospheric rates are used for the rest of the simulation. To properly handle the case where the datasets for different simulations are kept in the same file (separated by blank lines), follow the procedure described in (a). 5. If HSPATM is nonzero and not equal to 9999 (spatially homogeneous), each set of ATMINP data should consist of a single value which gets copied to all surface nodes. If HSPATM is zero (spatially variable), each set of ATMINP data should consist of NNOD values (note that we read in a value for each surface node, including surface nodes which may be designated as non‐atmospheric Dirichlet or Neumann boundary conditions. IFATM controls whether the atmospheric input for a given surface node is actually used). |

Table 24. Exchange Variables between FLOW3D and SURF\_ROUTE

|  |  |
| --- | --- |
| **ID** | **Description** |
| OVFLNOD (NNOD) | node‐wise overland flux produced by FLOW3D |
| OVFLP (NNOD) | node‐wise overland flux at previous time level |
| PONDNOD (NNOD) | node‐wise ponding pressure head produced by SURF\_ROUTE |
| OVFLCEL (NCELL) | cell‐wise overland flux produced by FLOW3D |
| PONDCEL (NCELL) | cell‐wise ponding pressure head produced by SURF\_ROUTE |
| ACTCEL (NCELL) | cell‐wise actual flux from subsurface (ATMACT transferred to the cells |
| CELLCOARSE(NCELL) | ... can probably be eliminated ... |

Table 25. Real Arrays for SURF\_ROUTE

|  |  |
| --- | --- |
| **ID** | **Description** |
| DEM\_MAP (NROW,NCOL) | Raster map of the catchment surface. Each value is the elevation of the corresponding cell. |
| DTM\_W\_1 (NCOL,NROW) | Raster map of the weights. Each value is the weight assigned to the cardinal drainage direction. |
| DTM\_W\_2 (NCOL,NROW) | Raster map of the weights. Each value is the weight assigned to the diagonal drainage direction. |
| DTM\_LOCAL\_SLOPE\_1 (NCOL,NROW) | Raster map of local slopes. Each value is the local slope assigned to the cardinal drainage direction. |
| DTM\_LOCAL\_SLOPE\_2 (NCOL,NROW) | Raster map of local slopes. Each value is the local slope assigned to the diagonal drainage direction. |
| DTM\_EPL\_1 (NCOL,NROW) | Raster map of elemental path length. Each value is the length of the elemental path assigned to the cardinal drainage direction. |
| DTM\_EPL\_2 (NCOL,NROW) | Raster map of elemental path length. Each value is the length of the elemental path assigned to the diagonal drainage direction. |
| DTM\_KSS1\_SF\_1 (NCOL,NROW) | Raster map of roughness coefficient. Each value is the roughness coefficient assigned to the cardinal drainage direction. |
| DTM\_KSS1\_SF\_2 (NCOL,NROW) | Raster map of roughness coefficient. Each value is the roughness coefficient assigned to the diagonal drainage direction. |
| DTM\_WS1\_SF\_1 (NCOL,NROW) | Raster map of surface water width. Each value is the surface water width of the cells scaled in space and assigned to the cardinal drainage direction. |
| DTM\_WS1\_SF\_2 (NCOL,NROW) | Raster map of surface water width. Each value is the surface water width of the cells scaled in space and assigned to the diagonal drainage direction. |
| DTM\_B1\_SF (NCOL,NROW) | Raster map of exponent of the at‐a‐station relationship for the water surface width (i.e., surface water width is scaled in time with a power‐law function with that exponent). |
| DTM\_Y1\_SF (NCOL,NROW) | Raster map of exponent of the at‐a‐station relationship for the roughness coefficient(i.e., roughness coefficient is scaled in time with a power‐law function with that exponent) |
| DTM\_NRC (NCOL,NROW) | Raster map of containing the number of rivulets per hillslope cell. |

