TELEMAC-3D

User Manual

Version v8p5
December 1, 2023

Contents

1. Introduction

The TELEMAC-3D code solves such three-dimensional equations as the free surface flow equations (with or without the hydrostatic pressure hypothesis) and the transport-diffusion equations of intrinsic quantities (temperature, salinity, concentration). Its main results, at each point in the resolution mesh in 3D, are the velocity in all three directions and the concentrations of transported quantities. Water depth is the major result as regards the 2D surface mesh. The TELEMAC-3D's prominent applications can be found in free surface flow, in both seas and rivers; the software can take the following processes into account:

- Influence of temperature and/or salinity on density,
- Bottom friction,
- Influence of the Coriolis force,
- Influence of weather elements: air pressure, rain or evaporation and wind,
- Consideration of the thermal exchanges with the atmosphere,
- Sources and sinks for fluid moment within the flow domain,
- Simple or complex turbulence models (k-ε) taking the effects of the Archimedean force (buoyancy) into account,
- Dry areas in the computational domain: tidal flats,
- Current drift and diffusion of a tracer, with generation or disappearance terms,
- Oil spill modelling.

The code is applicable to many fields. The main ones are related to the marine environment through the investigations of currents being induced either by tides or density gradients, with or without the influence of such an external force as the wind or the air pressure. It can be applied either to large extent areas (on a sea scale) or to smaller domains (coasts and estuaries) for the impact of sewer effluents, the study of thermal plumes or even sedimentary transport. As regards the continental waters, the study of thermal plumes in rivers, the hydrodynamic behaviour or natural or man-made lakes can be mentioned as well.

TELEMAC-3D is developed by the LNHE (Laboratoire National d'Hydraulique et Environnement) of the Research and Development Division of EDF (EDF-R&D). As for previous releases, the 7.1 release of the code complies with the Quality Assurance procedures of scientific and technical softwares of EDF-R&D. It is a process of construction and verification of the product quality in the different phases of his life. In particular, a software following the Quality Assurance procedures comes with a Validation Folder that describes the intended use of the software and a set of test cases. This document allows you to judge the performance and limitations of the software, situating the field of application. These tests are also used in development of the software and are checked at every new release.

1.1 Position of the TELEMAC-3D code within the telemac modelling system

The TELEMAC-3D software is part of the TELEMAC modelling system developed by the LNHE of EDF R&D. TELEMAC is a set of modelling tools allowing to treat every aspects of natural free surface hydraulics: currents, waves, transport of tracers and sedimentology.
The pre-processing and post-processing of simulations can be done either directly within the TELEMAC system or with different software that present an interface of communication with the system. We can particularly mention the following tools:

- The FUDAA-PREPRO software, developed from the FUDAA platform by the CEREMA's Recherche, Informatique et Modélisation Department, covers all the pre-processing tasks involved by the achievement of a numerical hydraulic study, as well as a graphical post-processing tool,
- The Blue Kenue software, developed the Hydraulic Canadian Center, proposes a powerful mesh generation tool and a user-friendly post-processing tool,
- The Janet software, developed by Smile Consult GmbH, which offers among others, a mesh generation tool,
- The ParaView software, developed by Sandia National Laboratories, Los Alamos National Laboratory and Kitware, which enables to visualise 3D results, big data in particular and is open source,
- The SALOME-HYDRO software based on the SALOME platform, developed by EDF, CEA and OPENCASCADE which enables to handle raw data (bathymetry, maps, pictures, LIDAR...) until the mesh generation. The post-processing tool ParaViS available in the SALOME platform is based on the ParaView software and can visualise 1D, 2D or 3D results. A first release of SALOME-HYDRO has been available since Spring 2016,
- The Tecplot 360 software, developed by Tecplot which enables to visualise 2D and 3D results,
- The QGIS software, which is an open source Geographic Information System.

1.2 Software environment

All the simulation modules are written in FORTRAN 90, with no use of the specific language extensions in a given machine. They can be run on all the PCs (or PC "clusters") under Windows and Linux operating systems as well as on the workstations under the Unix operating system.

1.3 User programming

When using a simulation module from the TELEMAC system, the user may have to program specific subroutines which are not in the code's standard release. In particular, that is made through a number of so-called « user » subroutines. These subroutines are written so that they can be modified, provided that the user has a basic knowledge in FORTRAN language, with the help of the « Guide for programming in the Telemac system » [8].
The procedure to be carried out in that case comprises the steps of:

- Recovering the standard version of the user subroutine(s) as supplied in the distribution and copying it into the current directory,
- Amending the subroutine(s) according to the model to be constructed,
- Concatenating the whole set of subroutines into a single FORTRAN file or putting every file in one directory which, in both cases, will be compiled during the TELEMAC-3D launching process.

During that programming stage, the user can gain access to the various variables of the software through the FORTRAN 90 structures.
All the data structures are gathered within FORTRAN files, which are known as modules. For TELEMAC-3D, the file name is **DECLARATION_TELEMAC3D**. To gain access to the TELEMAC-3D data, just insert the command **USE DECLARATIONS_TELEMAC3D** into the beginning of the subroutine. Adding the command **USE BIEF** may also be necessary in order to reach the structure in the BIEF library.
Nearly all the arrays which are used by TELEMAC-3D are declared in the form of a structure. For example, the access to the water depth array will be in the form **H%R**, **%R** meaning it is a real-typed pointer. In case of an integer-typed pointer, the **%R** is replaced by a **%I**. However, in order to avoid having to handle too many **%R** and **%I**, a number of aliases are defined, such as, the **NPOIN3**, **NELEM3** and **NPTFR2** variables. For further details, the user can refer to the programming guide in TELEMAC [8].

2. Theoretical aspects

2.1 Notations

TELEMAC-3D is a three-dimensional computational code describing the 3D velocity field (U, V, W) and the water depth h (and, from the bottom depth, the free surface S) at each time step. Besides, it solves the transport of several tracers which can be grouped into two categories, namely the so-called "active" tracers (primarily temperature and salinity[1]), which change the water density and act on flow through gravity), and the so-called "passive" tracers which do not affect the flow and are merely transported.

2.2 Equations

The reader will refer to the J.-M. Hervouet's book [5] for a detailed statement of the theoretical aspects which TELEMAC-3D is based on.

2.2.1 Equation with the hydrostatic pressure hypothesis

In its basic release, the code solves the three-dimensional hydrodynamic equations with the following assumptions:

- Three-dimensional Navier-Stokes equations with a free surface changing in time,
- Negligible variation of density in the conservation of mass equation (incompressible fluid),
- Hydrostatic pressure hypothesis (that hypothesis results in that the pressure at a given depth is the sum of the air pressure at the fluid surface plus the weight of the overlying water body),
- Boussinesq approximation for the momentum (the density variations are only taken into account as buoyant forces).

Due to these assumptions, the three-dimensional equations being solved are:

[1] Sediment transport with TELEMAC-3D is not described in this manual.

$$\frac{\partial U}{\partial x}+\frac{\partial V}{\partial y}+\frac{\partial W}{\partial z}=0 \tag{2.1a}$$

$$\frac{\partial U}{\partial t}+U\frac{\partial U}{\partial x}+V\frac{\partial U}{\partial y}+W\frac{\partial U}{\partial z}=-g\frac{\partial Z_S}{\partial x}+\upsilon\,\Delta(U)+F_x \tag{2.1b}$$

$$\frac{\partial V}{\partial t}+U\frac{\partial V}{\partial x}+V\frac{\partial V}{\partial y}+W\frac{\partial V}{\partial z}=-g\frac{\partial Z_S}{\partial y}+\upsilon\,\Delta(V)+F_y \tag{2.1c}$$

$$p=p_{atm}+\rho_0 g\,(Z_s-z)+\rho_0 g\int_z^{Z_S}\frac{\Delta\rho}{\rho_0}dz' \tag{2.1d}$$

$$\frac{\partial T}{\partial t}+U\frac{\partial T}{\partial x}+V\frac{\partial T}{\partial y}+W\frac{\partial T}{\partial z}=\mathrm{Div}\,(\upsilon\,\mathbf{Grad}\,(T))+Q \tag{2.1e}$$

wherein:

- h (m) water depth,
- Z_S (m) free surface elevation,
- U, V, W (m/s) three-dimensional components of velocity,
- T (°C, g/L…) passive or active (acting on density) tracer,
- p (X) pressure,
- p_{atm} (X) atmospheric pressure,
- g (m/s^2) acceleration due to gravity,
- υ (m^2/s) cinematic viscosity and tracer diffusion coefficients,
- Z_f (m) bottom depth,
- r_0 (X) reference density,
- $\Delta\rho$ (X) variation of density around the reference density,
- t (s) time,
- x, y (m) horizontal space components,
- z (m) vertical space component,
- F_x, F_y (m/s^2) source terms,
- Q (tracer unit) tracer source of sink.
- h, U, V, W and T are the unknown quantities, also known as computational variables.
- F_x and F_y are source terms denoting the wind, the Coriolis force and the bottom friction (or any other process being modelled by similar formulas). Several tracers can be taken into account simultaneously. They can be of two different kinds, either active, i.e. influencing the flow by changing the density, or passive, without any effect on density and then on flow.

The TELEMAC-3D basic algorithm can be split up in three computational steps (three fractional steps).
The first step consists in finding out the advected velocity components by only solving the advection terms in the momentum equations.
The second step computes, from the advected velocities, the new velocity components taking into account the diffusion terms and the source terms in the momentum equations. These two solutions enable to obtain an intermediate velocity field.
The third step is provided for computing the water depth from the vertical integration of the continuity equation and the momentum equations only including the pressure-continuity terms (all the other terms have already been taken into account in the earlier two steps). The resulting two-dimensional equations (analogous to the Saint-Venant equations without diffusion, advection and source terms) are written as:

$$\frac{\partial h}{\partial t}+\frac{\partial (uh)}{\partial x}+\frac{\partial (vh)}{\partial y}=0 \tag{2.2a}$$

$$\frac{\partial u}{\partial t}=-g\frac{\partial Z_S}{\partial x} \tag{2.2b}$$

$$\frac{\partial v}{\partial t}=-g\frac{\partial Z_S}{\partial y} \tag{2.2c}$$

The u and v in lower case denote the two-dimensional variables of the vertically integrated velocity.
These two-dimensional equations are solved by the libraries in the TELEMAC-2D code and enable to obtain the vertically averaged velocity and the water depth.
The water depth makes it possible to recompute the elevations of the various mesh points and then those of the free surface.
Lastly, the computation of the U and V velocities is simply achieved through a combination of the equations linking the velocities. Finally, the vertical velocity W is computed from the continuity equation.

2.2.2 Non-hydrostatic Navier-Stokes equations

The following system (with an equation for W which is similar to those for U and V is then to be solved:

$$\frac{\partial U}{\partial x}+\frac{\partial V}{\partial y}+\frac{\partial W}{\partial z}=0 \tag{2.3a}$$

$$\frac{\partial U}{\partial t}+U\frac{\partial U}{\partial x}+V\frac{\partial U}{\partial y}+W\frac{\partial U}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\upsilon\,\Delta(U)+F_x \tag{2.3b}$$

$$\frac{\partial V}{\partial t}+U\frac{\partial V}{\partial x}+V\frac{\partial V}{\partial y}+W\frac{\partial V}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial y}+\upsilon\,\Delta(V)+F_y \tag{2.3c}$$

$$\frac{\partial W}{\partial t}+U\frac{\partial W}{\partial x}+V\frac{\partial W}{\partial y}+W\frac{\partial W}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial z}-g+\upsilon\,\Delta(W)+F_z \tag{2.3d}$$

In order to share a common core as much as possible with the solution of the equations with the hydrostatic pressure hypothesis, the pressure is split up into a hydrostatic pressure and a "dynamic" pressure term.

$$p=p_{atm}+\rho_0 g\,(Z_S-z)+\rho_0 g\int_z^{Z_S}\frac{\Delta\rho}{\rho_0}dz+p_d \tag{2.4}$$

The TELEMAC-3D algorithm solves a hydrostatic step which is the same as in the previous subsection, the only differences lying in the continuity step ("projection" step in which the dynamic pressure gradient changes the velocity field in order to provide the required zero divergence of velocity) and the computation of the free surface.

2.2.3 The law of state

Two laws of state can be used by default through TELEMAC-3D.
In most of the simulations, salinity and temperature make it possible to compute the variations of density. The first law expresses the variation of density from only these two parameters. The second law is more general and enables to construct all the variations of density with the active tracers being taken into account in the computation.
The **first law** is written as:

$$\rho = \rho_{ref}\left[1-\left(7\left(T-T_{ref}\right)^2-750S\right)10^{-6}\right] \tag{2.5}$$

With T_{ref} as a reference temperature of 4°C and ρ_{ref} as a reference density at that temperature when the salinity S is zero, then ρ_{ref} = 999.972 kg/m^3. That law remains valid for 0°C < T < 40°C and 0 g/L < S < 42 g/L.
The **second law** is written as:

$$\rho = \rho_{ref}\left[1-\sum_i \beta_i\left(T_i-T_i^0\right)_i\right] \tag{2.6}$$

ρ_{ref}, the reference density can be modified by the user together with the volumetric expansion coefficients β_i related to the tracers T_i.

2.2.4 k-ε model

The turbulent viscosity can be given by the user, as determined either from a mixing length model or from a k-ε model the equations of which are:

$$\frac{\partial k}{\partial t}+U\frac{\partial k}{\partial x}+V\frac{\partial k}{\partial y}+W\frac{\partial k}{\partial z}=\frac{\partial}{\partial x}\left(\frac{\nu_t}{\sigma_k}\frac{\partial k}{\partial x}\right)+\frac{\partial}{\partial y}\left(\frac{\nu_t}{\sigma_k}\frac{\partial k}{\partial y}\right)+\frac{\partial}{\partial z}\left(\frac{\nu_t}{\sigma_k}\frac{\partial k}{\partial z}\right)+P-G-\varepsilon \tag{2.7a}$$

$$\frac{\partial \varepsilon}{\partial t}+U\frac{\partial \varepsilon}{\partial x}+V\frac{\partial \varepsilon}{\partial y}+W\frac{\partial \varepsilon}{\partial z}=\frac{\partial}{\partial x}\left(\frac{\nu_t}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x}\right)+\frac{\partial}{\partial y}\left(\frac{\nu_t}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial y}\right)+\frac{\partial}{\partial z}\left(\frac{\nu_t}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial z}\right) \tag{2.7b}$$

$$+C_{l\varepsilon}\frac{\varepsilon}{k}\left[P+\left(1-C_{3\varepsilon}\right)G\right]-C_{2\varepsilon}\frac{\varepsilon^2}{k} \tag{2.7c}$$

wherein:

- $k=\frac{1}{2}\overline{u_i' u_i'}$ denotes the turbulent kinetic energy of the fluid,
- $u_i' = U_i - \overline{u_i}$ denotes the i^{th} component of the fluctuation of the velocity $\boldsymbol{U}(U,V,W)$,
- $\varepsilon = \upsilon\overline{\frac{\partial u_i'}{\partial x_j}\frac{\partial u_i'}{\partial x_j}}$ is the dissipation of turbulent kinetic energy,
- P is a turbulent energy production term,

- G is a source term due to the gravitational forces,

$$P = \nu_t \left(\frac{\partial \overline{U_i}}{\partial x_j} + \frac{\partial \overline{U_j}}{\partial x_i} \right) \frac{\partial \overline{U_i}}{\partial x_j}; G = -\frac{\nu_t}{\mathrm{Pr}_t} \frac{g}{\rho} \frac{\partial \rho}{\partial z} \tag{2.8}$$

  and ν_t verifies the equality: $\nu_t = C_\mu \frac{k^2}{\varepsilon}$,

- C_μ, Pr_t, $C_{1\varepsilon}$, $C_{2\varepsilon}$, $C_{3\varepsilon}$, σ_k, σ_ε are constants in the k-ε model.

2.2.5 Equations of tracers

The tracer can be either active (it affects hydrodynamics) or passive in TELEMAC-3D. Temperature, salinity and in some cases a sediment are active tracers. The tracer evolution equation is formulated as:

$$\frac{\partial T}{\partial t} + U\frac{\partial T}{\partial x} + V\frac{\partial T}{\partial y} + W\frac{\partial T}{\partial z} = \frac{\partial}{\partial x}\left(\nu_T \frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(\nu_T \frac{\partial T}{\partial y}\right) + \frac{\partial}{\partial z}\left(\nu_T \frac{\partial T}{\partial z}\right) + Q \tag{2.9}$$

with:

- T (tracer unit) tracer either passive or affecting the density,
- n_T (m^2/s) tracer diffusion coefficients,
- t (s) time,
- x, y, z (m) space components,
- Q (tracer(s) unit) tracer source or sink.

2.3 Mesh

2.3.1 The discretization

The TELEMAC-3D mesh structure is made of prisms (possibly split in tetrahedrons). In order to prepare that mesh of the 3D flow domain, a two-dimensional mesh comprising triangles which covers the computational domain (the bottom) in a plane is first constructed, as for TELEMAC-2D. The second step consists in duplicating that mesh along the vertical direction in a number of curved surfaces known as "planes". Between two such planes, the links between the meshed triangles make up prisms.
The computational variables are defined at each point of the three-dimensional mesh, inclusive of bottom and surface. Thus, they are "three-dimensional variables" except, however, for the water depth and the bottom depth which are obviously defined only once along a vertical. Thus, they are "two-dimensional variables". Some TELEMAC-3D actions are then shared with TELEMAC-2D and use the same libraries, such as the water depth computation library. Therefore, it is well understood that TELEMAC-3D should manage a couple of mesh structures: the first one is two-dimensional and is the same as that used by TELEMAC-2D, and the second one is three-dimensional. That implies managing two different numberings a detailed account of which is given below.

2.3.2 The two-dimensional mesh

The two-dimensional mesh, which is made of triangles, can be prepared using a mesh generator software compatible with the TELEMAC system (SALOME-HYDRO, Janet, Blue Kenue, MATISSE...).

Using a mesh generator that does not belong to the TELEMAC chain involves converting the resulting file, through the STBTEL interface for instance, to the SERAFIN format, which can be read by TELEMAC as well as by ParaView, FUDAA-PREPRO, Tecplot 360, Blue Kenue, RUBENS post-processors. In addition, STBTEL checks such things as the proper orientation of the local numbering of the mesh elements.

The two-dimensional mesh (included in the `GEOMETRY FILE`) consists of **NELEM2** elements and **NPOIN2** vertices of elements which are known through their X, Y, and Z_f co-ordinates (the BOTTOM variable). Each element is identified by a code known as **IELM2** and includes **NDP** nodes (3 for a triangle with a linear interpolation). The nodes on an element are identified by a local number ranging from 1 to **NDP**. The link between that element-wise numbering (local numbering) and the mesh node numbering ranging from 1 to **NPOIN2** (global numbering) is made through the connectivity table **IKLE2**. The global number of the IDPIDP local-number node in the **IELEM2** element is **IKLE2(IELEM2,IDP)**.

2.3.3 The three-dimensional mesh

The three-dimensional mesh, which is made of prisms, is automatically constructed by TELEMAC-3D from the previous mesh. The data in the three-dimensional mesh of finite elements are as follows:

- **NPOIN3**: the number of points in the mesh (NPOIN3 = NPOIN2 $\times$ NPLAN),
- **NELEM3**: the number of elements in the mesh,
- **NPLAN**: the number of planes in the mesh,
- **X, Y, Z: NPOIN3** dimensional arrays. **X** and **Y** are obtained by merely duplicating the above-described arrays of the two-dimensional mesh. Dimension Z obviously depends on the mesh construction being selected (keyword `MESH TRANSFORMATION`),
- `IKLE3`: dimensional arrays (`NELEM3,6`). `IKLE3(IELEM3,IDP)` provides with the global number of the `IDP` point in the `IELEM3` element. `IKLE3` defines a numbering of the 3D elements and a local numbering of the points in each element, it provides for the transition from that local numbering to the global numbering.

From these data, TELEMAC-3D constructs other arrays, such as the edge point global address array.

Figure 2.1 herein below illustrates a TELEMAC-3D three-dimensional mesh.

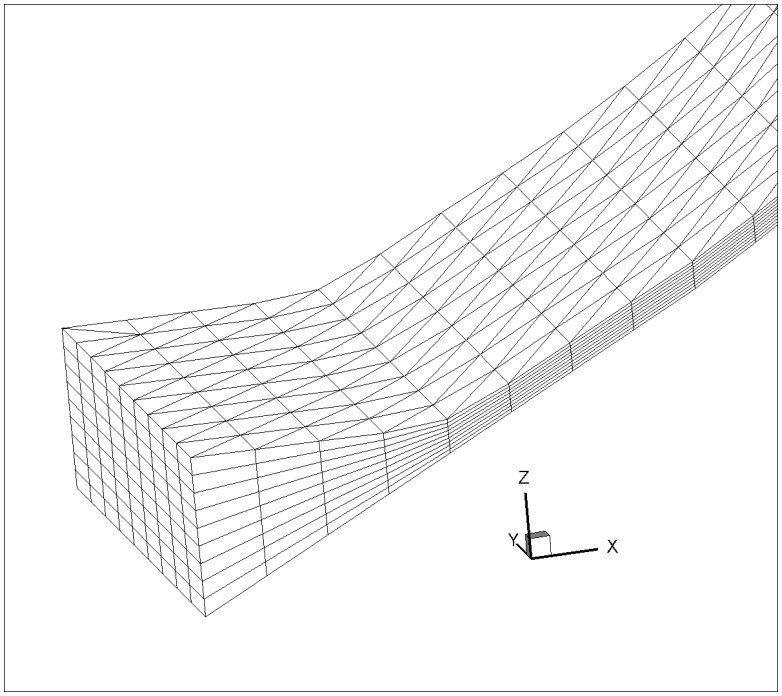

Figure 2.1: A three-dimensional mesh.

3. The inputs / outputs

3.1 Preliminary comments

In a computation, the TELEMAC-3D code uses a number of input and output files, some of which are optional. Most of these files are similar or identical to their counterparts in TELEMAC-2D.
The input files are:

- The steering file (mandatory), which contains the "configuration" of the simulation,
- The geometry file (mandatory), which contains the information regarding the mesh,
- The boundary conditions file (mandatory), which contains the description of the type of each boundary,
- The FORTRAN file, which contains the specific subroutines of the simulation (modified TELEMAC subroutine or specifically created),
- The bottom topography file, containing the elevation of the bottom. Generally, the topographic data are already available in the geometry file and the bottom topography file is generally not used,
- The liquid boundaries file, which contains the information on the imposed values at liquid boundaries,
- The previous computation file, which provides the initial state of the calculation in the case of a restart calculation,
- The reference file, which contains the calculation of "reference" used for the validation process,
- The stage-discharge curves file, which contains the information on the imposed values at liquid boundaries in case of height / flow rate law,
- The sources file, which contains the information regarding the sources,
- The binary boundary data file, which contains boundary conditions data varying in time and space,

- The time series coordinates file which contains coordinates and periods of time where time series are extracted and written in the 2D and 3D time series files,
- The sections input file, which contains the description of the control sections of the model (sections across which the flowrate is computed).

The output files are:

- The 3D result file, which contains the graphical results associated to the 3D mesh,
- The 2D result file, which contains the graphical results associated to the 2D mesh,
- The output listing, which is simulation report. In case of difficulty in performing a calculation, the user can request the printing of additional information by activating the logical keyword `DEBUGGER` (default = 0). `DEBUGGER = 1` provides the calling sequences of subroutines from the main program **TELEMAC3D**. This technique is useful in case of critical computation crash to identify the responsible subroutine,
- The restart file, allowing to perform a restart computation without information loss,
- The 2D and 3D time series files which contain time series at locations defined in the time series coordinates file,
- The sections output file, which contains the results of the "control sections" computation.

In addition, the user may have to manage additional files:

- The binary data files 1 and 2, in input,
- The formatted data files 1 and 2, in input,
- The binary results file, in output,
- The formatted results file, in output.

3.1.1 Binary files format

The binary files used within the TELEMAC system can have various formats. The most common format is the SERAFIN format (also known as SELAFIN, for misidentified historical reasons) which is the standard internal TELEMAC system format (described in Appendix C). This SERAFIN format can be configured so as to store the real-typed values in single or double precision. The other available format is the MED format which is used by the SALOME platform jointly developed by EDF and the French atomic energy commission (CEA). The full description of the MED format is available on the SALOME website http://www.salome-platform.org. Depending on the specified format, the binary files may be treated by different software. However, in the current release of TELEMAC, only the SERAFIN single precision format can be read by the post-processing tools like RUBENS or Blue Kenue. It is not the case for Paraview or FUDAA-PREPRO.
The selection of the format of a file is done by the corresponding keyword. Thereby, the keyword `GEOMETRY FILE FORMAT` specifies the format of the geometry file. Those keywords may take 3 different values (as an 8 character string): the value 'SERAFIN ' corresponds to the standard single precision SERAFIN format, which is the default and recommended value (do not forget the space in last position). The 'SERAFIND' value corresponds to the double precision SERAFIN format which allows to increase the precision of the results, especially in case of a

restart or reference file. Finally, the 'MED' value corresponds to the HDF5 MED format in double precision.
The keywords involved are:

- `2D RESULT FILE FORMAT`,
- `2D TIME SERIES FILE FORMAT`,
- `3D RESULT FILE FORMAT`,
- `3D TIME SERIES FILE FORMAT`,
- `BINARY ATMOSPHERIC DATA FILE FORMAT`,
- `BINARY DATA FILE 1 FORMAT`,
- `BINARY RESULTS FILE FORMAT`,
- `FILE FOR 2D CONTINUATION FORMAT`,
- `GEOMETRY FILE FORMAT`,
- `PREVIOUS COMPUTATION FILE FORMAT`,
- `REFERENCE FILE FORMAT`,
- `RESTART FILE FORMAT`,
- `TIDAL MODEL FILE FORMAT`.

3.2 The steering file

The steering file contains all the data about the selection of computational options (physical, numerical, etc.). It is an ASCII file which can be generated either through the FUDAA-PREPRO or EFICAS software or directly using a text editor. In a way, it serves as the computation dashboard. It includes a set of keywords to which values are assigned. If a keyword does not occur in that file, then TELEMAC-3D will assign it the default value as defined in the dictionary file. If such a default value is not defined in the dictionary, then the computation will be interrupted with an error message. For instance, the command `TIME STEP = 10.0` specifies that the computation time step value is 10 seconds.
TELEMAC-3D reads the steering file at the beginning of the computation.
Both dictionary file and steering file are read using the **damocles** library which is included in the TELEMAC chain. It is therefore necessary, when generating the steering file, to observe the **damocles** syntax rules (what is performed automatically if the file is generated using FUDAA-PREPRO).
The writing rules are as follows:

- The keywords can be of the Integer, Real, Logical or Character type,
- The sequence order in the steering file is of no importance,
- Several keywords can be on the same line,
- Each line cannot exceed 72 characters. However, one can start a new line as many times as one wishes provided that the keyword name is not astride two lines,

- For the array-types keywords, the character separating successive values is the semicolon. For example:

```
PRESCRIBED FLOWRATES =   10.0;20.0
```

- The symbols ":" and "=" can both be used in order to separate a keyword name and its value. They can be either preceded or followed by any number of blanks. The value itself can occur on the following line. For example:

```
TIME STEP =   10.
```

or

```
TIME STEP : 10.
```

or else

```
TIME STEP =
10
```

- The characters occurring between two "/" on one line are taken as comments. Likewise, those characters occurring between a "/" and a line ending are considered as comments. For example:

```
/ k-epsilon model
VERTICAL TURBULENCE MODEL = 3
```

- A line beginning with a "/" in a first column is wholly considered as a comment, even though there is another "/" on the line. For example:

```
/ The geometry file is ./mesh/geo
```

- Writing the integers: do not exceed the maximum size allowed by the machine (in a machine with a 32 bit architecture, the extreme values range from -2 147 483 647 to +2 147 483 648. Do not insert any blank between the sign (optional for the sign +) and the number. A point after the end of the number is tolerated,
- Writing the reals: the point and the comma are accepted as a decimal separator, as well as the FORTRAN formats E and D. (1.E-3 0.001 0,001 1.D-3 represent the same value),
- Writing the logical values: the values 1, OUI, YES, .TRUE., TRUE, or VRAI on the one hand, and 0, NON, NO, .FALSE., FALSE, or FAUX on the other hand, are accepted,
- Writing the character strings: the strings including blanks or reserved symbols ("/",":", "=", "&") should be inserted between quotes (’). The value of a character keyword may contain up to 144 characters. As in FORTRAN, the quotes included in a string should be doubled. A string may not begin or end with a blank. For example:

```
TITLE = 'COASTAL ENVIRONMENT STUDY'
```

In addition to the keywords, a number of pseudo instructions or metacommands which are interpreted during the sequential reading of the steering file can also be used:

- The `&FIN` command specifies the file end (even though the file is not over). Thus, some keywords can be disabled simply by placing them behind that command so that they can easily be reactivated later on. However, the computation keeps running,

- The &ETA command prints the list of keywords and the values which are assigned to them when **damocles** meets that command. That display will take place at the head of the output listing,
- The &LIS command prints the list of keywords. That display will take place at the head of the output listing,
- The &IND command prints the detailed list of keywords. That display will take place at the head of the output listing,
- The &STO command causes the program to be halted, whereas the computation does not keep running.

