final

Prepared for:

DG Rijkswaterstaat

RIKZ

On-line Sediment Transport within Delft3D-FLOW

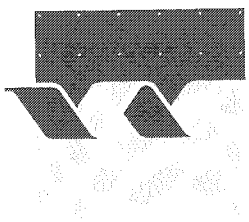
October 2000

Z2899

WL | delft hydraulics

On-line Sediment Transport within Delft3D-FLOW

Giles Leser
Jan van Kester
J.A. (Dano) Roelvink



WL | delft hydraulics

WL | delft hydraulics

CLIENT: DG Rijkswaterstaat; Rijks-Instituut voor Kust en Zee / RIKZ

TITLE: On-line Sediment transport within Delft3D-FLOW

ABSTRACT:

Overview of Report
This report describes the changes made to the standard DELFT3D-FLOW module in order that on-line, three-dimensional morphological simulations may be carried out, the simulations may include the three dimensional effect of surface waves.

This report is in two parts. Part I describes the changes made to allow the on-line computation of three-dimensional suspended sediment transport and associated morphological changes. This part of the report has been previously published (jointly by WL | DELFT HYDRAULICS| and IHE Delft) as the MSc thesis of Giles Lesser (May 2000).

Part II of the report describes changes implemented subsequent to the publication of Giles Lesser's MSc thesis. These changes include the addition of a bed-load sediment transport formulation, inclusion of the modified FLOW module into the MORSYS morphological system, and a number of further improvements. The activities described in this part were carried out as part of the present study under number RKZ-837.

Each of the two parts describes, in some detail, the changes made to the flow module in order to implement the desired features, also, they each contain a section describing the validation tests that have been carried out to test the modified computer code.

As this report has been compiled in two pieces, we would like to point out to readers that the appendices for Part I of this report are located at the end of Part I; not at the end of the report. We also point out that Part I appendix A 'Changes to the computer code' refers only to the changes made in order to implement the features described in Part I (code version 007). A similar appendix describing the changes made for Part II (version 009) has yet to be compiled.

REFERENCES: Contract RKZ-837

VER.	ORIGINATOR	DATE	REMARKS	REVIEW	APPROVED BY
1	J.A. Roelvink	30 August 2000		A. Roelfzema	
2	J.A. Roelvink	30 October 2000		A. Roelfzema 30/10	T. Schilperoort 30/10

PROJECT IDENTIFICATION:		Z2899						
KEYWORDS:		3D-Sand Transport 3D-Morphology						
CONTENTS:	TEXT PAGES	100	TABLES		FIGURES	49	APPENDICES	4
STATUS:	☐ PRELIMINARY		☐ DRAFT		☒ FINAL			

On-line Sediment Transport within DELFT3D-FLOW

Giles Lesser
Jan van Kester
J.A. (Dano) Roelvink

August, 2000

WL | delft hydraulics

Overview of Report

This report describes the changes made to the standard DELFT3D-FLOW module in order that on-line, three-dimensional morphological simulations may be carried out, the simulations may include the three dimensional effect of surface waves.

This report is in two parts. Part I describes the changes made to allow the on-line computation of three-dimensional suspended sediment transport and associated morphological changes. This part of the report has been previously published (jointly by WL | DELFT HYDRAULICS| and IHE Delft) as the MSc thesis of Giles Lesser (May 2000).

Part II of the report describes changes implemented subsequent to the publication of Giles Lesser's MSc thesis. These changes include the addition of a bed-load sediment transport formulation, inclusion of the modified FLOW module into the MORSYS morphological system, and a number of further improvements. This work has been funded by RIKZ, in the framework of the cooperation between RIKZ and WL | DELFT HYDRAULICS, under project number RKZ-837, within Project K2005*Zeebodem.

In the cycle of testing-evaluation-development, this work comprises the development (of a 3D morphodynamic model) and the testing of it. Although more testing is certainly desirable, the model in its present status is ready to also be evaluated in the field.

Each of the two parts describes, in some detail, the changes made to the flow module in order to implement the desired features, also, they each contain a section describing the validation tests that have been carried out to test the modified computer code.

As this report has been compiled in two pieces, we would like to point out to readers that the appendices for Part I of this report are located at the end of Part I; not at the end of the report. We also point out that Part I appendix A 'Changes to the computer code' refers only to the changes made in order to implement the features described in Part I (code version 007). A similar appendix describing the changes made for Part II (version 009) has yet to be compiled.

Part I

Computation of Three-dimensional Suspended Sediment Transport within the DELFT3D-FLOW Module

Giles Lesser

May 2000

WL | Delft Hydraulics

Report Z2396

IHE Delft

Thesis HE 066

Foreword

This study has been performed by Giles Lesser, under the supervision of Dr ir J.A Roelvink of WL | DELFT HYDRAULICS, as part of the requirements for the attainment of a Master of Science degree from IHE Delft.

This thesis was presented at 1pm on Wednesday 17 May, 2000 at the premises of IHE Delft. The committee responsible for assessing the content of this thesis consisted of:

1. Prof. dr. ir. L.C. van Rijn[1]
2. Dr. ir. J.A. Roelvink[2]
3. Ir. H.J. Verhagen[3]

This document is jointly published as:

IHE thesis number HE 066
WL | DELFT HYDRAULICS report number Z2396

[1] *University of Utrecht and WL | DELFT HYDRAULICS.*
[2] *WL | DELFT HYDRAULICS.*
[3] *International Institute for Infrastructural, Hydraulic and Environmental Engineering (IHE Delft).*

Abstract

Computer modelling of sediment transport patterns is generally recognised as a valuable tool for understanding and predicting morphological developments. In practice, state-of-the-art computer models are two-dimensional (depth averaged) and have a limited ability to model many of the important three-dimensional flow phenomena found in nature. This study reports the implementation and validation of suspended sediment transport formulations within a proven three-dimensional (hydrostatic) flow solver. The implementation considers both cohesive (mud) and non-cohesive (sand) sediment, in situations with and without waves. The bulk of the validation is concerned with non-cohesive sediment in current-only situations, although a limited wave and current validation is also included. The study concludes that the implementation of the non-cohesive sediment transport model performs well in current-only situations, and shows promise in situations with waves. Recommendations include adding a bed-load transport formulation and further testing, prior to a public release.

Acknowledgements

I would like to acknowledge the financial and logistical support of WL | DELFT HYDRAULICS for the duration of my MSc thesis investigation, without which I would have been unable to carry out this study. In addition, I would like to take this opportunity to thank a number of people, without whom the process of carrying out this study wouldn't have been half the pleasure it has turned out to be.

In an order roughly the reverse of that in which I was introduced to them, I would like to thank the following people:

Rob Uittenbogaard, for his help and genuine interest in my struggles with DELFT3D-FLOW's k-epsilon turbulence model.

Jan van Kester, for the many hours of good humoured and highly skilled assistance he provided while I slowly came to grips with DELFT3D-FLOW's numerical scheme.

Heleen Leepel, for the amount of her valuable time that I took up with endless questions about the structure of DELFT3D, and about FORTRAN programming in general.

Dano Roelvink, for his guidance, endless patience, and for continually showing a level of enthusiasm for what I was doing that made even the most mundane and frustrating debugging exercise feel like worthwhile and pioneering research.

Lauren Morgan, for effectively arranging her life around mine for the last 18 months.

and first, but certainly not least,

My parents, for first introducing me to the joys of playing with sand and water, and for then instilling in me a never-ending desire to understand "why?".

List of Symbols

Symbol	Description	Unit
a	Reference height as defined by van Rijn	m
$\hat{A}_\delta$	Peak orbital excursion at the bed	m
c	Volumetric concentration	(-)
c_a	Volumetric concentration at reference height a	(-)
d_{50}	Median diameter of sediment	m
d_s	Representative diameter of suspended sediment	m
g	Acceleration due to gravity	m/s^2
h	Local water depth	m
k_a	Increased (apparent) bed roughness height felt by the current in the presence of waves	m
$k_{s,c}$	Current related bed roughness height	m
$k_{s,w}$	Wave related bed roughness height	m
(ℓ)	Index number of sediment fraction	(-)
s	Relative density of sediment fraction. $(=\rho_s/\rho_w)$	(-)
S	Salinity (ppt)	(-)
$S_{\max}$	Maximum salinity for flocculation calculations	(-)
T_a	Dimensionless bed-shear stress for reference concentration	(-)
T_p	Peak wave period	s
u	Fluid velocity in the x direction	m/s
$\hat{U}_\delta$	Peak orbital velocity at the bed	m/s
u_*	Bed shear velocity	m/s
v	Fluid velocity in the y direction	m/s
w	Fluid velocity in the z direction	m/s
$w_{s,0}$	Particle settling velocity in clear water (non-hindered)	m/s
w_s	Particle (hindered) settling velocity in a mixture	m/s
$W_{s,f}$	User specified particle settling velocity in fresh water.	m/s
$W_{s,max}$	User specified floc settling velocity in water with salinity = $S_{\max}$	m/s
x	Horizontal co-ordinate	m
y	Horizontal co-ordinate	m
z	Vertical co-ordinate	m
β	Ratio of sediment and fluid mixing	(-)
δ_m	Thickness of wave boundary mixing layer	m
Δ_r	Ripple height	m
δ_w	Thickness of wave boundary layer	m
ε_f	Fluid diffusion in the z direction $(=\varepsilon_{f,z})$	m^2/s
$\varepsilon_{f,x}$, $\varepsilon_{f,y}$, $\varepsilon_{f,z}$	Fluid diffusion coefficients in the x, y, z directions respectively.	m^2/s
ε_s	Sediment diffusion in the z direction $(=\varepsilon_{s,z})$	m^2/s

$\varepsilon_{s,x}$, $\varepsilon_{s,y}$, $\varepsilon_{s,z}$	Sediment diffusion coefficients in the x, y, z directions respectively.	m^2/s
υ	Kinematic viscosity coefficient	m^2/s
ρ_{mix}	Density of the sediment-water fluid mixture	kg/m^3
ρ_s	Density of the solid sediment particles	kg/m^3
ρ_w	Density of water (adjusted for temperature and salinity)	kg/m^3
$\tau_{b,c}$	Bed shear stress due to current	N/m^2
$\tau_{b,cw}$	Bed shear stress due to current in the presence of waves	N/m^2
$\tau_{b,w}$	Bed shear stress due to waves	N/m^2
τ_{cr}	Critical bed shear stress	N/m^2
$\tau_{cr,d}$	User specified critical deposition shear stress	N/m^2
$\tau_{cr,e}$	User specified critical erosion shear stress.	N/m^2
τ_{cw}	Mean bed shear stress due to current and waves	N/m^2
τ_{mean}	Mean (cycle averaged) bottom shear stress with wave-current interaction	N/m^2
τ_{max}	Maximum bottom shear stress with wave-current interaction	N/m^2

Contents

Appendices

List of Figures

I Summary

Computer modelling of hydrodynamic flows and sediment transport is widely accepted as a valuable tool for understanding and predicting morphological developments. In practical applications suspended sediment transport calculations are generally carried out in one or, at most, two dimensions; generally without taking into account the density effects of the sediment on the hydrodynamic flow calculations. The validity of applying such computer models to strongly three-dimensional situations where density gradients or the hydrodynamic flow pattern have significant three-dimensional properties is questionable. Furthermore, in an attempt to model these three-dimensional situations, a wide range of extending expressions are frequently added to the basic one- and two-dimensional computer models. Unfortunately the cumulative effect of several such additions can easily result in models that contain such a complicated array of internal dependencies and limitations that validation and calibration are difficult to achieve. This study follows a different approach to this problem by building sediment transport formulations directly into an existing three-dimensional flow solver. The advantage of this approach is that most three-dimensional phenomena can be automatically resolved by the flow solver itself, therefore requiring very few additional expressions and a minimum of user calibration.

