# Chapter 1

# General Overview

## 1.1 Introduction

During the last 25 years several two- and three-dimensional numerical model codes have been developed and made available for applications in the global ocean, shelf and regional seas, the coastal zones and estuaries. Without being complete we can mention public domain codes such as the global ocean MOM model (Modular Ocean Model; Pacanowski & Griffies, 2000), the POM (Princeton Ocean Model; Blumberg & Mellor, 1987) and ROMS (Regional Ocean Model System) models for regional seas, FVCOM (Chen *et al.*, 2006) and GETM (Burchard & Bolding, 2002) for application in coastal and estuarine environments. Besides these free software codes there exist several licensed commercial softwares like TELEMAC, MIKE, Delft3d. The advantage of a public domain code is that the source code is freely available and can be modified by an experienced user. This is obviously not the case for commercial codes. On the other hand, developers of free codes have no obligation for providing user support which should, obviously, be the case for the commercial ones. However, the lack of a substantial support can be compensated by providing a detailed manual and model documentation to the users.

A special category of the open source codes are the integrated "Community Models" which couple a core physical component with modules for biology, sediments, waves, .... The construction of such models is a multidisciplinary task which requires the collaboration of a variety of specialists and consumes a substantial number of years of scientific and programming effort for designing, developing and testing the model code and to validate the model against observational data.

COHERENS (coupled hydrodynamical-ecological model for regional and

shelf seas) is a three-dimensional hydrodynamic multi-purpose model for coastal and shelf seas, which resolves mesoscale to seasonal scale processes. The program has been developed initially over the period of 1990–1998 by a multinational European group as part of the MAST projects PROFILE, NOMADS and COHERENS funded by the European Union. During its development it was applied for studies in the North Sea and regions of fresh water influence (ROFIs) (Huthnance, 1997; Proctor, 1997; Luyten, 1999).

The first version, COHERENS V1[1], has been released in April 2000 as open source code. Besides the source code, the release contains an extensive User Documentation (Luyten *et al.*, 1999) and a series of test cases which can be used to test the installation and as an illustration of the different options available in the program.

The program was written in FORTRAN 77 and has four major components:

1. A physical part with a general module for solving advection-diffusion equations.

2. A simple microbiological module which deals with the dynamics of microplankton, detritus, dissolved inorganic nitrogen and oxygen.

3. An Eulerian sediment module which deals with deposition and resuspension of inorganic as well as organic particles.

4. A component with both an Eulerian and a Lagrangian transport model for contaminant distributions.

Its ease of implementation across a range of computing platforms made the program attractive to groups with a sufficient expertise in modelling and in need of sophisticated model products. Important advantages of the model are its transparency because of its modular structure and its flexibility because of the possibility to select different processes, specific schemes or different types of forcing for a particular application. The program structure allows users to perform process as well as predictive and operational setups without knowledge of the detailed model structure. Since the official release more than 1000 registrations have been recorded on the COHERENS home Web page.

`http://www.mumm.ac.be/coherens`

[1]The original version number 8.4 has been replaced by V1 to be compatible with the currently adopted versioning system.

where a publication list can be found with applications of COHERENS.

The work on a fully updated version of COHERENS started in 2003 as part of the EU-FP5 and FP6 projects ODON (2003–2005) and ECOOP (2006-2009). The first objectives were to produce an optimised code so that it can be implemented on parallel machines and to provide user interfaces for model setup independent of the main source code and for user-defined formats for external files representing model forcing. A first version was presented at the 2006 EUROGOOS meeting (Luyten *et al.*, 2006). In response to various user requests new algorithms have been implemented or existing ones have been improved (e.g. drying/wetting scheme, additional types of open boundary conditions, turbulence schemes, baroclinic pressure gradient, astronomical forcing, one-way nesting ...). More flexible model grids can be selected using orthogonal curvilinear coordinates in the horizontal and extended $\sigma$-coordinates (also known as $s$-coordinates) in the vertical.