3.3 The geometry file

It is the same file as that used by TELEMAC-2D. This is a binary file. The format of this file is given by the keyword GEOMETRY FILE FORMAT. There are 3 possibilities:

- SERAFIN: classical single precision format in TELEMAC,
- SERAFIND: classical double precision format in TELEMAC,
- MED: MED double precision format based on HDF5.

If it is a SERAFIN-formatted binary file, it can be read by Paraview, FUDAA-PREPRO, Tecplot 360, Blue Kenue or RUBENS and is generated either by SALOME-HYDRO, Janet, Blue Kenue or MATISSE, or by the STBTEL module (from the file(s) originating from of mesh generator). The SERAFIN format structure is described in Appendix C.
If it is a MED-formatted binary file, it can be read by Paraview only and can be generated by SALOME-HYDRO or by the STBTEL module (from the file(s) originating from of mesh generator).
That file contains all the data about the two-dimensional mesh (see in subsection 2.3.2). It includes the number of points in the mesh (**NPOIN2** variable), the number of elements (**NELEM2** variable), the number of vertices per element (**NDP** variable), the **X** and **Y** arrays containing the co-ordinates of all the points and, lastly, the **IKLE** array containing the connectivity table.
That file may also contain bathymetric and bottom friction data for each point in the mesh.
NOTE: TELEMAC-3D retrieves the geometry data at the beginning of the 2D result file. Any computational 2D result file can therefore be used as a geometry file when a further simulation on the same mesh is desirable.
That file name is provided using the keyword: GEOMETRY FILE and its format is specified by the keyword GEOMETRY FILE FORMAT (default value: 'SERAFIN '), see subsection 3.1.1.

3.4 The boundary conditions file

It is the same file as that used by TELEMAC-2D. It is a formatted file which is generated automatically by most of the mesh generators compatible with TELEMAC or STBTEL and can be amended using FUDAA-PREPRO or a text editor. Each line in that file is dedicated to a point at the 2D mesh boundary. The edge point numbering is that of the file lines; it first describes the domain outline in the counter clockwise direction from the bottom left-hand point (point where the sum of horizontal co-ordinates of which is minimum), then the islands in the clockwise direction.
For a thorough description of that file, refer to the specific subsection 4.3.4.
That name of file is provided using the keyword: BOUNDARY CONDITIONS FILE.

3.5 The FORTRAN file

The FORTRAN file (a single file or a directory containing one or several file(s)) may include a number of subroutines (so-called "user" subroutines) available under the TELEMAC-3D tree structure which the user can modify as well as those subroutines which have been specifically developed for the computation.
The user subroutines from the various libraries used by TELEMAC-3D are listed in Appendix B. Since TELEMAC-3D is available in Open Source, every subroutine can be freely used. Every user subroutine copied into the user FORTRAN file is automatically substituted for the same named subroutine occurring in the TELEMAC-3D compiled libraries.
Upon the creation and every amendment of the FORTRAN file, a new executable program is generated (compilation and link) for the simulation.
That file name or directory name is provided using the keyword `FORTRAN FILE`.

3.6 The liquid boundaries file

It is an ASCII file enabling the user to specify time-varying boundary conditions values (flow rate, depth, velocity, tracer concentration) at all the liquid boundaries. That file can be generated under the FUDAA-PREPRO software interface.
For a thorough description of that file, refer to the specific subsection 4.3.8.
That name of file is provided using the keyword: `LIQUID BOUNDARIES FILE`.

3.7 The previous computation file

It is a TELEMAC-3D result file which is used for initializing a new computation. In order to activate the optional use of that file, the keyword `COMPUTATION CONTINUED` should be activated (default = NO). In order to specify the previous computation file, its name should be stated through the keyword: `PREVIOUS COMPUTATION FILE`. The initial conditions of the new computation are defined by one recorded time step in the previous computation file given by the keyword `RECORD NUMBER FOR RESTART`. By default, its value is -1 which means it is the last time step to start from in the `PREVIOUS COMPUTATION FILE`. The whole set of data from the steering file is read and it makes possible to redefine or amend the variables (time step, turbulence model, addition or deletion of a tracer...).
A computation can also be initialized from a TELEMAC-2D result. In order to activate that option, the `2D CONTINUATION` keyword should be validated (default = NO). The TELEMAC-2D result file should then be associated with the `FILE FOR 2D CONTINUATION FORMAT` keyword (default value: 'SERAFIN '), see subsection 3.1.1.

3.8 The reference file

During the validation step of a calculation, this file contains the reference result. At the end of the calculation, the result of the simulation is compared to the last time step stored in this file. The result of the comparison is given in the control printout in the form of a maximum difference in elevation and the three components of velocity.
The name of this file is given by the keyword `REFERENCE FILE` and its format is specified by the keyword `REFERENCE FILE FORMAT` (default value: 'SERAFIN '), see subsection 3.1.1.

3.9 The stage-discharge curves file

This text file enables the user to configure the evolution of the prescribed value on specific open boundaries. This file is used when the prescribed elevation is determined by a discharge-

elevation law. The descriptions of the appropriate laws are given through this file.
See subsection 4.3.10 for a complete description of this file.
The name of this file is specified with the keyword: STAGE-DISCHARGE CURVES FILE.

3.10 The sections input file

This ASCII file enables the user to configure the control sections used during the simulation.
See section 4.4 for a complete description of this file.
The file name is specified with the keyword: SECTIONS INPUT FILE.

3.11 The sources file

This text file enables the user to specify values for time-dependent conditions for sources (discharge, tracers concentration).
See section 5.4 for a complete description of this file.
The file name is specified with the keyword: SOURCES FILE.

3.12 The binary boundary data file

This binary file enables the user to specify boundary conditions data (e.g. elevations, horizontal components of velocity, tracer values) which may vary in time and space.
See the Python script **converter.py** in the scripts/python3 directory with the positional argument generate_bnd. It may help to generate such a file from a global model which will be interpolated on a geometry file for every liquid node given by the boundary conditions file.
The name of this file is given by the keyword BINARY BOUNDARY DATA FILE and its format is specified by the keyword BINARY BOUNDARY DATA FILE FORMAT (default value: 'SERAFIN '), see subsection 3.1.1.

Caution:
To use the BINARY BOUNDARY DATA FILE, some keywords have to be added in the steering file:

- PRESCRIBED ELEVATIONS with at least as many values as the number of liquid boundaries if the BINARY BOUNDARY DATA FILE contains elevations at the boundaries,
- PRESCRIBED VELOCITIES with as at least many values as the number of liquid boundaries if the BINARY BOUNDARY DATA FILE contains horizontal components of velocity at the boundaries. Moreover, the two velocity components U and V must both exist in the file,
- PRESCRIBED TRACERS VALUES with at least as many values as the number of liquid boundaries times the number of tracer(s) if the BINARY BOUNDARY DATA FILE contains tracer(s) values at the boundaries,

3.13 The 3D result file

It is the file into which TELEMAC-3D stores the information during the computation. It is a SERAFIN-formatted file (refer to Appendix C) or a MED-formatted file. It first contains all the data about mesh geometry, then the names of the stored variables. It also contains, for each graphic printout and for each mesh point, the values of the various recorded variables.

Its content varies according to the values of the following keywords:
`NUMBER OF FIRST TIME STEP FOR GRAPHIC PRINTOUTS`: provided to set from which time step onwards the information will be stored, in order to prevent too large files, especially when the computation begins with an uninteresting transient stage related to the definition of unrealistic initial conditions (e.g. invariably zero currents). The default value is 0 (writing of the graphic printouts at the beginning of the simulation).
`GRAPHIC PRINTOUT PERIOD`: sets the period (in number of time steps) of printouts in order to prevent a too large file. Default value is 1 (writing at every time step). For instance:

```
TIME STEP   =  60.0
GRAPHIC PRINTOUT PERIOD =  30
```

The results will be backed up every $1,800^{\text{th}}$ second, i.e. 30^{th} minute.
`VARIABLES FOR 3D GRAPHIC PRINTOUTS`: keyword specifying the list of variables which will be stored in the result file. Each variable is identified by means of a name from the list below.

- Z elevation (m),
- U velocity along x axis (m/s),
- V velocity along y axis (m/s),
- W velocity along z axis (m/s),
- TA concentrations for tracers (TA1 for the 1^{st} one, TA2 for the 2^{nd} one...),
- NUX viscosity for U and V along x axis (m^2/s),
- NUY viscosity for U and V along y axis (m^2/s),
- NUZ viscosity for U and V along z axis (m^2/s),
- NAX viscosity for tracers along x axis (m^2/s),
- NAY viscosity for tracers along y axis (m^2/s),
- NAZ viscosity for tracers along z axis (m^2/s),
- RI Richardson number in case of a mixing length model,
- K turbulent energy for k-ε model (J/kg),
- EPS dissipation of turbulent energy (W/kg),
- DP dynamic pressure (multiplied by DT/RHO),
- PH hydrostatic pressure (Pa),
- RHO relative density,
- P1 private variable 1,
- P2 private variable 2,
- P3 private variable 3,
- P4 private variable 4,

- US : Stokes velocity along x axis (m/s),
- VS : Stokes velocity along y axis (m/s),
- WS : Stokes velocity along z axis (m/s),
- CSO2 oxygen saturation concentration (mgO2/l) if coupling with WAQTEL and using O2 or EUTRO model,
- O2SAT percentage of oxygen saturation (ratio oxygen concentration over oxygen saturation concentration C_s in %) if coupling with WAQTEL and using O2 or EUTRO model.

That file name is provided using the keyword: `3D RESULT FILE` and its format using `3D RESULT FILE FORMAT` (default value: 'SERAFIN '). Stored variables by default are 'Z, U, V, W'.
Using the private arrays requires updating the keyword `NUMBER OF PRIVATE ARRAYS` in order to perform the required memory allocations (see section 11.4).

3.14 The 2D result file

It is the file into which TELEMAC-3D stores the specifically two-dimensional data during the computation (such as the free surface, the depth-averaged horizontal components of velocity and the depth-averaged tracers). It has a SERAFIN format. The free surface and the horizontal components of velocity will then physically correspond to the same data as those being supplied by TELEMAC-2D. The obtained values, however, may be different from an analogue computation being directly made with TELEMAC-2D when the flow is specifically three-dimensional.
Its content varies according to the values of the following keywords:
`NUMBER OF FIRST TIME STEP FOR GRAPHIC PRINTOUTS`: same keyword as that described in section 3.13.
`GRAPHIC PRINTOUT PERIOD`: same keyword as that described in section 3.13.
`VARIABLES FOR 2D GRAPHIC PRINTOUTS`: keyword specifying the list of variables which will be stored in the result file. Each variable is identified by means of a name from the list below.

- U average velocity along x axis (m/s),
- V average velocity along y axis (m/s),
- C celerity (m/s),
- H water depth (m),
- S free surface elevation (m),
- B bottom elevation (m),
- TA averaged concentrations for tracers (TA1 for the 1^{st} one, TA2 for the 2^{nd} one...),
- F Froude number,
- Q scalar discharge (m^2/s),
- I discharge along x (m^2/s),
- J discharge along y (m^2/s),

- M norm of velocity (m/s),
- X wind along x axis (m/s),
- Y wind along y axis (m/s),
- P air pressure (Pa),
- W friction coefficient,
- RB non erodible bottom elevation (m),
- HD thickness of the sediment bed layer (m),
- EF erosion rate (kg/m^2/s),
- DF deposition flux (kg/m^2/s),
- DZF bed evolution,
- PRIVE1 private array PRIVE 1,
- PRIVE2 private array PRIVE 2,
- PRIVE3 private array PRIVE 3,
- PRIVE4 private array PRIVE 4,
- QS solid discharge (m^2/s),
- QSX solid discharge along x (m^2/s),
- QSY solid discharge along y (m^2/s),
- US friction velocity (m/s),
- MAXZ maximum value of the water elevation during the computation (m),
- TMXZ time corresponding to this maximum elevation (s),
- TAIR air temperature (°C),
- USURF surface velocity along x axis (m/s),
- VSURF surface velocity along y axis (m/s),
- WSURF surface velocity along z axis (m/s),
- MSURF magnitude of surface velocity (m/s),
- TASURFi concentrations for tracers at the surface, i is the tracer number,
- TASURF* concentrations for tracers at the surface from 1 to 9,
- TASURF** concentrations for tracers at the surface from 10 to 99,
- CSO2 average oxygen saturation concentration (mgO2/l) if coupling with WAQTEL and using O2 or EUTRO model,

- O2SAT average percentage of oxygen saturation (ratio oxygen concentration over oxygen saturation concentration C_s in %) if coupling with WAQTEL and using O2 or EUTRO model.

That file name is provided by means of the keyword: `2D RESULT FILE` and its format using `2D RESULT FILE FORMAT` (default value: 'SERAFIN '). Stored variables by default are 'U, V, H, B'.
Using the private arrays requires updating the keyword `NUMBER OF 2D PRIVATE ARRAYS` in order to perform the required memory allocations (see section 11.4).
If the user is only interested in the values of velocities at the surface and/or tracer values at the surface rather than the whole 3D domain, he/she can save them in the `2D RESULT FILE` with the dedicated variables with SURF suffix. That enables to save memory in the `3D RESULT FILE`.

3.15 The time series files

Time series files are files (in SERAFIN or MED format), in which time series are written for certain points in the domain of the simulation. This feature has been introduced since release 8.5. In this way, one can obtain results with a high time frequency, without using a large amount of disk space. The time series file module interpolates data (using linear interpolation) to the required output coordinates.

- `2D TIME SERIES FILE` The name of the output file to write 2D results. The time and coordinates correspond to those given in `TIME SERIES COORDINATES FILE`,
- `3D TIME SERIES FILE` The name of the output file to write 3D results for time and coordinates given in `TIME SERIES COORDINATES FILE`,
- `2D TIME SERIES FILE FORMAT` The file format of `2D TIME SERIES FILE` (single or double precision SERAFIN, or MED, default = SERAFIN),
- `3D TIME SERIES FILE FORMAT` The file format of `3D TIME SERIES FILE` (single or double precision SERAFIN, or MED, default = SERAFIN),
- `TIME SERIES COORDINATES FILE` The file which contains the output times and output coordinates for 2D and 3D `TIME SERIES FILE`. The format is described next.

The `TIME SERIES COORDINATES FILE` has following format:

- On the first line, there are two numbers indicating:
  - The number of time periods for which data is written to the `TIME SERIES FILE`,
  - The number of coordinates for which output is generated.
- On the next lines, for each output period, there are three numbers indicating:
  - Start of the output period (in seconds since the start of the computation),
  - End of the output period (in seconds since the start of the computation),
  - Output time interval (in seconds).
- For each coordinate, three numbers and a string, containing respectively:
  - x-coordinate of the output point,

  - y-coordinate of the output point,
  - Unique ID of each station (in order to easily encounter the output data in the output files). This ID will be written to the IKLE array in the output file, which has a size [NPOINTS $\times$ 1]. Note that the time series file consists of a list of points. There is no mesh information in a time series file. Therefore, the length of the IKLE array is the same as the number of points in the time series file,
  - The name of the station (for better readability of the coordinate file).

- Anything after the list of coordinates is considered comment and ignored by TELEMAC.

An example of a `TIME SERIES COORDINATES FILE` is given below.

```
2 3
0 100 10
500 1000 20
250  175     01 output_point_01
500  175     02 output_point_01
750  175     03 output_point_02

Description of the file
1st line: number of periods and number of points
2nd line: period 1: start time, end time and interval (in seconds)
3rd line: period 2: start time, end time and interval (in seconds)
4th-6th line: output points; x and y coordinates, unique ID, and station name
```

3.16 Results in longitude-latitude

If the geometry file is given in longitude-latitude, the result files can also be written with the coordinates in longitude-latitude by setting the keyword `RESULT FILE IN LONGITUDE-LATITUDE` at YES (default value).

3.17 The output listing

It is a formatted file which can be created by TELEMAC-3D during the computation (program launched with the -s option). It contains the report of a TELEMAC-3D running. The default value of the keyword `LISTING PRINTOUT` is YES, which means the listing is written. If NO is selected, the listing only includes the heading and the phrase "NORMAL END OF PROGRAM". It is not recommended to change the default value. In addition, the options `MASS-BALANCE` and `VALIDATION` are inhibited.
Its contents vary according to the values of the following keywords:

- `NUMBER OF FIRST TIME STEP FOR LISTING PRINTOUTS`: provided to set at which time step it is desired to begin printing the data, in order to prevent too large files. The default value is 0 (writing of the listing printouts at the beginning of the simulation),
- `LISTING PRINTOUT PERIOD`: sets the period between two time step printings. The value is given as a time step number. By default, it is equal to 1, i.e. every time step. For instance, the following sequence:

```
TIME STEP = 30.0
LISTING PRINTOUT PERIOD = 2
```

prints the output listing every minute of simulation,

- `MASS-BALANCE`: if it is requested, the user will get information about the mass flow (or rather the volumes) and the errors (primarily linked to the precision achieved by the solvers) of that computation in the domain. This is not done by default,
- `INFORMATION ABOUT MASS-BALANCE FOR EACH LISTING PRINTOUT`: if they are requested (that is done by default), the user will get, at each time step, information about the flows within the domain.

  The file name is directly managed by the TELEMAC-3D launching procedure. Generally, the file has a name which is created from the name in the steering file and the number of the processors used in the computation, followed by the suffix "**.sortie**".

3.18 The sections output file

Rather than reading the flowrates computed on control section(s) in the listing, the user can store these results in a file with the keyword `SECTIONS OUTPUT FILE`. It is an ASCII file, written by the master in case of parallel run.

3.19 The auxiliary files

Other files can be used by TELEMAC-3D:

- One or two binary data files, as specified by the keywords `BINARY DATA FILE 1` and `BINARY DATA FILE 2`. These files can be used to provide data to the program, the user has to handle their reading within the `FORTRAN FILE`. The data from these files shall be read using the **GET_FREE_ID** subroutine, the **T3DBI1** logic unit for binary data file 1 and the **T3DBI2** logic unit for binary data file 2,
- One or two formatted data files, as specified by the keywords `FORMATTED DATA FILE 1` and `FORMATTED DATA FILE 2`. These files can be used to provide data to the program, the user has to handle their reading within the `FORTRAN FILE`. The data from these files shall be read using the **GET_FREE_ID** subroutine, the **T3DFO1** logic unit for formatted data file 1 and the **T3DFO2** logic unit for formatted data file 2,
- A binary results file, as specified by the keyword `BINARY RESULTS FILE`. This file can be used to store additional results, the user has to handle their writing within the FORTRAN program using the **GET_FREE_ID** subroutine and the **T3DRBI** logic unit,
- A formatted results file, as specified by the keyword `FORMATTED RESULTS FILE`. This file can be used to store additional results (for instance, results readable by an external 1D simulation code used for a coupling with TELEMAC), the user has to handle their writing within the FORTRAN program using the **GET_FREE_ID** subroutine and the **T3DRFO** logic unit.

The read or write operations from/into these files should be thoroughly managed by the user (the files are opened and closed by the program). That management can be performed from any point which the user can gain access to. The logic unit numbers are stated in the **DECLARATIONS_TELEMAC3D** module and the user can access them through a **USE DECLARATIONS_TELEMAC3D** command at the beginning of a subroutine. For instance, using a file for providing the initial conditions will result in managing that file within the **CONDIM** subroutine or the subroutines called by it (e.g.: **USER_CONDI3D_H** etc.). Likewise, using a file

for inserting boundary conditions will be possible at the **USER_BORD3D** subroutine. In case of conflicting statements, one can use, for example: **USE DECLARATIONS_TELEMAC3D, ONLY: T3DBI1**.

3.20 The dictionnary file

This is a file containing all the information on the keywords (French name, English name, default value, type, keywords documentation). This file can be read by the user, but it should not be modified in any case.

3.21 Topographic and bathymetric data

Topographic and bathymetric data may be supplied to TELEMAC-3D at three levels:

- Either directly in the `GEOMETRY FILE` by a topographic or bathymetric value associated with each mesh node. In this case, the data are processed while the mesh is being built using SALOME-HYDRO, Janet, Blue Kenue or MATISSE, or when STBTEL is run before TELEMAC-3D is started. STBTEL reads the information in one or more bottom topography files (5 at most) and interpolates at each point in the domain,
- Or in the form of a cluster of points with elevations that have no relation with the mesh nodes, during the TELEMAC-3D computation. TELEMAC-3D then makes the interpolation directly with the same algorithm as STBTEL. The file name is provided by the keyword `BOTTOM TOPOGRAPHY FILE`. Unlike STBTEL, TELEMAC-3D only manages one single bottom topography file. This may be in SINUSX format or more simply a file consisting of three columns X, Y, Z. The SINUSX format is described in the RUBENS user manual,
- Or using the **USER_T3D_CORFON** subroutine. This is usually what is done for schematic test cases.

In all cases, TELEMAC-3D offers the possibility of smoothing the bottom topography in order to obtain a more regular geometry. The smoothing algorithm can be iterated several times depending on the degree of smoothing required. The keyword `NUMBER OF BOTTOM SMOOTHINGS` then defines the number of iterations carried out in the **T3D_CORFON** subroutine. The default value of this keyword is 0.

4. General setup of the hydrodynamic computation (Navier-Stokes equations)

The general setup of the computation is only performed at the steering file level.
The time data is provided by the two keywords `TIME STEP` (real, set at 1. by default) and `NUMBER OF TIME STEPS`. The first keyword sets the period of time between two consecutive computational moments (but not necessarily two outputs in the result file). The global duration of the computation is provided through a number of time steps (keyword `NUMBER OF TIME STEPS`, set at 1 by default) or a duration in seconds (keyword `DURATION`, set at 0. by default). If both are given, TELEMAC-3D follows the instruction leading to the longest computation.
By default the initial time is equal to 0. s. Since release v8p5, it can be changed by setting the keyword `INITIAL TIME` (default = 0. s). It avoids to change previous hard-coded value 0. in **CONDIM** subroutine. If input data is defined from a specific time and the user wants to start the computation from another time, it is a way to do it without changing input data.
Both date and hour corresponding to the initial time of the computations can be specified using the keywords `ORIGINAL DATE OF TIME` (AAAA, MM, JJ format; default value = 1900; 1; 1) and `ORIGINAL HOUR OF TIME` (HH, MM, SS format, default value = 0; 0; 0). These two data are mandatory if using the tidal data bases. They can be taken into account in programming by means of the **MARDAT** and **MARTIM** variables.
The computation title is specified by the keyword `TITLE`.

4.1 Mesh definition

The three-dimensional mesh, consisting of prisms possibly cut into tetrahedrons, is automatically constructed by TELEMAC-3D from the two-dimensional mesh. This construction is done in the **CALCOT** subroutine from the information given by the subroutine defining the mesh transformation **USER_MESH_TRANSF** or, by default if using a classical sigma transformation, in the **CONDIM** subroutine.
The number of prisms is specified in the steering file by means of the keyword `NUMBER OF HORIZONTAL LEVELS` (default value = 2). That number of levels is equivalent to the number of stacked prisms plus 1. Its minimum value is 2 (1 prism in the vertical direction).
TELEMAC-3D uses a change of variables in order to freeze the mesh on a time step (without such a change, the mesh dimensions z would vary in accordance with the free surface evolution). The frequently adopted change of variables is the sigma transform which consists in shifting from the $z(x,y,t)$ co-ordinate to the $z^*(x,y)$ co-ordinate. The user should enter the z^* co-ordinates in the **USER_MESH_TRANSF** subroutine. The normalized co-ordinates will then range from 0 (the bottom) to 1 (the surface).

The numbering of levels is made according to upward vertical. Level 1 follows the bottom and level N corresponds to the free surface (N being specified by the keyword `NUMBER OF HORIZONTAL LEVELS`).

The vertical mesh definition is based on the **TRANSF_PLANE** table which allows defining the behaviour of each level.

The keyword `MESH TRANSFORMATION` sets the kind of level distribution along the vertical. The value 0 corresponds to a distribution directly defined by the user in the **CALCOT** subroutine. The default value 1 corresponds to the classical sigma transformation.

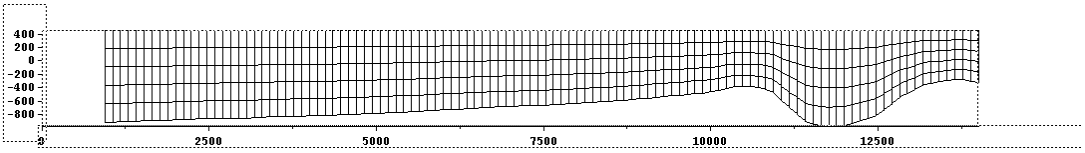

Figure 4.1: Effect of the `MESH TRANSFORMATION` keyword – Value 1: sigma.

The default value is 1 (Figure 4.1) and results in a homogeneous distribution of levels in the vertical direction (classical sigma transformation). The height of the levels varies depending on the water depth, all planes can move (except for the bottom). In this case, no programming in the **USER_MESH_TRANSF** subroutine is required.

A value of 2 (Figure 4.2) will allow the user to define the distribution of levels (e.g. refinement near surface) while maintaining the levels mobility (sigma transformation with given proportions). The latter choice implies that the user will program his/her distribution in the **USER_MESH_TRANSF** subroutine to define the **ZSTAR** array that describes the distribution of levels along the vertical as a percentage of the water depth. Changes to make are:

- Specifying the variable **TRANSF_PLANE** with a value of 2 for every level,
- Specifying the level distribution along the vertical through the array **ZSTAR** which describes the distribution along the vertical as a percentage of the water depth (the values are between 0. and 1.).

For example (Figure 4.2):

```
DO IPLAN = 1,NPLAN
  TRANSF_PLANE%I(IPLAN)=2
ENDDO

ZSTAR%R(1)=0.D0
ZSTAR%R(2)=0.02D0
ZSTAR%R(3)=0.1D0
ZSTAR%R(4)=0.4D0
ZSTAR%R(5)=0.8D0
```

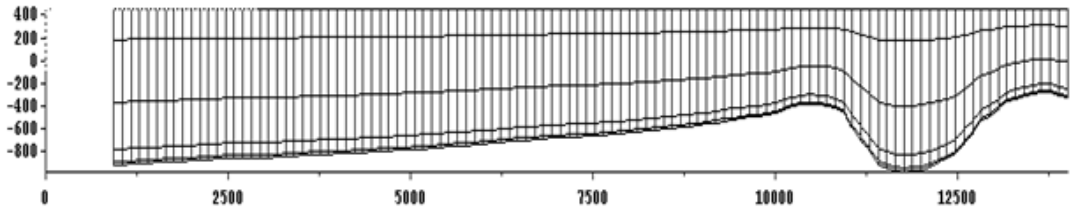

Figure 4.2: Effect of the `MESH TRANSFORMATION` keyword – Value 2: zstar.

In order to better represent the densimetric stratification areas (thermoclines, halocline and/or outfall), prescribing a maximum number of horizontal "levels" (particularly those where the gradients are the highest) is sometimes suitable. For that purpose, the user can select the value 3 for the keyword `MESH TRANSFORMATION`. In this configuration, the user can freely use the 3 types of level definition available in TELEMAC-3D to create the mesh along the vertical:

- Fixed levels at a given altitude (correspond to value 3 of the **TRANSF_PLANE** variable, the altitude is specified by the **ZPLANE** variable),
- Irregularly distributed movable levels between two fixed levels (correspond to value 2 of the **TRANSF_PLANE** variable, the distribution is specified by the **ZSTAR** variable),
- Evenly distributed movable levels between two fixed levels (correspond to value 1 of the **TRANSF_PLANE** variable).

This latter choice requires the user to make changes in the **USER_MESH_TRANSF** subroutine:

- Specifying the variable **TRANSF_PLANE** at value 1, 2 or 3 for each level,
- Specifying the level distribution along the vertical through the array **ZSTAR** for levels of type 2,
- Specifying the level altitude through the array **ZPLANE** for levels of type 3.

For example (Figure 4.3):

```
DO IPLAN = 1,5
  TRANSF_PLANE%I(IPLAN)=2
ENDDO
ZSTAR%R(1)=0.D0
ZSTAR%R(2)=0.2D0
ZSTAR%R(3)=0.5D0
ZSTAR%R(4)=0.7D0
ZSTAR%R(5)=0.8D0

DO IPLAN = 7,NPLAN
 TRANSF_PLANE%I(IPLAN)=1
ENDDO

TRANSF_PLANE%I(6)=3
ZPLANE%R(6)=0.D0
```

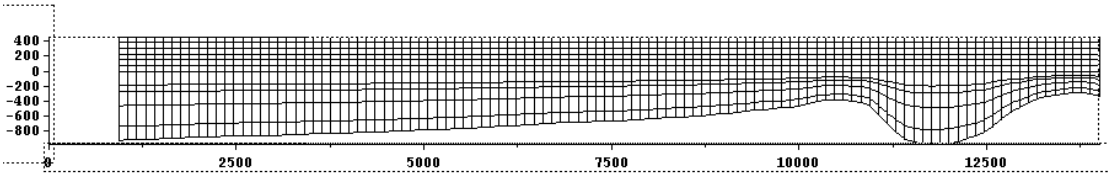

Figure 4.3: Effect of the `MESH TRANSFORMATION` keyword – Value 3: user defined.