This paper reports the implementation and validation of suspended sediment transport within the proven DELFT3D-FLOW hydrodynamic module. The implementation of formulations to describe the transport and bed interaction of cohesive and non-cohesive sediments under the action of waves and/or current is reported in detail, as are the modifications made to update the bathymetry used by the flow calculations at each computational time-step. Also presented are the results of a validation study of the non-cohesive sediment implementation, in which the results of computer simulations are compared with the results of a number of laboratory experiments available in the literature. A brief discussion of the initial results of applying the modified DELFT3D-FLOW module to real-life situations is also included.

The paper concludes that the implementation of suspended non-cohesive sediment transport in the DELFT3D-FLOW module has been successfully achieved and that the modified FLOW module is capable of accurately predicting the results of the laboratory experiments investigated. It also concludes that the modified flow module appears to be capable of carrying out real-life simulations over useful morphological time frames, and that the preliminary results of such simulations show promise; although this cannot be fully confirmed as the modified FLOW module still lacks a description of the bed-load transport which occurs in real-life situations. The paper recommends that the modified FLOW module form the basis of a future public release of a version of DELFT3D capable of carrying out three-dimensional morphological computations. To this end, the paper recommends the implementation of bed-load transport, a number of additional tests, and further validation against a number of real-life test cases.

2 Introduction

2.1 Calculation of Sediment Transport

2.1.1 Why calculate sediment transport ?

Sediment transport is one of those deceptively simple natural processes. Water moves and carries along sediment.

So why should we be interested in sediment ? Sediment is land. Changes in coastal land-form, or "coastal geomorphology", are caused by variations in the rate of sediment transport. If more sediment comes in than goes out then land is created. If more sediment leaves than arrives, then land leaves with it. This is frequently the crux of our problem. From this gross over-simplification of the study of geomorphology we begin to see the need to predict the rates and patterns of sediment transport. It is to improving the quality of these predictions that this study is dedicated.

2.1.2 Point estimates of sediment transport

Researchers have worked for many years attempting to predict the rate of sediment transport under well-known, controlled, and stable conditions. Even this is far from simple. Ancient (trial and error) research in this area dates back to the solution of sediment transport problems in the irrigation channels of the ancient civilisations of China, Mesopotamia, Egypt, and the Roman Empire. Of more immediate importance to this study is the work of researchers since the beginning of the 20th century. Shields' (1936) work on the initiation of motion of sediment particles is built, almost directly, into the sediment transport formulations used in this study. The work of Shields and several other notable researchers in the field is well summarised by van Rijn (1993).

For the purpose of analysis, the sediment transported by a flow is frequently divided into two components; bed-load transport, and suspended-load transport. Although the split between bed-load and suspended-load is artificially introduced, it is generally accepted that bed-load consists of the particles that are sliding, rolling, and jumping along the bed. Suspended sediment transport consists of those particles that are more or less continuously suspended by the turbulent motion of the water. Obviously, the total sediment transport rate is the sum of the bed-load transport and the suspended-load transport.

Although many alternative formulations exist, this study is primarily concerned with implementing the predictive formulations of van Rijn (1984). In a series of three papers van Rijn set out a method for predicting bed-load transport, suspended-load transport, and bed forms and alluvial roughness. Since 1984 these, semi-empirical, formulas have become widely accepted as the most accurate sediment transport predictors available. This study

focuses on the implementation of a number of van Rijn's formulations relating to suspended sediment transport into an existing three-dimensional hydrodynamic flow modelling system.

2.1.3 Modelling spatially varying sediment transport

As noted above, we are frequently not as interested in the actual quantity of sediment being carried by a flow of water as in any changes that may occur in the quantity of sediment being carried. For example, if water in a channel, initially at rest, is accelerated then the quantity of sediment carried by the water will increase with time (causing erosion of the bed). Such a situation occurs twice in every tidal cycle in coastal environments. Similarly a steady flow of water may be locally accelerated around a groyne in a river or at the coast, this acceleration will also cause a local increase in sediment transport and erosion near the tip of the groyne - a scour hole. In order to predict these types of erosion (and deposition) we must accurately represent the changes in sediment transport in space and with time.

The simplest sediment transport calculations involve analysing the changes of sediment transport in one spatial dimension. An example of this type of model is WL | DELFT HYDRAULICS' Unibest CL. This model evaluates the rate of long-shore sediment transport at intervals along a coastline, and then predicts erosion or accretion between these points based on the difference between adjacent sediment transport rates. One-dimensional models also exist where the single spatial dimension is in a vertical direction (so-called 1DV models). These "point" models are usually used as a research tool to investigate the time dependent change in a sediment concentration profile to a change in local flow conditions, or to investigate the effects of vertical density gradients (stratification) on the sediment concentration profile.

The next level of modelling sophistication is provided by two-dimensional computer models. These models work in a similar manner to the one-dimensional Unibest CL model described above. Effectively, they calculate the sediment transport magnitude and direction at a large number of points in space, and calculate the erosion or deposition by looking at the difference between adjacent points. DELFT3D-MOR (a module of the DELFT3D package) is an example of such a model (Figure 1). Two-dimensional models have a much wider applicability than one-dimensional models as they are capable of modelling spatial changes in bathymetry and wave climate much more realistically. Two-dimensional morphodynamic computer models have been practically applied since the early 1990s and are still regarded,

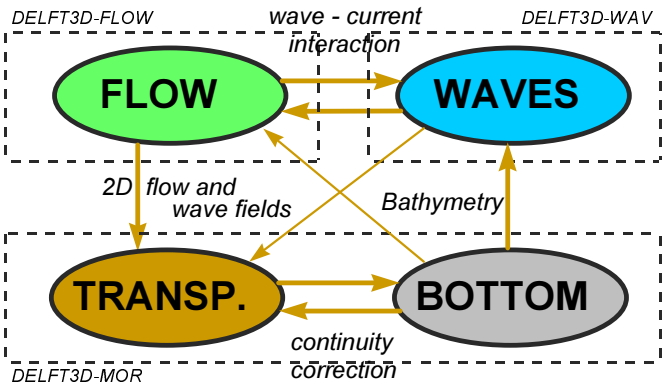


Figure 1 - Conventional morphological computation

by most practitioners, as the state-of-the-art in morphodynamic modelling.

While the boundaries of two-dimensional morphodynamic modelling are constantly being pushed back (predominantly by adding to the refinement of the hydrodynamic wave and flow models incorporated within the models) they will always suffer from the fact that they rely on depth-averaged estimates of the sediment transport rates. These depth-averaged estimates are, in turn, based on assumptions regarding the vertical velocity profile and turbulent mixing profile. The simplest of the two-dimensional models assume a logarithmic velocity profile and a parabolic, or parabolic-constant turbulent mixing profile. More advanced models, so-called "quasi three-dimensional" models, contain additional parameters in order to attempt to adapt these general velocity and turbulent mixing profiles in situations where they are no longer valid, for example in highly stratified flows, or in areas with significant wave action or with rapidly accelerating or decelerating flows.

The next logical step is the development of fully three-dimensional morphological models. These models calculate the velocity of water flows, along with the transport of sediment in all three spatial dimensions. This means that relatively few parameterisations are required to describe flow phenomena that occur in three dimensions - spiral flow in river bends for example - and thus the model actually becomes simpler, as well as being physically more accurate. The main disadvantage of three-dimensional modelling is the relatively large amount of computational effort required to solve the much larger systems of equations. While computing power is obviously a consideration that must be kept in mind when contemplating a large computer model, the ongoing rapid increases in computing power readily available to the professional user mean that desk-top personal computers are now capable of running moderate size three-dimensional simulations in an acceptable time. This is a situation that can only be expected to improve over the coming months and years.

This study focuses on the development and validation of just such a three-dimensional morphodynamic computer model; it is developed by incorporating sediment transport formulations into the heart of the already advanced DELFT3D hydrodynamic flow modelling system.

2.2 The DELFT3D Software Package

DELFT3D is a computer modelling framework for two- and three-dimensional flow and transport in coastal, river, and estuarine environments. The DELFT3D package consists of a number of integrated modules which, together, allow the computation of water flow, water quality, sediment transport, ecological and chemical parameters, short wave generation and propagation, wave-current interaction, and morphological simulation. Models may be formulated on a rectilinear, curvilinear, or spherical grid.

DELFT3D is a commercial software package developed and supported by WL | DELFT HYDRAULICS. It is used both in-house by WL | DELFT HYDRAULICS staff, and by an ever-increasing number of clients around the world. As part of their commitment to the continuous development of their computer models, WL | DELFT HYDRAULICS supports a standard public release version of DELFT3D as well as a number of research versions which are used, at a number of institutes, for developing and testing new model formulations. This

study focuses on one such research version (DELFT3D-FLOW v03.05.007) which is based on the general DELFT3D-FLOW version (03.05) which was the current general release as of April 1999.

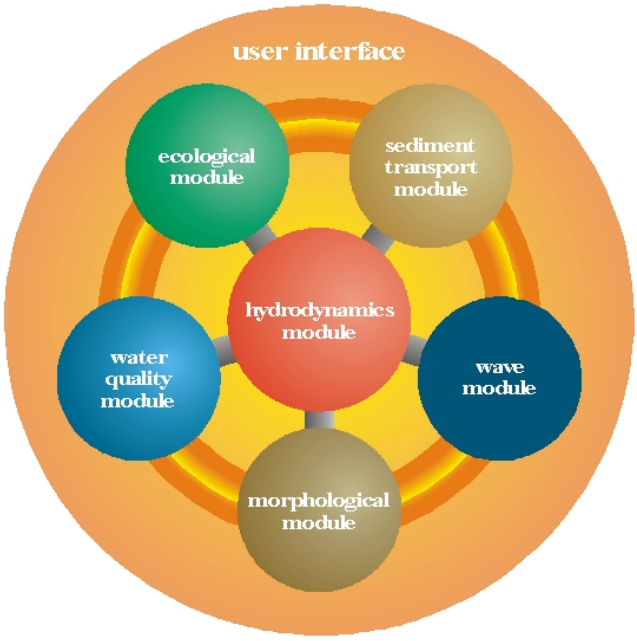


Figure 2 - The DELFT3D software package

The reader should be aware that at the time of writing version 03.05.007 is regarded as a non-commercial research version that is not formally supported by WL | DELFT HYDRAULICS.

2.3 The Standard FLOW Module

2.3.1 Applicability

The main purpose of the standard hydrodynamic module DELFT3D-FLOW (v 03.05) is to simulate non-steady two-dimensional (2D, depth averaged) and three-dimensional (3D, hydrostatic) tidal and wind-driven flows. The calculations can include transport phenomena and the effects of density differences due to non-uniform temperature and/or salinity distributions (density driven flows). DELFT3D-FLOW aims to model flow phenomena in which the horizontal length scales are significantly larger than the vertical; examples of these include shallow seas, coastal areas, estuaries, rivers, and lakes.

DELFT3D-FLOW solves the unsteady shallow-water equations in two or three dimensions. The system of equations consists of the horizontal momentum equations, the continuity equation, and transport equations. The equations are formulated in either a Cartesian frame of reference or spherical co-ordinates on the globe. The transport of salt, heat, and conservative constituents can be modelled by a conservative transport equation including source and sink terms to model discharges and withdrawals.

DELFT3D-FLOW includes formulations that take into account:

- Tidal forcing
- Free surface gradients (barotropic effects)
- The effect of the Earth's rotation (Coriolis force)
- Tidal potential (tide generating forces)
- Water with variable density (equation of state)
- Horizontal density gradients in pressure (baroclinic effects)
- Turbulence induced mass and momentum fluxes (turbulence closure models)
- Transport of salt, heat, and other conservative elements
- Space and time varying wind shear stress on the water surface
- Bed shear stress on the base of the flow
- Space and time varying atmospheric pressure on the water surface
- Time varying sources and sinks (e.g. river discharges)
- Drying and flooding of tidal flats
- Lateral shear stress due to rough walls
- Heat exchange through the free surface
- Effects of secondary flow (spiral motion intensity) on depth averaged momentum equations (Only necessary when modelling in 2D)
- Influence of surface waves on the bed shear stress
- Wave induced stresses and mass fluxes (2D only)

In order to solve the 2D or 3D shallow water equations a number of simplifying assumptions are made, the most important of which are summarised below:

1. The shallow water assumption: The vertical momentum equation is reduced to the hydrostatic pressure relation. This implies that vertical accelerations are assumed to be small compared with the acceleration due to gravity, and may be neglected.
2. The fluid (water) is assumed to be incompressible
3. The immediate effect of density differences (buoyancy) on vertical acceleration is not considered. However density differences are taken into account in the horizontal pressure gradients and in the vertical exchange coefficients (turbulence models). This limits the application of DELFT3D-FLOW to mid- and far-field dispersion simulations.
4. In 3D computations the effect of 3D turbulence on the vertical exchange of momentum and mass is modelled through a vertical eddy viscosity and eddy diffusivity coefficient (eddy viscosity concept) by means of an algebraic, k-L or k-ε turbulence model.