The new version, denoted as COHERENS V2.0, is now available for the scientific commnunity as open source code. Despite the numerous additional implementations the new code only contains a physical component, in contrast to COHERENS V1. In the framework of ongoing projects the development of a fully integrated version is now in progress. This includes in the first place the coupling with a generic biological model, a sediment transport model taking account of different size classes and a bottom boundary layer, a morphological module, a Langrangian tracer module wich can also be applied for the dispersion of oil slicks and contaminants in general, and with surface wave models such as WAM and SWAN. Additional improvements are an implicit time integration scheme and schemes allowing to represent inondation over a land topography.

## 1.2 Description of the program

The characteristics of COHERENS V2.0 are summarised below where a (*) marks a new implementation or a change with respect to COHERENS V1.

1. Model code

   - The code has been re-written in FORTRAN 90(*) replacing the FORTRAN 77 used in the previous version. Important advantages are:
     - dynamic allocation of model arrays. This permits a more efficient usage of internal memory and makes it easier to implement a domain decomposition for parallel applications.

  - modules replacing the INCLUDE statements in FORTRAN 77 and module routines used to create "internal libraries".
  - derived type variables for storing information of different "entities" (files, model variables, external grid information) in a structured format.
  - array instead of element-wise assignment within DO loops.
  - generic routines.
- Simulations can (optionally) be performed on parallel clusters using the MPI (Message Passing Interface) communication library (*). Advantages are a much faster computing speed and the possibility to distribute internal memory over different processes.
- CPP preprocessing for conditional compilation via #ifdef statements (*). The method is primarely used for an easy implementation of external libraries. In this way the user can decide to include the MPI (for parallel processing) and netCDF libraries (for portable data formats) by setting a compiler switch.
- The program is (in principle) compiled with the UNIX/LINUX `make` utility. A generic (machine-independent) *Makefile* is provided (*). The program can, in principle, be compiled on a Windows platform. This is however not recommended.
- Model setup has been significantly changed with respect to COHERENS V1. Definitions of model parameters and input of model forcing data are made by calls to "usrdef_" routines (*) which can be considered as a kind of user interfaces. These routines have to provided by the user. Although this requires some programming skill of the user, model forcing can now be given in any kind of user-defined format and independently from the main source code. An option is foreseen to re-write each kind of forcing into a standard COHERENS format, including netCDF files (*). Most setup parameters have default values which, for most applications, do not need to be defined explicitly.
- The definitions of all model parameters (including default ones) can optionally be obtained from a Central Input File (CIF) instead of calling usrdef_ routines (*).
- A total of 84 switches are implemented for the selection of different physical processes (barotropic or baroclinic, turbulence closures, density effects, tidal forcing, nesting ...), type of model grid, numerical schemes (advection and diffusion, open boundary conditions, surface and bottom fluxes, drying and wetting algorithm,

time-integration, ...), type of parallel communication, netCDF output.

- A series of monitoring utilities is optionally available:
  - The execution of the program can be traced in a "log" file.
  - The program is checked for setup errors. When the program traps an error, the program aborts with an explanatory message. The level of error trapping can be selected by the user.
  - Suspicious values of setup parameters are reported in a "warning" file (*).
  - The program optionally writes a timer report containing the % of time spent by different model compartments (*). The utility can be applied to optimise parallel applications.
- Several simulations can be set up and executed within one run (*).
- Restart times can be defined where the program writes initial conditions for an eventual restart of the program (*).

2. Model grid

- A rectangular or a curvilinear (*) grid can be selected in the horizontal directions.
- The program uses $\sigma$-coordinates in the vertical. The grid size in $\sigma$-space can either be uniform, non-uniform vertically and non-uniform in both vertical and horizontal directions (*) (making it equivalent to the "so-called" $s$-coordinate system).
- The model can either be set up on a 3-D grid, in 2-D (depth-averaged) mode (*) or in 1-D (water column) mode.
- One-way sub-grid nesting (*) can be applied for 2-D and 3-D model quantities using an off-line procedure. The advantage is that multiple sub-grids can be simultaneously defined without any restrictions of grid sizes and sub-grid time steps.
- Interpolation of model grid arrays at a different node is performed either with uniform or non-uniform (*) weight factors.

3. Numerics

- The model equations are discretised on an Arakawa C-grid using either Cartesian or spherical coordinates as selected by the user.