Various programming examples are provided as comments in the **USER_MESH_TRANSF** subroutine.

The triangular based prismatic elements can optionally be split into tetrahedrons. This option is enabled using the `ELEMENT` keyword can take the value 'PRISM' (default value) or 'TETRAHEDRON'.

There are some divisions by 3D volumes in the **MESH_PROP** subroutine. A minimum volume value can be set with the keyword `MINIMUM VOLUME OF 3D ELEMENTS` expressed in m^3 (default = 10^{-6} m^3 = 1 cm^3). This value can be changed, for example when modelling scale experiments where dimensions are not big (total length of the order 1 m or less).

When using `MESH TRANSFORMATION` = 3, for all or several planes, the distance between this kind of planes are spaced with a minimum value. Close to the bottom, this minimum distance between planes can be set with the keyword `MINIMUM DISTANCE BETWEEN PLANES CLOSE TO THE BOTTOM` whereas close to the free surface, this minimum distance can be set with the keyword `MINIMUM DISTANCE BETWEEN PLANES CLOSE TO THE FREE SURFACE`. By default, each distance is set to 0.2 m.
The keyword `THRESHOLD HEIGHT BEFORE CRUSHED ELEMENTS` can be used to decide below which height the 3D elements are treated as crushed. This is not done for the free surface plane. The default value 0 m means there are no elements treated as crushed (outside tidal flats). With constant elevation planes, to have crushed elements at the bottom rather than elements at the bottom with a minimum height of `MINIMUM DISTANCE BETWEEN PLANES CLOSE TO THE BOTTOM`, this last keyword should be set to 0 m and `THRESHOLD HEIGHT BEFORE CRUSHED ELEMENTS` to a value greater than 0 m.

Another mesh transformation was introduced in release 6.1: the Adaptive Mesh Refinement (or Redistribution) aka AMR. It can be used by setting `MESH TRANSFORMATION` = 5. The mesh is changed following criteria for a tracer. The interested user can read the TUC 2011 proceeding [2] for more details. The keyword `NUMBER OF TRACER FOR AMR` enables to change the number of tracer given to AMR algorithm in order to adapt the vertical mesh (default = 1).

4.2 Prescribing the initial conditions

The initial conditions aim at defining the model condition at the beginning of the simulation.
In case of a continuing computation, the initial conditions are provided at one time step in the result file of the previous computation (refer to section 3.7). The mandatory variables (at least the velocity components) when resuming the computation should then have been stored into the file being used as `PREVIOUS COMPUTATION FILE`.
Otherwise, the default initial condition is defined as follows:

- Free surface set to an elevation equal to 0,
- Zero velocities,
- Steady zero active and passive tracers.

If that initial condition is not suitable for a computation, then it should be changed using keywords in the simple cases or through a programming as described in the subsequent subsections.

4.2.1 Prescription through keywords

In all the cases, the kind of initial conditions is set by the keyword `INITIAL CONDITIONS`. That keyword can have one of the following six values:

- 'ZERO ELEVATION': Initializes the free surface elevation to 0 (default value). The initial water depths are then computed from the bottom elevation,
- 'CONSTANT ELEVATION': Initializes the free surface elevation to the values as supplied by the keyword `INITIAL ELEVATION` (default value = 0.). The initial water depths are then computed by getting the difference between the free surface elevation and the bottom elevation. In those areas where the bottom elevation exceeds the initial elevation, the initial water depth is zero,
- 'ZERO DEPTH': All the water depths are initialized with a zero value (free surface coinciding with bottom). In other words, the whole domain is "dry" at the beginning of the computation,
- 'CONSTANT DEPTH': Initializes the water depths to the value as supplied by the keyword `INITIAL DEPTH` (default value = 0.),
- 'TPXO SATELLITE ALTIMETRY': The initial conditions are set using information provided by the OSU harmonic constants database (TPXO for instance) or HAMTIDE model in the case of the use of this database for the imposition of maritime boundary conditions (see subsection 4.3.15),
- 'PARTICULAR' or 'SPECIAL': The initial conditions for water depth/ free surface are defined as programmed by the user in the **USER_CONDI3D_H** subroutine (refer to the next subsection). That procedure should be used whenever the initial conditions of the model do not correspond to one of above five cases.

4.2.2 Prescribing particular initial conditions with user subroutines

The **USER_CONDI3D_H**, **USER_CONDI3D_UVW**, **USER_CONDI3D_TRAC**... subroutines should be programmed whenever the initial conditions programmed by default are to be modified.
By default, the standard version of the **USER_CONDI3D_H** subroutine stops the computation if the keyword `INITIAL CONDITIONS` is set to 'PARTICULAR' or 'SPECIAL' without any actual amendment of the subroutine.
The **CONDIM** subroutine successively initializes the two-dimensional variables, then the three-dimensional variables:

- The water depth,
- The 3D component of velocities,

- The active and passive tracers.

The user can quite freely fill that subroutine. For instance, he/she can retrieve information in a formatted or binary file, using the corresponding keywords.

4.2.3 Resuming the computation

TELEMAC-3D enables the user to resume a computation by taking as the initial condition one time step of a computation which was previously computed on the same mesh, or possibly with a different number of levels. Thus, some computational parameters such as the time step, some boundary conditions, the turbulence model can be modified, or else a computation can be initiated once a steady state is achieved.
The file to be retrieved shall then inevitably contain all the data required for TELEMAC-3D, i.e. not only the co-ordinates of the X, Y and Z computational points which it necessarily contains, but also the 3D velocities and the tracers.
If some variables do not appear in the `PREVIOUS COMPUTATION FILE`, then they are automatically set to zero values. A usual application consists in using the result of a hydrodynamic computation in order to perform a tracer transport computation. Generally, the `PREVIOUS COMPUTATION FILE` does not include any result for the tracer.
To resume a computation, it is required to use two keywords into the steering file.
The keyword `COMPUTATION CONTINUED` should be set to the YES value (default value = NO).
The keyword `PREVIOUS COMPUTATION FILE` should provide the name of the file which will provide the initial conditions.
Optionally, the keyword `RECORD NUMBER FOR RESTART` can be used to define the record number to read if it is not the last one (defined by a default value set to -1).

Warning:
The two-dimensional mesh on which the useful results have been computed should be strictly identical to the mesh of the case to be handled.

Resuming the computation usually leads to small differences in results compared to the same calculation without interruption. This difference is mainly due to the fact that the velocity advection is not treated properly at the first time step, because this operation requires information from the previous time step. To correct this, the user has a specific recovery procedure to improve the accuracy of calculations, using double precision format SERAFIN files:

- In the first computation, the keyword `RESTART MODE` is set to YES (default = NO), which generates a specific file containing the full information at one or a few time step(s) of the simulation (in particular information on the advection field of the last time step). The name of this file is specified using the keyword `RESTART FILE`,
- In the second computation, this specific file must be used as `PREVIOUS COMPUTATION FILE` specifying the `PREVIOUS COMPUTATION FILE FORMAT` is 'SERAFIND' (SERAFIN double precision). If the restart should be done from the last time step saved in the `RESTART FILE`, the keyword `RECORD NUMBER FOR RESTART` is to be let to default value (= -1), otherwise it should be changed.

In the first computation, there are 2 keywords which can enable to tune the resuming of the computation (since release 8.4):

- If wanting to generate a `RESTART FILE` at a specific number of time step different from the last one, the keyword `RECORD NUMBER IN RESTART FILE` is to be used (default =

-1 means the RESTART FILE is only written at the last time step or periodically at the period RESTART FILE PRINTOUT PERIOD),

- If wanting to generate a RESTART FILE periodically to secure a file to resume computation in case of crash e.g., the keyword RESTART FILE PRINTOUT PERIOD defines the printout period in number of time steps. Default = 0 means no periodic writing and variables are only written at the last time step of the computation or at the time step number RECORD NUMBER IN RESTART FILE if not equal to -1).

However, it has to be mentioned that even if it is not advisable, the creation of specific restart file can be done not only SERAFIND format, but also with any other available format in the TELEMAC system, especially in single precision. In this case, the keyword RESTART FILE FORMAT (by default set at 'SERAFIND') must be set to the proper value.
A particular aspect of the resuming technique of computation is the value of the start time of the second simulation. By default, the start time of the second calculation is equal to the value of the last time step of restart file. This can be changed by using the logical keyword INITIAL TIME SET TO ZERO if the user wants to start from zero (default value is NO).

It is also possible to resume a computation from a 2D results file. This is generally useful in river hydraulic offering the possibility to initialise the model in 2D before shifting to 3D simulation. In this case, the horizontal velocities are considered as constant on the vertical and equal to the 2D velocities and the vertical velocities are initialised to zero. This possibility is activated using the 2D CONTINUATION logical keyword (default value = NO). The 2D results file must be given using FILE FOR 2D CONTINUATION. The keyword FILE FOR 2D CONTINUATION FORMAT gives the format of the file and can takes the following values: 'SERAFIN ' (default value), 'SERAFIND' and 'MED'.
Since release 8.5, if 2D CONTINUATION = YES and the FILE FOR 2D CONTINUATION contains bottom friction data, these values are taken into account rather than the potential ones in the GEOMETRY FILE.

4.3 Prescribing the boundary conditions

The boundary conditions are handled through types of conditions which are related to the computational variables. The combination of these types (from a list of possible choices) describes whether the boundary is liquid or solid and how it should be processed.
In TELEMAC-3D, the water depth H, the horizontal velocities U and V and the tracers are the only variables which necessarily involve defining their type of boundary conditions. Those types of boundary conditions applicable to the vertical velocity and the k and ε functions are managed by TELEMAC-3D by the user directly in the FORTRAN source files of TELEMAC-3D and therefore do not have a type. If the computation takes tracers into account, then a single type (common to all the tracers) should also be defined for a given boundary.
Once all the types of the boundary are defined, the user should enter the related values for the computational variables (at least H, U and V).
For example, the user may want to set the sea level and leave the velocity field free (e.g. the tide case). The type of boundary will be: "prescribed depth and free velocity". The values required for that type are only water depth at every instant at that boundary. The values of velocities (if they are entered) are not taken into account for that boundary.
Thus, for each TELEMAC-3D boundary, the computational variables (at least H, U and V) are necessarily associated with one type and each type may be associated with one value (either used or not).

The maximum number of boundaries is set to 30 by default but it can be changed by the user with the keyword `MAXIMUM NUMBER OF BOUNDARIES`. This avoids changing the previously hardcoded values (until release 7.0), which required recompiling the whole package.
After such a description of what is a boundary in TELEMAC-3D, we will describe the types, then the related values.

4.3.1 The boundaries in TELEMAC-3D

Water depth is the only two-dimensional variable computed. Its processing at the boundaries is like that being performed by TELEMAC-2D. The boundary points to be handled are those of the two-dimensional mesh (refer to Figure 4.4).
For the other variables (velocities and tracers), the boundary conditions should be handled over all the boundaries of the three-dimensional mesh which includes:

- The lateral boundary points (vertical column points linked to the boundaries of the two-dimensional mesh), whether it is a liquid or solid boundary,
- The points belonging either to the free surface or the bottom.

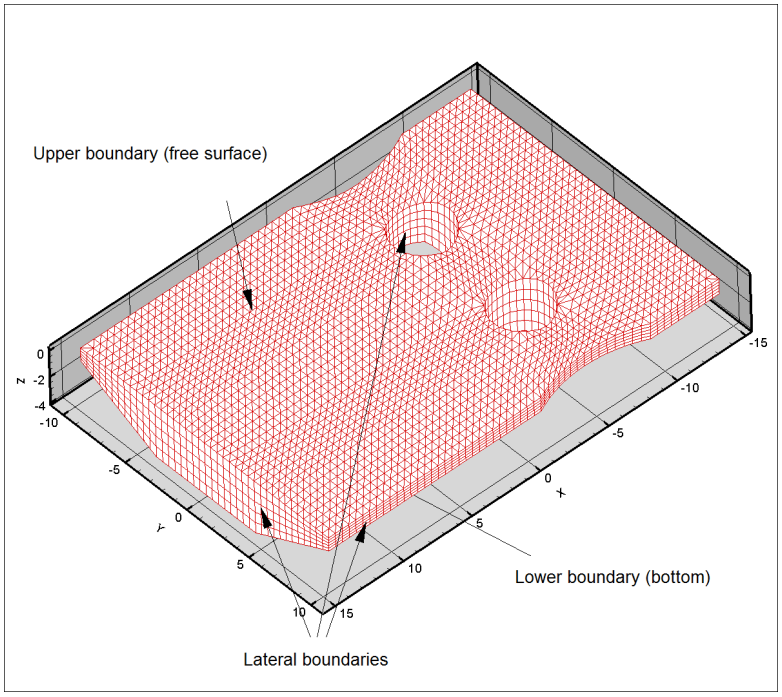

Figure 4.4: The various boundaries in TELEMAC-3D (bridge piers case study).

By default, TELEMAC-3D automatically handles all the surface and bottom points which do not belong to the side walls. The user, however, can modify them. This can be done by modifying the FORTRAN sources.
All the remaining points (on the lateral boundaries) are linked to the two-dimensional mesh boundaries at each horizontal level. Thus, they will be processed in a similar fashion as in TELEMAC-2D. The number of required data, however, increases so much that a full external handling (either through the steering file or the boundary conditions file) would become excessively complex. That is why the range of options offered to the user for dealing with these

boundary conditions is narrower than in TELEMAC-2D and definitely implies programming the sources of the user-available software. The next following subsections describe the way the boundary nodes are handled.

4.3.2 The boundary-related types

The boundary condition type for H, U, V and T of the edge points is read in the boundary conditions file. It can be either modified or directly defined by the user in the **USER_LIMI3D** subroutine.
The various types of boundary conditions can be combined in order to prescribe the conditions of different physical kinds (liquid inflow or outflow in supercritical conditions, open sea, wall, etc.). Some combinations, however, are not physical (refer to subsection 4.3.3 hereinafter).
Some boundary conditions are applicable to such facts as friction at the walls or wall impermeability. However, the wall definition is ambiguous if one only retains a definition of point wise boundary conditions. The following convention is then observed in order to determine the nature of a segment lying between two different kinds of points: a liquid segment is a segment linking two liquid-types points. Thus, under that convention, the connecting point between the shore and the marine boundary (or between the river and the bank) is preferably of the liquid type. Therefore, liquid + solid = solid.
Any sequential arrangement of the boundary types may exist along an outline (for instance, one may have a liquid boundary with a prescribed depth followed by a liquid boundaryliquid boundary with a prescribed velocity). The only condition to be met is that a boundary should consist of at least two points (it is a computational requirement, a number of at least four points being highly advisable from a physical point of view).

4.3.3 Description of the various types

The type of boundary condition at a given point is provided, in the boundary conditions file, in the form of four integers which are referred to as **LIHBOR**, **LIUBOR**, **LIVBOR** and **LITBOR**, with values which can range from 0 to 6.
The available options are as follows:

- Depth condition:
  - Prescribed depth liquid boundary: **LIHBOR** = 5,
  - Free depth liquid boundary: **LIHBOR** = 4,
  - Solid boundary (wall): **LIHBOR** = 2.

It is noteworthy that a depth/rate law is considered as a prescribed depth condition. The flow rate value should then explicitly be computed, according to the water depth, by programming the **USER_Q3** subroutine.

- Rate or velocity condition:
  - Prescribed flow rate liquid boundary: **LIUBOR/LIVBOR** = 5,
  - Prescribed velocity liquid boundary: **LIUBOR/LIVBOR** = 6,
  - Free velocity liquid boundary: **LIUBOR/LIVBOR** = 4,
  - Solid boundary with sliding or friction: **LIUBOR/LIVBOR** = 2,
  - Solid boundary with one or two zero velocity components: **LIUBOR** and/or **LIVBOR** = 0.

- Tracer condition:
  - Prescribed tracer liquid boundary: **LITBOR** = 5,
  - Free tracer liquid boundary: **LITBOR** = 4,
  - Solid boundary (wall): **LITBOR** = 2.

4.3.4 The boundary conditions file

That file is provided as standard by MATISSE, Janet, Blue Kenue or STBTEL, but it can be created or amended by means of FUDAA-PREPRO or a text editor. Each line of that file is dedicated to one point at the two-dimensional mesh boundary. The boundary point numbering is the same as that of the file lines, it first describes the domain outline in the counter clockwise direction, then the islands in the opposite direction.
The convention being observed in TELEMAC implies that the first liquid boundary is that which is defined, within the boundary conditions file, by the first two liquid-typed consecutive numbers. In the example below (channel case study), the first liquid boundary is defined by the nodes 42-47 (edge numbering) and corresponds to a prescribed depth (codes 5 4 4 at the beginning of lines). The second boundary begins at number 76 and ends at number 1 and corresponds to a prescribed rate (codes 4 5 5).

```
 4  5  5   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        1
1
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        5
2
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        6
3
...
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       44
41
 5  4  4   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        2
42
 5  4  4   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       45
43
 5  4  4   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       86
44
...
 5  4  4   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        3
47
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       87
48
...
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       73
74
 2  2  2   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       74
75
 4  5  5   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000        4
76
 4  5  5   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       88
77
...
```

```
 4 5 5   0.000   0.000   0.000   0.0    2   0.000   0.000   0.000       89
85
```

For each point, and each line in the boundary conditions file, the following values are entered:
LIHBOR, LIUBOR, LIVBOR, HBOR, UBOR, VBOR, AUBOR, LITBOR, TBOR, ATBOR, BTBOR, N, K

- **LIHBOR, LIUBOR, LIVBOR** and **LITBOR** are boundary-typed integers for each of the variables.
- **HBOR** (real) denotes the prescribed depth value when the **LIHBOR** value is set to 5,
- **UBOR** (real) denotes the prescribed U velocity value when the **LIUBOR** value is set to 6,
- **VBOR** (real) denotes the prescribed V velocity value when the **LIVBOR** value is set to 6,
- **AUBOR** denotes the value of the boundary friction coefficient when the **LIUBOR** or **LIVBOR** value is set to 2. The friction law is then written as:

$$\upsilon_T \frac{dU}{dn} = AUBOR \times U \quad \text{and/or} \quad \upsilon_T \frac{dV}{dn} = AUBOR \times V \tag{4.1}$$

The **AUBOR** coefficient is applicable to the segment included between the edge point being considered and the next point (in the counter clockwise direction for the outside outline and in the clockwise direction for the islands). By default, the **AUBOR** value is 0. A friction corresponds to a negative value. With the k-ε model, the value of **AUBOR** is automatically computed by TELEMAC-3D, the indications in the boundary conditions file will then be ignored.

- **TBOR** (real) denotes the prescribed tracer value when the **LITBOR** value is set to 5,
- **ATBOR** and **BTBOR** denote the coefficient values of the flux law which is written as:

$$\upsilon_T \frac{dT}{dn} = ATBOR \times T + BTBOR \tag{4.2}$$

The **ATBOR** and **BTBOR** coefficients are applicable to the segment included between the edge point considered and the next point (in the counter clockwise direction for the outside outline and in the clockwise direction for the islands).

- **N** denotes the edge point global number,
- **K** denotes the point number in the edge point numbering. This number also represents a node colour (as an integer). This number named **BOUNDARY_COLOR**, can be used in parallel simulations to simplify the implementation of specific cases. Without particular modification, this value is the rank of the border point in the global numbering. For example, a test like **IF (I.EQ.144) THEN** can be replaced by **IF (BOUNDARY_COLOUR%I(I).EQ.144) THEN** which is compatible with the parallel mode. However, this only concerns the 2D mesh (Table **BOUNDARY_COLOUR** is only given for level 1). Be careful not to modify the last column of the boundary conditions file that contains this **BOUNDARY_COLOUR** table, when using tidal harmonic constants databases (cf. [6]).

As regards the horizontal velocities, all the points in one water column will have the same type of boundary condition defined by **LIUBOR** or **LIVBOR**. That principle is intrinsic to the TELEMAC-3D formulation. Prescribing a different type of boundary condition in the vertical direction (e.g. for a subterranean stream) may, indeed, induce severe inconsistencies with the hydrostaticity hypothesis and generate, for instance, unrealistic vertical velocities. It is then advisable that the user will follow that principle. Nonetheless, the boundary condition type in the vertical direction can be altered through direct programming in the **USER_LIMI3D** subroutine.

The so-called **LIHBOR**, **LIUBOR** and **LIVBOR** integers (which define the boundary type) can assume a value ranging from 0 to 6. The available options are as follows:

- Depth-related condition:
  - Prescribed depth liquid boundary: **LIHBOR** = 5,
  - Free depth liquid boundary: **LIHBOR** = 4,
  - Solid boundary (wall): **LIHBOR** = 2,
- Velocity-related condition:
- Prescribed velocity liquid boundary: **LIUBOR/LIVBOR** = 6,
- Prescribed rate liquid boundary: **LIUBOR/LIVBOR** = 5,
- Free velocity liquid boundary: **LIUBOR/LIVBOR** = 4,
- Solid boundary with sliding or friction: **LIUBOR/LIVBOR** = 2,
- Solid boundary with one or two zero velocity components: **LIUBOR** and/or **LIVBOR=0**.

The boundary conditions of physical nature are defined by the relationship among the types of variables. In most cases, the boundary type can be set by a mesh generator (e.g. MATISSE or Janet) in the TELEMAC chain. The table below summarizes the physical relationship among the boundary types.

LIHBOR	LIUBOR	LIVBOR	LITBOR	
2	2	2	2	Solid wall.
2	0	2	2	Solid wall with zero U.
2	2	0	2	Solid wall with zero V.
2	0	0	2	Solid wall with zero U and V.
4	4	4	4	Free H, free velocities, free T.
5	4	4	4	Prescribed H, free velocities, free T.
5	4	0	4	Prescribed H, free U, zero V, free T.
5	0	4	4	Prescribed H, zero U, free V, free T.
1	1	1	4	Incident wave, free tracer.
4	5	5	5	Free H, prescribed Q, prescribed T.
4	5	0	5	Free H, prescribed Q with zero V, prescribed T.
4	0	5	5	Free H, prescribed Q with zero U, prescribed T.
4	6	6	5	Free H, prescribed velocities, prescribed T.
5	5	5	5	Prescribed H and Q, prescribed T.
5	6	6	5	Prescribed H and velocities, prescribed T.

4.3.5 Programming the boundary conditions type

The **USER_LIMI3D** subroutine can be programmed to handle specific boundary conditions, for the edge points as well as the surface and bottom points.
That subroutine is called upon each time step. Therefore, it can be used to change the boundary condition type in time, if required.

4.3.6 Prescribing values through keywords

In most simple cases, the boundary conditions are prescribed using keywords. However, if the values to be prescribed vary in time, it is necessary to program the adequate functions or to use the liquid boundaries file.
The appropriate keywords to prescribe the boundary values are as follows:

- `PRESCRIBED ELEVATIONS`: provided to set the elevation value of a prescribed elevation liquid boundary (free surface). It is an array that can contain up to **MAXFRO** reals, and therefore up to **MAXFRO** boundaries of that kind can be handled. The values defined by that keyword overwrite the depth values read from the `BOUNDARY CONDITIONS FILE`,

> Warning:
> The value given here is the free surface level, whereas the value given in the `BOUNDARY CONDITIONS FILE` is the water depth.

- PRESCRIBED FLOWRATES: provided to set the flow rate value of a prescribed flow at a liquid boundary. It is an array which can contain up to **MAXFRO** reals, and therefore up **MAXFRO** boundaries of that kind can be handled. A positive value corresponds to a domain inflow rate. The values defined by that keyword overwrite the velocity values read from the BOUNDARY CONDITIONS FILE,

- PRESCRIBED VELOCITIES: provided to set the velocity value of a prescribed velocity liquid boundary. The scalar value is the wall normal velocity. A positive value corresponds to a domain inflow. It is an array which can contain up to **MAXFRO** reals, and therefore up **MAXFRO** boundaries of that kind can be handled. The values defined by that keyword overwrite the values read from the BOUNDARY CONDITIONS FILE.

In addition, several simple rules should be observed:

- The boundary type as specified in the BOUNDARY CONDITIONS FILE should obviously be in accordance with the keywords in the steering file (do not insert the keyword PRESCRIBED FLOWRATES if there are no boundary points the **LIUBOR** and **LIVBOR** of which are set to 5). The keyword, however, is ignored if no type matches it,

- For each keyword, the number of specified values should be equal to the whole number of liquid boundaries, whatever their types may be. When a boundary is inconsistent with the keyword, then the specified value is ignored (a 0.0 value or, on the contrary, a very high value such as 999.0 may be systematically inserted).

  For example, in the channel test case, the first boundary (downstream boundary) is of prescribed level type whereas the second one (upstream boundary) is of prescribed flow rate type. The steering file contains a sequence of the following type:

```
PRESCRIBED ELEVATIONS = 0.5, 0.0
PRESCRIBED FLOWRATES  = 0.0, 50.0
```

4.3.7 Boundary condition on the bottom

By default, the boundary condition on the bottom is an impermeable slip boundary (Neumann condition of the same type as vertical conditions).
However, bottom velocities can be set to zero by using value 2 of the keyword BOUNDARY CONDITION ON THE BOTTOM (Default value of 1 corresponds to a slip condition). This option is valid only if the vertical mesh is refined at bottom level.
Since release 7.1, it is possible to prescribe a flux on the bed in TELEMAC-3D (e.g.: a flow rate on several liquid boundaries placed on the bed). To do so, it is necessary to define the imposed flow rates using the keywords OPEN BOUNDARY CONDITIONS ON THE BED set to YES (default value = NO) and PRESCRIBED FLOWRATES ON THE BED with values following the same structure as for other prescribed flow rates in the TELEMAC-MASCARET system. It should be a list of numbers separated by a semi-colon, one number per liquid boundary on the bed must be given. The maximum number of boundaries on the bed is set to 30 by default but it can be changed by the user with the keyword MAXIMUM NUMBER OF BOUNDARIES ON THE BED. At the moment, the BOUNDARY CONDITIONS FILE only deals with horizontal boundaries, therefore the user has to define the liquid boundary on the bed by hand. This can be done by modifying the subroutine **USER_LIMI3D** in the FORTRAN FILE. For example to add a circular boundary of radius 5 m centred around coordinate (2 000, 2 000) m, the following modifications can be done:

```
...
!      BOUNDARY CONDITIONS ON VELOCITIES
!      *********************************
!
!      BOTTOM
!      ======
!
!      DEFAULT: IMPERMEABILITY AND LOG LAW (SEE ALSO BORD3D)
!
       IF(BC_BOTTOM.EQ.1) THEN
!
         DO IPOIN2 = 1,NPOIN2
           LIUBOF%I(IPOIN2) = KLOG
           LIVBOF%I(IPOIN2) = KLOG
           LIWBOF%I(IPOIN2) = KLOG
!          USEFUL ? SHOULD NOT BE USED ANYWAY
           UBORF%R(IPOIN2)  = 0.D0
           VBORF%R(IPOIN2)  = 0.D0
           WBORF%R(IPOIN2)  = 0.D0
           IF(SQRT((X(IPOIN2)-2000.D0)**2
     &            +(Y(IPOIN2)-2000.D0)**2)
     &        .LE.50.D0) THEN
             !5: IMPOSED FLOW RATE
             LIUBOF%I(IPOIN2) = 5
             LIVBOF%I(IPOIN2) = 5
             LIWBOF%I(IPOIN2) = 5
             NLIQBED%I(IPOIN2) = 1
             WRITE(LU,*) '========================'
             WRITE(LU,*) 'FOR POINT ',IPOIN2
             WRITE(LU,*) 'BEDFLO',BEDFLO(1)
           ENDIF
         ENDDO
!
...
```

In this example, it should be noted that **NLIQBED%I(IPOIN2)** = 1 defines the position for the first liquid boundary defined in the steering file. This is all that needs to be defined by the user to deal with fluxes on the bed. However, in release 7.1, only constant velocity profile is available. It should also be noted that it has not been possible to prescribe a tracer or turbulence yet.

4.3.8 Using the liquid boundaries file

In case of time variable values, which are nonetheless constant in space along the relevant liquid boundary, the prescription can be done using the liquid boundaries file (as an alternative to programming).
It is a user-edited ASCII file the name of which should be given by the keyword `LIQUID BOUNDARIES FILE`. That file has the following format:

- The optional line(s) begin(s) with the sign # (1st character on the line) will be treated as comments,
- It should contain a header line beginning with **T** for identifying the supplied time dependent value(s) within that file. The identification is performed through mnemonic means which are identical to the variable names: **Q** for the flow rate, **SL** for the level, **VIT** for

the velocity (giving the magnitude) and **TR** for the tracer. These characters are directly followed by an integer in between brackets which is used to specify the current boundary. That line is necessarily followed by another line indicating the unit of the variables (lines of comments can be inserted, but the line of units should be present). The units are given for information only and TELEMAC-3D does not handle the conversion of units (thus, the user has to enter the values using the standard unit),

- The values to be prescribed are provided through a sequence of lines the format of which should be consistent with the identification line. The time value should be increasing and the last time value supplied should be higher than or equal to the value of the last time step, otherwise the computation is suddenly interrupted.