A complete list of assumptions is contained in the DELFT3D-FLOW User Manual.

2.3.2 Computational procedure

The computational procedure of the DELFT3D-FLOW module is illustrated diagrammatically in Figure 3. This procedure is repeated every computational time-step. It should be noted that if a 3D simulation is specified (i.e. the number of computational layers used is greater

than one) then the computed turbulent mixing, velocity, and constituent concentration fields are also three-dimensional.

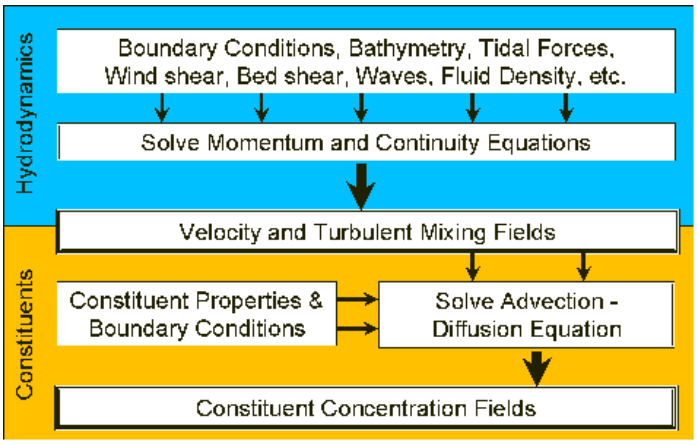


Figure 3 - Operation of the FLOW module

2.4 Development of the "Sediment Version" of DELFT3D-FLOW

2.4.1 Purpose of the sediment version

The implementation of sediment transport in the hydrodynamic (FLOW) module of DELFT3D is intended to be useful in situations where large sediment concentrations affect the density distribution to such an extent that significant hydrodynamic forces exist and/or a significant interaction with turbulence occurs. Such large concentrations, typically exceeding several 100 mg/l, are observed in the Rhine-Meuse estuary.

A further advantage of the sediment version of DELFT3D-FLOW is that it computes sediment transport in a fully three-dimensional manner (with the same limitations as the hydrodynamic calculations), without resorting to the use of shape functions for concentration or flow velocity profiles. As such, it accurately computes the suspended sediment transport in situations where significant 3D flow effects exist. For example the effects of density currents, stratification, and spiral flow are automatically taken into account without requiring any special parametric formulations.

In addition, the formulations used do not assume equilibrium sediment concentration profiles, and thus the variation of sediment transport in regions of accelerating and decelerating flow is also accurately accounted for.

2.4.2 Work by earlier developers

As reported by Vatvani (June 1999), the initial stages of including sediment transport in the DELFT3D-FLOW module were based on experience gained by Uittenbogaard et al. (1996) using a one-dimensional, vertical co-ordinate only, (1DV) flow and transport computer model. Although the results obtained from this model indicated that 50 to 100 equal

thickness layers were required to achieve a convergent solution, the formulations used by Uittenbogaard et al. were copied into the first sediment version of DELFT3D-FLOW (v 03.05.002).

This (002) version also included formulations describing the erosion of bed material into the flow, and the deposition of suspended material onto the bed. For cohesive sediment the well-known Partheniades-Krone formulations (Partheniades 1965) were used, and for non-cohesive sediment (i.e. sand) van Rijn's pickup function (van Rijn 1984) was applied.

Vatvani concluded his development of the computer code by confirming that the code produced the same results as the standard code in a test case without sediment, and by qualitatively testing the settling behaviour of four different types of mud (cohesive sediment) in a simple closed basin.

Van Kessel (August 1999) carried out a number of tests on the performance of the non-cohesive (sand) formulations in the sediment version of DELFT3D-FLOW. These tests were carried out by modelling a long (8km) straight flume with a movable sand bed. The roughness and slope of the flume were such that a constant flow depth and depth-averaged velocity were achieved along the length of the flume.

Van Kessel reported that version 002 was able to accurately reproduce the expected sediment concentration profiles under equilibrium conditions. However he also noted that the use of an equilibrium flux bottom boundary condition resulted in an adaptation distance required to reach an equilibrium sediment concentration profile that was significantly longer than expected from analytical calculations. Furthermore, this adaptation length appeared to be dependent on the chosen vertical grid spacing. To counter this he changed the bottom boundary condition to a concentration boundary condition (v 03.05.005), and noted that the distance for the flow to reach an equilibrium sediment concentration profile was much improved. He noted, however, that the equilibrium sediment concentration profile obtained was still dependent on the vertical grid spacing chosen and that this needed to be remedied.

The current study commenced by following the recommendations of van Kessel and further developed the concentration boundary condition model (v 03.05.006) before a decision was made to revert to the flux bottom boundary condition (v 03.05.007) once a number of improvements had been made to the earlier formulations. It is our opinion that the flux bottom boundary condition offers a number of significant advantages over the concentration boundary condition, as discussed in section 3.3 of this paper.

3 Modifications to the Standard FLOW Module

3.1 General

Three-dimensional transport of sediment is calculated by solving the three-dimensional advection-diffusion (mass-balance) equation for the suspended sediment

$$\frac{\partial c}{\partial t}+\frac{\partial}{\partial x}(uc)+\frac{\partial}{\partial y}(vc)+\frac{\partial}{\partial z}\left[(w-w_s)c\right]-\frac{\partial}{\partial x}\left[\varepsilon_{s,x}\frac{\partial c}{\partial x}\right]-\frac{\partial}{\partial y}\left[\varepsilon_{s,y}\frac{\partial c}{\partial y}\right]-\frac{\partial}{\partial z}\left[\varepsilon_{s,z}\frac{\partial c}{\partial z}\right]=0$$

This equation is solved taking into account the local sediment concentration (c), the local flow velocities (u, v, w), the local turbulent sediment mixing ($\varepsilon_{s,x}$, $\varepsilon_{s,y}$, $\varepsilon_{s,z}$), and the local (hindered) particle fall velocity (w_s). The local flow velocities and turbulent mixing components are based on the results of the main hydrodynamic computations in the FLOW module.

Computationally, the DELFT3D-FLOW module handles the three-dimensional transport of sediment in exactly the same way as it calculates the transportation of any other conservative constituent. Examples of other such constituents are salt, heat, and user-specified contaminants.

There are, however, a number of important differences between sediment and other constituents. For example, the exchange of sediment between the bed and the flow, and the fall velocity of sediment under the action of gravity are obviously of critical importance. Other processes such as the effect that sediment has on the local mixture density, and hence on turbulence damping, must also be taken into account. In addition, if a net flux of sediment from the bed to the flow, or vice versa, occurs then the resulting change in the bathymetry should influence subsequent hydrodynamic calculations.

The following paragraphs describe, at a conceptual level, the changes made to the standard flow module to account for the differences between the transport of sediment and the transport of other conservative constituents. These modifications come into operation whenever the user selects a constituent of type “Sediment”. For a detailed description of the changes made to the actual computer code the reader is referred to Appendix A

A number of modifications have also been incorporated in the FLOW module to account for the three-dimensional effects of waves on the computed velocity and turbulent mixing values. These effects are important when computing the transport of sediment in wave and current situations. The modifications required to model these effects have been made by another researcher and are fully described elsewhere, however a brief overview of the changes made is located in section 3.12 of this report.

3.2 Classification of Sediment Fractions

The sediment version (v03.05.007) of the DELFT3D-FLOW module requires the user to classify sediment fractions as one of either “sand”, “floc”, or “mud”; as different formulations are used to calculate the fall velocity and erosion or sedimentation of each of these three types of sediment.

Multiple fractions are also allowed, with up to five sediment fractions allowed in a simulation. This, in principle, means that calculations may be performed with, for example, more than one “sand” fraction, or a number of “sand” and “mud” sediment fractions present at one location. The simultaneous presence of multiple sediment fractions has implications for the calculation of the local hindered fall velocity of any one sediment fraction as well as for the resulting mixture density.

The presence of multiple sediment fractions also considerably complicates the calculation of the density of the bed, and the availability of a particular class of sediment at the bed. These last two effects are not yet accurately implemented in the sediment version of DELFT3D-FLOW.

3.3 Sediment Bed Boundary Condition.

The exchange of material between suspension and the bed is a phenomenon unique to modelling of sediment transport (when compared with modelling the transport of other conservative constituents). In version 007 this exchange is modelled by calculating the sediment fluxes from the bottom computational layer to the bed, and vice versa. These fluxes are then applied to the bottom computational layer by means of a sediment source and/or sink term in each computational cell. The calculated fluxes are also applied to the bed in order to update the bed level. Refer to section 3.11 for more detail of this process. The advection and diffusion of sediment particles through the bottom of the bottom computational layer is prevented in order to avoid double counting these sediment fluxes.

The reader should be aware that, as an alternative, a concentration may be set as the bottom boundary condition. This approach is partially implemented in research version 03.05.006 of the DELFT3D-FLOW module. The pros and cons of this approach are briefly discussed in section 3.3.3 below.

The details of the formulations used to calculate the deposition and erosion fluxes for both cohesive and non-cohesive sediments follow.

3.3.1 Cohesive sediment

For cohesive sediment fraction (ℓ), the fluxes between the bed and the bottom computational cell are calculated with the well-known Partheniades-Krone formulations (Refer to Partheniades (1965) for a full description of these formulations):

$$E^{(\ell)} = M^{(\ell)} \quad S\left(\frac{\tau_{cw}}{\tau_{cr,e}^{(\ell)}} - 1\right)$$
$$D^{(\ell)} = w_s^{(\ell)} c_b^{(\ell)} S\left(1 - \frac{\tau_{cw}}{\tau_{cr,d}^{(\ell)}}\right) \qquad (1)$$
$$c_b^{(\ell)} = c^{(\ell)}\left(z = \frac{\Delta z_b}{2}, t\right)$$

Where:

$E^{(\ell)}$	=	Erosion flux (Kg/m^2/s)
$M^{(\ell)}$	=	User specified erosion parameter (EROPAR[L])
$S\left(\frac{\tau_{cw}}{\tau_{cr,e}^{(\ell)}} - 1\right)$	=	Erosion step function.
	=	$\left(\frac{\tau_{cw}}{\tau_{cr,e}^{(\ell)}} - 1\right)$ if $\tau_{cw} > \tau_{cr,e}^{(\ell)}$
	=	0 if $\tau_{cw} \le \tau_{cr,e}^{(\ell)}$
$D^{(\ell)}$	=	Deposition flux (Kg/m^2/s)
$w_s^{(\ell)}$	=	Fall velocity (hindered)
$c_b^{(\ell)}$	=	Average sediment concentration in bottom computational layer.
$S\left(1 - \frac{\tau_{cw}}{\tau_{cr,d}^{(\ell)}}\right)$	=	Deposition step function.
	=	$\left(1 - \frac{\tau_{cw}}{\tau_{cr,d}^{(\ell)}}\right)$ if $\tau_{cw} < \tau_{cr,d}^{(\ell)}$
	=	0 if $\tau_{cw} \ge \tau_{cr,d}^{(\ell)}$
τ_{cw}	=	Mean bed shear stress due to current and waves as calculated by the wave-current interaction model selected by the user. Refer to the DELFT3D-FLOW user manual for further details.
$\tau_{cr,e}^{(\ell)}$	=	User specified critical erosion shear stress. (TCRERO[L])
$\tau_{cr,d}^{(\ell)}$	=	User specified critical deposition shear stress. (TCRSED[L])

Note: Superscript (ℓ) implies that this quantity applies to sediment fraction number (ℓ)

The calculated erosion or deposition flux is then applied to the bottom computational cell by setting the appropriate sink and source terms for that cell. Advection, particle settling, and diffusion through the bottom of the bottom computational cell are all set to zero to prevent double counting these fluxes.