Table 26. Real Arrays for Nudging and EnKF/SIR

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDTIM (NUDT) | Observation times |
| ENKFTIM(ENKFT) | Observation times for EnKF/SIR |
| NUDVAL (NUDC,NUDN) | Soil moisture content observation values |
| ENKFVAL(ENKFT,NOBS) | Soil moisture content or pressure head or streamflow or ERT observation values (see ENKNOD) in EnKF/SIR |
| EN\_ERT(NERT,NENS) | ERT measures associated with the ensemble |
| NUDSMC (NUDN) | Computed soil moisture or pressure head values, interpolated to the observation points |
| NUDTAU (NUDT,NUDN) | Half period of the temporal influence window for the Cressman‐type weighting function W(t) or temporal integral scale of the correlation function W(t) |
| NUDRXY (NUDN) | Horizontal radius of influence for the Cressman‐type weighting function W(x,y) or Gaussian correlation function W(x,y) |
| NUDRZ (NUDN) | Vertical radius of influence for the Cressman‐type weighting function W(z) or Gaussian correlation function W(z) |

(Continuing from Table 26)

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDEPS (NUDN) | Quality factor "epsilon" of the observation data |
| NUDX (NUDN) | x‐coordinates of the observation data points |
| NUDY (NUDN) | y‐coordinates of the observation data points |
| NUDZ (NUDN) | z‐coordinates of the observation data points |
| NUDDIF (NUDN) | Dynamical (Newtonian) relaxation or nudging term without the weighting function contribution (i.e., NUDG \* NUDEPS \* (NUDVAL - NUDSMC)) |
| NUDNOD (N) | Total nudging term contribution (with weighting functions) to the RHS system vector (units L^3/T) |
| NUDCUM (N) | Cumulative (over time) NUDNOD |
| ENDTM\_KSS1\_SF\_1/2 (NCOL,NROW, NENS) | Store the corresponding DTM\_KSS1\_SF\_1 and DTM\_KSS1\_SF\_2 for each ensemble realization. |
| ENDTM\_WS1\_SF\_1/2 (NCOL,NROW, NENS) | Store the corresponding DTM\_WS1\_SF\_1 and DTM\_WS1\_SF\_2 for each ensemble realization. |
| WSIR(NENS) | Weights of the particles in the ENKF/SIR schemes. For ENKF, WSIR (i)=1/NENS. |
| ENH (NOBS,N) | Stores the matrix H of the EnKF scheme |
| ENRETC (3,NENS) | Stores the retention curves parameters VGN, VGRMC, and VGPSAT, respectively, for eachn ensemble realization in the EnKF/SIR scheme. |
| ENDSRETC(3) | Coefficient of variation of VGN,VGRMC,VGPSAT |
| ENKSX (NSTR,NZONE, NENS) ENKSX (NSTR,NZONE,NENS)  ENKSX (NSTR,NZONE,NENS)  ENPOROS (NSTR,NZONE,NENS)  ENELSTOR (NSTR,NZONE,NENS) | Store values of PERMX, PERMY, PERMZ, ELSTOR and POROS for each realization of the ensemble in the EnKF/SIR scheme. |
| PERMX0 (NSTR,NZONE)  PERMY0 (NSTR,NZONE)  PERMZ0 (NSTR,NZONE)  ELSTOR0 (NSTR,NZONE)  POROS0 (NSTR,NZONE) | Ensemble average PERMX,PERMY, PERMZ,ELSTOR, POROS |
| ENDSKS(NSTR,NZONE, 3) | Ensemble coefficient of variation of KSX,KSY,KSZ |
| ENDSSTOR(NSTR,NZONE) | Ensemble coefficient of variation of ELSTOR |
| ENDSPOROS(NSTR,NZONE) | Ensemble coefficient of variation of POROS |
| ENSNODI(N,NENS) ENPNODI(N,NENS) | Store porosity and specific storage at each node for each realization of the ensemble in the EnKF/SIR scheme. |
| ENPTIMEP(N,NENS)  ENPOLD (N,NENS)  ENPNEW (N,NENS)  ENPTOLD(N,NENS)  ENPTNEW(N,NENS) | Store values of PTIMEP, POLD, PNEW, PTOLD, and PTNEW (see description below) for each realization of the ensemble in the EnKF/SIR scheme. |
| ENATMOLD(NNOD,NENS)  ENATMACT(NNOD,NENS)  ENATMPOT(NNOD,NENS) | Store values of ATMOLD, ATMACT and ATMPOT for each realization of the ensemble in the EnKF/SIR scheme. |
| ENOVFLNOD(N,NENS) ENOVFLP(N,NENS) ENPONDNOD(N,NENS) | Store values of OVFLNOD, OVFLP, and PONDNOD (see description above) for each realization of the ensemble in the EnKF/SIR scheme. |
| QNEW(NENS)  QTIMEP(NENS) | Time correlated observation errors for the atmospheric input at the current and previous time step, respectively. |
| ENDELTAT(NENS) | Store the DELTAT of each realization at the output/update times; it is initialize with the initial DELTAT value. |
| ENDTBIG(NENS) | Store the DTBIG for each realization |
| ENTBIG(NENS) | Store the TBIG for each realization |
| ENDTSMAL(NENS) | Store the DTSMAL for each realization |
| ENDTAVG(NENS) | Store the DTAVG for each realization |
| ENTSMAL(NENS) | Store the TSMAL for each realization |
| TIMEOUT(ENNOUT) | Store the time output time for EnKF/SIR |
| ENTIMEOUT(ENNOUT, NENS) ENOUTFLOW(ENNOUT, NENS) | Store the output time of each realization store the flow of each realization at the outlet at the time ENTIMEOUT. |
| ENOUTVOL(ENNOUT, NENS) | Store the volume of each realization at the time ENTIMEOUT |
| ENOUTATMPOT(ENNOUT, NENS) | Store the atmospheric potential of each realization at the time ENTIMEOUT |