Upon the retrieval of that file, TELEMAC-3D performs a linear interpolation in order to compute the value prescribed at a particular time step. The value which is actually prescribed by the code is printed on the check listing.
An example of a liquid boundaries file is given below.

```
#  Example of liquid boundaries file
#  2 boundaries are managed
#
T Q(1) SL(2)
s m3/s m
0. 0. 135.0
25. 15. 135.2
100. 20. 136.
500. 20. 136.
```

In that example, the flow rate is prescribed at the first boundary and the free surface is prescribed at the second boundary.
Since release 8.2, a time reference can be given: If a #REFDATE with a date + hour in YYYY-MM-DD HH:MM:SS in year, month, day, hour, minute, second format is written in ASCII files related to time, the date+hour will be added to the times in these ASCII files when using TELEMAC-2D and TELEMAC-3D.

4.3.9 Prescribing values through programming

Still in the case of time variable values that are constant in time along the liquid boundary processed, the prescription can be done simply by programming particular subroutines:

- **USER_VIT3** subroutine for prescribing a velocity,
- **USER_Q3** subroutine for prescribing a flow rate,
- **USER_SL3** subroutine for prescribing an elevation.

USER_Q3, USER_VIT3 and **USER_SL3** subroutines are similarly programmed. In each case, the user knows the time, the boundary rank (e.g. to determine whether the first or the second prescribed flow rate boundary is processed). By default, the subroutines prescribe values that are read from the boundary conditions file or provided by the keywords.
For instance, the body of subroutine **USER_Q3** used to prescribe a flow rate ramp for the first 1,000 seconds from 0 to 400 m^3/s can take such a form as:

```
IF (AT.LT.1000.D0) THEN
  Q3 = 400.D0 * AT/1000.D0
ELSE
  Q3 = 400.D0
ENDIF
```

Or

```
Q3 = 400.D0 * MIN(1.D0,AT/1000.D0)
```

4.3.10 Stage-discharge curves

TELEMAC-3D allows managing liquid boundaries for which the prescribed water elevation value is a function of local flow rate. This situation is encountered particularly in river hydraulics.
First, it is necessary to specify which boundaries are concerned with the keyword STAGE-DISCHARGE CURVES. This keyword provides an integer value for each boundary. This value can be:

- 0: no stage-discharge curves (default),
- 1: elevation as a function of local flow rate.

The keyword STAGE-DISCHARGE CURVES FILE provides the name of the text file containing information about the curves. An example is shown below:

```
#
#   STAGE-DISCHARGE CURVE BOUNDARY 1
#
Q(1)          Z(1)
m3/s          m
61.           0.
62.           0.1
63.           0.2
#
#   STAGE-DISCHARGE CURVE BOUNDARY 2
#
Z(2)          Q(2)
m             m3/s
10.           1.
20.           2.
30.           3.
40.           4.
50.           5.
```

The order of curves has no significance. The column order can be reversed, as is the case for the second boundary in the example. Lines beginning with # are comments. The lines defining the units are mandatory but the units are not checked. The number of points of each curve is completely free and need not be the same for each curve.
Warning: at initial conditions, the flow at the exit can be null. The initial level must correspond to that of the calibration curve otherwise a sudden change is imposed. To avoid extreme situations, the curve should be limited to a certain level of flow rate. In the example of boundary 1 above, the flow rates below 61 m^3/s generate a water elevation of 0 m above the flow of 63 m^3/s to produce an elevation equal to 0.2 m.

When using value 1 for the keyword STAGE-DISCHARGE CURVES, the relation between elevation and discharge may not be exactly the expected values. Indeed, there is a delay (relaxation) to avoid triggering resonances and high oscillations by means of a relaxation coefficient. This coefficient allows to smooth high gradients of the prescribed values. If set to 1., the elevation is instantaneously prescribed corresponding to the stage-discharge curve, but this may lead to instabilities. Setting a value between 0. and 1. is a compromise between the goal of the stage-discharge curve and possible instabilities.

Since release 8.3, the relaxation coefficient can be changed with the keyword STAGE-DISCHARGE CURVES RELAXATION COEFFICIENT (default value = 0.02). This keyword substitutes the old hard-coded value of 0.02 at the end of function **STA_DIS_CUR** (this value could have been changed manually).

4.3.11 Prescribing complex values

If the values to be prescribed vary in both space and time, then a programming in the **USER_BORD3D** subroutine becomes necessary, since that subroutine can be used to prescribe the values in a node wise way.

Its calling subroutine **BORD3D** describes all the liquid boundaries (loop on **NPTFR2**). For each boundary point, it determines the boundary type in order to prescribe the adequate value (velocity, elevation or flow rate). **USER_BORD3D** or **BORD3D** programming to prescribe a flow rate, however, hardly makes any sense, since the flow rate value is generally known for the whole boundary rather than on each boundary segment.

If a prescribed flow rate inlet is surrounded by walls with an adherence, then the corner velocities are cancelled.

Note that, the **USER_BORD3D** subroutine also makes it possible to prescribe the complex boundary values of the tracers.

4.3.12 Prescribing a profile

Horizontal profile

When processing a prescribed flow rate or prescribed velocity boundary, the user has the keyword VELOCITY PROFILES to specify which "horizontal" velocity profile should be prescribed by TELEMAC-3D (one value per liquid boundary). The following options are possible:

- 1: The profile is normal and homogeneous along the boundary (default option),
- 2: The values of U and V are read from the boundary conditions file (**UBOR** and **VBOR** values). In case of a prescribed flow rate, these values are multiplied by a constant in order to get the desirable flow rate,
- 3: The velocity vector is normal to the boundary and its norm is read from the boundary conditions file as the **UBOR** value. In case of a prescribed flow rate, that value is multiplied by a constant in order to get the desirable flow rate,
- 4: The velocity vector is normal to the boundary and its norm is proportional to the square root of the water depth,
- 5: The velocity vector is normal to the boundary and its norm is proportional to the square root of a virtual water depth computed from lowest point of the free surface on the boundary.

Vertical profile

When processing a prescribed flow rate or prescribed velocity boundary (for this last one only since release 8.5), the user has the keyword `VELOCITY VERTICAL PROFILES` to specify which "vertical" velocity profile should be prescribed by TELEMAC-3D (one value per liquid boundary). The options for that keyword are:

- 0: programmed by the user,
- 1: constant (default value for all the liquid boundaries),
- 2: logarithmic.

The user programming is done within the **USER_VEL_PROF_Z** subroutine.
Activating the keyword `DYNAMIC BOUNDARY CONDITION` (default value = FALSE) also enables to prescribe a velocity at the free surface coherent with the dynamic boundary condition.

4.3.13 Thompson conditions

In some cases, not all the necessary information concerning the boundary conditions is available. This is usual for coastal domains where only the values of the sea level on several points are known. This kind of model is referred to as an ”under-constrained” model.
To solve this problem, the Thompson method uses the theory of characteristics to calculate the missing values. For example, TELEMAC-3D will compute the velocity at the boundary in the case of a prescribed elevation.
This method can also be used for ”over-constrained” models. In this case, the user specifies too much information at the boundary. If the velocity information and the level information are not consistent, too little or too much energy is going into the model. For this, the Thompson method computes a new value for the velocity and performs small adjustments to cancel the inconsistencies in the information.
For this, the user can use the keyword `OPTION FOR LIQUID BOUNDARIES`, which offers two values (the user must specify 1 value for every open boundary):

- 1: strong setting (default value for all boundaries),
- 2: Thompson method.

Taking a simplified view, it may be said that, in the case of the first option, the values are “imposed”, in the case of the second option, the values are “suggested”.
However it is important to note that, given the two-dimensional aspect, the Thompson method can only be used in the case of a zero velocity gradient imposed on the vertical (uniform velocity along the vertical).

Note:

1. This keyword should be given for ALL liquid boundaries like given in the following example:
   `OPTION FOR LIQUID BOUNDARIES`= 2;1;2;2
   which means that you are asking TELEMAC-3D to apply Thompson boundary condition to boundaries number 1,3 and 4; unlike boundary number 2, where TELEMAC-3D will use a strong prescription.
2. This option will trigger the computation of characteristics’ trajectories in order to get informations from inside the domain.

4.3.14 Soft boundary conditions

Jetting flows can occur on liquid boundaries where the elevation is prescribed. To prevent this, a so called soft boundary can be used since release 8.5. On a soft boundary, the prescribed level is modified by a multiple of the speed normal to the boundary or the speed squared.

The keyword to set soft boundaries is `OPTION FOR SOFT BOUNDARIES`. The user should give one integer for each open boundary:

- 0: Not a soft boundary (default value for all boundaries),
- 1: Method 1. Proportional to speed,
- 2: Method 2. Proportional to speed squared.

Each soft boundary needs a coefficient. These coefficients are specified by the keyword `COEFFICIENT FOR SOFT BOUNDARIES` (default = 0.).

4.3.15 Tidal harmonic constituents databases

General parameters

To prescribe the boundary conditions of a coastal boundary subject to tidal evolution, it is generally necessary to have the information characterizing this phenomenon (harmonic constants). One of the most common cases is to use the information provided by large scale models.
5 databases of harmonic constants are interfaced with TELEMAC-3D:

- The JMJ database resulting from the LNH Atlantic coast TELEMAC model by Jean-Marc JANIN [9],
- The global TPXO database and its regional and local variants from the Oregon State University (OSU) [4],
- the HAMTIDE global model [15] since release 8.5,
- the regional North-East Atlantic atlas (NEA) [12, 13] and the global atlas FES (e.g. FES2004 or FES2014 [10]) coming from the works of Laboratoire d'Etudes en Géophysique et Océanographie Spatiales (LEGOS),
- The PREVIMER atlases [14].

However it is important to note that, in the current release of the code, the latter 2 databases are not completely interfaced with TELEMAC-3D and their use is recommended only for advanced users.
The keyword `OPTION FOR TIDAL BOUNDARY CONDITIONS` activates the use of one of the available database when set to a value different from 0 (the default value 0 means that this function is not activated). Since release 7.1, this keyword is an array of integers separated by semicolons (one per liquid boundary) so that the user can describe whether tidal boundary conditions should be computed or not (e.g. a weir) on a liquid boundary. When this keyword is activated, every tidal boundary is treated using the prescribed algorithms for the boundaries with prescribed water depths or velocities, with the same option for tidal boundary conditions (the values not equal to 0 have to be the same). The databases provide only a single value of the depth-averaged velocity, thus TELEMAC-3D prescribes the same value of the velocity at each point along the vertical (for more information, see Méthodologie pour la simulation de la

marée avec la version 6.2 de TELEMAC-2D et TELEMAC-3D, C.-T. Pham et al., EDF report H-P74-2012-02534-FR [3]).
The database used is specified using the keyword `TIDAL DATA BASE` which can take the values:

- 1: JMJ,
- 2: TPXO or HAMTIDE,
- 3: MISCELLANEOUS (LEGOS-NEA, FES20XX, PREVIMER...).

If using HAMTIDE model, the keyword `VELOCITIES IN BINARY DATABASE 2 FOR TIDE` has to be set to YES (default = NO, e.g. if using OSU tidal solution like TPXO), see below.

Depending on the database used, some keywords have to be specified.

- If using the JMJ database, the name of the database (typically bdd_jmj) is given by the keyword `ASCII DATABASE FOR TIDE` and the corresponding mesh file is specified using the keyword `TIDAL MODEL FILE`,
- If using the TPXO database or HAMTIDE model, the name of the water level database is given by the keyword `BINARY DATABASE 1 FOR TIDE` (for example h_tpxo7.2) and the name of the velocity database is given by the keyword `BINARY DATABASE 2 FOR TIDE` (for example u_tpxo7.2). Moreover, it is possible to activate an interpolation algorithm of minor constituents from data read in the database using the logical keyword `MINOR CONSTITUENTS INFERENCE`, activation not done by default. If using HAMTIDE model, as input data are velocity components rather than transports (water depth times velocity components), the keyword `VELOCITIES IN BINARY DATABASE 2 FOR TIDE` has to be set to YES (default = NO).

The keyword `OPTION FOR TIDAL BOUNDARY CONDITIONS` specifies the type of tide to prescribe. The default value 0 means no prescribed tide or that the tide is not treated by standard algorithms. Value 1 corresponds to prescribing a real tide considering the time calibration given by the keywords `ORIGINAL DATE OF TIME` (YYYY ; MM ; DD format) and `ORIGINAL HOUR OF TIME` (HH ; MM ; SS format). Other options are the following, available for every tidal database (JMJ, TPXO-type from OSU, HAMTIDE, LEGOS-NEA, FES, PREVIMER . . .). They are called “schematic tide” for values from 2 to 6:

- 2: exceptional spring tide (French tidal coefficient approximately equal 110),
- 3: mean spring tide (French tidal coefficient approximately equal 95),
- 4: mean tide (French tidal coefficient approximately equal 70),
- 5: mean neap tide (French tidal coefficient approximately equal to 45),
- 6: exceptional neap tide (French tidal coefficient approximately equal to 30),
- 7: real tide (before 2010 methodology, only available with JMJ).

In the case of options 2 to 6 (schematic tides), the boundary conditions are imposed so that the reference tide is approximately respected. In order to shift the phases of the waves of the tidal constituents so that the computation starts close to a High Water, two keywords are available. If using a TPXO-type tidal database from Oregon State University or HAMTIDE

model, the keyword GLOBAL NUMBER OF THE POINT TO CALIBRATE HIGH WATER has to be filled with the global number of the point (between 1 and the number of boundary nodes in the 2D mesh) with respect to which the phases are shifted to start with a high water (mandatory, otherwise the computation stops). This point has to be a maritime boundary node. If using one of the other tidal databases (JMJ, NEA/FES, PREVIMER) the keyword LOCAL NUMBER OF THE POINT TO CALIBRATE HIGH WATER should be filled in with the local number between 1 and the number of tidal boundary points of the HARMONIC CONSTANTS FILE; If not filled in (default value = 0), a value is then automatically calculated. However, it is usually necessary to wait for the second or third modelled tide in order to overcome the transitional phase of start-up of the model. It is also necessary to warn the user that the French tidal coefficients shown are approximate.
During a simulation, data contained in the tidal database are interpolated on boundary points. When using the JMJ database, this spatial interpolation can be time consuming if the number of boundary points is important, and is not yet available in case of parallel computing. It is therefore possible to generate a file containing harmonic constituents specific to the model treated. The principle is at a first step, to perform a calculation on a single time step whose only goal is to extract the necessary information and to generate a file containing for each boundary point of the model, the harmonic decomposition of the tidal signal. Subsequent calculations directly use that specific file rather than directly addressing to the global database. The harmonic constants specific file is specified using the keyword HARMONIC CONSTANTS FILE, this file is an output file in the first calculation, and an input file in subsequent calculations.

If using tidal solutions coming from OSU (e.g. TPXO), to get velocity components, it is necessary to divide the transports terms (water depth times velocity components) by water depth, contrary to HAMTIDE model which directly provides velocity components. A minimum value of water depth to get them is taken to avoid divisions by 0. Since release 8.2, it is possible to change this old hard coded value of 0.1 m, both for boundary conditions and initial conditions, with the keywords MINIMUM DEPTH TO COMPUTE TIDAL VELOCITIES BOUNDARY CONDITIONS and MINIMUM DEPTH TO COMPUTE TIDAL VELOCITIES INITIAL CONDITIONS. Both default values are 0.1 m. Moreover, for initial conditions, if water depth is below MINIMUM DEPTH TO COMPUTE TIDAL VELOCITIES INITIAL CONDITIONS, the velocity components are set to 0. These 2 keywords enable to decrease artificially too high velocities in particular at open boundaries with shallows or if the tidal solutions have shallows at the same location. If this effect of the latter keyword is not sufficient, the user can initialise the computation with no velocities with INITIAL VELOCITIES COMPUTED BY TPXO = NO (default = YES, i.e. the tidal velocities are computed by algorithms to use OSU tidal solutions).

Horizontal spatial calibration

In order to perform the spatial interpolation of the tidal data, it is imperative to provide to TELEMAC-3D information on the spatial positioning of the mesh model relative to the grid of the tidal database. To do this, the user has two keywords:
The first keyword specifies the geographic system used to establish the coordinates of the 2D mesh of TELEMAC-3D. This keyword GEOGRAPHIC SYSTEM, which has no default value, may take the following values:

- 0: User Defined,
- 1: WGS84 longitude/latitude in real degrees,
- 2: WGS84 UTM north,

- 3: WGS84 UTM south,
- 4: Lambert,
- 5: Mercator projection.

The second keyword is used to specify the area of the geographic system used to establish the coordinates of the 2D mesh of TELEMAC-3D. This keyword `ZONE NUMBER IN GEOGRAPHIC SYSTEM` which has no default value, may take the following values:

- 1: Lambert 1 north,
- 2: Lambert 2 center,
- 3: Lambert 3 south,
- 4: Lambert 4 Corsica,
- 22: Lambert 2 extended,
- 93: Lambert 93,
- *X*: UTM zone value of the WGS84 (*X* is the number of the zone).

If using the Lambert 93 projection, the user has to copy the file provided in the tide examples of TELEMAC-2D called gr3df97a.txt which is used for the conversion in the Lambert 93 projection. The keyword `LAMBERT 93 CONVERSION FILE` has to indicate the path and the name of the gr3df97a.txt file.

Since release 8.2, it is possible to use *x* and *y* origin coordinates stored in the geometry file to decrease the number of digits of coordinates when modelling tide, e.g. when using UTM or Lambert projections. The two numbers are stored in the **I_ORIGIN** and **J_ORIGIN** variables reachable with the help of **GET_MESH_ORIG** subroutine. Caution: these two numbers are integers, not floats as the preliminary structure available in the SERAFIN format expected this type for these 2 variables. The tidal computations automatically take into account this offset for UTM + Lambert projections when generating the `HARMONIC CONSTANTS FILE` for JMJ database or when interpolating harmonic constants to compute boundary or initial conditions for solutions coming from OSU (e.g. TPXO) or HAMTIDE. In particular for the operations to locate the nodes correctly (a simple translation) but TELEMAC-3D still continues to compute other steps in a local coordinate system.

Calibration of the information

The transfer of information between a large scale model and the boundaries of a more local model generally requires calibration.
To do this, the user has three keywords:

- The keyword `COEFFICIENT TO CALIBRATE SEA LEVEL` (default real value 0.) is used to calibrate the mean tide level (the harmonic decomposition of information provided by the various databases are used to generate the tidal signal oscillating around mean tide level). The calibration of the mean tide level must obviously be made depending on the altimetric reference used in the model,

- The keyword `COEFFICIENT TO CALIBRATE TIDAL RANGE` (default real value 1.) is used to specify a calibration coefficient applied on the amplitude of the tidal wave. This coefficient is applied to the amplitude of the overall signal, and not on the amplitude of each of the elementary waves,

- The keyword `COEFFICIENT TO CALIBRATE TIDAL VELOCITIES` (default real value 999,999.0) is used to specify the coefficient applied on velocities. The default value (999,999.0) means that the square root of the value specified by the keyword `COEFFICIENT TO CALIBRATE TIDAL RANGE` tidal is used.

For more information, the reader may refer to the methodological guide for tide simulation with version 6.2 [3].

4.4 Control sections

A control section offers the possibility of obtaining the instantaneous and cumulated flow rates through a specific surface of the domain.

Warning:
For this feature in 3D, the flow rates are computed through sections from bottom to free surface, not between specific elevations. In that case, it is assumed that depth averaged velocities U_{2D} and V_{2D} are OK to compute flow rates through these control sections.

The weak formulation of the no-flux boundary condition through solid boundaries raises a theoretical problem for computing the flow rates. Either they are compatible with the results file, or they are compatible with the weak formulation. To be compatible with the weak formulation, use the keyword `COMPATIBLE COMPUTATION OF FLUXES` (default = NO). The difference may reach a few percents.
It is also possible to obtain the cumulated flow rates in the listing for each control section by activating the logical keyword `PRINTING CUMULATED FLOWRATES` (default value = NO). In that case, to improve the quality of results, the treatment of the control section is done at each time step and not only at each time step concerned by a printing on output listing.
Contrary to TELEMAC-2D, the control sections can only be managed using one single procedure in TELEMAC-3D since release 8.5. It is the same as the second procedure in TELEMAC-2D (available since release 6.0). It is based on an external configuration file and is compatible with the parallel mode. TELEMAC-3D uses depth-averaged velocity components to compute control section(s) as it is done in 2D.
The user must supply the name of the section(s) configuration file using the keyword `SECTIONS INPUT FILE`.
In parallel mode, this file will be modified by the mesh partitioner so that it corresponds locally to every sub-domain.
The file format is the following:

- one comment line (free but must be here),

- two integers: number of sections and steering integer. For the steering integer, the convention is as follow:

  - if negative (< 0): node numbers (global numbers) are given,

  - if equal to 0: coordinates are given. Note that if positive (> 0) and parallel mode, coordinates can also be given, BUT the combo positive steering integer (> 0) and serial mode does not work!

- two lines per section:

  - 24 characters for a section name, followed by:
  - begin and end node number or begin and end coordinates. The line coordinates should be given as follow: x1 y1 x2 y2

Example:

```
# Control sections definition
5  -1
Wesxan_outflow
46 70
Wesxan_Middle
639 263
Wesxan_Inflow
480 414
Wesxan_crazy
142 147
Wesxan_even_worse
144 7864
```

Headers and printouts on control sections may be modified in **FLUXPR_TELEMAC3D** subroutine (TELEMAC-3D library).
The printouts can be written in the file named by the keyword `SECTIONS OUTPUT FILE` if a name is given to this keyword.

Inverted barometer effect

Atmospheric pressure can be taken into account at tidal boundaries by using inverted barometer method (adding a head loss computed from the difference of pressures divided by water density and gravity acceleration) when modelling storm surges e.g. (with wind and atmospheric pressure fields). From release 8.5, the keyword `ATMOSPHERIC PRESSURE AT TIDAL BOUNDARIES` is to be activated (default = NO) to do so.

5. Physical setup of the hydrodynamic computation

In addition to the general setup given in the steering file, a number of physical parameters may or should be specified upon a simulation.

5.1 Hydrostatic pressure hypothesis

Firstly, it should be specified whether one wants to use the hydrostatic pressure hypothesis or not. That choice is made using the keyword `NON-HYDROSTATIC VERSION` which, by default, has been set to YES since release 8.0. As a reminder, the hydrostatic pressure hypothesis (`NON-HYDROSTATIC VERSION` = NO) consists in simplifying the W vertical velocity, ignoring the diffusion, advection and other terms. Therefore, the pressure at a point is only related to the weight of the overlying water column and to the atmospheric pressure at the surface. Without the hydrostatic pressure hypothesis (`NON-HYDROSTATIC VERSION` = YES), TELEMAC-3D solves a W vertical velocity equation which is similar to the U and V equations, with the additional gravity term.

5.2 Modelling turbulence

The Reynolds numbers ($Re = U.L/\upsilon$) reached for tidal flows or in an estuary are excessively high and illustrate basically turbulent flows (L, the scale of eddies, for example, assumes the value of water depth h for a vertically homogeneous flow). For such a kind of flow, the turbulence-induced momentum prevails (in relation to the molecular diffusion). That diffusion is strictly defined by a tensor which could be anisotropic.
The concept of eddy scale, however, is spatially constrained by the horizontal and vertical scales of the modelled domain. At sea, for example, a one kilometre long cape can generates eddies the dimensions of which are horizontally related to that scale. Vertically, however, the eddy size is constrained by the water depth or even more by possible effects of stratifications. Synthetically, a common practice consists in separating the vertical and horizontal turbulence scales which are not relevant to the same dynamics for the standard applications of TELEMAC-3D. That involves defining horizontal as well as vertical viscosities rather than a single viscosity. On the open sea, for instance, the horizontal and vertical viscosities differ by several orders of magnitude.
Thus, the implementation of TELEMAC-3D requires defining two models of horizontal and vertical turbulence (`HORIZONTAL TURBULENCE MODEL`, `VERTICAL TURBULENCE MODEL`).
Turbulence modelling is an awkward task and TELEMAC-3D offers the user several approach options which are different, but also increasingly complex, and are applicable to velocities as

well as to active and passive tracers.
The Von Karman constant and Prandtl number (ratio between eddy viscosity and eddy diffusivity) can be changed with the keywords KARMAN CONSTANT (default value = 0.4) and PRANDTL NUMBER (default value = 1.).

5.2.1 Constant viscosity

The simplest turbulence model consists in using a constant viscosity coefficient (option for parameters: 1="CONSTANT VISCOSITY", default value). In that case, the latter includes the effects of molecular viscosity and dispersion (refer to Theoretical note [5]). The horizontal and vertical turbulent viscosities are then constant throughout the domain. The global (molecular + turbulent) viscosity coefficients are provided by the user by means of the keywords COEFFICIENT FOR HORIZONTAL DIFFUSION OF VELOCITIES and COEFFICIENT FOR VERTICAL DIFFUSION OF VELOCITIES, set by default to 10^{-6} m^2/s.
The value of that coefficient has a definite effect on both size and shape of the recirculations and eddies. A low value will tend to only dissipate the small-sized eddies, a high value will tend to dissipates large-sized recirculations. The user shall then carefully select that value according to the case studied. Usually, that value becomes a model calibration data by comparison with measurements. Besides, it is worth mentioning that a value bringing about the dissipation of recirculations of a smaller than two mesh cell extent has nearly no influence on the computation (i.e. there is a threshold beyond which the viscosity or turbulence value has substantially no effect).
TELEMAC-3D makes it possible to get a space- and time-variable coefficient. The **VISCOS** subroutine will necessarily be programmed. Within that subroutine, geometrical information, basic hydrodynamic information (water depth, velocity components) and time are made available to the user.
That option theoretically aims at enabling the user to define the turbulent viscosity by programming the **VISCOS** subroutine.

5.2.2 Mixing length (vertical model)

The user also has the opportunity to use a vertical mixing length model (VERTICAL TURBULENCE MODEL: 2="MIXING LENGTH"). The vertical diffusivity of velocities is then automatically computed by TELEMAC-3D by means of the selected mixing length model taking or not taking the effects of density into account. The mixing length model expresses the turbulent viscosity (or diffusion coefficient) as a function of the mean velocity gradient and the mixing length (Prandtl's theory):

$$\nu = L_m^2 \sqrt{2 D_{ij} D_{ij}} \text{ where } D_{ij} = \frac{1}{2}\left(\frac{\partial \bar{U}_i}{\partial x_j} + \frac{\partial \bar{U}_j}{\partial x_i}\right) \tag{5.1}$$

Due to that choice, the user should enter the following options for the mixing length model (MIXING LENGTH MODEL):

- 1: Prandtl (default). Standard Prandtl's model. That formulation suits such flows with a strong barotropic component as the tidal flows,
- 3: Nezu and Nakagawa. Nezu and Nakagawa model,
- 5: Quetin [11]. Better representation of wind drift. In windy weather, a surface boundary layer is formed and viscosity decreases,
- 6: Tsanis [6]. Better representation of wind drift.

The graph below shows the variations of the mixing length for the various models.

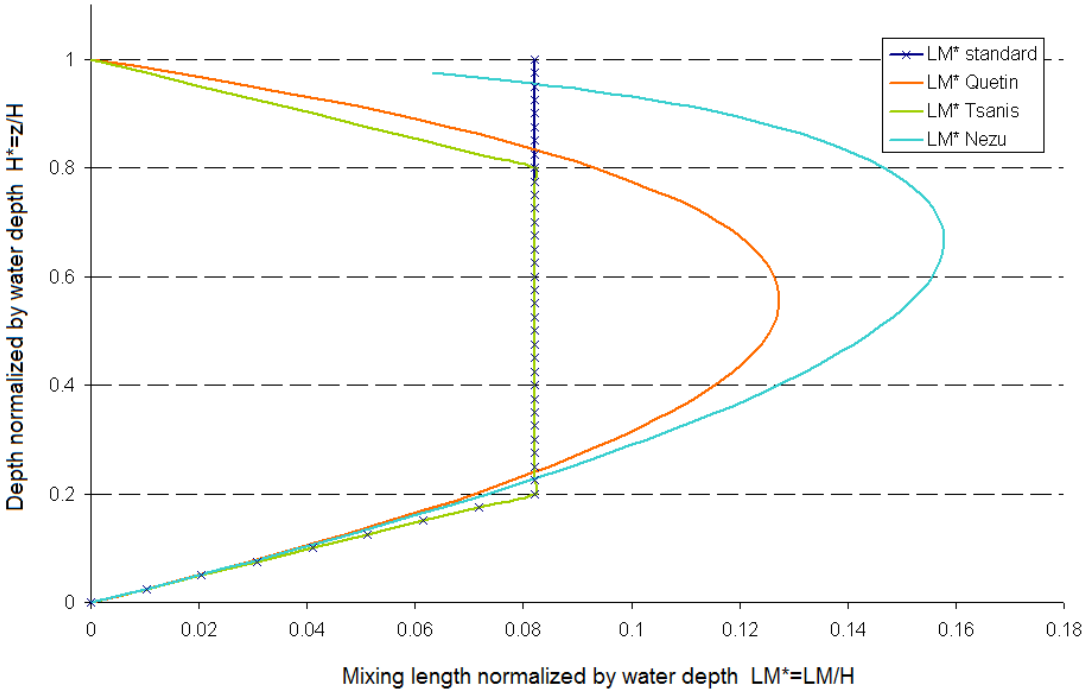

Figure 5.1: Mixing lengths versus depth.