3.3.2 Non-cohesive sediment

For non-cohesive sediment (e.g. sand), we follow the method of van Rijn (1993) for the combined effect of waves and currents (as used in the TRANSPOR 1993 model). In order to apply this method, the following calculations are made:

1. Calculation of van Rijn's reference height

$$a = \max\left[\left[\mathbf{AKSFAC}\right]k_s, \frac{\Delta_r}{2}, 0.01h\right] \tag{2}$$

Where

a	=	Van Rijn's reference height.
[**AKSFAC**]	=	User-specified proportionality factor specified in the **morph.inp** file.
k_s	=	User-specified current-related effective roughness height (space varying).
Δ_r	=	Wave-induced ripple height, refer equation (3).
h	=	Water depth

In addition, a is also limited to a maximum of 20% of the water depth. This precaution is only likely to come into effect in very shallow areas.

When waves are present, Δ_r (the ripple height) is estimated using the method of van Rijn (1993):

$$\begin{aligned} \Delta_r &= 0.22\,\hat{A}_\delta && \text{for } \Psi \le 10 \\ \Delta_r &= 2.8\times10^{-13}(250-\Psi)^5\,\hat{A}_\delta && \text{for } 10 < \Psi \le 250 \\ \Delta_r &= 0 && \text{for } \Psi > 250 \end{aligned} \tag{3}$$

In this expression:

$\Psi = \dfrac{(\hat{U}_\delta)^2}{(s-1)gd_{50(\mathrm{av})}}$	=	Mobility parameter
$\hat{A}_\delta = \dfrac{T_p\hat{U}_\delta}{2\pi}$	=	Peak orbital excursion at the bed
$\hat{U}_\delta$	=	Peak orbital velocity at the bed
	=	$\sqrt{2}\times$ RMS bed velocity from the wave module
T_p	=	Peak wave period from the wave module
$d_{50(\mathrm{av})}$	=	Average sediment size for all sediment fractions

This estimated ripple height is used not only for setting the reference height, but also for setting the wave-related roughness height. Refer to equation (15) below for the detail of this calculation.

We note, at this point, that this solution is not ideal as the above expression for estimating the ripple height is of questionable accuracy, and is only strictly valid for wave-only situations. We argue, however, that this method is the best available to us at this point in time, for the following reasons:

1. A DELFT3D computer model that includes waves requires that the wave-related roughness be specified, or estimated, at every grid point in the model (in two

dimensions). This wave related roughness is also likely to be a function of flow velocity, water depth, etc. and therefore, of time. We feel that it is highly unlikely that any user will seriously contemplate specifying a space-varying wave-related roughness, let alone one that also varies with time.

2. The generally accepted range of probable values for a wave-related roughness is 0.01m to 0.1m. We believe that the above expression will provide a more reasonable guess of the wave-related roughness within this range than users will, in the vast majority of cases. There is, however, some justification for allowing the user the option of specifying a uniform, constant, ripple height or wave-related roughness value. This might be appropriate in the case of simulating a laboratory experiment, for instance.
3. The user is still given the opportunity to tune the wave-related roughness by specifying the [**RWAVE**] parameter in the **morph.inp** file. Refer to equation (15) below for more detail.

Our limited experience of testing this expression in wave and current situations leads us to believe that it does provide a reasonable estimate of the ripple height, in the situations we have tested. We acknowledge that it is desirable that further testing of this expression is carried out, under a range of conditions. We also suggest that the expression be modified somewhat to take into account the effect of current on ripple formation. We believe that this should be possible to carry out in a logical (even if not scientifically justifiable) manner. Ensuring that wave-related ripples are smoothed out at high current velocities, would be one example of this.

2. *Identification of the reference layer*

The index of the lowest computational layer that is *entirely above* van Rijn's reference height is identified and stored. This is the lowest layer for which three-dimensional sediment transport calculations will be carried out for any "sand" fractions. From this point we will refer to this layer as *kmx*. This arrangement is indicated in Figure 4 below.

The sediment concentration in the layer(s) that lie below the *kmx* layer does not directly affect the three-dimensional sediment transport calculations occurring in the main body of the flow (as no advection or diffusion is allowed to occur through the bottom of the *kmx* layer) and changes in the concentration of these lower layer(s) are not considered when mass fluxes to and from the bed are calculated. However, the concentration in these layer(s) does have an indirect effect on the higher layers as the vertical density gradient in the vicinity of van Rijn's reference height has a significant impact on the intensity of turbulent mixing available to lift sediment from the reference height into the flow. For this reason the concentration of these lower layer(s) is set according to the following procedure:

1. If $\partial c/\partial z > 0$ in the vicinity of van Rijn's reference height, i.e. if (unrealistic) downward diffusion is expected, then the concentration of cells below *kmx* is set equal to the concentration of the cell *kmx*. In effect this sets $\partial c/\partial z = 0$ below van Rijn's reference height.

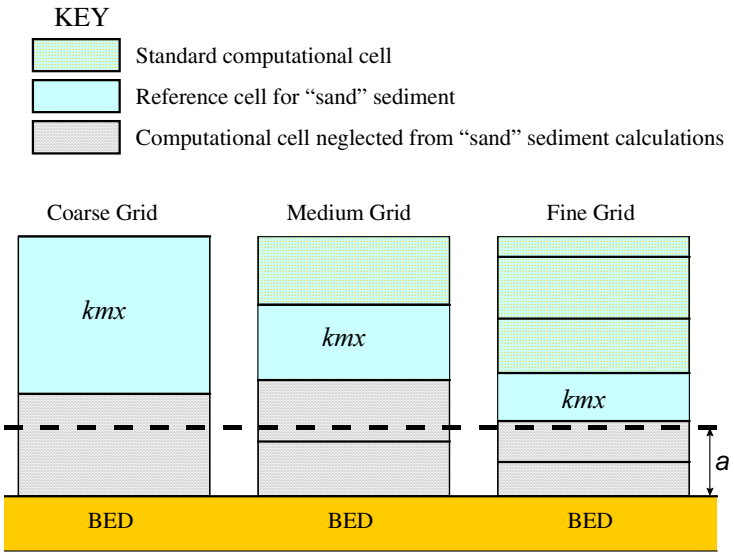


Figure 4 - Selection of the *kmx* layer

2. Otherwise, as long as sediment supply at the bed is not limited (by being less than the user-specified threshold above the fixed layer - refer to section 3.10 below) then the concentration of these layers is set equal to the equilibrium sediment concentration, calculated according to the method described in section 3.5. If the supply of sediment at the bed is limited by the proximity of a fixed layer, then a gradual transition from setting the equilibrium concentration to setting $\partial c/\partial z = 0$ layer occurs.

3. Calculation of the reference concentration

The reference concentration is calculated in accordance with van Rijn (1993), although an additional factor η is introduced to reflect the presence of multiple sediment fractions. The resulting expression is:

$$c_a^{(\ell)} = \eta^{(\ell)}\ 0.015\frac{d_{50}^{(\ell)}}{a}\frac{\left(T_a^{(\ell)}\right)^{1.5}}{\left[D_*^{(\ell)}\right]^{0.3}} \tag{4}$$

$c_a^{(\ell)}$ = Volumetric concentration at reference height a.

In order to evaluate this expression the following quantities must be calculated:

$\eta^{(\ell)}$ = Relative availability of sediment fraction

$$\eta^{(\ell)} = \frac{\text{mass of fraction } (\ell) \text{ in mixing layer}}{\text{total mass of sediment in mixing layer}} \tag{5}$$

$D_*^{(\ell)}$ = Dimensionless particle diameter

$$D_*^{(\ell)} = d_{50}^{(\ell)} \left[\frac{(s^{(\ell)} - 1)g}{\upsilon^2} \right]^{1/3} \tag{6}$$

$T_a^{(\ell)}$ = Dimensionless bed-shear stress

$$T_a^{(\ell)} = \frac{(\mu_c^{(\ell)} \tau_{b,cw} + \mu_w^{(\ell)} \tau_{b,w}) - \tau_{cr}^{(\ell)}}{\tau_{cr}^{(\ell)}} \tag{7}$$

$\mu_c^{(\ell)}$ = Efficiency factor current

$$\mu_c^{(\ell)} = \frac{f_c'^{(\ell)}}{f_c} \tag{8}$$

$f_c'^{(\ell)}$ = Grain related friction factor

$$f_c'^{(\ell)} = 0.24 \left[\log_{10} \left(\frac{12h}{3d_{90}^{(\ell)}} \right) \right]^{-2} \tag{9}$$

$f_c^{(\ell)}$ = Total current-related friction factor

$$f_c^{(\ell)} = 0.24 \left[\log_{10} \left(\frac{12h}{k_s} \right) \right]^{-2} \tag{10}$$

$\tau_{b,cw}$ = Bed shear stress due to current in the presence of waves. Note that the bed shear velocity u_* is calculated in such a way that van Rijn's wave-current interaction factor α_{cw} is not required. Refer to Appendix B for further details. If waves are not included then $\tau_{b,cw} = \tau_{b,c}$

$$\tau_{b,cw} = \rho_w u_*^2 \tag{11}$$

$\mu_w^{(\ell)}$ = Efficiency factor waves. $\mu_w^{(\ell)} = 0$ if waves are not included.

$$\mu_w^{(\ell)} = \frac{0.6}{D_*^{(\ell)}} \tag{12}$$

$\tau_{b,w}$ = Bed shear stress due to waves. $\tau_{b,w} = 0$ if waves are not included.

$$\tau_{b,w} = \frac{1}{4} \rho_w f_w \left(\hat{U}_\delta \right)^2 \tag{13}$$

f_w = Total wave-related friction factor.

$$f_w = \exp\left[-6 + 5.2\left(\frac{\hat{A}_\delta}{k_{s,w}}\right)^{-0.19}\right] \tag{14}$$

To avoid the need for excessive user input, the wave related roughness $k_{s,w}$ is related to the estimated ripple height, using the relationship

$$\begin{aligned} &k_{s,w} = [\text{RWAVE}]\Delta_r \\ &\text{with the limits:} \quad (0.01\text{m} \le k_{s,w} \le 0.1\text{m}) \end{aligned} \tag{15}$$

where: $[\text{RWAVE}]$ is the user-specified wave roughness adjustment factor. Recommended to be in range 1-3, default = 2. Refer to page 3–4 above for a discussion of the estimation of the ripple height and wave-related roughness.

$\tau_{cr}^{(\ell)}$ = Critical bed shear stress

$$\tau_{cr}^{(\ell)} = (\rho_s^{(\ell)} - \rho_w)\, g\, d_{50}^{(\ell)}\ \theta_{cr}^{(\ell)} \tag{16}$$

$\theta_{cr}^{(\ell)}$ = Threshold parameter $\theta_{cr}^{(\ell)}$ is calculated according to the classical Shields curve as modelled by Van Rijn (1993) as a function of the non-dimensional grain size D_*. This avoids the need for iteration. Note that, for clarity, in this expression the symbol D_* has been used where $D_*^{(\ell)}$ would be more correct.

$$\begin{aligned} \theta_{cr}^{(\ell)} &= 0.24\, D_*^{-1} &,\quad 1 &< D_* \le 4 \\ \theta_{cr}^{(\ell)} &= 0.14\, D_*^{-0.64} &,\quad 4 &< D_* \le 10 \\ \theta_{cr}^{(\ell)} &= 0.04\, D_*^{-0.1} &,\quad 10 &< D_* \le 20 \\ \theta_{cr}^{(\ell)} &= 0.013\, D_*^{0.29} &,\quad 20 &< D_* \le 150 \\ \theta_{cr}^{(\ell)} &= 0.055 &,\quad 150 &< D_* \end{aligned} \tag{17}$$

a = Van Rijn's reference height

$\hat{A}_\delta$ = Peak orbital excursion at the bed

= $\dfrac{T_p \hat{U}_\delta}{2\pi}$

$d_{50}^{(\ell)}$ = Representative sediment diameter

$d_{90}^{(\ell)}$ = 90% sediment passing size

= $1.5\, d_{50}^{(\ell)}$

h = Water depth

k_a = Apparent bed roughness felt by the flow when waves are present. Calculated by DELFT3D-FLOW using the wave-current interaction formulation selected by the user. Refer to the DELFT3D-FLOW User Manual for further details.