(Continuing from Table 26)

|  |  |
| --- | --- |
| **ID** | **Description** |
| ENSTORE\_SAT(NENS) | Saturation volume of each realization |
| DSMEAS(NUDT) | Coefficients of variation of the measurements for each update. |

Table 27. Real Arrays for Material Properties and Hydraulic Characteristics

|  |  |
| --- | --- |
| **ID** | **Description** |
| PNEW (N) | Pressure heads at current time level, current nonlinear iteration |
| POLD (N) | Pressure heads at previous nonlinear iteration |
| PDIFF (N) | Difference in pressure heads between nonlinear iterations |
| PTNEW (N) | Weighted pressure heads at current nonlinear iteration |
| PTOLD (N) | Weighted pressure heads at previous nonlinear iteration |
| PTIMEP(N) | Pressure heads at previous time level (initial conditions on input; initial guess for steady state case) |
| PTIME0(N) | Initial conditions stored for the calculation of swelling/shrinkage in saturated zone |
| UNOD (N) | Darcy velocity ‐x at each node |
| VNOD (N) | Darcy velocity‐y at each node |
| WNOD (N) | Darcy velocity‐z at each node |
| UU (NT) | Darcy velocity‐x for each element |
| VV (NT) | Darcy velocity‐y for each element |
| WW (NT) | Darcy velocity‐z for each element |
| RECNOD(NNOD) | Recharge flux computed at nodes immediately above the water table, reconducted to surface nodes. |
| TIMPR(NPRT) | Time values for detailed output. Detailed output is produced at initial conditions (TIME=0), at time values indicated in TIMPRT, and at the end of the simulation (TIME=TMAX). Detailed output consists of: values of pressure head, velocity, water saturation, and relative conductivity (depending on setting of IPRT) at all nodes; velocity, and water saturation (depending on setting of IPRT) at all elements; vertical profiles of pressure head, water saturation, and relative conductivity for the NODVP surface nodes; pressure head, water saturation, and SATSUR values at the surface nodes. |