In the presence of a vertical density gradient, the environment stability (respectively the instability) hinders (enhances) the vertical exchanges of mass and momentum.

In order to quantify the effects of the gravity terms in the turbulent power balance, the dimensionless Richardson number is commonly used. It is a local number the value of which can obviously be different at each flow point.

In order to take the mixture reduction into a stable stratified flow into account, a damping law is introduced into the turbulence model according to the Richardson number. The user can set the damping function through the keyword `DAMPING FUNCTION`. The available options are:

- 0: nothing (default value),
- 1: user-performed: Law programming in the **USER_DRIUTI** subroutine,
- 2: Viollet,
- 3: Munk and Anderson.

The graph below illustrates the variation of the Munk and Anderson damping function according to the Richardson number for velocity and salinity. In the case of a stable stratification, the pressure fluctuations more readily transmit a momentum flux than a mass flux and the diffusion coefficient becomes higher for the velocities than for the mass.

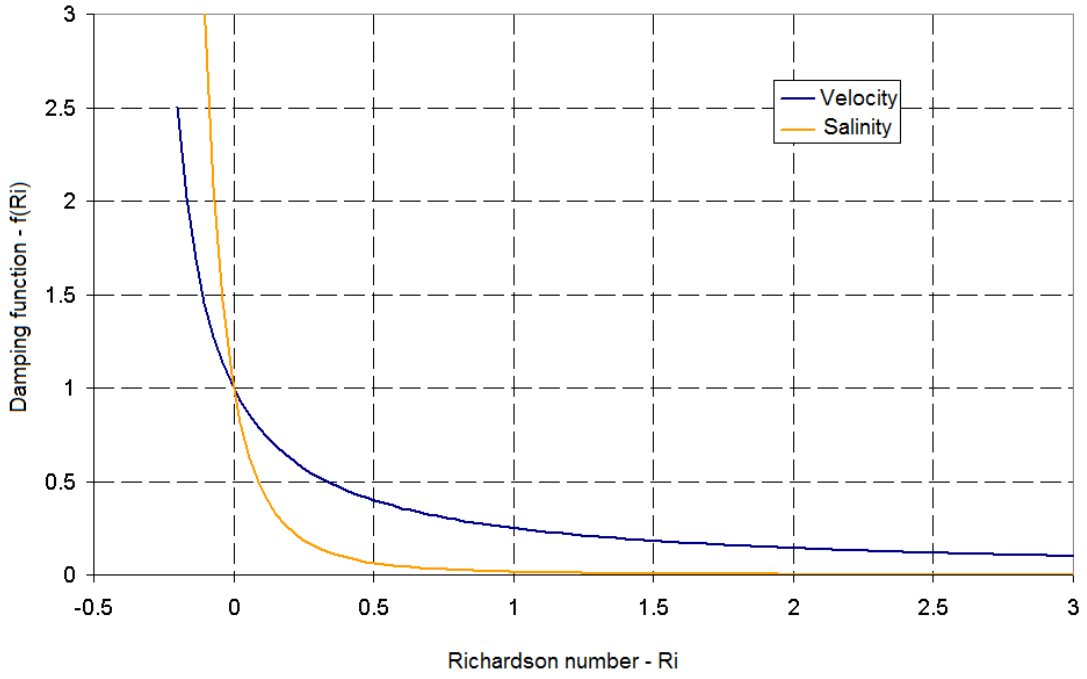

Figure 5.2: Munk and Anderson damping function.

When using the mixing length model, the user can also configure the calculation of the vertical derivative of the velocities with the keyword `VERTICAL VELOCITY DERIVATIVES`. Default value 1 corresponds to a vertical derivative that is linear. Value 2 activates a logarithmic computation between the bottom and 0.2 times the water depth. This allows getting better results when modelling the velocity profile near the bottom. This algorithm is implemented in the subroutine **VISCLM**.

5.2.3 Smagorinsky

That option is activated by setting the horizontal or vertical turbulence model to 4 (Smagorinsky). If `VERTICAL TURBULENCE MODEL` = 4, `HORIZONTAL TURBULENCE MODEL` is automatically set to 4 even though the user has set to a different value before. On the contrary, a Smagorinski model in the horizontal directions (`HORIZONTAL TURBULENCE MODEL` = 4) can be used with every possible value for `VERTICAL TURBULENCE MODEL`.

The Smagorinsky scheme is recommended, in particular, in the presence of a highly non-linear flow [7].

5.2.4 k-ε

TELEMAC-3D gives an opportunity to use the so-called k-ε model as proposed by Rodi and Launder for solving the turbulence equations. That model is activated by setting the keywords of turbulence models (`HORIZONTAL TURBULENCE MODEL` and `VERTICAL TURBULENCE MODEL`) to the value 3.

The k-ε model is defined through a couple of equations solving the balance equations for k (turbulent energy) and ε (turbulent dissipation). Applying the k-ε model often requires using a finer two-dimensional mesh than the constant viscosity model and then increases the computation times.

For detailed information about the formulation of the mixing length and k-ε models, the user can refer to the TELEMAC-3D Theoretical Note.
Strictly speaking, except for the constant viscosity model, the diffusion coefficient should be equal to the molecular diffusion of water:

```
COEFFICIENT FOR HORIZONTAL DIFFUSION OF VELOCITIES = 1.D-6
COEFFICIENT FOR VERTICAL DIFFUSION OF VELOCITIES = 1.D-6
```

The default value of that viscosity may have to be increased to ensure a minimum diffusion, especially during the first time steps of the computation.
There are two options to compute the lateral boundary conditions of k and ε (in **KEPCL3** subroutine) with the keyword `OPTION FOR THE BOUNDARY CONDITIONS OF K-EPSILON`:

- 1 = No turbulence (k and ε takes the minimum values **KMIN** and **EMIN** defined in subroutine **CSTKEP**), which is the default value,
- 2 = Hans and Burchard's formula (introduced in release 7.0).

Since release 8.1, TELEMAC-3D can be coupled with the GOTM library (General Ocean Turbulence Model). It enables to use a turbulence model library over the vertical by setting the keyword `VERTICAL TURBULENCE MODEL` to 6="STANDARD GOTM" and providing an ASCII file containing GOTM parameters with the keyword `GOTM STEERING FILE`. An example of such a file can be found in the TELEMAC-3D example stratification. As GOTM is a 1DV water column model, the turbulence model can only be used over the vertical and the user can choose another turbulence model in the x and y directions with the keyword `HORIZONTAL TURBULENCE MODEL`.

5.2.5 $k-\omega$ model

TELEMAC-3D gives an opportunity to use a k-ω model for solving the turbulence equations. That model is activated by setting the keywords of turbulence models (`HORIZONTAL TURBULENCE MODEL` and `VERTICAL TURBULENCE MODEL`) to the value 7. It is not possible to mix $k-\varepsilon$ and $k-\omega$ in two different directions.
Parameters definition for the $k-\omega$ model is common with the $k-\varepsilon$ model one.

5.2.6 Spalart-Allamaras and DES models

The use of Spalart-Allmaras model (resp. DES for Detached Eddy Simulation) is activated by giving the value 5 (resp. 9) to the keywords `HORIZONTAL TURBULENCE MODEL` and `VERTICAL TURBULENCE MODEL`. In that casem it is mandatory the same turbulence model for both directions `HORIZONTAL TURBULENCE MODEL` = `VERTICAL TURBULENCE MODEL` = 5.
Parameters definition for the Spalart-Allmaras model is common with the $k-\varepsilon$ model one.

5.3 Setting up the friction

The bottom or sidewall friction reflects the continuity of the constraint at the fluid-solid interface. Knowing the constraint involves knowing the flow in the vicinity of the bottom. The turbulence models provide a modelling for that flow.
The constraint can be written in several forms:

$$\boldsymbol{\tau} = \mu \frac{\partial \boldsymbol{U}}{\partial n} = -\frac{1}{2}\rho C_f \sqrt{U^2+V^2}\boldsymbol{U} \tag{5.2}$$

$$= -\rho U^{*2} \tag{5.3}$$

where U^* denotes the friction, C_f - a dimensionless friction, U - the velocity of current recorded far enough from the wall.
The friction condition is then provided:

- Either by a turbulence model which indicates the constraint through a friction velocity formulation,
- Or by the knowledge of the friction coefficient C_f and the related velocity U (here, the vertically-averaged velocity). That approach will then use the Chézy, Strickler, Manning... laws.

The same approach is adopted for both sidewalls and bottom.

5.3.1 Bottom friction

The friction law used for the bottom friction modelling is set by the keyword `LAW OF BOTTOM FRICTION` which can assume the following values:

- 0: No friction,
- 1: Haaland law,
- 2: Chézy law,
- 3: Strickler law,
- 4: Manning law,
- 5: Nikuradse law (default value since release 8.1).

As regards the 1-5 values, the value of the friction coefficient corresponds to the selected law, and shall be given by means of the keyword `FRICTION COEFFICIENT FOR THE BOTTOM`. This is of course only valid if the friction is constant in time and space. The default value for that parameter is 0.01 which corresponds to 1 cm for a Nikuradse law (old default value = 60 for Chézy law until release 8.0).
The computation of turbulent constraint at the bottom depends on the velocity profile above the bottom (within the boundary layer). The profile depends on the ratio of the wall asperity size to the viscous sub-layer thickness (for further details, refer to the Theoretical Manual). When the asperities are larger than the viscous sub-layer thickness, the latter cannot be established and the friction regime is rough. On the other hand, when there is a viscous sub-layer, the friction regime is smooth.
The computation of the turbulent constraint depends on the keyword `TURBULENCE REGIME FOR THE BOTTOM`. The available options are:

- 1: smooth regime,
- 2: rough (default value).

In smooth friction regime conditions, the friction law is not used and the constraint is computed from Reichard law of velocity profile (a law giving the friction velocity value U^*.
In rough friction regime conditions and for the bottom friction laws 0, 2, 3 and 4, the constraint is computed from the friction velocity U^* and its relation to the C_f coefficient. For law 5, the friction velocity is computed from the velocity profile within the logarithmic layer and from the asperity size k_s (`FRICTION COEFFICIENT FOR THE BOTTOM`).

5.3.2 Sidewall friction

The friction law used to model the sidewall friction is set by the keyword LAW OF FRICTION ON LATERAL BOUNDARIES which can take the following values:

- 0: No friction (default value),
- 5: Nikuradse law.

The size of asperities (which is used in the Nikuradse law) is given by the FRICTION COEFFICIENT FOR LATERAL SOLID BOUNDARIES (default value 0.01 m since release 8.1, old default value = 60! until release 8.0).
The friction is activated using the keyword TURBULENCE REGIME FOR LATERAL SOLID BOUNDARIES.
The available options are:

- 1: smooth regime,
- 2: rough (default value).

That option changes the formulation of the velocity profile and consequently, the friction velocity. See previous section for more information.

5.4 Punctual source terms

TELEMAC-3D offers an opportunity to place momentum sources or sinks in any point of the domain.
The user has two options to place horizontally the various sources:

1. using the keywords ABSCISSAE OF SOURCES and ORDINATES OF SOURCES. They are arrays of reals giving the co-ordinates of sources, in meters. Actually, TELEMAC-3D will position a source at the closest mesh point to the point as specified by these keywords. The software will determine itself the number of sources according to the number of values given to each keyword,
2. using the keyword GLOBAL NUMBERS OF SOURCE NODES. This is an array of integers which contains the global numbers of nodes in the 2D mesh that correspond to source point locations. The software will determine itself the number of sources according to the number of values given to the keyword.

The vertical positioning of the sources is done using the keyword ELEVATIONS OF SOURCES. TELEMAC-3D places the sources on the nearest mesh level. In this case, it is recommended to use fixed levels at sources elevations in order to avoid unwanted vertical movement of the sources during the simulation. Also note that the sources cannot be placed on the bottom level (level number 1). This is to ensure an impermeable bottom boundary.
At each source, the user should specify the liquid flow rate. That liquid flow rate is given (in m^3/s) using the keyword WATER DISCHARGE OF SOURCES.
In case of sources with variable characteristics, the user can then either use specific programming in the **USER_T3D_DEBSCE** subroutine (and **USER_T3D_TRSCE** in the presence of tracer), or use the source file whose name is given by the keyword SOURCES FILE. This file has exactly the same structure as the liquid boundaries file. An example is shown below with two sources and two tracers. Between two specified times, the information used by TELEMAC-3D at sources is obtained by linear interpolation.

```
#
#  FLOW RATES ANS TRACERS CONCENTRATIONS AT SOURCES 1 ET 2
#
#  T IS THE TIME
#
#  Q(1) IS FLOW RATE AT SOURCE 1
#  Q(2) IS FLOW RATE AT SOURCE 2
#
#  TR(1,1) IS TRACER 1 CONCENTRATIONS AT SOURCE 1
#  TR(1,2) IS TRACER 2 CONCENTRATIONS AT SOURCE 1
#  TR(2,1) IS TRACER 1 CONCENTRATIONS AT SOURCE 2
#  TR(2,2) IS TRACER 2 CONCENTRATIONS AT SOURCE 2
#
#
T      Q(1)      TR(1,1)     TR(1,2)     Q(2)       TR(2,1)    TR(2,2)
s      m3/s        C           C         m3/s         C          C
0.      0.        99.         20.         0.         30.        40.
2.      1.        50.         20.         2.         30.        20.
4.      2.        25.         80.         4.         30.        20.
```

Besides, TELEMAC-3D is capable of taking into account an injection velocity (in m/s) at the sources in the dynamic equations. By default, the injection takes places without any momentum input. The user may prescribe a particular velocity. If the latter is constant throughout the simulation, then its value can be given using the keywords `VELOCITIES OF THE SOURCES ALONG X`, `VELOCITIES OF THE SOURCES ALONG Y` and `VELOCITIES OF THE SOURCES ALONG Z` (default values = 0.). Otherwise, the user should program the **USER_SOURCE** subroutine in order to amend **USCE** (for the velocity at sources along X) and **VSCE** (for the velocity at sources along Y). The user can use the time and all the parameters of the sources within that subroutine.

From a theoretical point of view, complete mass conservation can only be ensured if the source is treated as a Dirac function and not as a linear function. The type of treatment is indicated by the user with the keyword `TYPE OF SOURCES`, which can have a value of 1 (linear function, default value) or 2 (Dirac function). It should be noted that in the second case, the solutions are of course less smoothed. It is the same implementation as in TELEMAC-2D and the Dirac option is recommended with a big number of sources. The maximum number of sources is set to 20 by default but it can be changed by the user with the keyword `MAXIMUM NUMBER OF SOURCES`. This avoids changing the previously hardcoded values (until release 7.0), which required recompiling the whole package.

5.5 Setting up the water-atmosphere exchanges

5.5.1 The wind

TELEMAC-3D can be used to simulate flow while taking into account the influence of a wind blowing on the water surface. The logical keyword `WIND` is used first of all for determining whether this influence is taken into account (default value = NO, i.e. no influence).

If so, the coefficient is then provided with the keyword `COEFFICIENT OF WIND INFLUENCE` (default value = 1.55×10^{-6} since release 8.0, 0. before), see below.

Wind effect can be managed using the keyword `OPTION FOR WIND` (default = 0). This keyword can have the following values:

- 0: this means no wind effect (this is equivalent to put the keyword `WIND` to FALSE),

- 1: wind is constant in time and space, wind speed in directions x and y are supplied with the keywords `WIND VELOCITY ALONG X` and `WIND VELOCITY ALONG Y`. Default values for these three coefficients are 0.,
- 2: wind variable in time, constant is space, it is given through the formatted file `ASCII ATMOSPHERIC DATA FILE`. Shortnames WINDS and WINDD (for wind velocity magnitude and direction) or WINDX and WINDY (for x and y wind velocity components) are to be written in the headline of this file (see the section 3.1 of the WAQTEL user manual),
- 3: wind is variable in time and space, this option is not implemented for `ASCII ATMOSPHERIC DATA FILE` as there are multiple choices of implementation. In this case, the user must program him/herself the **METEO** subroutine. If the `ASCII ATMOSPHERIC DATA FILE` does not follow the format expected by the **METEO_TELEMAC** module and implementation is required to handle them outside this module, the keyword `FREE FORMAT FOR ATMOSPHERIC DATA FILE` has to be set to YES (default = NO). An example of implementation is given in the TELEMAC-2D validation test case "wind_txy" (in folder examples/telemac2d/wind_txy).

  If using a `BINARY ATMOSPHERIC DATA FILE` not following the format expected by the **METEO_TELEMAC** module and implementation is required to handle them outside this module, the keyword `FREE FORMAT FOR ATMOSPHERIC DATA FILE` has to be set to YES. Until release 8.4, for the validation test case "wind_txy" (in folder examples/telemac2d/wind_txy), it has to be done that way but since release 8.5, no specific personal implementation is to be done by user provided that variables names for wind, pressure and air temperature (+ other meteo data common with WAQTEL, see WAQTEL user manual) are among WINDX, WINDY, WINDS, WINDD, PATM, TAIR. The name of the variables can be changed in the `BINARY ATMOSPHERIC DATA FILE` by running:
  run_telfile.py alter –rename 'old name var=new name var' name_bin_meteo_file
  one variable by one variable e.g. Reference times are to be added in the keywords `ORIGINAL DATE OF TIME` and `ORIGINAL HOUR OF TIME` and `FREE FORMAT FOR ATMOSPHERIC DATA FILE` let to its default value (= NO). Last, but not least for big files, time of interpolation of `BINARY ATMOSPHERIC DATA FILE` has been optimised since release 8.5 when using **METEO_TELEMAC** module.

  For users who still want to use the old way with subroutines **METEO_FROM_BINARY_FILE**, **METEO_SET_VAR_NAMES** and **READ_BIN_2D** in particular who do not want to change the format of the `BINARY ATMOSPHERIC DATA FILE` (e.g. the name of the wind and pressure variables), it is still possible by adding these 3 subroutines + modified **METEO** subroutine in `FORTRAN FILE` as done for wind_txy/t2d_wind_txy_bin_old.cas and user_fortran-bin_old folder. In that case, set the keyword `FREE FORMAT FOR ATMOSPHERIC DATA FILE` to YES.

The coefficient of wind influence hides complex phenomena. In fact, the influence of the wind depends on the smoothness (or, lack of it) of the free surface and the distance over which it acts (called the "fetch"). The coefficient value can be obtained from many different formulas.
This is the formula used by the Institute of Oceanographic Sciences (United Kingdom):

if $\|\boldsymbol{U}_{wind}\| < 5$ m/s $\quad a_{wind} = 0.565 \times 10^{-3}$
if $5 < \|\boldsymbol{U}_{wind}\| < 19.22$ m/s $\quad a_{wind} = (-0.12 + 0.137\|\boldsymbol{U}_{wind}\|)10^{-3}$
if $\|\boldsymbol{U}_{wind}\| > 19.22$ m/s $\quad a_{wind} = 2.513 \times 10^{-3}$

This formula can be activated by setting the keyword `COEFFICIENT OF WIND INFLUENCE VARYING WITH WIND SPEED` to YES (default value is YES since release 8.2). If YES, the value of `COEFFICIENT OF WIND INFLUENCE` is overwritten.

The parameter COEFFICIENT OF WIND INFLUENCE asked for by TELEMAC-3D is: $a_{wind}(\rho_{air}/\rho)$ and not only a_{wind}. ρ_{air} is approximately 1.2 kg/m^3 and ρ is approximately 1,000 kg/m^3. Thus it is necessary to divide the value of a_{wind} by 1,000 to obtain the value of the TELEMAC-3D keyword. The default value of COEFFICIENT OF WIND INFLUENCE has been set to 1.55×10^{-6} since release 8.0.

The whole formulation used to consider the wind effects, through the keyword COEFFICIENT OF WIND INFLUENCE (refer to the Theoretical Note for the definition of that coefficient), on the surface flows is fully stated in the **BORD3D** subroutine.

If the wind velocity is space- or time-variable, the user should modify the **METEO** subroutine.

Warning:
The **METEO** subroutine is provided to define the wind velocity and direction even though they are space- or time-variable. The **BORD3D** subroutine describes the law of wind-induced drift of water bodies.

If there are tidal flats or dry zones in the domain, the wind may trigger unphysical velocities as it becomes the only driving term in the equations. To avoid this, the influence of the wind is cancelled below a threshold value of depth, with the keyword THRESHOLD DEPTH FOR WIND (default value = 1 m). Be careful if the model includes shallow waters, lower than this value.

5.5.2 The pressure

Atmospheric pressure is taken into account by setting the keyword AIR PRESSURE to YES (the default value is NO). The pressure value is set in the **METEO** subroutine with the keyword of VALUE OF ATMOSPHERIC PRESSURE. By default, the latter initializes a pressure of 10^5 Pa ($\approx$ 1 atm) over the whole domain.

Atmospheric pressure can be given through the formatted file ASCII ATMOSPHERIC DATA FILE. Shortname PATM is to be written in the headline of this file (see the section 3.1 of the WAQTEL user manual)

5.5.3 Rain or evaporation

The modelling of the influence of precipitation or evaporation is activated with the logical keyword RAIN OR EVAPORATION (default value = NO). The value of the contribution or the loss of water at the surface is specified using the keyword RAIN OR EVAPORATION IN MM PER DAY which default value is 0. (a negative value reflects an evaporation).

Rain and evaporation can also vary in time and space. They can be introduced through the ASCII ATMOSPHERIC DATA FILE or BINARY ATMOSPHERIC DATA FILE. Shortnames RAINI and RAINC (with a final I for rain classically interpolated as other meteo variables or with a final C as a cumulated variable during a certain period of time) are to be written in the headline of this file (see the section 3.1 of the WAQTEL user manual). Also see water quality, wind and rain validation test cases (in folders examples/telemac2d or examples/waqtel).

In case of calculation with consideration of tracers, it is possible to specify the contribution related to the rain with the keyword VALUES OF TRACERS IN THE RAIN (default value is 0.). It is important to note that, in the case of evaporation, no tracer is taken into account in the water loss, which is incorrect if the tracer is the temperature.

5.5.4 Atmosphere-water exchange models

Before release 7.0, heat exchange between water and atmosphere could have been done with a linearised formula of the balance of heat exchange fluxes at the free surface. An example of an exchange with a constant atmosphere temperature and a constant sea salinity used to be given as standard (as comments) through a direct programming in the **BORD3D** subroutine.
A much more elaborated model has been introduced in TELEMAC-3D. Since release 7.2, this model has been integrated in WAQTEL and TELEMAC-3D has to be coupled with WAQTEL to activate it (`COUPLING WITH = 'WAQTEL'`), with `WATER QUALITY PROCESS = 11` (since release 8.0, 5 previously) in the WAQTEL steering file. This module calculates the complete balance of exchanged fluxes involved:

- The solar radiation,
- The atmospheric radiation,
- The water radiation,
- The latent heat due to evaporation,
- The sensitive heat of conductive origin.

It takes into account the solar radiation penetration in the water column.
The choice of the heat exchange model can be done with the keyword `ATMOSPHERE-WATER EXCHANGE MODEL` in the WAQTEL steering file (default value = 0: no exchange model). Value 1 will use with the linearised formula at the free surface, whereas value 2 will use with the model with complete balance.
These calculations require additional data (wind magnitude and direction, air temperature, atmospheric pressure, relative humidity, nebulosity and rainfall, all these variables may vary in time) in a standard format, rather defined in the `ASCII ATMOSPHERIC DATA FILE` of the TELEMAC-3D steering file, see the example "heat_exchange". Since release 8.2, the format of this file is flexible, i.e. the order of columns is free but the user has to use right shornames for every meteo variable:

- WINDS and WINDD (resp. WINDX and WINDY) for wind velocity magnitude and direction (resp. x and y components of wind velocity),
- TAIR for air temperature (in °C),
- PATM for atmospheric pressure (in hPa = mbar),
- HREL for relative humidity (in %),
- CLDC for cloud cover or nebulosity (in octas for WAQTEL or tenths for KHIONE),
- RAINI or RAINC for rain (in mm/s), see previous section for the meaning of the I and C additional charactere,

The format may be changed but the user has to change the implementation of the reading if not standard and the interpolation of the meteorological data (see **METEO** subroutine, and/or the **METEO_TELEMAC** module since release 8.2). When using the complete module, evaporation is calculated by TELEMAC-3D, but the user has to provide rainfall data with units homogeneous with length over time. The user can also define metocean data with the `BINARY ATMOSPHERIC DATA FILE` but has to change some subroutines to deal with it.
The main developments of this module are implemented in the module **EXCHANGE_WITH_ATMOSPHERE**.

Some physical parameters or formulae can be changed with the helps of some keywords in the WAQTEL steering file (default values are the mean values): e.g. the type of sky related to the luminosity of the site with the WAQTEL keyword `LIGHTNESS OF THE SKY` (1: very pure sky, 2: mean pure sky or 3: industrial zone sky) or the `FORMULA OF ATMOSPHERIC RADIATION` (1: Idso and Jackson (1969), 2: Swinbank (1963) which is the default formula, 3: Brutsaert (1975) or 4: Yajima Tono Dam (2014)). Some other physical parameters have been hard-coded in the module, often imposed as the mean value: e.g. the type of cloud (cirrus, cirro stratus, alto cumulus, alto stratus, stratus), but these values can be changed in the module **EXCHANGE_WITH_ATMOSPHERE**).
Because the site of a study may not be equipped with local wind measurements and these kinds of data are available at a different location, possibly far from the studied site, a wind function is used. This is a linear function with a single coefficient of calibration b : $f(U_2) = b(1+U_2)$ where U_2 is the wind velocity at 2 m high.
To get the wind velocity at 2 m high from classical wind data at 10 m high, a roughness length of $z_0 = 0.0002$ m has been chosen in the code, that leads to $U_2 \approx 0.85U_{10}$. This value of 0.85 (or the roughness length) may be changed by the user if needed.
Examples of solar radiation penetration are given in the **CALCS3D_THERMICV** subroutine. Two laws are suggested: the first one uses the *in situ* measurements of Secchi length and is recommended if available; the second one uses two exponential laws that may be difficult to calibrate and require an estimation of the type of water from turbidity.
Except for the coefficient to model the penetration of solar radiation in the water column, the parameter b that appears in the wind function is the single calibration parameter of this module. Its value is given by the keyword `COEFFICIENT TO CALIBRATE THE ATMOSPHERE-WATER EXCHANGE MODEL` in the WAQTEL steering file (default value = 0.0025 but recommended values are between 0.0017 and 0.0035). This keyword is both used for the linearised formula at the free surface and the model with complete balance (values 1 and 2 for the keyword `ATMOSPHERE-WATER EXCHANGE MODEL` in the WAQTEL steering file).

5.6 Astral potential

When modelling large maritime areas, it is sometimes necessary to take into account the effect of astral forces generating tide inside the area. For this, the user has several keywords at his disposal.
First of all, the logical keyword `TIDE GENERATING FORCE` (default value NO) allows these phenomena to be taken into account. If YES, the keyword `SPHERICAL COORDINATES` has to be activated, it is impossible to account tide generating force in cartesian coordinates.
The keyword `LONGITUDE OF ORIGIN POINT` must be positioned at the right value (default = 0 degrees).
Lastly, the two keywords `ORIGINAL DATE OF TIME` (format YYYY;MM;DD) and `ORIGINAL HOUR OF TIME` (format HH;MM;SS) must be used to give the beginning time of the simulation. This information is necessary for TELEMAC-3D to compute the respective position of the moon and the sun.

5.7 Consideration of wave driven currents

It is possible to take into account the wave driven currents by retrieving information calculated by a wave propagation module of the TELEMAC modelling system (mainly TOMAWAC but also ARTEMIS). In the current release, only taking into account a steady state is possible. The procedure is as follows:

- Perform a calculation of wave propagation on the same mesh as the TELEMAC-3D calculation requesting the storage of the driving forces. In the case of TOMAWAC, the variables **FX** and **FY**,
- Get the wave results file and provide its name through the keyword `BINARY DATA FILE 1`,
- Activate the keyword `WAVE DRIVEN CURRENTS` (default = NO),
- Fill the keyword `RECORD NUMBER IN WAVE FILE`. This value corresponds to the iteration number stored in the wave file which must be taken into account by TELEMAC-3D. Usually, this is the last iteration stored, by default this number is set to 1. The name of the variables to read is "FORCE FX" and "FORCE FY", but this can be changed within the **TRISOU** subroutine.

In the current release of TELEMAC-3D, this driving force is considered as being constant over the vertical.
If the user wishes to take into account several results of the wave propagation module (e.g. to take into account changes in the level of the sea), FORTRAN programming is required.