$(k_a \le 10k_s)$

k_s = User specified current-related effective roughness height (space varying).

$k_{s,w}$ = Wave-related roughness, calculated from ripple height. See equation (15)

u_z = Velocity magnitude taken from a near-bed computational layer. In a current-only situation the velocity in the bottom computational layer is used. Otherwise, if waves are active, the velocity is taken from the layer closest to the height of the top of the wave mixing layer δ

$\hat{U}_\delta$ = Peak orbital velocity at the bed.
$= \sqrt{2} \times$RMS orbital velocity at bed, taken from the wave module.

z_u = Height above bed of the near-bed velocity (u_z) used in the calculation of bottom shear stress due to current.

Δ_r = Estimated ripple height. Refer to equation (3).

δ_m = Thickness of wave boundary mixing layer following van Rijn (1993)
= $3\delta_w$ (and $\delta_m \ge k_a$)

δ_w = Wave boundary layer thickness
= $0.0782\hat{A}_\delta \left(\frac{\hat{A}_\delta}{k_{s,w}} \right)^{0.25}$

Note: Superscript (ℓ) implies that this quantity applies to sediment fraction number (ℓ)

We emphasise the following points regarding this implementation:

1. The bottom shear stress due to currents is based on a near-bed velocity taken from the hydrodynamic calculations, rather than the depth-averaged velocity used by van Rijn. Refer to appendix B for further details.
2. All sediment calculations are based on hydrodynamic calculations from the previous half time-step. We find that this is necessary to prevent unstable oscillations developing.
3. The apparent roughness felt by the flow (k_a) is dependent on the hydrodynamic wave-current interaction model chosen by the user. At this time, van Rijn's wave-current interaction model is not available in DELFT3D-FLOW. This means that it is not possible for a user to exactly reproduce results obtained using van Rijn's full formulations for waves and currents.

4. Calculation of the required sediment flux terms

The transfer of sediment between the bed and the flow is modelled using sink and source terms acting on the *kmx* layer. Each half time-step these terms model the quantity of sediment entering the flow due to upward diffusion from the reference level, and the quantity of sediment dropping out of the flow due to particle settling. The mathematical

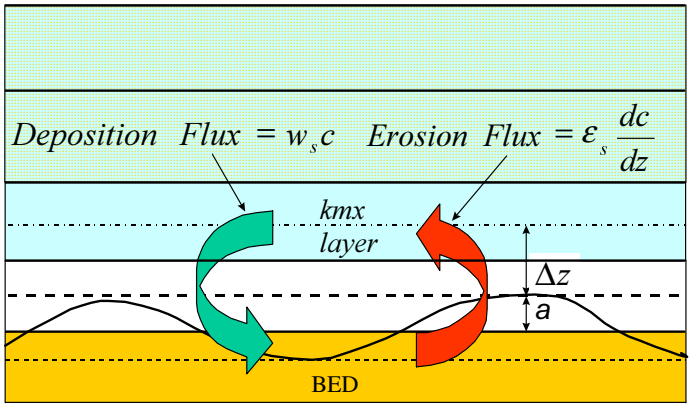


Figure 5 - Schematic arrangement of flux bottom boundary condition

treatment of sink and source terms in DELFT3D-FLOW is discussed more fully in the DELFT3D-FLOW User Manual. Here it is sufficient to remark that a sink term removes sediment at a volumetric rate multiplied by the calculated concentration of the cell and thus must be included in an implicit manner when the advection-diffusion equation is solved. This is quite different from the mathematical treatment of a source term which is an explicit addition of sediment in the advection-diffusion equation. The required sink and source terms for each *kmx* grid cell are calculated in the following manner:

1. Erosive flux due to upward diffusion.

The upward diffusion of sediment through the bottom of the *kmx* cell is given by the expression

$$E = \varepsilon_s \frac{\partial c}{\partial z} \tag{18}$$

Where ε_s and $\frac{\partial c}{\partial z}$ are evaluated at the bottom of the *kmx* cell.

We approximate this expression by (refer to Figure 6)

$$E \approx \varepsilon_s \left(\frac{c_a - c_{kmx}}{\Delta z} \right) \tag{19}$$

Where:

ε_s = Sediment diffusion coefficient evaluated at the bottom of the *kmx* cell.

c_a = Reference concentration

Note: strictly this should be written $c_a^{(\ell)}$ however the super-scripted sediment fraction has been omitted in all cases for clarity.

c_{kmx} = Average concentration of the *kmx* cell
Note: strictly this should be written $c_{kmx}^{(\ell)}$ however the super-scripted sediment fraction has been omitted in all cases for clarity.

Δz = Difference in elevation between the centre of the *kmx* cell and van Rijn's reference height
= $z_{kmx} - a$

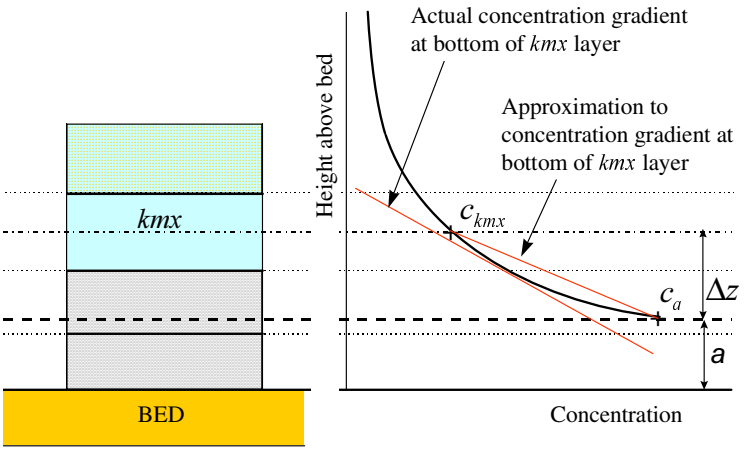


Figure 6 - Estimation of concentration gradient

This is then split into two terms as follows:

$$E \approx \frac{\varepsilon_s c_a}{\Delta z} - \frac{\varepsilon_s c_{kmx}}{\Delta z} \tag{20}$$

The first of these terms can be evaluated explicitly, and is implemented as a sediment source term. The second can only be evaluated implicitly and is implemented as a (positive) sink term. Thus:

$$\begin{aligned} Source_{erosion} &= \frac{\varepsilon_s c_a}{\Delta z} \\ Sink_{erosion} &= \frac{\varepsilon_s c_{kmx}}{\Delta z} \end{aligned} \tag{21}$$

2. Deposition flux due to downward settling of particles.

The downward settling of sediment through the bottom of the *kmx* cell is given by the expression

$$D = w_s c_{kmx(bot)} \tag{22}$$

Where w_s and $c_{kmx(bot)}$ are evaluated at the bottom of the *kmx* cell.

As the value of $c_{kmx(bot)}$ is unknown, we assume a linear concentration gradient between the centre of the *kmx* cell and van Rijn's reference height (Figure 7). This allows us to approximate $c_{kmx(bot)}$ by assuming that:

$$c_{kmx(bot)} \approx c_{kmx} + \frac{\partial c}{\partial z}\frac{thick_{kmx}}{2} \tag{23}$$

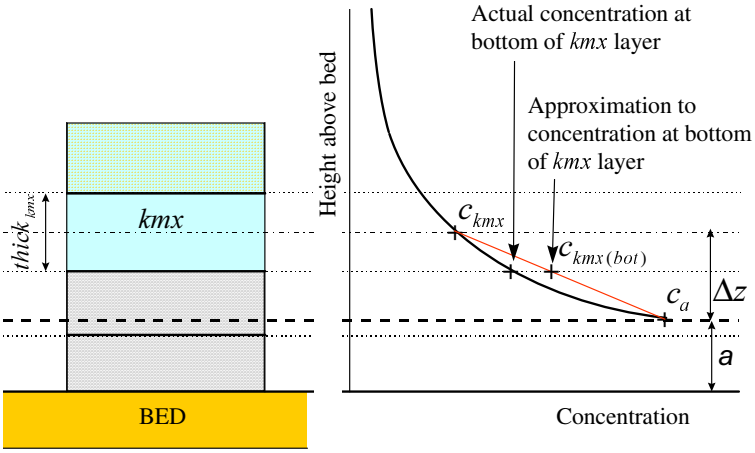


Figure 7 - Estimation of concentration at bottom of *kmx* layer

Where c_{kmx} and $\frac{\partial c}{\partial z}$ are defined as for the calculation of the erosive flux above, and $thick_{kmx}$ is the thickness of the *kmx* layer. This leads to the following approximation to the expression for the deposition flux:

$$D \approx c_a w_s \left(\frac{thick_{kmx}}{2\Delta z} \right) + c_{kmx} w_s \left(1 - \frac{thick_{kmx}}{2\Delta z} \right) \tag{24}$$

This, once again, has been split into two terms. As for the erosion expressions, the first term can be evaluated explicitly, and is implemented as a (negative) sediment source term. The second can only be evaluated implicitly and is implemented as a (positive) sink term. Thus:

$$\begin{aligned} Source_{deposition} &= -c_a w_s \left(\frac{thick_{kmx}}{2\Delta z} \right) \\ Sink_{deposition} &= c_{kmx} w_s \left(1 - \frac{thick_{kmx}}{2\Delta z} \right) \end{aligned} \tag{25}$$

3. Calculation of total source and sink terms

From equations (21) and (25) it follows that the expressions for the total source and sink terms for each cell are

$$\begin{aligned} Source &= c_a\left(\frac{\varepsilon_s}{\Delta z} - w_s\frac{thick_{kmx}}{2\Delta z}\right) \\ Sink &= c_{kmx}\left(\frac{\varepsilon_s}{\Delta z} + w_s\left(1-\frac{thick_{kmx}}{2\,\Delta z}\right)\right) \end{aligned} \tag{26}$$

From this equation we observe that if $\varepsilon_s < \frac{w_s\, thick_{kmx}}{2}$ then the source term will become negative. If this situation was allowed to occur then the negative source terms could produce negative sediment concentrations, which are physically unrealistic and numerically unstable. In order to prevent this from occurring the above condition is checked and if a negative source term would result then the simpler deposition flux

$$Sink_{deposition} = c_{kmx} w_s \tag{27}$$

is used in place of equation (25). This expression uses the average concentration of the *kmx* cell, rather than an estimate of the concentration at the bottom of the cell, to calculate the deposition flux. Use of this simpler expression for the settling flux results in the following expressions for the total source and sink terms

$$\begin{aligned} Source &= c_a\left(\frac{\varepsilon_s}{\Delta z}\right) \\ Sink &= c_{kmx}\left(\frac{\varepsilon_s}{\Delta z} + w_s\right) \end{aligned} \tag{28}$$

These source and sink terms are guaranteed to be positive, although they will result in slightly less deposition than the expressions given by equations (26) which are used in the rest of the simulation. We find that, in practice, conditions rarely require the use of equations (28) and when these simplified equations are used they do not introduce a discernible error into the solution.