Table 28. Real Arrays for System Components

|  |  |
| --- | --- |
| **ID** | **Description** |
| AI (4,NT) | Coefficients 'a‐i / 6' of the basis functions |
| BI (4,NT) | Coefficients 'b‐i / 6' of the basis functions |
| CI (4,NT) | Coefficients 'c‐i / 6' of the basis functions |
| DI (4,NT) | Coefficients 'd‐i / 6' of the basis functions |
| LMASS (4,4) | Local mass matrix, without the storage coefficient ETAE and without the volume term |
| COEF1 (NTERM) | Global stiffness matrix; also used to store the LHS system matrix, which is the Jacobian in the Newton case |
| COEF2 (NTERM) | Global mass matrix |
| COEF3 (NTERM) | Derivative term components of the Jacobian for Newton scheme; also used as a scratch vector |
| SCR (NTERM) | Scratch vector |
| RNSYM (IBOT) | Real scratch vector for NONSYM solver |
| TNOTI (N) | RHS system vector |
| BKCFLOW\_NODE (N) | Back calculated nodal flow |
| XT5 (N) | TNOTI before imposition of Dirichlet boundary conditions (needed for back‐calculation of fluxes used in mass balance calculations) |
| LHSP (NP) | Values of diagonal elements of LHS system matrix corresponding to non‐atmospheric, non‐seepage face Dirichlet nodes before imposition of Dirichlet BC's (needed for back‐calculation of fluxes used in mass balance calculations) |
| LHSSF (NSF,NNSFMX) | Values of diagonal elements of LHS system matrix corresponding to seepage face Dirichlet nodes before imposition of Dirichlet BC's (needed for back‐calculation of fluxes used in mass balance calculations, and for calculation of new position of the exit point along each seepage face) |

(Continuing from Table 28)

|  |  |
| --- | --- |
| **ID** | **Description** |
| LHSATM(NNOD) | Values of diagonal elements of LHS system matrix corresponding to atmospheric Dirichlet nodes before imposition of Dirichlet BC's (needed for back‐calculation of fluxes used in mass balance calculations, and for switching control of atmospheric BC's) |

Table 29. Real Scalars for Mass Balance

(Defined in common block and Hydrograph Calculations include file MB\_HGRAPH.H)