5.8 Other physical parameters

When modelling large areas, the influence of the Coriolis force of inertia has to be taken into account. This is done by activating the logical keyword `CORIOLIS` (which is set to NO by default). In such a case, the value of the Coriolis coefficient (refer to the Theoretical Note) is defined by the keyword `CORIOLIS COEFFICENT` (default value is 0.). The latter should be computed according to latitude λ through the formula:

- FCOR = $2\omega\sin(\lambda)$ where ω is the angular velocity of the Earth, equal to 7.2921 $\times$ 10^{-5} rad/s and λ is the average latitude of the model.

The components of the Coriolis force are thus:
FU = FCOR $\times$ V and FV = -FCOR $\times$ U
In the case of very large domains such as portions of oceans, it is necessary to carry out a simulation with spherical coordinates, in which case the Coriolis coefficient is adjusted automatically at each point of the domain by activating the keyword `SPHERICAL COORDINATES` (see 11.3). Its default value is NO.
Gravity acceleration can be changed with the keyword `GRAVITY ACCELERATION` whose default value is set at 9.81 m/s^2.

6. Numerical setup of the computation

The numerical setup is comparatively common to a hydrodynamic computation alone or with a tracer. Thus, in the following sections of this chapter, the numerical parameters as applied to the solution of a tracer equation are integrated into the hydrodynamic parameters.

6.1 General setup

TELEMAC-3D solves the Navier-Stokes equations in several stages, possibly through the three stages of the fractional step method (see the theoretical note). The first stage consists in finding out the advected velocity components by only solving the convection terms of the momentum equations. The second stage computes, from the advected velocities, the new velocity components by taking into account both diffusion and source terms of the momentum equations. These two solutions enable to get an intermediate velocity field. The third stage computes the water depth from the vertical integration of the continuity equation and momentum equations only including the pressure-continuity terms. This step is called the propagation step.
The user can activate or deactivate, either globally or individually, some of these stages.

6.1.1 Advection step

Whether the advection terms will be taken into account or not will be determined by the logical keyword `ADVECTION STEP` (default value YES). However, even though that keyword is set to YES, then some advection terms can be deactivated by using the following complete keywords (value 0 = "NO ADVECTION "):

- `SCHEME FOR ADVECTION OF VELOCITIES`: for the advection of velocities,
- `SCHEME FOR ADVECTION OF DEPTH`: for taking the advection of depth into account (default value = 5 which is the single choice and now mandatory),
- `SCHEME FOR ADVECTION OF K-EPSILON`: for the advection of power and turbulent dissipation for the $k-\varepsilon$ model or $\tilde{\nu}$ for Spalart-Allmaras model,
- `SCHEME FOR ADVECTION OF TRACERS`: for the advection of tracers (one value per tracer).

See section 6.2 for more information on the possible choices.

6.1.2 Diffusion step

Some diffusion terms can be deactivated by means of the following complete keywords by the following keywords:

- SCHEME FOR DIFFUSION OF VELOCITIES,
- SCHEME FOR DIFFUSION OF TRACERS (one value common to all tracers),
- SCHEME FOR DIFFUSION OF K-EPSILON for both $k-\varepsilon$ model and Spalart-Allmaras model.

The 0 value at each keyword cancels the diffusion, whereas the 1 value (default value) leads to the implicit calculation of diffusion.

6.1.3 Propagation step

Since the release 6.0 of TELEMAC, the propagation step (for velocity and water depth) is necessarily treated with TELEMAC-2D "wave equation" option.
The propagation step can be linearized by activating the keyword LINEARIZED PROPAGATION (default value = NO), in particular when running a test case for which an analytical solution is available in the linearized case. It is then necessary to set the water depth around which the linearization is to be performed, by using the keyword MEAN DEPTH FOR LINEARIZATION (default value = 0.).
With the "wave equation" option, the keyword FREE SURFACE GRADIENT COMPATIBILITY can be used. A value lower than 1. (which is the default value) makes it possible to delete the spurious oscillations of the free surface, but slightly alters the consistency between the water depth and the velocities in the continuity equation.
When using the non-hydrostatic version, it is possible to enable an estimation of the dynamic pressure gradient in the treatment of the wave equation. This option is enabled by the logical keyword DYNAMIC PRESSURE IN WAVE EQUATION but it leads a dilemma:

- If the keyword is set to YES, the dynamic pressure gradient is taken into account when calculating the evolution of the water depth (advantage), but the evolution of the water depth is not known when calculating the dynamic pressure (disadvantage),
- If the keyword is set to NO (which is the default value), the dynamic pressure is calculated taking into account the evolution of the water depth (advantage) but it is not taken into account in the calculation of the water depth (disadvantage).

Therefore, when the effect of the dynamic pressure is important (nonlinear waves), it is recommended to set this keyword to YES in combination with setting sub-iterations for nonlinearities (see test case NonLinearWave).

6.2 The advection scheme

The procedure for taking the advection terms into account is individualized for each of the variables liable to be processed. It has previously been explained that the zero option corresponds to a deactivation of the term.

6.2.1 Advection of the three-dimensional variables

The advection schemes of the three-dimensional variables (i.e. all the variables except the water depth) are (for further details about these options, the reader shall refer to the Theoretical Note):

- 0: Deactivation,
- 1: Method of characteristics. That method involves mutually independent advection and diffusion steps. The method consists in writing that the value of the advected variable is equal to the value of the same variable in the previous instant traced back on the path travelled during the time step,
- 2: Explicit scheme + SUPG (Streamline Upwind Petrov Galerkin). That method uses test functions which are deformed in the direction of the current for the variational method,
- 3: Explicit Leo Postma scheme,
- 4: Explicit scheme + MURD (Multidimensional Upwind Residual Distribution) N scheme,
- 5: Explicit scheme + MURD PSI scheme,
- 13: Explicit Leo Postma scheme for tidal flats,
- 14: Explicit scheme + MURD (Multidimensional Upwind Residual Distribution) N scheme for tidal flats.

The latter five schemes are primarily recommended for the tracers, since they are advantageous in being conservative and monotonic, i.e. they do not generate any numerical oscillation. On the other hand, they are more diffusive than SUPG. In that respect, scheme 5 is an improvement to scheme 4, being less diffusive perpendicularly to the flow but quite obviously somewhat more computation time-consuming. Nevertheless, it is still globally less time-consuming than SUPG. Distributive schemes (options 3, 4, 5, 13 or 14) are schemes whose stability is conditioned by a Courant number less than 1. When using one of these schemes, at each time step, TELEMAC-3D performs a test for checking the Courant number point by point. In case it exceeds the value 1, TELEMAC-3D will automatically execute sub-iterations to satisfy the stability criterion. However, if the number of sub-iterations exceeds 100, TELEMAC-3D considers that the treatment of the advection term is problematic and the calculation is interrupted, printing an error message in the list control.
The default value for both `SCHEME FOR ADVECTION OF VELOCITIES` and `SCHEME FOR ADVECTION OF K-EPSILON` is 5 since release 8.1 (old value until release 8.0 was 1 i.e. method of characteristics). Combined with the new default values of the keywords `SCHEME OPTION FOR ADVECTION OF VELOCITIES` = 4 and `SCHEME OPTION FOR ADVECTION OF K-EPSILON` = 4, that means that a LIPS scheme is used (see 6.2.3). That value is advisable, since it is satisfactory in many instances. The default value for the tracers is still 5 and also with the default value of `SCHEME OPTION FOR ADVECTION OF TRACERS` equal to 4. It is the most reliable scheme, since the "mass" conservation of the active tracers is a frequently essential point in TELEMAC-3D.
Since release 6.0, the value concerning the water depth advection scheme is ignored by TELEMAC-3D. The optimum advection scheme is automatically selected by the software (conservative scheme).
According to the schemes used, the mass conservation can be improved by performing sub-iterations. That consists in an updating, for one given time step, both advective field and propagative field during several sub-iterations. Upon the first sub-iteration, the field of velocities

is yielded by the results achieved during the previous time steps. Through that procedure, the non-linearities can be taken into account in a better way and the mass conservation can be significantly improved in the cases of schemes 2 and 3. The number of sub-iterations is set by the keyword NUMBER OF SUB ITERATIONS FOR NON LINEARITIES, the default value of which is 1 (also refer to the Theoretical Note).
The SUPG scheme can be configured using specific keywords (see 6.2.2).
The MURD schemes present several options described in 6.2.3.

6.2.2 Configuration of the SUPG scheme

When using the SUPG method, the user has to determine the type of upwinding desired with the keyword SUPG OPTION which is an array of 4 integers related to the velocities, the water depth, tracer(s) and k-ε model (or Spalart-Allmaras model) respectively. Default value = (1;0;1;1).
The possible values are:

- 0: no upwinding,
- 1: upwinding with the classical SUPG method, i.e. the upwinding is 1,
- 2: upwinding with the modified SUPG method, i.e. the upwinding is equal to the Courant number.

In theory, option 2 is more accurate when the Courant number is less than 1, but should not be used for large Courant numbers. Thus, option 2 should be used in models for which the Courant number remains low. If the Courant number cannot be estimated, it is strongly recommended to use option 1 (which can be considered as more ”universal”).
In practice, only the 1st coefficient is used (and applied to U, V and W). It is also this coefficient which is applied to tracer(s), k and ε if needed. The only possible value for the 2nd coefficient is 0, so no need to change it.
At the end, the user only needs to give one value to the keyword SUPG OPTION with TELEMAC-3D.

6.2.3 Configuration of the MURD schemes

The N and PSI residual distribution schemes present different options that allow to increase the accuracy of the numerical scheme or to deal with tidal flats. The keyword SCHEME OPTION FOR ADVECTION OF VELOCITIES (which can be found also for k-ε and tracers) specifies the desired option and must be used coupled with the keyword SCHEME FOR ADVECTION OF VELOCITIES = 4 (or 5). It can be set to:

- 1: explicit scheme,
- 2: first order predictor-corrector scheme,
- 3: second order predictor-corrector scheme,
- 4: locally semi-implicit predictor-corrector scheme (for tidal flats): LIPS (default value since release 8.1).

The default value for SCHEME OPTION FOR ADVECTION OF VELOCITIES is 4 (i.e. LIPS) since release 8.1 which works with tidal flats (old value = 1 i.e. explicit scheme until release 8.0). If there are no tidal flats in the domain, using option 2 (first order predictor-corrector scheme) may accelerate the computation but it does not work with tidal flats.

Like in TELEMAC-2D, the predictor-corrector schemes need an additional parameter which represents the number of iterations for every time step (or sub-time step) to converge to the solution. The keyword NUMBER OF CORRECTIONS OF DISTRIBUTIVE SCHEMES (default value = 1) plays this role and it is useful for unsteady cases. For quasi-steady flows, the number of corrections does not have a large impact on the solution, so it can be set to 0. On the other hand, for unsteady flows, it is suggested to set the keyword NUMBER OF CORRECTIONS OF DISTRIBUTIVE SCHEMES = 2 (at least), which is a good compromise between accuracy and computational time. Indeed, by increasing the number of corrections, the scheme is more accurate but the CPU time rapidly increases. The keyword NUMBER OF CORRECTIONS OF DISTRIBUTIVE SCHEMES can be used with advection schemes of type (3, 4, 5, LIPS or not).
The keyword MAXIMUM NUMBER OF ITERATIONS FOR ADVECTION SCHEMES enables to limit the number of solver iterations for the advection schemes of type NERD (SCHEME FOR ADVECTION OF... = 13 or 14). The default value is 50 (old default value = 10 until release 8.1).
The keyword NUMBER OF SUB-STEPS OF DISTRIBUTIVE SCHEMES can only be activated for locally semi-implicit predictor-corrector schemes aka LIPS (SCHEME FOR ADVECTION OF... = 4 or 5 + SCHEME OPTION FOR ADVECTION OF... = 4). The default value is 1. As the keyword mentions, it allows to subdivide the time step given by the user in the steering file, into several sub-steps. Again, it produces an effect on the precision of the scheme and it is convenient to set this keyword in order to have Courant numbers not too large (around 1).

Note:

- If present, the keyword SCHEME OPTION FOR ADVECTION OF VELOCITIES replaces and has priority over the following keywords: OPTION FOR CHARACTERISTICS and SUPG OPTION.
- The same remark are valid for advection of tracer, k, ε and $\tilde{\nu}$. However there are dedicated keywords: SCHEME FOR ADVECTION OF TRACERS and SCHEME FOR ADVECTION OF K-EPSILON,
- MATRIX STORAGE = 3 is mandatory with some distributive schemes for advection (= 3, 13, 14 or LIPS).

If using ELEMENT = 'TETRAHEDRON' and SCHEME FOR ADVECTION OF... = 3 aka Leo Postma (resp. 13 aka Leo Postma for tidal flats), this last keyword is automatically changed to 4 aka N (resp. 14).

6.2.4 Configuration of the weak characteristics

When choosing the method of characteristics, two forms can be used with the keyword OPTION FOR CHARACTERISTICS:

- 1: the strong form (by default),
- 2: the weak form.

If SCHEME FOR ADVECTION OF... = 1, and also the corresponding keyword SCHEME OPTION FOR ADVECTION OF... = 2, OPTION FOR CHARACTERISTICS is automatically set to 2.

None of them are recommended for the advection of tracers because they are not mass conservative. The weak form will decrease the diffusion. If the keyword MASS-LUMPING FOR WEAK CHARACTERISTICS = 1. (default value = 0. i.e. no mass-lumping), monotonicity of the scheme

appears. This weak form should be more conservative than the strong form. The NUMBER OF GAUSS POINTS FOR WEAK CHARACTERISTICS defines the number of Gauss points used to compute the weak characteristics. One single choice is possible for TELEMAC-3D: 6 points (default value).

6.3 Specific parameters in the non-hydrostatic version

The application of the software NON-HYDROSTATIC VERSION (default = YES) requires that complementary keywords be defined.
An equation for vertical velocity is initially solved in a same way as the U and V components. That equation is only written with the hydrostatic pressure which, under that hypothesis, is cancelled out with the gravity term. The convection scheme in that equation is identical to that chosen for U and V (keyword SCHEME FOR ADVECTION OF VELOCITIES).
Afterwards, TELEMAC-3D solves a Poisson equation for the dynamic pressure. The dynamic pressure gradient plays the part of a correction providing the required zero divergence on velocity. The solution of the linear system in that equation is managed by the following keywords:

- SOLVER FOR PPE refer to subsection 6.5.1 below,
- MAXIMUM NUMBER OF ITERATIONS FOR PPE refer to subsection 6.5.2 below,
- ACCURACY FOR PPE refer to subsection 6.5.2 below,
- OPTION OF SOLVER FOR PPE refer to subsection 6.5.1 below,
- PRECONDITIONING FOR PPE refer to subsection 6.5.3 below.

Once that pressure is computed, all the velocity components are updated with the dynamic pressure gradient which will ensure the zero divergence condition. Updating that "solenoidal" velocity does not require that a linear system be solved.

When activating the keyword CONTINUITY CORRECTION ON OPEN BOUNDARIES to YES (default value = NO), the free horizontal velocity components are modified on open boundaries in order to get a better divergence-free field during the final velocity projection step. This option is only possible if NON-HYDROSTATIC VERSION = YES.

6.4 Implicitation

Apart from the terms of the time derivative, the unknowns f (the velocity and water depth components) can be considered in both extreme instants t^n (the equation is then referred to as explicit) or t^{n+1} (the equation is then referred to as implicit). Strictly speaking, and for a 2-order solution in time, an approach consists in considering the terms in the intermediate instant$(f^n + f^{n+1})/2$. Practically, the latter approach is unstable and it becomes necessary to define an implicitation coefficient for which the unknowns are actually discretized in time in the following form:

$$\theta f^{n+1} + (1-\theta) f^n.$$

The implicitation coefficients are theoretically always higher than 0.5 (0.55 or 0.6 will generally yield good results).
The user can use the keyword IMPLICITATION FOR VELOCITIES (default value: 0.55 since release 8.1, old default value = 1. until release 8.0) which defines the value of the θ_u coefficient for the velocity components. The keyword IMPLICITATION FOR DEPTH (default value: 0.55)

is provided to set the value of "propagation height" multiplying coefficient θ_h Lastly, in order to make the construction of the various numerical schemes more versatile, a diffusion-specific coefficient θ_{ut}^d is provided (keyword `IMPLICITATION FOR DIFFUSION` (default value: 1.) and can be different from θ_{ut}).

6.5 Solution of the linear systems

Both discretization and variational formulation of the equations lead to a linear system which now has to be solved. The direct solution methods are often not suitable and are overly expensive as soon as there are many unknowns. Thus, the main ways developed in TELEMAC-3D consist in solving the linear systems by means of iterative solvers. Nevertheless for specific applications, a direct solver can be used.

6.5.1 Solvers

According to the relevant numerical parameters, various linear systems are liable to be solved. The solver used for solving one of these systems can be selected by the user through the following keywords:

- `SOLVER FOR DIFFUSION OF VELOCITIES` (default value: 1),
- `SOLVER FOR PROPAGATION` (default value: 7 since release 8.1, old default value = 1 until release 8.0),
- `SOLVER FOR PPE` (default value: 7 since release 8.1, old default value = 1 until release 8.0),
- `SOLVER FOR DIFFUSION OF TRACERS` (default value: 1, one value for each tracer),
- `SOLVER FOR DIFFUSION OF K-EPSILON` (default value: 1) for both $k-\varepsilon$ model and Spalart-Allmaras model.

Each of these keywords can assume a value ranging from 1 to 8, which values correspond to the following possibilities. Options 1 to 6 are all related to the conjugate gradient method:

- 1: conjugate gradient method (when the matrix of the system to solve is symmetric),
- 2: conjugate residual method,
- 3: conjugate gradient on a normal equation method,
- 4: minimum error method,
- 5: square conjugate gradient method,
- 6: CGSTAB (stabilized conjugate gradient) method,
- 7: GMRES (Generalised Minimum RESidual) method,
- 8: direct solver (YSMP, solver of the Yale university), does not work in parallel mode.

The conjugate gradient is generally recommended for symmetric linear systems, thus when solving the diffusion equations.
The GMRES method is well suited for improperly conditioned systems. This method requires that the dimension of the Krylov space be defined. That parameter is set using the following keywords:

- OPTION OF SOLVER FOR DIFFUSION OF VELOCITIES,
- OPTION OF SOLVER FOR PROPAGATION,
- OPTION OF SOLVER FOR PPE,
- OPTION OF SOLVER FOR DIFFUSION OF TRACERS (different values for each tracer possible since release 8.4),
- OPTION OF SOLVER FOR DIFFUSION OF K-EPSILON for both $k-\varepsilon$ model and Spalart-Allmaras model.

The default values are set to 5 since release 8.1 (old default values = 3 until release 8.0). The larger that parameter is, the higher the memory requirements and the number of matrix-vector products per iteration (and consequently the computational time as well) are, but the better the convergence is.

6.5.2 Accuracies

The principle of iterative methods consists in getting gradually closer to the exact solution during the iterations. The systems to be solved imply a relative accuracy within a range from 10^{-4} to 10^{-10} with a restricted number of iterations. Both accuracy and maximum number of iterations should be set for each system.
Accuracy is specified by the following keywords:

- ACCURACY FOR DIFFUSION OF VELOCITIES (default value: 1.E-8 since release 8.1, old default value = 1.E-5 until release 8.0),
- ACCURACY FOR PROPAGATION (default value: 1.E-8 since release 8.1, old default value = 1.E-6 until release 8.0),
- ACCURACY FOR PPE (default value: 1.E-4 since release 8.1, old default value = 1.E-4 until release 8.0),
- ACCURACY FOR DIFFUSION OF TRACERS (default value: 1.E-8 since release 8.0; different values for each tracer possible since release 8.4),
- ACCURACY FOR DIFFUSION OF K-EPSILON (default value: 1.E-8 since release 8.1, old default value = 1.E-6 until release 8.0) for both $k-\varepsilon$ model and Spalart-Allmaras model.

The maximum number of iterations is specified by the following keywords:

- MAXIMUM NUMBER OF ITERATIONS FOR DIFFUSION OF VELOCITIES (default value: 60),
- MAXIMUM NUMBER OF ITERATIONS FOR PROPAGATION (default value: 200),
- MAXIMUM NUMBER OF ITERATIONS FOR PPE (default value: 100),
- MAXIMUM NUMBER OF ITERATIONS FOR DIFFUSION OF TRACERS (default value: 60; different values for each tracer possible since release 8.4),
- MAXIMUM NUMBER OF ITERATIONS FOR DIFFUSION OF K-EPSILON (default value: 200) for both $k-\varepsilon$ model and Spalart-Allmaras model,

- MAXIMUM NUMBER OF ITERATIONS FOR ADVECTION SCHEMES (default value: 50 since release 8.2, old default value = 10 until release 8.1) only for schemes 13 and 14.

The user automatically gets information about the solvers upon each listing printout. The information provided in the listing can be of two types:

- Either the treatment converged before reaching the maximum allowable number of iterations, and then TELEMAC-3D will provide the number of actually performed iterations, as well as the achieved accuracy,
- Or the treatment did not converge early enough. TELEMAC-3D will then provide the message "**MAXIMUM NUMBER OF ITERATIONS IS REACHED**" and give the actually achieved accuracy. In some cases, and if the maximum number of iterations is already set to a high value (for instance over 100), convergence can then be improved by reducing the time step or, quite often, by improving the quality of the mesh.

6.5.3 Preconditionings

The iterative methods are sensitive to the "conditioning" of matrices, so that a complementary preconditioning is necessary in order to reduce the number of iterations for getting a prescribed accuracy.
TELEMAC-3D offers several opportunities for preconditioning. The selection is made by means of the following keywords:

- PRECONDITIONING FOR DIFFUSION OF VELOCITIES,
- PRECONDITIONING FOR PROPAGATION,
- PRECONDITIONING FOR PPE,
- PRECONDITIONING FOR DIFFUSION OF TRACERS (one value for each tracer),
- PRECONDITIONING FOR DIFFUSION OF K-EPSILON for both $k-\varepsilon$ model and Spalart-Allmaras model.

The available options are:

- 0: no preconditioning,
- 2: diagonal preconditioning,
- 3: diagonal preconditioning with the condensed matrix,
- 5: diagonal preconditioning with absolute values,
- 7: Crout preconditioning per elementCrout (downgraded in parallel),
- 11: Gauss-Seidel preconditioning per element (downgraded in parallel),
- 13: preconditioning matrix is provided by the user,
- 14: cumulated diagonal preconditioning and Crout preconditioning per element,
- 17: preconditioning through direct solution along each vertical direction,

- 21: cumulated diagonal preconditioning with the condensed matrix and Crout preconditioning per element,
- 34: cumulated diagonal preconditioning with direct solution along each vertical direction.

The default value is 2 for all the preconditionings. Some preconditionings can be cumulated, namely the diagonal preconditionings with other ones. Since the basic values are prime numbers, a couple of preconditionings are cumulated by giving to the keyword the value of the product of the two preconditionings which one wants to cumulate (e.g. numbers 14 and 21).

6.6 Tidal flats

TELEMAC-3D offers several treatment options as regards the tidal areas.
First, if the user has ascertained that his/her model has no tidal area throughout the simulation, the processing of such areas can be deactivated by setting the keywords `TIDAL FLATS` to NO (the default value is YES). That option makes it possible to save computational time (by deleting the tidal flat testing).
The tidal flats can be treated in two different ways:

- In the first case, the equations are treated all over the domain and in a thorough way. The tidal areas are detected and such terms as the free surface gradient (in the absence of water, the free surface gradient becomes the bottom gradient and generates spurious motive terms) are corrected in them,
- In the second case, the tidal areas are withdrawn from the computation. The exposed elements are always part of the mesh, but all their contributions to the computations are cancelled by a so-called "masking" array. Thus, the data structure and the computations remain formally unchanged, to within the masking coefficient. That method, however, raises issues as regards the mass conservation and the exposure and coverage dynamics.

The treatment will be selected by means of the keyword `OPTION FOR THE TREATMENT OF TIDAL FLATS` which can be set either to 1 or 2, the default value being 1.
The treatment of negative depths can be specified using the keyword `TREATMENT OF NEGATIVE DEPTHS`. A value of 1 (default), consists in a conservative smoothing of negative depths. The second option is to limit the flux between the elements to ensure strictly positive water depths. This second option should be used with advection schemes consistent with tidal flats (+ `MASS-LUMPING FOR DEPTH` = 1.). The value 0 means that no special treatment is performed.
The numerical advection schemes (`SCHEME FOR ADVECTION OF`...) for tidal flats are:

- 13 or 14: NERD scheme,
- 4 or 5 coupled with `OPTION FOR ADVECTION OF`... = 4: LIPS scheme.

Note:
NERD schemes (13 and 14) with `TIDAL FLATS` = YES require the keywords `OPTION FOR THE TREATMENT OF TIDAL FLATS` = 1 (default value) + `TREATMENT OF NEGATIVE DEPTHS` = 2 + `MASS-LUMPING FOR DEPTH`.

The keyword `MINIMAL VALUE FOR DEPTH`, the default value of which is -1,000, enables to set the threshold below which the smoothing is. It is only used with `OPTION FOR THE TREATMENT OF TIDAL FLATS` = 2 (default = 0) i.e. dry elements are frozen (tidal flats area are masked) or

with ELEMENTS MASKED BY USER = YES (default = NO). For example, MINIMAL VALUE FOR DEPTH set to 0.01 means the minimum depth is 1 cm.
The following three keywords are for setting, after coverage, the value of the variable which has been masked:

- TREATMENT ON TIDAL FLATS FOR VELOCITIES,
- TREATMENT ON TIDAL FLATS FOR TRACERS,
- TREATMENT ON TIDAL FLATS FOR K-EPSILON.

The available options for these keywords are:

- 0: that option corresponds to a setting to zero of the variable on the element (default value),
- 1: that option sets to its prior-to-masking value.

The keyword THRESHOLD FOR VISCOSITY CORRECTION ON TIDAL FLATS (default value is 0.2 m) allows to specify the minimum water depth from which the viscosity is gradually reduced (see programming within the **VISCLIP** subroutine).
When the three-dimensional mesh has crushed levels (null water depth or fixed level "hitting" the bottom), it is recommended to activate a specific treatment that prevents the transfer of very small amounts of water at the calculation points which have no volume (this situation also tends to degrade the mass conservation when using distributives PSI and N schemes). This algorithm is activated with the keyword logical BYPASS VOID VOLUMES (default value = NO). When using PSI and N schemes compatible with tidal flats, the option is automatically enabled, even if the keyword is set to NO.

6.7 Hydrostatic inconsistencies

Hydrostatic inconsistencies (linked to the truncature errors in the computation of the buoyancy terms) are liable to occur on the nearly-zero volume prisms. The keyword HYDROSTATIC INCONSISTENCY FILTER (default = NO) is provided for the forces caused by the spurious horizontal pressure gradients and the various diffusion coefficients on those prisms where at least one of the lower base nodes has a higher elevation than one of the upper base nodes.

6.8 Other parameters

6.8.1 Mass-lumping

Upon the solution of the linearized system, TELEMAC-3D makes it possible to perform a mass-lumping on the mass matrices. That procedure consists in partly or wholly returning the mass matrix to its diagonal and enables to substantially shorten the computational times. The resulting solutions, however, become smoothed, except for in steady flow conditions in which they are unchanged. The mass-lumping rate is set with the keywords MASS-LUMPING FOR DEPTH, MASS-LUMPING FOR DIFFUSION and MASS-LUMPING FOR WEAK CHARACTERISTICS. Value 1. means maximum mass-lumping (the mass matrices are diagonal), value 0. (default value) corresponds to the normal treatment without any mass-lumping.

If using OPTION FOR THE TREATMENT OF TIDAL FLATS = 1 (default value) and TREATMENT OF NEGATIVE DEPTHS = 2 (flux control), the keyword MASS-LUMPING FOR DEPTH must be equal to 1.

The keyword MASS-LUMPING FOR DIFFUSION used for the mass-matrix in the diffusion step is automatically set to 1. in **DIFF3D** if diffusion is explicit (IMPLICITATION FOR DIFFUSION < 0.001) or if the advection scheme is of type 3, 4, 5, 13 or 14 (N, PSI, NERD).
The keyword MASS-LUMPING FOR VELOCITIES is read but not used.

For further details, the reader shall refer to the TELEMAC-3D Theoretical Note.

6.8.2 Convergence aid

Another way to speed up the system convergence when solving the propagation step consists in acting upon the initial solution rather than the matrix proper. To that purpose, the initial value being set for h (actually, the unknown h^{n+1} is replaced by the incrementation $\delta h = h^{n+1} - h^n$) is modified at the beginning of computation, the user can take action at the keyword INITIAL GUESS FOR DEPTH which can assume the following values:

- 0: the initial value of $\delta h = h^{n+1} - h^n$ is zero,
- 1: the initial value of δh is equal to the value of δh at the previous time step (default value),
- 2: $\delta h = 2h^n - \delta h^{n-1}$ where δh^n is the value of δh at the previous time step, and δh^{n-1} is the value of δh two time steps earlier. It is actually an extrapolation.

If option 2 with the non-hydrostatic version, INITIAL GUESS FOR DEPTH is automatically set to 1.