Further inspection of equation (26) reveals that, as $\frac{thick_{kmx}}{2\Delta z}$ is restricted to the range 0 to 1 by the manner in which the location of the *kmx* layer is defined, the total sink term is always positive.

3.3.3 Concentration bottom boundary condition

As mentioned above, setting a known concentration at the bottom boundary is a possible alternative to setting a known sediment flux. This approach is attractive, for non-cohesive sediment, as van Rijn's pick-up formulations mean that a near-bottom reference concentration can easily be calculated. Accurate implementation of this alternative is somewhat more complicated, however, as the resulting equilibrium sediment concentration profile is very sensitive to the height at which the reference concentration is applied. Thus it is essential that if van Rijn's reference concentration is to be used as the bottom boundary

concentration it must be applied at the correct reference height, as stipulated by van Rijn (1984).

Unfortunately, due to the sigma co-ordinate system used in DELFT3D-FLOW, it is impossible to force the bottom cell to have its mid-point at exactly van Rijn's reference height; as the water depth, and thus the layer thickness, changes from place to place. This means that the reference concentration must be adjusted to suit the actual thickness of the bottom cell. Furthermore, it is unwise to attempt to set the bottom boundary concentration at a level much lower than van Rijn's reference level as the required concentration increases rapidly, and the solution becomes very sensitive to small variations in the available turbulent mixing (which approaches zero at the bed). Thus it is more prudent to adjust the reference concentration upwards from van Rijn's reference level until the next cell midpoint is reached, and set the boundary concentration in this cell (Figure 8). Obviously this implies that, depending on the vertical grid scheme chosen and the local water depth, one or more computational cells may fall below the concentration "boundary" cell, and therefore effectively fall out of the calculation.

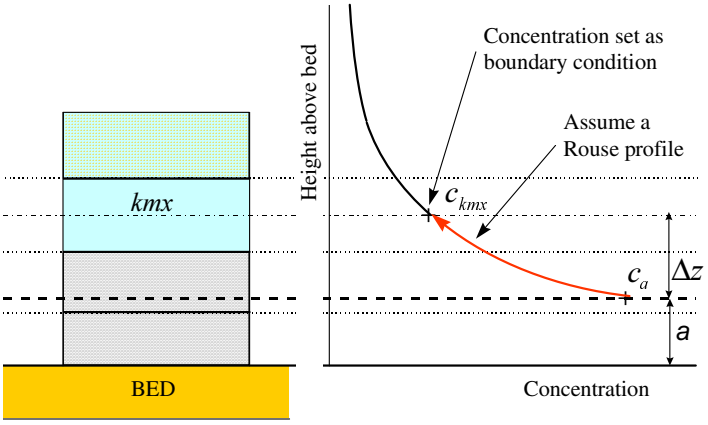


Figure 8 - Concentration boundary condition

This approach has been successfully implemented in research version 03.05.006 of the DELFT3D-FLOW routine; although this implementation is limited to current-only situations as it relies on using the standard Rouse profile to convert the reference concentration to the correct (equilibrium) concentration at the level corresponding to the centre of the "boundary" sediment transport cell. Although this limitation could be overcome, by using a simple form of numerical integration to calculate the equilibrium concentration at the higher level, this approach is not developed further as we believe that it would be rather more complicated to implement features such as fixed layers and multiple sediment fractions if a concentration boundary condition was used. Because of these limitations and concerns, the development of the concentration boundary condition version (006) has not been carried as far as that of the flux boundary condition (007). A useful comparison of the results achieved by both versions has been carried out however, and a summary of this is located in section 4.2 of this paper.

3.4 Other Sediment Boundary and Initial Conditions

In addition to the bottom boundary, the other sediment transport boundary conditions, and initial concentrations need to be considered. These are dealt with as follows.

3.4.1 Water surface boundary

The vertical diffusive flux through the free surface is set to zero for all conservative constituents (except heat, which can cross this boundary). This is left unchanged for suspended sediment.

3.4.2 Open lateral boundaries

DELFT3D-FLOW requires the user to specify boundary conditions for all conservative constituents at all open boundaries. When modelling in three dimensions the user may choose to specify boundary concentrations that have a uniform, linear, or step distribution over the vertical profile. The user may also choose to specify a "Thatcher-Harleman" return time to simulate the re-entry of material that flowed out of the model if the flow reverses direction.

All of these options are also available for sediment constituents, although they are probably more appropriate for fine, cohesive sediment than for sand-sized particles. To assist with modelling coarser material an additional option has been included in the sediment version of DELFT3D-FLOW. This allows the user to specify that, at all open boundaries, the flow should enter carrying all "sand" sediment fractions at their equilibrium concentration profiles. This means that the sediment load entering through the boundaries will be near-perfectly adapted to the local flow conditions, and very little accretion or erosion should be experienced near the model boundaries. This will generally be the desired situation if the model boundaries are well chosen.

The user may select this option by setting [**EQMBC = .true.**] in the **morph.inp** file.

3.4.3 Initial condition

Sediment constituents' initial concentrations are handled in exactly the same manner as those for any other conservative constituent. DELFT3D-FLOW allows a number of options for specifying the initial concentrations, namely, the user may elect to:

1. Specify one global initial concentration for each constituent.
2. Specify that the initial concentrations are read from a restart file generated by a previous run.
3. Specify space-varying initial concentrations directly from a user-defined input file.

It is our experience that sediment concentrations adapt very rapidly to equilibrium conditions and that, in the case of a cold start where the hydrodynamic model also takes some time to stabilise, a uniform zero concentration for the sediment fractions is usually adequate. Nevertheless, specifying an initial concentration distribution is usually required

in the case of a hot start from a restart file. Readers should consult the DELFT3D-FLOW user manual for more detail on how to perform this procedure.

3.5 Calculation of Equilibrium Sediment Concentration Profiles

In the sediment version of DELFT3D-FLOW the user has two options to calculate the equilibrium sediment concentration for each computational cell. Both methods are based on the use of van Rijn's reference concentration applied at the correct reference height (refer to section 3.3.2 above). The first method sets the concentrations of cells above the reference height to the concentration calculated using the well-known Rouse expression

$$\frac{c}{c_a} = \left[\frac{(a)(h-z)}{(z)(h-a)}\right]^z \tag{29}$$

Users should note that this expression assumes that concentrations are small (constant fall velocity), and that the turbulent mixing follows a parabolic distribution (zero at the bed and at the free surface). These assumptions are reasonable in the case of normal, current-only flows; however they do not adequately describe the conditions found if waves or significant stratification are present.

The second method (which is selected by setting [**ROUSE = .false.**] in the **morph.inp** file) calculates the equilibrium concentrations using the actual mixing values calculated by DELFT3D, and a Runga Kutta method of numerical integration to solve the stationary advection-diffusion equation (equation (42)). This method gives the correct results even when the turbulent mixing profile is clearly non-parabolic.

Users should note that in the current (007) implementation the calculated equilibrium sediment concentration profiles are only used for the calculation of sediment concentrations at open inflow boundaries (refer to section 3.4.2 above) and for output to the MAP and HISTORY files in order to give the user a benchmark against which to compare the non-equilibrium concentrations calculated by DELFT3D-FLOW. Therefore, users should decide for themselves whether having more accurately calculated equilibrium sediment concentration profiles at the boundaries, or a more widely recognised and relatively independent benchmark against which the DELFT3D-FLOW calculations can be compared, is more appropriate to their application.

3.6 The Effect of Sediment on Fluid Density

In its standard form (v 03.05), DELFT3D-FLOW uses an empirical relation (Eckart 1958) to adjust the density of water in order to take into account varying temperature and salinity. In the sediment version this is extended to include the density effect of the presence of sediment fractions in the fluid mixture. This is achieved by adding (per unit volume) the mass of all sediment fractions, and subtracting the mass of the displaced water. As a mathematical statement this translates as

$$\rho_{mix} = \rho_w + \sum_{\ell=1}^{lsed} c^{(\ell)} (\rho_s^{(\ell)} - \rho_w) \tag{30}$$

where *lsed* is the number of sediment fractions.

As this calculation is carried out for each computational cell at each time-step it has implications for creating density currents where horizontal gradients in density exist. Vertical density gradients can also have a significant effect on the amount of vertical turbulent mixing present, as discussed in section 3.9 below.

The decision to include of the effect of sediment on the fluid density may be made by the user by setting the [**DENSIN**] flag in the **morph.inp** file. This option has been included as it has been found that a secondary effect of including sediment in the density calculations is a reduction of the flow velocity in the lower computational layers (when compared with a standard logarithmic velocity profile) and a consequent reduction in the computed bed shear stress. This reduction in bed shear stress is particularly pronounced when the k-epsilon turbulence closure model is used, and leads to an increase in overall flow velocity and a consequent lowering of the free surface. Our experience shows that this change in the free surface level (even if very slight) can lead to calibration problems when converting an existing 2DH model to 3D if the model is driven using water level boundary conditions. A simple method of circumventing these problems can be achieved by setting [**DENSIN = .false.**] which has the effect of preventing the sediment from having any effect on the density of the water/sediment mixture. The user should use this flag with caution, however, as it will introduce other inaccuracies into the simulation, such as negating the damping effect that a vertical density gradient has on the vertical diffusion of sediment and other constituents. Refer to section 3.9 below for more detail on this process.

3.7 Particle Settling Velocity in Clear Water

The sediment version of DELFT3D-FLOW uses a simple first-order upwind numerical scheme to calculate the quantity of sediment falling out of the bottom of each computational cell at each time-step. This scheme works by simply multiplying the average concentration of the cell by the hindered settling velocity for the sediment fraction; calculated at the lower cell interface. The calculation of the non-hindered settling velocity is discussed in this section; the formulation of the hindering effect of high sediment concentrations is discussed in section 3.8.

While use of the upwind settling formulation does slightly under-predict the mass of sediment settling at each time-step, the magnitude of this error has been shown to be rather small (refer to section 4.2.1). Furthermore, the process of conducting these tests has confirmed that the upwind settling scheme remains far more stable than the, slightly more accurate, central difference scheme in situations where the upward diffusion of sediment particles reduces to zero.

3.7.1 Cohesive sediment

"Mud" fractions

If the sediment fraction type is defined as "**mud**" then the formation of sediment flocs is ignored and the (non-hindered) fall velocity of the fraction is simply set equal to the user specified input value [**WS0(L)**]. I.e.

$$w_{s,0} = [\ \mathbf{WS0(L)}\ ] \tag{31}$$

In this case, the user parameters [**WSM(L)**] and [**SALMAX(L)**] are not required.

"Floc" fractions

In salt water cohesive sediment tends to join together (flocculate) to form sediment "flocs", with the degree of flocculation depending on the salinity of the water. These flocs are much larger than the individual sediment particles, and settle at a faster rate. In order to model this change in settling velocity the user should define the sediment fraction as type "**floc**". In this case the user must also supply two settling velocities and a maximum salinity. The first velocity [**WS0(L)**] is taken to be the settling velocity of the fraction in fresh water (salinity = 0). The second [**WSM(L)**] is the settling velocity of the fraction in water having a salinity equal to [**SALMAX(L)**]. The settling velocity of the sediment flocs is calculated as follows:

If the salinity of the computational cell is less than [**SALMAX(L)**] then

$$w_{s,0} = \frac{W_{s,\max}}{2} * \left(1 - \cos(\pi \frac{S}{S_{\max}})\right) + \frac{W_{s,f}}{2} * \left(1 + \cos(\pi \frac{S}{S_{\max}})\right) \tag{32}$$

where: $w_{s,0}$ = The (non-hindered) settling velocity of the sediment fraction
$W_{s,max}$ = [**WSM(L)**] (User specified)
$W_{s,f}$ = [**WS0(L)**] (User specified)
S = Salinity of the cell
$S_{\max}$ = [**SALMAX(L)**]

If the salinity of the computational cell is greater than [**SALMAX(L)**] then

$$w_{s,0} = W_{s,\max} \tag{33}$$

Modelling the break-up of sediment flocs is not yet operational as the existing relations describing the process are not yet fully tested (they are included in the source code, but are inactive at present).