|  |  |
| --- | --- |
| **ID** | **Description** |
| NUDIN | Total "inflow" flux contribution from the nudging term |
| NUDOUT | Total "outflow" flux contribution from the nudging term |
| NUDINP | NUDIN at previous time level |
| NUDOUTP | NUDOUT at previous time level |
| VNUDIN | Total "inflow" volu from nudging term over curr time step |
| VNUDOUT | Total "outflow" volu from nudging term over curr time step |
| VNUDTOT | Cumulative (over all time steps) nudging term "flow" volume VNUDIN + VNUDOUT |
| OVFLOW | Total overland flow (surface runoff) flux produced at atmospheric surface nodes. Overland flow occurs during rainfall periods when the actual flux is less than the potential flux, and accounts for both Horton and Dunne saturation mechanisms. |
| REFLOW | Total return flow flux produced at atmospheric surface nodes. Return flow occurs during rainfall periods when the actual flux is negative (outflow rather than inflow).In this case all of the potential flux becomes overland flow, and the magnitude of the actual flux becomes the return flow component of surface runoff. |
| RECFLOW | Total recharge flux produced at nodes immediately above the water table. We consider recharge only when the Darcy velocity in the z‐direction is negative [L^3/T]. |
| RECVOL | Same as RECFLOW, but in terms of cumulated volume [L^3]. |
| SFFLW | Total subsurface flow flux produced at seepage faces at the current time level |
| SFFLWP | Total subsurface flow flux produced at seepage faces at the previous time level |
| VSFFLW | Total subsurface flow volume produced at seepage faces between current and previous time levels |
| APOT | Total atmospheric potential flux at the current time level, used for hydrograph output. Note that we disregard contribution of non‐atmospheric, non‐seepage face surface nodes in the calculation of APOT. |
| AACT | Total atmospheric actual flux at the current time level, used for hydrograph output.  AACT=ADIN+ADOUT+ANIN+ANOUT |
| AACTP | AACT at the previous time level |
| AACTAV | Average AACT between the current and previous time levels |
| ADIN | Tot inflow flux from atmosph Dir nodes at curr time level |
| ADOUT | Tot outflow flux from atmosph Dir nodes at curr time level |
| ADINP | Tot inflow flux from atmosph Dir nodes at prev time level |
| ADOUTP | Tot outflow flux from atmosph Dir nodes at prev time level |
| NDIN | Tot inflow flux from na, nsf Dir nodes at curr time level |
| NDOUT | Tot outflow flux from na, nsf Dir nodes at curr time level |
| NDINP | Tot inflow flux from na, nsf Dir nodes at prev time level |
| NDOUTP | Tot outflow flux from na, nsf Dir nodes at prev time level |
| ANIN | Tot inflow flux from atmosph Neu nodes at curr time level |
| ANOUT | Tot outflow flux from atmosph Neu nodes at curr time level |
| ANINP | Tot inflow flux from atmosph Neu nodes at prev time level |
| ANOUTP | Tot outflow flux from atmosph Neu nodes at prev time level |
| NNIN | Tot inflow flux from na, nsf Neu nodes at curr time level |
| NNOUT | Tot outflow flux from na, nsf Neu nodes at curr time level |
| NNINP | Tot inflow flux from na, nsf Neu nodes at prev time level |
| NNOUTP | Tot outflow flux from na, nsf Neu nodes at prev time level |
| VADIN | Tot inflow volu from atmosph Dir nodes over curr time step |
| VADOUT | Tot outflow volu from atmosph Dir nodes over curr time step |
| VNDIN | Tot inflow volu from na, nsf Dir nodes over curr time step |
| VNDOUT | Tot outflow volu from na, nsf Dir nodes over curr time step |

(Continuing from Table 29)

|  |  |
| --- | --- |
| **ID** | **Description** |
| VANIN | Tot inflow volu from atmosph Neu nodes over curr time step |
| VANOUT | Tot outflow volu from atmosph Neu nodes over curr time step |
| VNNIN | Tot inflow volu from na, nsf Neu nodes over curr time step |
| VNNOUT | Tot outflow volu from na, nsf Neu nodes over curr time stepnon‐atmospheric‐‐‐^ ^‐‐‐non‐seepage face |
| VIN | VADIN + VNDIN + VANIN + VNNIN + VNUDIN = total inflow volume between current and previous time levels (> 0) |
| VOUT | VADOUT + VNDOUT + VANOUT + VNNOUT + VSFFLW + VNUDOUT =total outflow volume between current and previous time levels (< 0) |
| VAPOT\_T | Cumulative (over all time steps) atmospheric potential flux volume |
| VAACT\_T | Cumulative (over all time steps) atmospheric actual flux volume |
| VSFTOT | Cumulative (over all time steps) seepage face flow volume |
| VSFFLWVNDTOT | Cumulative (over all time steps) non‐atmospheric, on‐seepage face Dirichlet flow volume VNDIN + VNDOUT |
| VNNTOT | Cumulative (over all time steps) non‐atmospheric, on‐seepage face Neumann flow volume VNNIN + VNNOUT |
| CVIN | Cumulative (over all time steps) net inflow volume VIN |
| CVOUT | Cumulative (over all time steps) net outflow volume VOUT |
| VTOT | Cumulative (over all time steps) total net flow volume VIN + VOUT |
| STORE1 | Volume of water in the subsurface at each time step calculated by integrating moisture content (SW \* PNODI)over the entire domain |
| STORE0 | Initial (time 0) volume of water in the subsurface |
| STORE2 | Volume of water in the subsurface at each time step calculated by accumulating to STORE0 the current value of DSTORE |
| DSTORE | Total volume of storage change between current and previous time levels (> 0 for net increase in storage) |
| CDSTOR | Cumulative (over all time steps) net storage change DSTORE (> 0 for an overall global net increase in storage) |
| ERRAS | Volume ("mass") balance error over the current time step (> 0 for (VIN + VOUT) > DSTORE) |
| CERRAS | Cumulative (over all time steps) mass balance error ERRAS |
| CAERAS | Cumulative (over all time steps) absolute value of the mass balance error ERRAS |
| ERREL | Relative (percent) mass balance error over the current time step |