- The mode-splitting technique is applied to solve the 2-D and 3-D momentum and continuity equations.
- The same time step is used for 3-D currents and 3-D scalars.
- Two different advection schemes are implemented: a first order upwind and a more accurate TVD (Total Variation Diminishing) scheme. A separate selection can be made for 2-D transports, 3-D currents, 3-D scalars and turbulence transport variables.

4. Physics

- The program runs by default in barotropic mode, i.e. without density effects. Temperature and salinity can be included via separate switches.
- Density effects in the momentum and turbulence equations are included via an equation of state. A correction has been made with respect to COHERENS V1 since the simulated temperature field represents potential temperature instead of the physical "in situ" one.
- The absorption of solar radiation in the upper part of the water column is implemented by an optical module.
- The program incorporates a variety of turbulence schemes ranging from simple algebraic formulations to one- or two-equation schemes (Mellor-Yamada, $k - \varepsilon$). Additional RANS models have been implemented (*).
- On request of the users, additional types of open boundary conditions have been implemented (*), while corrections have been made to some of the existing ones.
- Different formulations for the wind stress and surface heat fluxes are available. Additional schemes have been added (*).
- A drying/wetting algorithm is implemented in Version V2.0. An extended scheme for simulating the flooding of land areas and the flows over obstacles has been added in Version 2.3.
- A module has been added to simulate effects of hydraulic structures, i.e. dry cells, thin dams, weirs and barriers (*).
- A discharge module has been implemented (*).
- A wave-current interaction module has not yet been implemented in COHERENS V2.0.

5. Other compartments

   - An extended sediment model has been implemented in Version 2.5.

## 1.3 User experience

The COHERENS program has been designed such that it can be applied by users with a "low" or a "high" level of model experience. In the former case the program is set up by defining a limited number of parameters (e.g. time steps, a few model switches, attributes for model forcing, ...) whereas default values can be taken for most model parameters. Bathymetry, initial conditions, open boundary and surface forcing conditions are provided by the user via the usrdef_ model setup routines. To facilitate its use for experienced modellers, the program is supplied in modular form. This allows to replace a module by a user-defined version without affecting the core of the program. It is clear that this practice is not without danger especially when the module is linked with other program units. Changing the default settings of switches and model parameters is easily performed but not always without risk.

For a user it is important to know which specific experience is required to run COHERENS for a particular type of application. There are two basic criteria: programming experience and scientific background. They are further discussed below.

### 1.3.1 Programming experience

A basic knowledge of FORTRAN 90 is required. As explained in Part IV (User manual) a series of FORTRAN source code files need to be created for a user application. Generic example files have been made available to facilitate this. In this files the user only needs to replace or add a number of assignment statements or insert READ instructions for data input. All aspects of model setup are, in principle, covered by the usrdef_ interfaces. This makes this version more flexible than COHERENS V1, since there is no need to change the main code. However, if it is the intention to implement new schemes or to add new modules, the user must have, besides a good skill in programming with FORTRAN 90 have a good understanding of the general concept, including the parallellisation of the program, and internal structure of the COHERENS code. For this reason Part III (Model code) has been written for developers and covers all different aspects of the model code.

Installing and compiling COHERENS V2.0 only requires a limited knowledge of the UNIX/LINUX operating system. For installation and linking

with external libraries the user is referred to the manuals which are provided by the software developers and are freely available through the internet.

### 1.3.2 Scientific background

The default settings make it easier to run the model without a detailed knowledge of the underlying scientific basis. Only a limited number of default values need to be changed for most applications. Detailed instructions are given in this documentation together with a large number of examples (see Part V). A more specific scientific background is required if for example the user intends to perform experiments with different turbulence or numerical schemes or with alternative settings of model parameters. It is then recommended to read first the appropriate chapters of Part II (Model Description).

## 1.4 Contents of the documentation

### 1.4.1 Structure

This documentation is composed of six parts covering different aspects of the COHERENS program.

I. Introductory manual.

   Chapter 2 is a tutorial manual for beginning users. This chapter also presents various free-sofware for displaying COHERENS results. More detailed instructions for installing, preparing and running a pre-defined test case and a user application are given in Chapter 3.