3.7.2 Non-cohesive sediment

The settling velocity of a non-cohesive ("sand") sediment particle is computed following the method of van Rijn (1993). The formulation used depends on the diameter of the sediment in suspension, as follows:

$$
\begin{aligned}
w_{s,0}^{(\ell)} &= \frac{(s^{(\ell)}-1)g\, d_s^{(\ell)2}}{18\upsilon} && , \quad 1\ \mu m < d_s \le\ 100\ \mu m \\
w_{s,0}^{(\ell)} &= \frac{10\nu}{d_s}\left[\left(1+\frac{0.01(s^{(\ell)}-1)g\, d_s^{(\ell)3}}{\upsilon^2}\right)^{0.5}-1\right] && , \quad 100\ \ \mu m < d_s \le 1000\ \mu m \\
w_{s,0}^{(\ell)} &= 1.1\left[(s^{(\ell)}-1)g\, d_s^{(\ell)}\right]^{0.5} && , \quad 1000\ \mu m < d_s
\end{aligned}
\tag{34}
$$

The user should note that in this expression $d_s^{(\ell)}$ is the representative diameter of the suspended sediment, and is a user-defined property [**SEDDIA(L)**]. In the case of a uniform sediment size at the bed, this will be usually be equal to the median diameter (d_{50}) of the bed material. In the case of non-uniform bed material Van Rijn (1993) concluded that, on the basis of measurements, $d_s^{(\ell)}$ should be in the range of 60 to 100% of the (d_{50}) of the bed material. If the bed material is very widely graded (well sorted) consideration should be given to using several sediment fractions to model its behaviour more accurately.

3.8 Particle Settling Velocity in a Mixture

In high concentration mixtures, the settling velocity of a single particle is reduced due to the presence of other particles. In order to account for this hindered settling effect we follow Richardson and Zaki (1954) and determine the settling velocity in a fluid-sediment mixture as a function of the sediment concentration and the non-hindered settling fall velocity as follows:

$$
w_s = \left(1-\frac{c_m^{tot}}{\rho_s}\right)^5 w_{s,0} \tag{35}
$$

Note that in equation (35) c_m^{tot} is the total mass concentration of the computational cell. Mathematically

$$
c_m^{tot} = \sum_{\ell=1}^{lsed} \rho_s^{(\ell)} c^{(\ell)} \tag{36}
$$

where *lsed* is the number of sediment fractions and $c^{(\ell)}$ is the volumetric concentration of each sediment fraction.

As the fall velocity is now a function of the sediment concentration, this implies that each sediment fraction has a fall velocity which is a function of location and time. Thus the fall velocity for each fraction must be calculated at every time-step, for each computational cell.

3.9 Particle Diffusion

Turbulent mixing of water is a process which generally takes place on a spatial scale too small to be resolved on the grid size used in most DELFT3D-FLOW simulations. In order to model the important fluid mixing processes that take place on this sub-grid scale DELFT3D-FLOW uses a number of so-called "turbulence closure models" (TCM's). Four different TCM's of varying complexity are available to the user, they are:

1. Constant coefficient
2. Algebraic eddy viscosity closure model
3. $k - L$ turbulence closure model
4. $k - \varepsilon$ (k-epsilon) turbulence closure model

The first is a simple constant value which is specified by the user. A constant eddy viscosity will lead to parabolic vertical velocity profiles (laminar flow). The other three TCM's are based on the eddy viscosity concept of Kolmogorov (1942) and Prandtl (1945) and offer zero, first, and second order closures for the turbulent kinetic energy (k) and for the mixing length (L). All three of the more advanced TCM's take into account the effect that a vertical density gradient has on damping the amount of vertical turbulent mixing. A full description of the available turbulence models can be found in the DELFT3D-FLOW user manual.

The validation simulations performed as part of this study have been carried out using either the Algebraic TCM, to ensure logarithmic velocity profiles, or the k-epsilon TCM for maximum realism in stratified or other very three-dimensional situations.

The output of the selected TCM is the eddy viscosity at each layer interface; from this the vertical sediment mixing coefficient is calculated using the following expressions:

$$\varepsilon_s = \frac{vicmol}{sigmol} + \frac{\beta^{(\ell)}}{\sigma_c^{(\ell)}} \nu_v \tag{37}$$

$$\sigma_c^{(\ell)} = \sigma_{c0}^{(\ell)} F_\sigma(Ri) \tag{38}$$

where: $vicmol$ = Molecular viscosity of water (= 1 x 10^{-6})
$sigmol$ = Molecular Prandtl number (= 6.7)
$\beta^{(\ell)}$ = Van Rijn's β factor (= $\varepsilon_{s,\max} / \varepsilon_{f,\max}$)
$\sigma_c^{(\ell)}$ = Modified Prandtl-Schmidt number
ν_v = Vertical eddy viscosity (from the TCM)
$\sigma_{c0}^{(\ell)}$ = Basic Prandtl-Schmidt number
$F_\sigma(Ri)$ = Damping function

$F_\sigma(Ri)$ is a damping function which is only active in the case of the Algebraic TCM. The damping function simulates the turbulence damping caused by stratification of the flow and its value depends on the Richardson Numbers (Ri) of adjacent model layers. The damping function $F_\sigma = 1$ if the k-epsilon TCM is used, as the turbulence damping effects of vertical

density gradients are automatically accounted for in the k-epsilon model. This is one of the main advantages of the k-epsilon TCM. The DELFT3D-FLOW User Manual contains further detail regarding the implementation of the TCM's and damping functions in DELFT3D-FLOW.

The values used for $\beta^{(\ell)}$ and $\sigma_{c0}^{(\ell)}$ depend on the type of sediment, and are discussed below.

3.9.1 Cohesive sediment

The diffusion of cohesive sediment, i.e. sediment of type "**mud**" or "**floc**", is calculated using the standard values for a conservative constituent. In this case:

$\beta^{(\ell)}$ = 1.0 as van Rijn's β factor is only applied to sand-sized sediment
σ_{c0} = 0.7 which is standard in DELFT3D-FLOW for a constituent consisting of very fine "particles"

3.9.2 Non-cohesive sediment

In the case of a "**sand**" sediment fraction, the Prandtl-Schmidt number σ_{c0} is set to 1.0, and the $\beta^{(\ell)}$ factor is calculated in accordance with the method of van Rijn (1984). Namely

$$\beta^{(\ell)} = 1 + 2\left[\frac{w_s^{(\ell)}}{u_{*,c}}\right]^2, \qquad for \qquad 0.1 < \frac{w_s^{(\ell)}}{u_{*,c}} < 1 \tag{39}$$

Where $w_s^{(\ell)}$ is the settling velocity of the sediment fraction in clear water, and $u_{*,c}$ is the local bed shear stress due to currents.

This implies that the value of β is space (and time) varying, however it is constant over the depth of the flow. In addition, due to the limited knowledge of the physical processes involved, we follow van Rijn (1993) and limit β to the range $1 < \beta^{(\ell)} < 1.5$.

We note that in a wave and current situations van Rijn (1993) applies this β factor to only the current-related turbulent mixing, whereas we apply it to the total turbulent mixing calculated by the selected TCM (see equation (37) above). We believe, however, that as little is known about the dependence of the β factor on flow conditions this discrepancy is of little practical significance.

3.10 Availability of Sediment at the Bed

The sediment version of DELFT3D-FLOW (v 03.05.007) is capable of modelling areas of non-erodible material (fixed layers). Because of this, the user must specify the quantity of sediment available at the bed at the outset of the model run. This is achieved by specifying the total (dry) mass of all fractions of sediment above the fixed layer (in kg/m^2) at the outset of the model run. This may be specified to be a constant value for the entire model by

setting the value of [**SEDBUNI**] in the sediment input file or, alternatively, specifying [**FILSDB]** as the file name of a space-varying initial sediment file.

The thickness of the sediment above the fixed layer is calculated by dividing the mass of sediment available at the bed by the (user specified) dry bed density [**CDRYB(L)**]. At this point in time only one value can be specified for the dry bed density for the entire model, this is read from the [**CDRYB(L)**] value for the first sediment fraction. In the future, when multiple sediment fractions are properly implemented, we envisage that it will also be possible to specify a space-varying dry bed density file. The reader should refer to appendix D for more detail on the envisaged future development of the bottom-evolution model in DELFT3D-FLOW.

Fixed layers are implemented in DELFT3D-FLOW by means of reducing the sediment flux terms when the bed approaches the fixed layer. This is carried out in slightly different manners for cohesive and non-cohesive sediments, as described below.

3.10.1 Cohesive sediment fractions

In the case of cohesive sediment, the erosive sediment source term is simply compared with the quantity of sediment (of the appropriate fraction) available at the bed. If the source term would result in more sediment being eroded than is available, then the source term is reduced to the quantity of sediment available.

3.10.2 Non-cohesive sediment fractions

Because of the more complex, part explicit - part implicit, erosive flux terms used for "**sand**" type sediments it is not possible to use the simple source limitation technique used for cohesive sediments. Instead, the user is required to specify a threshold thickness [**THRESH**]. At each time-step the thickness of the bottom sediments in each computational cell are calculated as described above. If the remaining sediment thickness is less than the user-specified threshold *and erosive conditions are expected* then the source and sink sediment flux terms (generated as described in section 3.3 above) are reduced in the following manner:

$$\begin{aligned} Source_{total} &= Source_{total} * f_r \\ Sink_{total} &= Sink_{total} * f_r \end{aligned} \tag{40}$$

Where: f_r is a reduction factor determined by

$$\begin{aligned} f_r &= \frac{\Delta_{sed}}{[THRESH]} \\ 0 \le f_r &\le 1 \end{aligned} \tag{41}$$

where Δ_{sed} is the thickness of sediment at the bed, calculated by dividing the total mass of sediment at the bed (all fractions) by the density of the bed layer, as described above.

The likelihood of erosive conditions occurring is assessed by calculating the total sediment source and sink terms using the cell concentration from the previous time-step to evaluate the implicit sink term. If the sink term is greater than the source term, then deposition is expected, and f_r is set to 1.0 so that deposition close to the fixed layer is not hindered.

3.11 Morphological Updating

The sediment version of DELFT3D-FLOW (v 03.05.007) dynamically updates the elevation of the bed at each computational time-step. This is one of the distinct advantages of the sediment version of DELFT3D-FLOW as it means that the hydrodynamic flow calculations are always carried out using the correct bathymetry.

At each time-step, the FLOW module calculates the change in the mass of bottom sediment that has occurred as a result of the sediment sink and source terms. This change in mass is then translated into a change in thickness of the bottom sediment layer using the same (trivial) expression for the density of the bed material as discussed in section 3.10 above. This change in thickness is equivalent to a change in bed elevation, which is applied to the depth values stored at the ζ (zeta, or water-level) points. The new depths at the velocity points are then set by way of a simple upwind numerical scheme from the new depths at the ζ points[4]. The depths stored at the depth points (which are read directly from the bathymetry specified by the user) are not updated (as they are not used by the rest of the FLOW module).

We note that a limitation of the current implementation is that the time-varying depths held at the zeta and velocity points are not written out to the history, map, or communication files. This situation needs to be remedied, and may have further consequences for other DELFT3D modules, e.g. WAQ and GPP.

A number of additional features have been included in the morphological updating routine in order to increase the flexibility of the sediment version of DELFT3Dflow. These are as follows:

3.11.1 Morphological "switch"

The user can specify whether or not to update the calculated depths to the bed by setting the [**MORUPD**] flag in the **morph.inp** file. It may be useful to turn bottom updating off if only the initial patterns of erosion and deposition are required, or an investigation of sediment transport patterns with a constant bathymetry is desired. Note that use of [**MORUPD**] only affects the updating of the depth values (at ζ and velocity points) used by flow calculations at subsequent time-steps; the quantity of sediment available at the bed (BODSED and DPSED) will still be updated, regardless of the state of the [**MORUPD**] flag. If the user wishes to prevent any change in both the bottom sediments *and* flow depths from the initial condition then this may be easily achieved by either setting [**MORSTT**] to

[4] *Note that this is rather different from the method used in the standard FLOW module where the depth at velocity points is set once, at initiation, by interpolation between adjacent* ***depth*** *points and is not subsequently updated.*

a value larger than the total number of time-steps in the simulation, or by setting [**MORFAC**] = 0. See below for a description of these two user variables.