Table 30. Integer and Real Arrays for Calculation of Residual and Difference Norms

(Defined in common block include file NORMVL.H used in subroutines FLOW3D and CONVER)

|  |  |
| --- | --- |
| **ID** | **Description** |
| ITUMAX(ITUNS) | Nodes with largest pressure head difference in absolute value between current and previous nonlinear iterations for each FLOW3D nonlinear iteration |
| PIKMXV(ITUNS) | Values of pressure head difference at node with largest pressure head difference in absolute value between current and previous iterations (for bothFLOW3D and coupled FLOW3D/SURF\_ROUTE) |
| PCURRV(ITUNS) | "current iteration" pressure head value used in the calculation of PIKMXV(ITUNS) values for each nonlinear iteration |
| PPREVV(ITUNS) | "previous iteration" pressure head value used in the calculation of PIKMXV(ITUNS) values for each nonlinear iteration |
| PL2V (ITUNS) | Values of the square root of the sum of squares of pressure head differences over all nodes (i.e., L2norm of the convergence error), used in comparison with TOLUNS for convergence test in the case L2NORMnonzero, for each nonlinear iteration |
| FINFV (ITUNS) | values of residual error in the nonlinear FLOW3Dsolution calculated using the infinity norm (for the nonlinear system f(x)=0, the residual error at iteration "m" is the norm of f(x^m)) for each nonlinear iteration |
| FL2V (ITUNS) | values of residual error in the nonlinear FLOW3Dsolution calculated using the L2 norm for each nonlinear iteration |

Table 31. Real Scalars and Real Arrays for Unsaturated Soil Characteristics and for Chord and Tangent Slope Formulas

(Defined in common block include file SOILCHAR.H)