6.8.3 Matrix storage

TELEMAC-3D provides a couple of procedures for storing the various matrices it has to handle, namely the conventional EBE (Element By Element) method and the segment-wise storage. The second is faster (about 20%) in most cases.
The choice between the two storage methods can be done using the keyword MATRIX STORAGE, with the following values:

- 1: classical Element by Element (EBE) method,
- 3: edge-based storage method (default and recommended value).

MATRIX STORAGE = 3 is mandatory with some distributive schemes for advection (= 3, 13, 14 or LIPS).

6.8.4 Velocities projection

At the end of the time loop, it is possible to do a treatment which aim is to cancel the component of the normal velocity at the bottom or the normal velocity on the solid lateral walls. This check is activated with the logical keywords VELOCITY PROJECTED ON SOLID LATERAL BOUNDARIES and VELOCITY PROJECTED ON BOTTOM.
These two options are activated by default.

7. Tracer transport

The TELEMAC-3D software makes it possible to take into account the transport of passive or active tracers (active tracers affect the hydrodynamics), being either conservative or not.
This chapter discloses the tracer transport features.
The maximum number of tracers is set to 20 by default but it can be changed by the user with the keyword MAXIMUM NUMBER OF TRACERS. This avoids changing the previously hardcoded values (until release 7.0), which required recompiling the whole package.

7.1 General setup

The number of tracers is defined by the keywords NUMBER OF TRACERS. If that number is set to zero (default value), then the tracers will not be taken into account by TELEMAC-3D. Additional modules which can be coupled with TELEMAC-3D (e.g. WAQTEL or GAIA, see section 11.7) can add extra tracers to the initial set defined in the TELEMAC-3D steering file.
In addition to the number of tracers, the user should enter the NAMES OF TRACERS. A tracer name should be written with 32 characters (16 for the name and 16 for the unit).

```
NUMBER OF TRACERS : 2
NAMES OF TRACERS :
'TEMPERATURE     C               ';'SALINITY        G/L             '
```

Obviously, it is necessary to add the appropriate specifications in the keywords VARIABLES FOR 3D GRAPHIC PRINTOUTS and VARIABLES FOR 2D GRAPHIC PRINTOUTS. The name of the variables is a letter TA followed by the number of tracer. For example 'TA1,TA3' stand for first and third tracer. It is possible to use the character * as wildcards (replace any character). TA* stands for TA1 to TA9, and TA** stands for TA10 to TA99.

7.1.1 Prescribing the initial conditions

If the initial values of tracers are constant all over the domain, just insert, into the steering file, the keyword INITIAL VALUES OF TRACERS with the required value(s) separated with a semicolumn; if more than one. The number of supplied values must be equal to the number of declared tracers.
In more complex cases, it is necessary to work directly in the **USER_CONDI3D_TRAC** subroutine, in a similar way to that described in the subsection 4.2.2 dealing with the initial hydrodynamic conditions.
When resuming a computation, the initial condition of tracers corresponds to the condition of the last time step which was stored into the restart file. The tracer management sequence order

during the previous computation needs to be well known and the same sequence order shall be followed in the computational suite in order to prevent on confusion. If the restart file does not contain any information about the tracer, TELEMAC-3D will use the value as set by the keyword `INITIAL VALUES OF TRACERS`.

7.1.2 Prescribing the boundary conditions

Tracer boundary conditions are prescribed according to the same principle as the hydrodynamic boundary conditions (see section 4.3).
The boundary condition type will be given by the value of **LITBOR** in the boundary conditions file.
In case of an entering liquid boundary with one or several prescribed tracer(s) (**LITBOR** = 5), the tracer value can be given in various ways:

- If the value is constant along the boundary and in time, it is provided in the steering file by the keyword `PRESCRIBED TRACERS VALUES`. This is an array of real numbers for managing several boundaries and several tracers (100 at most, this number can be changed with the keyword `MAXIMUM NUMBER OF TRACERS`). The writing convention is as follows: value of tracer 1 at boundary 1, value of tracer 2 at boundary 1, ..., value of tracer N at boundary 1, value of tracer 1 at boundary 2, value of tracer 2 at boundary, ..., value of tracer N at boundary 2, etc. The boundary order is the same as in the case of hydrodynamic boundary conditions. The writing convention is the same as that used for the hydrodynamic boundary conditions. The values specified by the keyword cancel the values read from the boundary conditions file,
- If the value is constant in time but varies along the boundary, it will be set directly by the **TBOR** variable in the `BOUNDARY CONDITIONS FILE`,
- If the value is constant along the boundary but varies in time, the user may either use the `LIQUID BOUNDARIES FILE` or specify this with the **USER_TR3** subroutine. The latter will be programmed somewhat like the **USER_VIT3, USER_Q3** and **USER_SL3** subroutines. Note that the liquid boundaries file will not be taken into account if the keywords `PRESCRIBED ELEVATIONS` and `PRESCRIBED FLOWRATES` do not appear in the steering file.

```
T          SL(1)     TR(2,1)       TR(2,2)
s          m         C             C
0          0.47      24.7          38.0
1040400    0.57      28.0          36.7
```

In the above example of a liquid boundaries file, for the indices of tracers values, the first value is the number of the liquid boundary and the second the number of the tracer,

- If the value varies in both time and space, the user should then modify the **USER_BORD3D** subroutine, in the part regarding the tracer.

The keyword `TREATMENT OF FLUXES AT THE BOUNDARIES` enables, during the convection step (with the SUPG, PSI and N schemes), to set a priority among the tracer flux across the boundary and tracer value at that wall. Option 2 ("Priority to fluxes") will then induce a change in the tracer prescribed value, but will bring about a good assessment of the "mass" of tracer passing across the boundary. On the other hand, option 1 ("Priority to prescribed values", default value) sets the tracer value without checking the fluxes. There are as many values as the number

of liquid boundaries, contrary to what can be done in TELEMAC-2D (one single value for every liquid boundary).
Finally, in the case of an input boundary, it is possible to specify a concentration profile in the vertical using the keyword `TRACERS VERTICAL PROFILES`. There are as many values to be given as the product of the number of tracers and the number of open boundaries. The options are:

- 0: user programming (in **USER_BORD3D**),
- 1: constant profile (default),
- 2: constant profile (tracer diluted) or Rouse profile (sediment),
- 3: Rouse (normalised) and imposed concentration.

7.2 Physical setup

7.2.1 Active tracers

The active tracers affect the flow through the hydrostatic pressure gradient term. As a rule, indeed, the pressure is written as:

$$p = p_h + p_d = \rho g(Z_s - z) + \rho_0 g \int_z^{Z_s} \frac{\Delta\rho}{\rho_0} dz + p_d \tag{7.1}$$

where p_h and p_d denote the hydrostatic pressure and the dynamic pressure, respectively.
Thus, two elements should be defined, namely: ρ_0 and $\frac{\Delta\rho}{\rho_0}$.
The term ρ_0 is defined by the keyword `AVERAGE WATER DENSITY`. The default value is 1,025; it corresponds to sea (ocean) water.
The second term, which operates in the buoyancy source terms, directly depends on the values of the active tracers and is defined by the keyword `DENSITY LAW`.
The available values for that keyword `DENSITY LAW` are:

- 0: no interaction with the tracers, except if present sediment (default value),
- 1: variation of density according to temperature (and sediment if present),
- 2: variation of density according to salinity (and sediment if present),
- 3: variation of density according to temperature and salinity (and sediment if present),
- 4: variation as a function of the spatial expansion coefficients (and effect of the sediment with its own behaviour if present),
- 5: the sediment and other tracers are forced to be passive,
- 6: Jackett et al. law (2006) with variation of density according to temperature and salinity using a 25 term formula.

With the 1-3 options, the variations are given by the law as defined in TELEMAC-3D. In such a case, the name of the salinity tracer (expressed in kg/m^3) shall necessarily begin with SALINIT and the name of the temperature tracer (in °C) shall begin with TEMPERATURE.

With option 4, the term $\frac{\Delta\rho}{\rho_0}$ is described by a linear function of the T_i tracer of the type:

$$\frac{\Delta\rho}{\rho_0} = -\sum_i \beta_i (T_i - T_i^0)_i \tag{7.2}$$

The β_i coefficients (spatial expansion coefficients) are set by the values of the keyword `BETA EXPANSION COEFFICIENT FOR TRACERS` (default = 0.). They can be either positive (temperature) or negative (salinity, suspended sediment). The values T_i^0 are defined by the values of the keyword `STANDARD VALUES FOR TRACERS` (default = 0.).
The user shall enter the expansion coefficients, the standard values and the tracer names in the same sequence order to ensure that each tracer will have the correct parameters.

7.2.2 Punctual source terms

For each source, the user shall enter the tracer value at the sources by means of the keyword `VALUE OF THE TRACERS AT THE SOURCES`. Thus, it is an array of reals specifying the concentration of tracers at the source. The writing convention is as follows: source value 1 of tracer 1; source value 1 of tracer 2; ..., source value 1 of tracer n; source value 2 of tracer 1; ..., source value 2 of tracer n etc. In case of time dependent value, it is necessary to use the source file or a specific FORTRAN programming (**USER_T3D_TRSCE** subroutine).

7.2.3 General source terms

If one wants to take the tracer generation or disappearance source terms into account, then this has to be implemented within the **SOURCE_TRAC** subroutine.

In case of calculation with consideration of rain, it is possible to specify the contribution related to the rain with the keyword `VALUES OF TRACERS IN THE RAIN` (default value is 0.). It is important to note that, in the case of evaporation, no tracer is taken into account in the water loss, which is incorrect if the tracer is the temperature.

7.3 Numerical setup

As with hydrodynamics, the advection schemes `SCHEME FOR ADVECTION OF TRACERS`, option for these advection schemes `SCHEME OPTION FOR ADVECTION OF TRACERS` (refer to subsection 6.1.1) and the diffusion schemes `SCHEME FOR DIFFUSION OF TRACERS` (refer to subsection 6.1.2) can be modified.
Otherwise, the horizontal and vertical diffusion of the tracers can be set with the keywords `COEFFICIENT FOR HORIZONTAL DIFFUSION OF TRACERS` and `COEFFICIENT FOR VERTICAL DIFFUSION OF TRACERS`. Their default values are 10^{-6} m^2/s. These two keywords are arrays since release 7.1, with one value per tracer, separated by semicolons, so that different values can be given for different tracers.
`TREATMENT ON TIDAL FLATS FOR TRACERS` keywords is for setting, after coverage, the value of the variable which is masked at the diffusion step for tracer for tidal flatss. The available options for this keyword are:

- 0: that option corresponds to a setting to zero of the variable on the element (default value),
- 1: that option sets to its prior-to-masking value.

8. Drogues

During a hydrodynamic simulation, TELEMAC-3D offers an opportunity to follow the paths of a number of particles (drogues) which are released into the fluid from discharge points. The result is provided in the form of a formatted file which contains the various positions of the drogues. The default format for this file TECPLOT. Alternatively, PCL format can be chosen.

8.1 Configuration of simulation

Three parameters should be entered into the steering file. First, the user should mention the number of drogues by means of the keywords `MAXIMUM NUMBER OF DROGUES`, the default value of which is 0. Secondly, the user should enter the name of the file into which TELEMAC-3D will store the successive positions of the drogues. This is defined through the keyword `ASCII DROGUES FILE` or `BINARY DROGUES FILE`. Lastly, the user can configure the printout period within that file by means of the keyword `PRINTOUT PERIOD FOR DROGUES` (default value = 1). That value is expressed in a number of time steps and is quite independent from of the printout period of the other results in TELEMAC-3D.

The subroutine **ADD_PARTICLE** is called within the **USER_FLOT3D** subroutine to set the initial values of variables **XFLOT, YFLOT, ZFLOT** and **TAGFLO**, which are the three-dimensional coordinates of the release point and an identifier of the particle for each drogue. The release time step is also to be amended. A commented example can be found within the subroutine **USER_FLOT3D** or in the example "particles". This subroutine is to be inserted in the FORTRAN file.

TELEMAC-3D offers the possibility to introduce a stochastic diffusion coefficient. When setting the keyword `STOCHASTIC DIFFUSION MODEL` = 1 (default = 0), a stochastic model generates a diffusion coefficient which is computed using the turbulent viscosity. If no turbulence is activated, this stochastic diffusion is not considered during the particle transport.

The example below illustrates the programming of the steering file in the case of two drogues being released at different times.

```
MAXIMUM NUMBER OF DROGUES           = 2
ASCII DROGUES FILE                  = './drogues'
PRINTOUT PERIOD FOR DROGUES         = 10
```

8.2 Visualisation of results

The format of the drogues output file is specified by the keyword `DROGUES FILE FORMAT`. The default value for the file format is TECPLOT. In this case, the drogues are written to the file given by the keyword `ASCII DROGUES FILE`. Alternatively, the file format can be BKBINPCL. In this case, the drogues are written to the file given by the keyword `BINARY DROGUES FILE`. The output file is a PCL file which can be read by the BlueKenue software. The TECPLOT file is an ASCII file written in a format compatible with the TECPLOT software. If this tool is not available, it is quite easy to get the coordinates of the different drogue positions to export them to another viewer. It is also possible to develop a new drogue output format, but this must be done in the subroutine **UTIMP_DROGUES**.
Within TECPLOT, in order to add the Tecplot `ASCII DROGUES FILE` to TELEMAC result data already loaded, select "Add to current data" set in the **Load Data File Warning dialogue** (cf. Figure 8.1). The Load Data File Warning dialogue will appear after the user has selected the file and zones and/or variables to load.

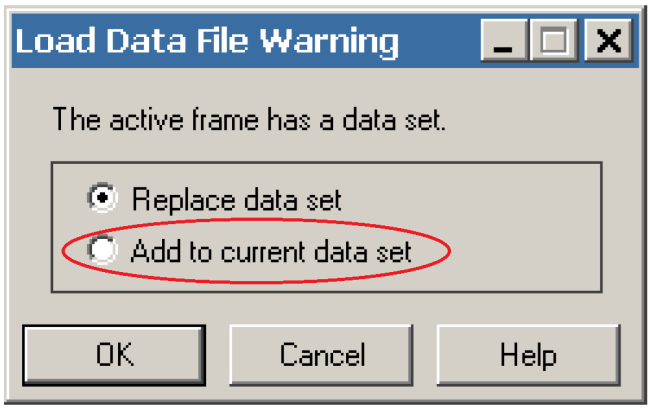

Figure 8.1: Load Data File Warning dialogue in TECPLOT.

Once you have loaded your data with TECPLOT, the drogue positions will be considered as "Scatter plots" by Tecplot Software. Scatter plots are plots of symbols centered at the data points in a field. To add a scatter layer to your plot, activate the "Scatter" toggle in the Sidebar. To be visible in your plot, the Scatter layer which contains the Tecplot `ASCII DROGUES FILE` must be turned on and the Scatter layer containing the `3D RESULT FILE` data must be turned off. This is done by selecting "Yes" or "No" from the [Scat Show] button drop-down menu on the Scatter page of the **Zone Style dialogue** (cf. Figure 8.2).

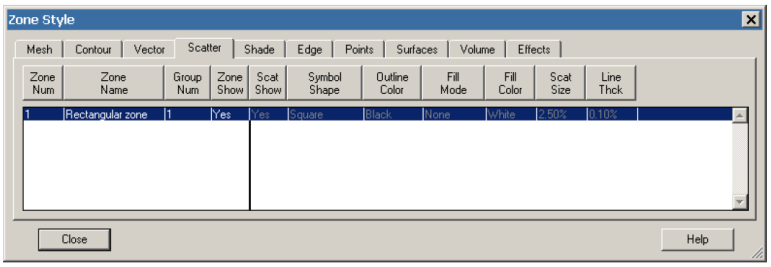

Figure 8.2: Zone Style dialogue in TECPLOT.

Then, you can modify your Scatter plot using the Scatter page of the **Zone Style dialogue** and

the Scatter submenu of the **Plot menu**. You can control any of the following attributes for a zone or group of zones from the Scatter page of the Zone Style dialogue. For complementary information on the Tecplot procedure, the reader may refer to the Tecplot user guide.

9. Oil spill modelling

During a hydrodynamic simulation, TELEMAC-3D offers an opportunity to follow the paths of an oil spill which is released into the fluid from a discharge point.
The oil spill model introduced here combines an Eulerian and a Lagrangian approach. The Lagrangian model simulates the transport of an oil spill near the surface. The oil slick is represented by a large set of hydrocarbon particles. Each particle is considered as a mixture of discrete non-interacting hydrocarbon components. Particles are therefore represented by component categories (soluble and unsoluble components), and the fate of each component is tracked separately. Each particle has associated to it, amongst other properties, an area, a mass, its barycentric coordinates within the element it is located in, and the physico-chemical properties of each of its components. The model accounts for the main processes that act on the spilled oil: advection, effect of wind, diffusion, evaporation and dissolution. Though generally considered to be a minor process, dissolution is important from the point of view of toxicity. To simulate soluble oil component dissolution in water, an Eulerian advection-diffusion model is used. The fraction of each dissolved component is represented by a tracer whose mass directly depends on the dissolved mass of oil particles. The hydrodynamic data required for either Lagrangian and Eulerian transport approach are provided by the TELEMAC-3D hydrodynamic model. The oil spill theoretical background is explained in [1].
The result of the oil spill modelling is provided in the form of a TECPLOT formatted file which contains the various positions of the oil particles and a TELEMAC-3D result file (SERAFIN format) storing the oil dissolved components in water column during the computation.

9.1 Input files

In addition to the minimum set of input files necessary to run a TELEMAC-3D case, an oil spill computation also needs an oil spill steering file. Furthermore, to run an oil spill model the subroutine **OIL_FLOT** needs to be modified in the FORTRAN file.

9.2 Steering file

In addition to the necessary information for running the TELEMAC-3D hydrodynamic model the following essential information must be specified in the TELEMAC-3D steering file to run an oil spill propagation model:

- The use of the oil spill model must be declared: `OIL SPILL MODEL` (= YES, default = NO),

- The name of the oil spill steering file which contains the oil characteristics: `OIL SPILL STEERING FILE` (= `name chosen by the user`),
- The number of oil releases during oil spill: `MAXIMUM NUMBER OF DROGUES` (= number chosen by the user),
- The frequency of the drogues printout period: `PRINTOUT PERIOD FOR DROGUES` (= number chosen by the user),
- The name of the tecplot oil file containing the oil displacement: `ASCII DROGUES FILE` (= name chosen by the user, default = 1).

With the oil spill module, it is possible to take into account the transport of soluble oil components in water (whose presence has no effect on the hydrodynamics). These may or may not be diffused within the flow but their characteristics have to be defined in the `OIL SPILL STEERING FILE`. If these components are allowed to diffuse in the flow, they are then treated with the tracer transport computations of TELEMAC-3D. This implies that the `NUMBER OF TRACERS` must be set to the number of the oil soluble components. In addition the TRACER keywords, described in chapter 7, can be specified.

9.3 Oil spill steering file

As seen previously, the `OIL SPILL STEERING FILE` name is given by the user in the TELEMAC steering file. This file contains all the informations for an oil spill calculations based on the composition considered by the user, i.e.:

- The number of unsoluble components in oil,
- The parameters of these components such as the mass fraction (%) and boiling point of each component (K),
- The number of soluble components in oil,
- The parameters of these components such as the mass fraction (%), boiling point of each component (K), solubility ($kg.m^{-3}$) and the mass transfer coefficient of the dissolution and volatilization phenomena ($m.s^{-1}$)
- The oil density,
- The oil viscosity ($m^2.s^{-1}$),
- The volume of the spilled oil (m^3),
- The water surface temperature (K),
- The spreading model chosen by the user:
  - Fay's model,
  - Migr'Hycar model,
  - Constant area model.

WARNING:

Warning:

- The parameters of soluble (or unsoluble) components need to be informed only if the number of these components is not null
- If the sum of all mass fraction components is not equal to 1, the run is interrupted and the following error message is displayed:

```
WARNING::THE SUM OF EACH COMPONENT MASS FRACTION IS NOT EQUAL TO 1.
PLEASE, MODIFY THE INPUT STEERING FILE
```

An example of the oil spill steering file is given.

```
NUMBER OF UNSOLUBLE COMPONENTS IN OIL
6
UNSOLUBLE COMPONENTS PARAMETERS (FRAC MASS, TEB)
5.1D-02          ,402.32D0
9.2D-02          ,428.37D0
3.16D-01         ,458.37D0
3.5156D-01       ,503.37D0
8.5D-02          ,543.37D0
9.4D-02          ,628.37D0
NUMBER OF SOLUBLE COMPONENTS IN OIL
4
SOLUBLE COMPONENTS PARAMETERS (FRAC MASS, TEB, SOL, KDISS, KVOL)
1.D-02    ,497.05D0,   0.018D0     , 1.25D-05  ,5.0D-05
3.2D-02   ,551.52D0,   0.00176D0   , 5.63D-06  ,1.51D-05
1.D-04    ,674.68D0,   2.0D-04     , 2.D-06    ,4.085D-07
2.D-05    ,728.15D0,   1.33D-06    , 1.33D-06  ,1.20D-07
OIL DENSITY
830.D0
OIL VISCOSITY
4.2D-06
OIL SPILL VOLUME
2.02D-05
WATER TEMPERATURE
292.05D0
SPREADING MODEL (1=FAY'S MODEL, 2=MIGR'HYCAR MODEL, 3=CONSTANT AREA)
2
```

If in the oil spill steering file, the SPREADING MODEL is set to 3, two lines must be added to the previous example:

```
CONSTANT AREA VALUE CHOSEN BY THE USER FOR EACH OIL PARTICLE
1 (example if the user wants area particle equal to 1 m2)
```

9.4 The OIL_FLOT subroutine

After inserting the **OIL_FLOT** subroutine in the FORTRAN file, it must be modified it in order to indicate the release time step, together with the coordinates of the release point. If the

release point coordinates are outside the domain, the run is interrupted and an error message is displayed. In addition, if a particle leaves the domain during the simulation, it is of course no longer monitored but its previous track remains in the results file for consultation.
An example of modifications in the **OIL_FLOT** subroutine is given.
The release time step in the first condition statement and the coordinates of the release point must be changed:

```
...
IF(LT.EQ.10000)THEN
  NUM_GLO=0
  NUM_MAX=0
  NUM_LOC=0
  COORD_X=0.D0
  COORD_Y=0.D0
  NUM_MAX=INT(SQRT(REAL(NFLOT_MAX)))
  DO K=1,NUM_MAX
    DO J=1,NUM_MAX
      COORD_X=336000.D0+REAL(J)
      COORD_Y=371000.D0+REAL(K)
      NUM_GLO=NUM_GLO+1
      NFLOT_OIL=0
      CALL ADD_PARTICLEADD_PARTICLE(COORD_X,COORD_Y,0.D0,NUM_GLO,NFLOT_OIL,
&                          1,XFLOTXFLOT,YFLOTYFLOT,YFLOT,TAGFLOTAGFLO,
&                          SHPFLO,SHPFLO,ELTFLO,ELTFLO,MESH,1,
&                          0.D0,0.D0,0.D0,0.D0,0,0)
...
    END DO
  END DO
END IF
```

9.5 Output files

During an oil spill computation, the TELEMAC-3D software produces at least two output files:

- The `3D RESULT FILE`,
- The output `ASCII DROGUES FILE`.

9.5.1 The 3d result file

This is the file in which TELEMAC-3D stores information during the computation. It is normally in SERAFIN format. First of all, it contains information on the mesh geometry, then the names of the stored variables. It then contains the time for each time step and the values of the different variables for all mesh points. For complementary information on the `3D RESULT FILE`, the reader may refer to 3.13.

9.5.2 The output drogues file

This is an ASCII file created by TELEMAC-3D during the computation. It stores drogue positions in TECPLOT format. To visualize the drogue positions with Tecplot software, the user must:

- Use the File>Load Data File(s) command to load the `3D RESULT FILE`
- Use the File>Load Data File(s) command to load the Tecplot drogue file

See section 8 for more information as it is the same file as for drogues.

10. Construction works modelling

10.1 Culverts

The keyword NUMBER OF CULVERTS (default value = 0) specifies the number of culverts to be treated as source terms. Culverts are described as couples of points between which flow may occur, as a function of the respective water level at these points. They must be described as sources in the domain. It is not necessary to describe each culvert inflow and outflow as a source point like in TELEMAC-2D before release 6.2.
There are two options to treat culverts in TELEMAC-3D. The choice can be done with OPTION FOR CULVERTS (default value = 1). For more information about this choice, the reader is invited to refer to the TELEMAC-3D theory guide.

Information about culvert characteristics is stored in the CULVERTS DATA FILE.
The following file gives an example of a culvert:

```
Relaxation , Number of culverts
0.2  1
I1   I2   CE1 CE2 CS1 CS2 LRG   HAUT1 CLP LBUS Z1   Z2   CV   C56 CV5 C5
CT   HAUT2 FRIC LENGTH CIRC D1   D2 A1 A2   AA
199  640  0.5  0.5  10   1.0  2.52  2.52   0    0.2   0.3  0.1  0.0  0.0  0.0  0.0  0.0  2.52
0.0   0.0      1      90.  0.  0.  90.  0
```

The relaxation coefficient is initially used to prescribe the discharge in the culvert on a progressive basis in order to avoid the formation of an eddy. Relaxation, at time T, between result computed at time T and result computed at previous time step. A relaxation coefficient of 0.2 means that 20% of time T result is mixed with 80% of the previous result. I1 and I2 are the numbers of each end of the culvert in the global point numbering system.
The culvert discharge is calculated based on the formulae given in the TELEMAC-3D theory guide: CE1 and CE2 are the head loss coefficients of 1 and 2 when they are operating as inlets. CS1 and CS2 are the head loss coefficients of 1 and 2 when they are operating as outlets. LRG is the width of the culvert. HAUT1 and HAUT2 are the heights of the construction work (in meters) at the inlet and outlet. The flow direction is also imposed through the keyword CLP:
CLP = 0, flow is allowed in both directions,
CLP = 1, flow is only allowed from section 1 to section 2,
CLP = 2, flow is only allowed from section 2 to section 1,
CLP = 3, no flow allowed.
LBUS is the linear head loss in the culvert, generally equal to $\lambda \frac{L}{D}$ where L is the length of

the pipe, D its diameter and l the friction coefficient. Z1 and Z2 are the levels of the inlet and outlet. CV refers to the loss coefficient due to the presence of a valve and C56 is the constant used to differentiate flow types 5 and 6 in the formulation by Bodhaine. C5 and CV5 represent correction coefficients to C1 and to CV coefficients due to the occurrence of the type 5 flow in the Bodhaine formulation. CT is the loss coefficient due to the presence of trash screens. FRIC is the Manning Strikler coefficient. LENGTH is the length of the culvert, and the culvert's shape can be specified through the parameter CIRC (equal to 1 in case of a circular section, 0 for a rectangular section). A1 and A2 are the angles with respect to the x axis. D1 and D2 are the angles that the pipe makes with respect to the bottom, in degrees. For a vertical intake, the angle with the bottom will therefore be 90°. They are used to account for the current direction at source or sink point. AA is a parameter which allows the user to choose whether A1 and A2 are automatically computed by TELEMAC-3D or whether the data file values are used to set these angles: AA=1 – automatic angle; AA=0 – user-set angle.

11. Other configurations

11.1 Modification of bottom topography (USER_T3D_CORFON)

Bottom topography may be introduced at various levels, as stated in section 3.21.

TELEMAC-3D offers the possibility of modifying the bottom topography at the beginning of a computation using the **USER_T3D_CORFON** subroutine. This is called up once at the beginning of the computation and enables the value of variable **ZF** to be modified at each point of the mesh. To do this, a number of variables such as the point coordinates, the element surface value, connectivity table, etc. are made available to the user.

By default, the **T3D_CORFON** subroutine (which calls the **USER_T3D_CORFON** subroutine) carries out a number of bottom smoothings equal to **LISFON**, i.e. equal to the number specified by the keyword `NUMBER OF BOTTOM SMOOTHINGS` for which the default value is 0 (no smoothing). The call to bottom smoothings can be done after or before the call to **USER_T3D_CORFON** in the **T3D_CORFON** subroutine with the keyword `BOTTOM SMOOTHINGS AFTER USER MODIFICATIONS`. By default, potential bottom smoothings are done after potential modifications of the bottom in **USER_T3D_CORFON** subroutine (default = YES).

The **T3D_CORFON** subroutine is not called up if a computation is continued. Neither is **USER_T3D_CORFON** subroutine. This avoids having to carry out several bottom smoothings or modifications of the bottom topography during the computation.

11.2 Modifying coordinates (USER_CORRXY)

TELEMAC-3D also offers the possibility of modifying the mesh point coordinates at the beginning of a computation. This means, for example, that it is possible to change the scale (from that of a reduced-scale model to that of the real object), rotate or translate the object.

The modification is done in the **USER_CORRXY** subroutine (BIEF library), which is called up at the beginning of the computation. This subroutine is empty by default and gives an example of programming a change of scale and origin, within commented statements.

It is also possible to specify the coordinates of the origin point of the mesh. This is done using the keyword `ORIGIN COORDINATES` which specify 2 integers (default = (0;0)). These 2 integers will be transmitted to the results file in the SERAFIN format, for a use by post-processors for superimposition of results with digital maps (coordinates in meshes may be reduced to avoid large real numbers). These 2 integers may also be used in subroutines under the names **I_ORIG** and **J_ORIG**. Otherwise they do not have a use yet.