II. Model description.
   The scientific background of COHERENS is described in Chapter 4. Numerical aspects are the subject of Chapter 5. In this chapter the discretisized forms of the model equations are written out in full detail. The implementation of hydraulic structures and discharges is discussed in Chapter 6. The sediment model is described in Chapter 7.

III. Description of the model code.
   This part is mainly intended for model developers, although some aspects can be useful for beginning users as well. Program conventions and model layout are explained in Chapter 8. Chapter 9 deals with

different aspects of model input and output: purpose of the files connected to the program (monitoring, forcing, model output defined by the user), implementation in the model, structure of a file in standard COHERENS format. Chapter 10 explains how arrays are defined on the model C-grid and the types, principles and implementation of interpolation (internal on the model grid, from an external grid to the model grid, from the model to external data locations as used in the nesting procedures). Chapter 11 is devoted to the parallelisation of the code (general principles, domain decomposition, array halos, implementation of process communication with the MPI library, index mapping). Chapter 12 provides a listing of all source code files with their purpose and illustrates the general structure of the code with flow diagrams.

IV. User manual.
This part is undoubtly the most important one of the documentation. The meaning and contents of each usrdef_ routine are discussed in the following chapters:

- Chapter 13: general priciples of model setup, listing of all usrdef_ routines and files
- Chapter 14: switches, model parameters, forcing attrributes, setup of monitoring and the parallel mode
- Chapter 15: model grid, bathymetry, initial conditions
- Chapter 16: open boundary conditions
- Chapter 18: structures and discharges
- Chapter 17: surface boundary conditions and nesting
- Chapter 20: different kinds of user output
- Chapter 19: sediment model

V. Test cases.
The test cases experiments which are provided with the distribution of the source code, are presented. The aim is to test the portability of the code, to illustrate how model setup is performed and to show different types of applications.The results are briefly discussed and illustrated with figures and tables which may be used by the user for comparison.

- Chapter 22: purpose of the test cases, CPU time table obtained on different computing platforms
- Chapter 23: numerical tests for advection schemes

- Chapter 24: turbulence schemes and heat flux formulations
- Chapter 25: density fronts in a channel, river plumes
- Chapter 27: shelf sea applications
- Chapter 28: hydraulic structures and discharges
- Chapter 29: tests for the sediment model

VI. Reference manual.
This part provides a listing of all program routines with their syntax and purpose and of all program variables with a global scope in the program.

- Chapter 30: external routines
- Chapter 31: module routines which are mainly used by the program as internal libraries
- Chapter 32: usrdef_ routines
- Chapter 33: program variables defined in the FORTRAN 90 modules
- Chapter 34: sediment routines (external, module, usrdef_) and sediment program variables

### 1.4.2 Suggestions for reading

The documentation is written, as much as possible, in a self-contained form. Appropriate links are made, where necessary, to chapters and sections where a specific topic is explained in more detail.

Beginning users whose prime intentions are to perform simple simulations with the model and have less interest in learning all the features of COHERENS, are not obliged to read this voluminous documentation as a whole, but may proceed as follows:

1. Read Part I.

2. Select a few test cases from Part V to test the portability of the model. Have a look at the source code files used to set up the selected test cases.

3. Read those sections of Part IV which are of interest for the intended application.

4. Create the source files for the application using the setup of an analogous test case. For more information about the model setup or the name of certain variables for model output consult the reference manual.

On the other hand, if the user wants to learn about the full model's capacities or has the intention to extend the code with a new implementation it is recommended to read this manual in more detail, especially Part III and Chapter 5 about the discretisation methods used in the program.

## Conventions used in the document

The text is typesetted in LaTeX2e. The following numbering conventions are adopted:

- Chapters and pages are numbered continuously over all Parts.
- Sections are numbered in a two-number format (Chapter-Section).
- Subsections are numbered in a three-number format (Chapter-Section-Subsection).
- Equations, figures and tables are numbered in a two-number format (Chapter-Equation/Figure/Table).

Specific names and keywords are outlined in different LaTeX fonts:

- sans serif style for the FORTRAN names of program variables and specific keywords
- **bold** for the names of directories
- *slanted* style for the names of files
- ***emphasised bold*** for the names of test cases
- `type-writer` font for sections of model code and UNIX command names