|  |  |
| --- | --- |
| **ID** | **Description** |
| PMIN units  [m sec] | 'air dry' pressure head value (for switching control of atmospheric boundary conditions during evaporation) |
| CBETA0, THETA0, CANG | Parameters for Camporese adaptation of Pyatt and John relation for peat soil deformation |
| VGN,  (VGM), VGRMC, VGPSAT | Parameters for van Genuchten and extended van Genuchten moisture curves (other 'VG' parameters ‐ specific storage, porosity, and VGPNOT ‐ are assigned nodally). VGM is derived from VGN. VGRMC is residual moisture content. For IVGHU=0, VGPNOT is (porosity ‐ VGRMC)/porosity, or (1 ‐ residual water saturation). For IVGHU=1, VGPNOT is a continuity parameter, derived by imposing a continuity requirement on the derivative of moisture content with respect to pressure head. |
| SMCREF, SMCWLT, SMCANA | Parameters for root water uptake module. SMCREF is field capacity, SMCWLT is wilting point, SMCANA is anaerobiosis point.  SMCANA 🡪 Read only if IVGHU=0 |
| HUN,  HUA,  HUB, HUALFA, HUBETA, HUGAMA, HUPSIA, HUSWR | Parameters for moisture curves from Huyakorn et al (WRR 20(8) 1984, WRR 22(13) 1986) (other 'HU' parameters ‐ specific storage and porosity ‐ are assigned nodally). HUN is only used for IVGHU=2; HUA and HUB are only used HUGAMA, for IVGHU=3. HUSWR is residual water saturation, which is equivalent to residual moisture content/porosity. |
| BCBETA, BCRMC, BCPSAT | Parameters for Brooks-Corey moisture curves (other 'BC' parameters ‐ specific storage and porosity ‐ are assigned nodally). BCRMC is residual moisture content |
| TOLKSL | Tolerance for chord slope formula. Whenever the chord slope formula is to be applied (for KSLOPE=1 or 2 at every iteration and at all nodes; for KSLOPE=3 at those nodes whose pressure heads fall within given ranges), it is applied only if the absolute pressure head difference (between the current and previous nonlinear iterations) is larger than TOLKSL. If the difference is smaller than TOLKSL, then differentiation is done either analytically (KSLOPE=1, 3) or with a centered difference formula (KSLOPE=2). |
| PKRL, PKRR | Left and right endpoints of the pressure head range within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the derivative of relative hydraulic conductivity |
| PSEL, PSER | Left and right endpoints of the pressure head range within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the derivative of effective saturation (moisture content for the case of extended van Genuchten curves, IVGHU=1) |
| PDSE1L PDSE1R PDSE2L PDSE2R | Left and right endpoints of the two pressure head ranges within which the chord slope (case KSLOPE=3) or tangent slope (case KSLOPE=4) formula is used to evaluate the second derivative of effective saturation (moisture content for the case of extended van Genuchten curves, IVGHU=1). (Two ranges are specified since in general d(Se)/dP is non‐monotonic.) |
| DKRTAN | Tangent slope approximation of d(Kr)/dP, the derivative of relative hydraulic conductivity Kr wrt to pressure head P. i.e., DKRTAN = (Kr(PKRR) ‐ Kr(PKRL))/(PKRR ‐ PKRL) |
| VGPNOT (N) | (porosity ‐ VGRMC)/porosity for van Genuchten curves (IVGHU=0); continuity parameter 'PNOT' for extended van Genuchten curves (IVGHU=1) |
| BCPORM (N) | (porosity ‐ BCRMC)/porosity for Brooks‐Corey curves (IVGHU=4 |
| DSETAN (N) | Tangent slope approximation of d(Se)/dP, the derivative of effective saturation Se (moisture content for case IVGHU=1) wrt to pressure head P. i.e., DSETAN = (Se(PSER) ‐ Se(PSEL)) /(PSER ‐ PSEL) |
| DDSE1T, DDSE2T (N) | Tangent slope approximations of dd(Se)/dPP, the second DDSE2T derivative of effective saturation Se (moisture content for case IVGHU=1) wrt to pressure head P.  i.e.,  DDSE1T = (DSe(PDSE1R) ‐ DSe(PDSE1L)) / (PDSE1R ‐ PDSE1L)  DDSE2T = (DSe(PDSE2R) ‐ DSe(PDSE2L)) /(PDSE2R ‐ PDSE2L)  where DSe is the derivative of Se.  (DSETAN, DDSE1T, and DDSE2T contain tangent slope values at each node only for the case IVGHU=1; for the other IVGHU cases the tangent slope values are constant for all nodes and are stored in DSETAN(1), DDSE1T(1), and DDSE2T(1).) |

Table 32. Real, Scalars and Real Arrays for Surface Water Routing and Watershed Characteristics

(Defined in common block include files RIVERNETWORK.H and SURFWATER.H)