11.3 Spherical coordinates (LATITU)

If a simulation is performed over a large domain, TELEMAC-3D offers the possibility of running the computation with spherical coordinates.
This option is activated when the keyword `SPHERICAL COORDINATES` is set to YES (default value = NO). In this case, TELEMAC-3D calls a subroutine named **LATITU** through the subroutine **INBIEF** at the beginning of the computation. This calculates a set of tables depending on the latitude of each point. To do this, it uses the Cartesian coordinates of each point provided in the geometry file, and the latitude of origin point of the mesh provided by the user in the steering file with the keyword `LATITUDE OF ORIGIN POINT` (default value = 0 degrees).
The spatial projection type used for the mesh is then specified with the keyword `SPATIAL PROJECTION TYPE`. That can take the following values:

- 1 : Lambert Cartesian not geo-referenced,
- 2 : Mercator (default value),
- 3 : Latitude/longitude (in degrees).

In this case of option 3, the coordinates of the mesh nodes should be expressed with latitude and longitude in degrees. TELEMAC-3D then converts with the information with the help of the Mercator's projection. It is important to notice here that, if option `SPHERICAL COORDINATES` = YES, `SPATIAL PROJECTION TYPE` has to be 2 or 3.
The **LATITU** subroutine (BIEF library) may be modified by the user to introduce any other latitude-dependent computation.

11.4 Adding new variables

A standard feature of TELEMAC-3D is the storage of some computed variables. In some cases, the user may wish to compute other variables and store them in the results file (the number of variables is currently limited to four).
Since TELEMAC-3D uses 2D and 3D variables, the treatments linked to these variables may differ and call three subroutines:

- **NOMVAR_2D_IN_3D**: to manage 2D variables names,
- **NOMVAR_TELEMAC3D**: to manage 3D variables names,
- **USER_PRERES_TELEMAC3D**: to compute new variables (2D and 3D).

TELEMAC-3D has a numbering system in which, for example, the array containing the Froude number has the number 7. The new variables created by the user may have the numbers 25, 26, 27 and 28 (for 3D variables) and 27, 28, 29 and 30 (for 2D variables).
In the same way, each variable is identified by a letter in the keywords `VARIABLES FOR 2D GRAPHIC PRINTOUTS` and `VARIABLES FOR 3D GRAPHIC PRINTOUTS`. The new variables are identified by the strings **PRIVE1, PRIVE2, PRIVE3** and **PRIVE4** for 2D variables and **P1, P2, P3** and **P4** for 3D variables.
At the end of the **NOMVAR_TELEMAC3D** or **NOMVAR_2D_IN_3D** subroutines, it is possible to change the abbreviations (mnemonics) used for the keywords `VARIABLES FOR 2D GRAPHIC PRINTOUTS` and `VARIABLES FOR 3D GRAPHIC PRINTOUTS`. Sequences of 8 letters may be used. Consequently, the variables must be separated by spaces, commas or semicolons in the keywords, e.g.:

```
VARIABLES FOR 2D GRAPHIC PRINTOUTS : 'U, V, H, B'
```

In the software data structure, these four variables correspond to the tables **PRIVE%ADR(1)%P%R(X), PRIVE%ADR(2)%P%R(X), PRIVE%ADR(3)%P%R(X)** and **PRIVE%ADR(4)%P%R(X)** (in which **X** is the number of nodes in the mesh). These may be used in several places in the programming, like all TELEMAC variables. For example, they may be used in the subroutines **USER_CORRXY**, **USER_CORSTR**, **USER_BORD3D**, etc. If a **PRIVE** table is used to program a case, it is essential to check the value of the keyword `NUMBER OF PRIVATE ARRAYS`. This value fixes the number of tables used (0, 1, 2, 3 or 4) and then determines the amount of memory space required. The user can also access the tables via the aliases **PRIVE1, PRIVE2, PRIVE3** and **PRIVE4**.
An example of programming using the second **PRIVE** table is given below. It is initialised with the value 10.

```
DO I=1,NPOIN2
  PRIVE%ADR(2)%P%R(I) = 10.D0
ENDDO
```

New variables are programmed in two stages:

- Firstly, it is necessary to define the name of these new variables by filling in the **NOMVAR_TELEMAC3D** (or **NOMVAR_2D_IN_3D**) subroutine. This consists of two equivalent structures, one for English and the other for French. Each structure defines the name of the variables in the results file that is to be generated and then the name of the variables to be read from the previous computation if this is a restart. This subroutine may also be modified when, for example, a file generated with the English version of TELEMAC-3D is to be continued with the French version. In this case, the **TEXTPR** table of the French part of the subroutine must contain the English names of the variables,
- Secondly, it is necessary to modify the **USER_PRERES_TELEMAC3D** subroutine in order to introduce the computation of the new variable(s). The variables **LEO, SORG2D** and **SORG3D** are also used to determine whether the variable is to be printed in the printout file or in the results file at the time step in question.

User arrays can be handled to store extra variables in 2D with the help of two keywords to define the number and the name of the extra variables in the 2D private arrays: `NUMBER OF 2D PRIVATE ARRAYS` (up to 4, default value = 0) and `NAMES OF 2D PRIVATE VARIABLES`. It is the names of the user arrays **PRIVE%ADR(1)%P, PRIVE%ADR(2)%P** ... up to 4, that will be seen in the results files. The great advantage is that these variables will be read if present in the `GEOMETRY FILE`.

11.5 Array modification or initialization

When programming TELEMAC-3D subroutines, it is sometimes necessary to initialize a table or memory space to a particular value. To do that, the BIEF library furnishes a subroutine called **FILPOL** that lets the user modify or initialize tables in particular mesh areas.
A call of the type **CALL FILPOL (F, C, XSOM, YSOM, NSOM, MESH)** fills table **F** with the **C** value in the convex polygon defined by **NSOM** nodes (coordinates **XSOM, YSOM**). The variable **MESH** is needed for the **FILPOL** subroutine but has no meaning for the user.

11.6 Validating a computation (BIEF_VALIDA)

The structure of the TELEMAC-3D software offers an entry point for validating a computation, in the form of a subroutine named **BIEF_VALIDA**, which has to be filled by the user in accordance with each particular case. Validation may be carried out either with respect to a reference

file (which is therefore a file of results from the same computation that is taken as reference, the name of which is supplied by the keyword REFERENCE FILE), or with respect to an analytical solution that must then be programmed entirely by the user.
When using a reference file, the keyword REFERENCE FILE FORMAT specifies the format of this binary file ('SERAFIN ' by default).
The **BIEF_VALIDA** subroutine is called at each time step when the keyword VALIDATION has the value YES, enabling a comparison to be done with the validation solution at each time step. By default, the **BIEF_VALIDA** subroutine only does a comparison with the last time step. The results of this comparison are given in the output listing.

11.7 Coupling

The principle of coupling two (or in theory more) simulation modules involves running the two calculations simultaneously and exchanging the various results at each time step. For example, the following principle is used to couple a hydrodynamic module and a sediment transport module:

- The two codes perform the calculation at the initial instant with the same information (in particular the mesh and bottom topography),
- The hydrodynamic code runs a time step and calculates the water depth and velocity components. It provides this information to the sediment transport code,
- The sediment transport code uses this information to run the solid transport calculation over a time step and thus calculates a change in the bottom,
- The new bottom value is then taken into account by the hydrodynamic module at the next time step, and so on.

Several modules can be coupled in the current release of the code: the sediment transport modules GAIA the sea state computational module TOMAWAC with 4 options (see below to see the descriptions of the options) and the water quality module WAQTEL (and even DELWAQ). The time step used for the two calculations is not necessarily the same and is managed automatically by the coupling algorithms and the keyword COUPLING PERIOD FOR TOMAWAC with the default value 1 (coupling at every iteration).
This feature requires two keywords. The keyword COUPLING WITH indicates which simulation code is to be coupled with TELEMAC-3D. The values of this keyword can be:

- COUPLING WITH = 'GAIA' for coupling with the GAIA module,
- COUPLING WITH = 'TOMAWAC' for coupling with the TOMAWAC module, forces are constant along the vertical (like with TELEMAC-2D),
- COUPLING WITH = 'TOMAWACT3D' for coupling with the TOMAWAC module but forces induced by waves are 3D,
- COUPLING WITH = 'TOMAWAC2' for coupling with the TOMAWAC module, forces are constant along the vertical, but using the feature tel2tom, the meshes and the domains of TELEMAC-3D and TOMAWAC can be different,
- COUPLING WITH = 'TOMAWACT3D2' for coupling with the TOMAWAC module, forces induced by waves are 3D, but using the feature tel2tom, the meshes and the domains of TELEMAC-3D and TOMAWAC can be different,

- COUPLING WITH = 'WAQTEL' for coupling with the WAQTEL module.

If wanting to couple with 2 modules or more, the differents modules are to be written in the COUPLING WITH separated with semicolon.
Depending on the module(s) used, the keywords GAIA STEERING FILE, TOMAWAC STEERING FILE and WAQTEL STEERING FILE indicate the names of the steering files of the coupled modules.

If coupling with the water quality module WAQTEL, the integer keyword WATER QUALITY PROCESS must be set to a value different from 1 (default = 1) in the TELEMAC-3D steering file. The possible choices are:

- 0: all available processes,
- 1: nothing (default value),
- 2: O_2 module,
- 3: BIOMASS module,
- 5: EUTRO module,
- 7: MICROPOL module,
- 11: THERMIC module,
- 13: AED2 model,
- 17: degradation law.

Several modules can be combined by giving the multiplication of the process choices, e.g. 55 = 5 $\times$ 11 activates EUTRO and THERMIC modules. It is noted that AED2 should be used on its own, for the time being, without possible combination with other processes. Please refer to the WAQTEL documentation for additional informations for WAQTEL.

If coupling with the TOMAWAC in a 3D way (the keyword COUPLING WITH including TOMAWACT3D or TOMAWACT3D2), the keyword BOTTOM FRICTION DUE TO WAVES enables to take into account the momentum lost by waves due to bottom friction (default = NO). A fine mesh around the bottom is needed to be accurate.

The use of the feature tel2tom to have different meshes with TELEMAC-3D and TOMAWAC requires to calculate the weights of interpolation between the two meshes before the simulation. Those weights can be obtained and put in the meshes through the use of the command:

```
run_telfile.py tel2tom
```

An example of use of this command is done in the notebook $HOMETEL/pretel/tel2tom.ipynb

The keyword COUPLING WITH is also used if the computation has to generate the appropriate files necessary to run a water quality simulation with DELWAQ. In that case, it is necessary to specify COUPLING WITH = 'DELWAQ'. Please refer to Appendix D for all informations concerning communications with DELWAQ.

11.8 Checking the mesh (CHECKMESH)

The **CHECKMESH** subroutine of the BIEF library is available to look for errors in the mesh, e.g. superimposed points ... The keyword `CHECKING THE MESH` (default value = NO) should be activated to YES to call this subroutine.

12. Parallelism

For simulations requiring a high computational power, it can be advisable to run the computations in multi-processor machines, or in clusters of workstations. TELEMAC-3D is available in a parallel version in order to take advantage of that kind of computational architecture.
The TELEMAC-3D parallel version uses the MPI library which has to be installed beforehand to be implementable. The interface between TELEMAC-3D and that MPI library is achieved through the **parallel** library which is common to all the TELEMAC system modules.
Lots of pieces of information concerning the implementation of the parallel version can be found in the system's installation literature.
The user shall initially specify the number of processors used by means of the keyword `PARALLEL PROCESSORS`. That integer type keyword can take the following values:

- 0: Use of the conventional TELEMAC-3D version (default),
- 1: Use of the parallel TELEMAC-3D version on one processor,
- 2 ...: Use of the parallel TELEMAC-3D version using the specified number of processors.

Different partitioning tools may be used if installed. They can be selected by the keyword `PARTITIONING TOOL` (default value is METIS). The possible choices are:

- METIS,
- SCOTCH,
- PARMETIS,
- PTSCOTCH ...

A. API

Information on the TELEMAC-3D API can be found in the telapy user documentation.

A. Launching the computation

A computation is launched through the **telemac3d.py** command. That command activates the execution of a script which is common to all the computation modules in the TELEMAC system. Depending on the platform, some options may be unavailable.
The syntaxes in that command are as follows:

```
telemac3d.py [cas]    [--options]
```

- cas: name of the steering file,
- –ncsize=NCSIZE: specifies the number of processors forced in parallel mode, default = the number defined in the steering file,
- -c CONFIGNAME or –configname=CONFIGNAME: specifies the configuration name, default is randomly found in the configuration file,
- -f CONFIGFILE, –configfile=CONFIGFILE: specifies the configuration file, default = systel.cfg,
- -s, –sortiefile: specifies whether there is a sortie file, default is no,
- -t or –tmpdirectory: the temporary work directory is not destroyed on completion of computation.

By default, the procedure runs the computation in an interactive mode and displays the control listing on the monitor.
The operations performed by that script are as follows:

- Creation of a temporary directory (**name_cas_YYYY-MM-DD_HHhMMminSSs**),
- Duplication of the dictionary and the input files into that directory,
- Execution of the DAMOCLES software in order to determine the work file names,
- Creation of the computation launching script,
- Allocation of the files,
- Compilation of the FORTRAN file and link editing (as required),

- Launching of the computation,
- Retrieval of the results files and destruction of the temporary directory.

The procedure takes place with slight differences according to the selected options.
The detailed description of that procedure can be obtained through the help command:

```
telemac3d.py --help
telemac3d.py -h
```

B. List of user subroutines

Even though all subroutines can be modified by the user, some subroutines have been specifically designed to define complex simulation parameters. They are listed below:

CALCOT	Preparation of the array of mesh elevations between the bottom and the free surface
CONDIS	Initialization of the arrays of the physical sedimentological quantities
DECLARATIONS_TELEMAC3D	Statement of the TELEMAC-3D structures
DRSURR	Computation of density (equation of state)
NOMVAR_TELEMAC3D	Definition of the names of variables for the graphic printouts
SOURCE_TRAC	User source term in the equations of tracers
TRISOU	Source terms for the velocity components
USER_BORD3D	Management of the boundary conditions
USER_CONDI3D_H	Management of the initial conditions for water depth
USER_CONDI3D_KEP	Management of the initial conditions for k and ε
USER_CONDI3D_P	Management of the initial conditions for pressure
USER_CONDI3D_TRAC	Management of the initial conditions for tracer(s)
USER_CONDI3D_UVW	Management of the initial conditions for velocity components
USER_CORRXY	Modification of the mesh node coordinates
USER_CORSTR	Correction of the bottom friction coefficient when it is time-variable
USER_DRIUTI	User damping function
USER_FLOT3D	Management of drogues
USER_KEPINI	Management of the initial conditions for k and ε
USER_LIMI3D	Management of the boundary conditions
USER_MESH_TRANSF	Definition of the mesh transformation (distribution of the horizontal planes)
USER_PRERES_TELEMAC3D	Computation of variables which will be written in the results file or listing
USER_Q3	Management of the flow rates in a boundary condition (may be time-dependent)
USER_SL3	Management of the free surface elevation in a boundary condition (may be time-dependent)
USER_SOURCE	User source term in the hydrodynamic equation

USER_STRCHE	Management of space-dependent friction coefficient
USER_T3D_CORFON	Modification of bottom elevations
USER_T3D_DEBSCE	Management of discharge at source points (possible time dependent)
USER_T3D_TRSCE	Management of tracers at source points (possible time dependent)
USER_TR3	Management of the tracers in a time-dependent boundary condition
USER_TRA_PROF_Z	Definition of the vertical tracer profile
USER_UTIMP	Additional variable writing
USER_VEL_PROF_Z	Definition of the vertical velocity profile
USER_VIT3	Management of the velocities in a time-dependent boundary condition
VISCLM	Computation of the viscosity in the mixing length models
VISCOS	Computation and initialization of the constant viscosity

C. The SELAFIN format

Note: historically, this format was called SERAFIN and its file extension was often ".srf". At some point in the TELEMAC SYSTEM development history, it has been decided to switch to the MED format. As a joke, SERAFIN was then renamed to SELAFIN, to reflect French "C'est la fin", meaning "This is the end" (of the SERAFIN format). However, MED never replaced SELAFIN, which remains the most widely used of the two formats.

The SELAFIN file format is binary-based.

This format can be 'SELAFIN', for single precision storage, or 'SELAFIND' for double precision storage. Double precision storage can be used for cleaner restarts, but may not be understood by all post-processors.

All strings in a SELAFIN file must be utf-8 encoded (See for `https://en.wikipedia.org/wiki/UTF-8` for the exact list).

The records are listed below. Records are given in the FORTRAN sense. It means that every record corresponds to a FORTRAN WRITE:
1 record containing the title of the study (80 characters), The last 8 characters must contain the format of the file (SELAFIN or SELAFIND)
1 record containing the two integers NBV(1) and NBV(2) (NBV(1) the number of variables, NBV(2) with the value of 0),
NBV(1) records containing the names and units of each variable (over 32 characters),
1 record containing the integers table IPARAM (10 integers, of which only 4 are currently being used).
If IPARAM (3) is not 0: the value corresponds to the x-coordinate of the origin in the mesh
If IPARAM (4) is not 0: the value corresponds to the y-coordinate of the origin in the mesh
These coordinates in metres may be used by post-processors to retrieve geo-referenced coordinates, while the coordinates of the mesh are relative to keep more digits.
If IPARAM (7) is not 0: the value corresponds to the number of planes on the vertical (in prisms.)
If IPARAM (8) is not 0: the value corresponds to the number of boundary points (in parallel).
If IPARAM (9) is not 0: the value corresponds to the number of interface points (in parallel).
if IPARAM (10) = 1: a record containing the computation starting date in 6 integers: year, month, day, hour, minute, second

1 record containing the integers NELEM,NPOIN,NDP,1 (number of elements, number of points, number of points per element and the value 1),
1 record containing table IKLE (integer array of dimension (NDP,NELEM) which is the connectivity table. Beware: in TELEMAC-2D, the dimensions of this array are (NELEM,NDP)),
1 record containing table IPOBO (integer array of dimension NPOIN); the value is 0 for an internal point, and gives the numbering of boundary points for the others. This array is never used (its data can be retrieved by another way). In parallel the table KNOLG is given instead, keeping track of the global numbers of points in the original mesh.
1 record containing table X (real array of dimension NPOIN containing the abscissas of the points),
1 record containing table Y (real array of dimension NPOIN containing the ordinates of the points),
Next, for each time step, the following are found:

- 1 record containing time T (real),
- NBV(1)+NBV(2) records containing the results arrays for each variable at time T.

D. Generating output files for DELWAQ

The TELEMAC-3D software is able to generate the appropriate files necessary to run a DELWAQ water quality simulation. This generation is managed only through the following keywords:

```
BOTTOM SURFACES DELWAQ FILE
DELWAQ PRINTOUT PERIOD
DELWAQ STEERING FILE
DIFFUSION FOR DELWAQ
DIFFUSIVITY DELWAQ FILE
EXCHANGE AREAS DELWAQ FILE
EXCHANGES BETWEEN NODES DELWAQ FILE
NODES DISTANCES DELWAQ FILE
SALINITY DELWAQ FILE
SALINITY FOR DELWAQ
TEMPERATURE DELWAQ FILE
TEMPERATURE FOR DELWAQ
VELOCITY DELWAQ FILE
VELOCITY FOR DELWAQ
VERTICAL FLUXES DELWAQ FILE
VOLUMES DELWAQ FILE
```

More information about these keywords can be found in the TELEMAC-3D reference manual. For more information, please refer to the DELWAQ user documentation.

E. Some recommendations for TELEMAC-3D simulations

If it is not too bad for you to increase your computational time, here are some recommendations about the options to use.

E.1 Advection

- use the predictor-corrector distributive scheme with local implicitation for tidal flats, based on the PSI scheme, for the velocity and all tracers, including k and ε if you have them:
  / PSI scheme
  `SCHEME FOR ADVECTION OF VELOCITIES` = 5
  `SCHEME FOR ADVECTION OF K-EPSILON` = 5
  `SCHEME FOR ADVECTION OF TRACERS` = 5
  / predictor-corrector with local implicitation implicite
  `SCHEME OPTION FOR ADVECTION OF VELOCITIES` = 4
  `SCHEME OPTION FOR ADVECTION OF K-EPSILON` = 4
  `SCHEME OPTION FOR ADVECTION OF TRACERS` = 4
  These are the new default values since release 8.1.

- without tidal flats, use the predictor-corrector without tidal flats treatment to gain some time
  / Predictor-corrector without tidal flats treatment
  `SCHEME OPTION FOR ADVECTION OF VELOCITIES` = 2
  `SCHEME OPTION FOR ADVECTION OF K-EPSILON` = 2
  `SCHEME OPTION FOR ADVECTION OF TRACERS` = 2

E.2 Friction

- use the Nikuradse law, which is the only one that is really 3D compatible: the other options have been implemented to ensure an easy correspondance between 2D and 3D simulations, but their use is not recommended for fully 3D studies:
  `LAW OF BOTTOM FRICTION` = 5,

- use a friction coefficient that is representative of the physics of your case: with the Nikuradse law, the friction coefficient corresponds to the characteristic height of the roughness
  FRICTION COEFFICIENT FOR THE BOTTOM = 3 × d90.
  For a natural bed with sediment of characteristic size d90 and without riddles or dunes, its value should be taken from van Rijn, equal to 3 × d90. In the presence of dunes, van Rijn proposes another formula,

- depending on the nature of the lateral walls, either use a Nikuradse law or no friction
  LAW OF FRICTION ON LATERAL BOUNDARIES = 0 or 5,

- as for the bed, use a physical value for FRICTION COEFFICIENT FOR LATERAL SOLID BOUNDARIES,

- remark: contrary to the friction coefficients used with the laws derived from the shallow water context (Manning, Strickler, Chézy, Haaland), the friction coefficient with the Nikuradse law does not "hide" physical effects like diffusion-dispersion, like it does with the shallow-water equations. Thus, it is less prone to being used as a calibration parameter in the studies.

E.3 Mass-lumping

- avoid using mass-lumping, unless you really have a strong constraint on computational time (this is the default behaviour of TELEMAC-3D): it introduces artificial dispersion in the results.

E.4 Semi-implicitation

- do not change the default value for
  IMPLICITATION FOR DEPTH (= 0.55)
  and use
  IMPLICITATION FOR VELOCITIES = 0.55, new default value since release 8.1.
  Warning: never use a value lower or equal to 0.5 for these coefficients, otherwise the scheme becomes unconditionnally unstable. Avoid using the value 1 too, we have observed a strange behaviour of the scheme in this case: it seems to introduce quite a lot of spurious diffusion, even though 0.99 does not... We should perform more testing and analysis to better understand this behaviour.

- the keyword IMPLICITATION FOR DIFFUSION can be left to its value by default, 1.

E.5 Linear solvers

- use the GMRES solver (= 7) for the matrices inversion, except diffusion (SOLVER FOR PPE, SOLVER FOR PROPAGATION). For the diffusion, the conjugate gradients should be good enough (solver number = 1). These are the new default values since release 8.1,

- use an accuracy equal to 10^{-8} or lower for all solvers (ACCURACY FOR PPE, ACCURACY FOR PROPAGATION, ACCURACY FOR DIFFUSION OF VELOCITIES, ACCURACY FOR DIFFUSION OF K-EPSILON, ACCURACY FOR DIFFUSION OF TRACERS, ACCURACY FOR DIFFUSION OF SEDIMENT). These are the new default values since release 8.1,

- be careful to always check in your listing that your solvers have actually converged, otherwise the simulation may complete but with wrong results. In case you do not reach

the accuracy you asked for with these options, try increasing the value of `OPTION OF SOLVER FOR [...]` to 10. It sets the size of the Krylov subspaces used in the GMRES solver. You can also try using conjugate gradients or Bi-CGSTAB, or increasing the maximum number of iterations of the solvers, or try using the preconditionning number 34 to make the matrix inversions faster.

E.6 Fractional steps method

- the keyword `DYNAMIC PRESSURE IN WAVE EQUATION` can be left to its value by default NO, but setting it to YES may slightly reduce the numerical diffusion (altough we need to test it further to really characterize the behaviour of these two choices).

E.7 Free surface

- if you observe free-surface instabilities (wiggles) in your results, you can try using a lower value for the
  `FREE SURFACE GRADIENT COMPATIBILITY`
  than the one by default, for example 0.9.

E.8 Liquid boundaries

- we would suggest trying not to use the Thompson boundary conditions, because they are actually only valid in the shallow water context. However, they may help stabilise the simulations involving liquid boundaries, so it is not always that easy not to use them. You should try without it before trying to use it.

E.9 Tidal flats

- use `TREATMENT OF NEGATIVE DEPTHS` = 2 if possible,
- use `TREATMENT ON TIDAL FLATS FOR TRACERS` = 1 to ensure conservation.

E.10 Turbulence

- trying to use the k-ε model (= 3) on the vertical and the horizontal directions, or the Spalart-Allmaras one (= 5), also in both directions, instead of the mixing length model which is not flexible, since it is a zero-equation model. However, for stratification simulations, the mixing length model may provide better results than any other (the $k-\varepsilon$ model tends to mix too much).

E.11 Other numerical options

- do not change the matrix storage, simply do not include it from the steering file.

[1] JOLY A., GOEURY C., and HERVOUET J.-M. Adding a particle transport module to telemac-2d with applications to algae blooms and oil spills. Technical Report H-P74-2013-02317-EN, EDF R&D-LNHE, 2013.

[2] Cawthorn C., Bourban S., Mead C., Wood M., and Turnbull S. Adaptive vertical layering in telemac-3d. In *XVIIIth TELEMAC-MASCARET User Club*, Chatou, France, 2011.

[3] PHAM C.-T., BOURBAN S., DURAND N., and TURNBULL M. Méthodologie pour la simulation de la marée avec la version 6.2 de telemac-2d et telemac-3d. Technical Report H-P74-2012-02534-FR, EDF R&D-LNHE, 2012.

[4] G. D. Egbert and S. Y. Erofeeva. Efficient inverse modeling of barotropic ocean tides. *Journal of Atmospheric and Oceanic Technology*, 19(2):183–204, 2002. doi: https://doi.org/10.1175/1520-0426(2002)019<0183:EIMOBO>2.0.CO;2. URL `https://journals.ametsoc.org/view/journals/atot/19/2/1520-0426_2002_019_0183_eimobo_2_0_co_2.xml`.

[5] J-M. HERVOUET. *Hydrodynamics of free surface flows. Modelling with the finite element method.* John Wiley & Sons, Ltd, Paris, 2007.

[6] TSANIS I. Simulation of wind-induced water currents. *Journal of hydraulic Engineering*, 115(8):1113–1134, 1989.

[7] SMAGORINSKY J. General simulation experiments with the primitive equations. *Monthly Weather Review*, 91(3):99–164, March 1963.

[8] HERVOUET J.-M. Guide to programming in the telemac system version 6.0. Technical Report H-P74-2009-00801-EN, EDF R&D-LNHE, 2009.

[9] J.-M. Janin and X. Blanchard. Simulation des courants de marée en manche et proche atlantique. Technical Report HE-42/92.58, EDF DER-LNH, 1992.

[10] F. H. Lyard, D. J. Allain, M. Cancet, L. Carrère, and N. Picot. Fes2014 global ocean tide atlas: design and performance. *Ocean Science*, 17(3):615–649, 2021. doi: 10.5194/os-17-615-2021. URL `https://os.copernicus.org/articles/17/615/2021/`.

[11] GAUTHIER M. and QUETIN B. Modèles mathématiques de calcul des écoulements induits par le vent. In *17e congrès de l'AIRH*, Baden-Baden, August 1977.

[12] I. L. Pairaud, F. Lyard, F. Auclair, T. Letellier, and P. Marsaleix. Dynamics of the semi-diurnal and quarter-diurnal internal tides in the bay of biscay. part 1: Barotropic tides. *Continental Shelf Research*, 28(1011):1294–1315, 2008.

[13] I. L. Pairaud, F. Auclair, T. Letellier, P. Marsaleix, F. Lyard, and A. Pichon. Dynamics of the semi-diurnal and quarter-diurnal internal tides in the bay of biscay. part 2: Baroclinic tides. *Continental Shelf Research*, 30(3-4):253–269, 2010.

[14] L. Pineau Guillou. Previmer. validation des atlas de composantes harmoniques de hauteurs et courants de marée. Technical report, Ifremer, June 2013. URL `https://marc.ifremer.fr/content/download/7861/41823/file/2013_06_12_rap_valid_atlas_V1.pdf`. ODE/DYNECO/PHYSED/2013-02 1.0.

[15] E. Taguchi, D. Stammer, and Zahel W. Inferring deep ocean tidal energy dissipation from the global high-resolution data-assimilative hamtide model. *J. Geophys. Res.Oceans*, 119(7):4573–4592, 2014. doi: https://doi.org/10.1002/2013JC009766. URL `https://agupubs.onlinelibrary.wiley.com/doi/epdf/10.1002/2013JC009766`.