|  |  |
| --- | --- |
| **ID** | **Description** |
| NORTH, SOUTH, EAST, WEST | DEM boundaries as given by GRASS |
| FACTOR | Multiplicative factor for DEM values (e.g. to change the units of the elevation) |
| DELTA\_X, DELTA\_Y | Cell dimensions (assumed equal!) |
| AK\_MAX | Maximum wave celerity |
| AK\_MAX\_ SAV | Variable defined to store AK\_MAX in case of back‐stepping |
| AK\_MAX\_P | Maximum wave celerity at previous FLOW3D time level |
| CUTRGT | Courant number "target" (defines the Courant routing module; usually = 1.0; a very large value, e.g. 1.E+40, suppresses the Courant criterion so that NSURF=1, i.e. DTSURF = DELTAT where DTSURF is the time step size for the surface routing module) |
| CU\_MAX | Maximum Courant number |
| SURFACE\_WATER\_INP (2,NCELL) | Input local contribution to surface runoff such as read in the effective rainfall input file (IIN22) |
| SURFACE\_WATER\_SN (NCELL) | SURFACE\_WATER\_INP multiplied by the cell area (DELTA\_X\*DELTA\_Y) |
| ELTRIA (NTRI) | Elevation assigned to each triangle of the cell (the two triangles of the same cell have the same cell elevation value) |
| H\_WATER\_KKP1\_SN (NCELL) | Surface water height calculated at the cell |
| Q\_IN\_KK\_SN (NCELL) | Discharge entering the cells previous SURF\_ROUTE time level |
| Q\_IN\_KK\_SN\_SAV (NCELL) | Variable defined to store Q\_IN\_KK\_SN in case of back‐stepping |
| Q\_IN\_KK\_SN\_P (NCELL) | Discharge entering the cells at previous FLOW3D time level |
| Q\_IN\_KKP1\_SN (NCELL) | Discharge entering the cells at actual time |
| Q\_OUT\_KK\_SN\_1 (NCELL) | Discharge exiting the cells along the cardinal direction at previous SURF\_ROUTE time level |
| Q\_OUT\_KK\_SN\_1\_SAV (NCELL) | Variable defined to store Q\_OUT\_KK\_SN\_1in case of back‐stepping |
| Q\_OUT\_KK\_SN\_1\_P (NCELL) | Discharge exiting the cells along the cardinal direction at previous FLOW3D time level |
| Q\_OUT\_KKP1\_SN\_1 (NCELL) | Discharge exiting the cells along the cardinal direction at actual time |
| Q\_OUT\_KK\_SN\_2 (NCELL) | Discharge exiting the cells along the diagonal direction at previous SURF\_ROUTE time level |
| Q\_OUT\_KK\_SN\_2\_SAV (NCELL) | Variable defined to store Q\_OUT\_KK\_SN\_2in case of back‐stepping |
| Q\_OUT\_KK\_SN\_2\_P (NCELL) | Discharge exiting the cells along the diagonal direction at previous FLOW3D time level |
| Q\_OUT\_KKP1\_SN\_2 (NCELL) | Discharge exiting the cells along the diagonal direction at actual time |
| Q\_OVERLAND (NCELL) | Lateral inflow |
| VOLUME\_KK\_SN (NCELL) | Volume of surface water at previous SURF\_ROUTE time level |
| VOLUME\_KK\_SN\_SAV (NCELL) | Variable defined to store VOLUME\_KK\_SN in case of back‐stepping |
| VOLUME\_KK\_SN\_P (NCELL) | Volume of surface water at previous FLOW3D time level |
| VOLUME\_KKP1\_SN (NCELL) | Volume of surface water at actual time |
| H\_POOL\_KK\_VEC (NUMRES) | Water levels in reservoirs at previous SURF\_ROUTE time level |
| H\_POOL\_KK\_VEC\_SAV (NUMRES) | Variable defined to store H\_POOL\_KK\_VEC in case of back‐stepping |
| H\_POOL\_KK\_VEC\_P (NUMRES) | Water levels in reservoirs at previous FLOW3D time level |
| H\_POOL\_KKP1\_VEC (NUMRES) | Water levels in reservoirs at actual time |
| HRES (NUMRES,30) | … |
| ARES (NUMRES,30) | … |
| H\_SOGLIA (NUMRES) | … |
| H\_FONDO (NUMRES) | … |

**References**

Montgomery DR and Dietrich WE (1988) Where do channels begin? Nature, 336(6196): 232-234.

O’Callaghan J and Mark DM (1984) The extraction of drainage networks from digital elevation data, Computer Vision, Graphics, and Image Processing, 28(3), 323-344.

Tarboton G, Bras RL and Rodriguez-Iturbe I (1988) The fractal nature of river networks. Water Resources Research, 24(8): 1317-1322.