3.11.2 Morphological delay

Frequently, a hydrodynamic simulation will take some time to stabilise after transitioning from the initial conditions to the (dynamic) boundary conditions. It is likely that during this stabilisation period the patterns of erosion and accretion that take place do not accurately reflect the true morphological development and should be ignored. This is made possible by use of [**MORSTT**] whereby the user can specify a time-step number after which the morphological bottom updating will begin. Before the simulation reaches the [**MORSTT**] time all other calculations will proceed as normal (sediment will be available for suspension for example) however the effect of the sediment fluxes on the available bottom sediments will not be taken into account.

We acknowledge that it would be simpler for users if the user-specified value was a *time* at which to start morphological calculations, rather than a time-step number, this should be improved before a public release of this version is considered.

3.11.3 Morphological time scale factor

One of the complications inherent in carrying out morphological projections on the basis of hydrodynamic flows is that morphological developments take place on a time scale several times longer than typical flow changes (for example, tidal flows change significantly in a period of hours, whereas the morphology of a coastline will usually take weeks, months, or years to change significantly). One technique for approaching this problem is to use a "morphological time scale factor" whereby the speed of the changes in the morphology is scaled up to a rate that it begins to have a significant impact on the hydrodynamic flows. This can be achieved in the sediment version of DELFT3D-FLOW by specifying a non-unity value for the variable [**MORFAC**] in the **morph.inp** file.

The implementation of the morphological time scale factor in version 03.05.007 of DELFT3D-FLOW is achieved by simply multiplying the erosion and deposition fluxes from the bed to the flow and vice-versa by the [**MORFAC**] factor, at each computational time-step. This allows accelerated bed-level changes to be incorporated dynamically into the hydrodynamic flow calculations.

While the maximum morphological time scale factor that can be included in a morphodynamic model without affecting the accuracy of the model will depend on the particular situation being modelled, and will remain a matter of judgement, tests have shown that the sediment version of DELFT3D-FLOW can remain stable in moderately morphologically active situations even with [**MORFAC**] factors in excess of 1000. We also note that setting [**MORFAC**] = 0 is often a convenient method of preventing both the flow depth and the quantity of sediment available at the bottom from updating, if an investigation of a steady state solution is required.

3.12 Three-dimensional Wave Effects

In the standard version of DELFT3D-FLOW wave effects are only incorporated in a depth-averaged manner via a (breaking) wave induced shear stress at the surface, a wave induced mass flux and an increased bed shear stress. Important wave effects such as streaming in the wave boundary layer and wave induced turbulence are not accounted for. A separate research project, carried out concurrently with this study, aims to improve the representation of these three-dimensional effects in the DELFT3D-FLOW module. The main areas of the improvement recommended by this study are as follows:

1. The wave induced mass flux is corrected with the second order Stokes drift.
2. The production of turbulent energy associated with wave breaking is incorporated by introducing an extra source terms in the kinetic energy and dissipation equations of the k-epsilon turbulence model.
3. The production of turbulent energy associated with dissipation in the near-bed wave boundary layer is incorporated by introducing an extra source terms in the kinetic energy and dissipation equations of the k-epsilon turbulence model.
4. Streaming (a wave induced current in the wave boundary layer directed in the wave propagation direction) is modelled as a time averaged shear stress.

More detail of the formulations developed by, and the preliminary results of, the three-dimensional wave research project may be found in Walstra and Roelvink 2000.

As the implementation of suspended sediment transport in the DELFT3D-FLOW module is valid for wave and current situations it is valuable to combine the results of the two research projects in order to assess the manner in which they interact. The simulations described in section 4.4 of this thesis are designed to test this interaction.

4 Model Validation

4.1 Validation Methodology

The validation of the sediment version (03.05.007) of DELFT3D-FLOW is achieved by performing a number of simulations of hypothetical and real sediment transport problems. The simulations are selected from the large number available in the literature, and those chosen have been selected because they represent a range of complexity of problems, and because they have well documented analytical, computational, or physical solutions against which the results of the DELFT3D-FLOW simulation are compared. A "real-life" test case is also presented and, although there is insufficient measurement data to confirm the results are accurate, it provides a useful demonstration that the modified version of DELFT3D-FLOW can be successfully applied to full-scale real-life problems. The validation is carried out in two distinct phases, with and without waves present, as detailed in the following sections.

Section 4.2 and section 4.3 present the results of using the modified DELFT3D-FLOW module in a range of situations where the stirring effects of waves either do not exist, or are negligible compared to the turbulent mixing effect of the current. Section 4.2 first presents three hypothetical flow experiments that are simple enough for an analytical solution to exist, followed by a more complex physical experiment involving the downstream migration of a trench in a flume. Section 4.3 presents the results of two three-dimensional simulations; the first is hypothetical, the second is real. These simulations do not have any accepted "correct" solution so the results can only be judged qualitatively. Section 4.4 presents the results of an extension to the quantitative validation, this time in a situation where the stirring effects of waves are important. The results of three experiments are reported, the first with waves-only, the subsequent include waves with following currents of differing strengths.

4.2 Quantitative Current-only Validation

The following sections show the results of simulation runs carried out in situations where the stirring effect of waves can be neglected. The simulations start with the rather simple, one-dimensional vertical (1DV), problem of checking the creation of an accurate equilibrium sediment concentration profile under stationary conditions, and describing the settlement of sediment under the influence of gravity alone (zero flow velocity). Following these rather straightforward tests is the result of a 2DV simulation testing the lateral development of an equilibrium sediment concentration profile in an initially clear flow. A more complicated simulation is reported in section 4.2.4 where the migration of a trench under the influence of a transverse current is investigated.

4.2.1 Test 1 - Equilibrium sediment concentration profiles

Under equilibrium (stationary) conditions the advection-diffusion equation reduces to

$$cw_s + \varepsilon_s \frac{dc}{dz} = 0 \tag{42}$$

as presented by Rouse (1937). This equation is valid as long as the fluid and sediment mixing coefficients are approximately equal, as is generally the case for fine sediment particles. This expression can be solved analytically under the additional assumptions of small concentrations (constant w_s) and a parabolic sediment mixing profile, resulting in the expression for the well-known "Rouse" sediment concentration profile

$$\frac{c}{c_a} = \left[\frac{(a)(h-z)}{(z)(h-a)} \right]^z \tag{43}$$

as mentioned in section 3.5 above. Alternatively, a more general solution to equation (42), which does not require the assumption of a constant fall velocity or parabolic sediment mixing profile, may be obtained by using a simple numerical integration technique. Both these approaches are used to check the solution computed by DELFT3D-FLOW.

Model set-up

In order to achieve equilibrium conditions in DELFT3D-FLOW, a very long flume is simulated. The flume is arranged so that the bed slope exactly matches the friction losses of the flow at the desired flow velocity of 2 m/s (refer to Figure 9). This arrangement provides a constant water depth and depth-averaged flow velocity along the length of the flume and allows equilibrium sediment transport conditions to develop towards the downstream end of the flume.

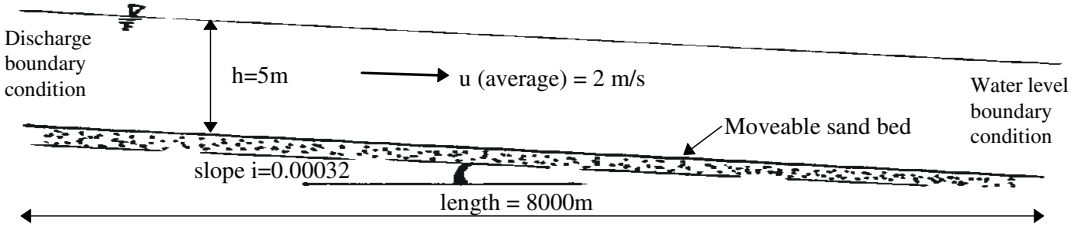


Figure 9 - Long flume for equilibrium sediment concentrations

Note: $d_{50} = 200\mu\text{m}$ and Chezy $C = 50\,\text{m}^{0.5}/\text{s}$ for all simulations

The calibration runs are performed with both up and downstream concentration boundary conditions of $c = 0$, however the concentration boundary condition at the downstream end of the flume is never actually used by the calculation as the flow is always directed outward through this boundary. Sediment concentration and flow velocities are recorded for all computational layers at a point located on the longitudinal centreline of the flume approximately 6.5km from the upstream boundary.

Some of the more important computational parameters relevant to this simulation are as follows:

Length of computational cells in longitudinal direction :	160m
Width of computational cells in transverse direction :	20m
Computational time-step :	2 minutes
Courant number in longitudinal direction :	5
Courant number in transverse direction :	42
Simulation time required to reach steady hydrodynamic state :	≅ 120 min.
Total simulation time :	240 min.

Several simulation runs are performed with this model in order to test the sensitivity of the calculated sediment concentration profile to changes in the model configuration.

Computational results

Figure 10 shows the results of the base simulation run for the sensitivity comparisons. The base settings for the variables tested are as follows:

Number of layers :	20
Bottom Layer thickness :	1% (50mm)
Layer spacing :	Logarithmic
Turbulence model :	Algebraic
Particle settling scheme :	Central Difference
Bottom boundary condition type :	Flux

The results of this run show that the computed sediment concentration profile is very close to that predicted by the Rouse profile, although DELFT3D slightly over-predicts sediment

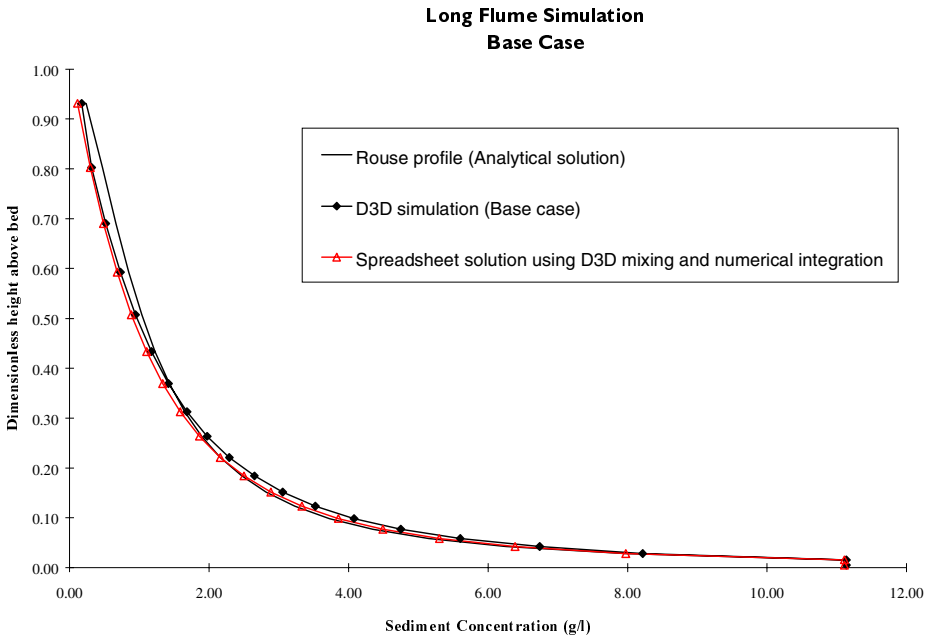


Figure 10 - Equilibrium sediment concentration profiles base case

concentrations in the lower 40% of the water column and slightly under-predicts concentrations above this level. This slight discrepancy can be partially explained by the difference between the turbulent mixing distribution assumed by van Rijn, and that calculated by DELFT3D (Figure 11). Figure 10 also shows a line calculated using equation (42) and the actual turbulent mixing values calculated by DELFT3D, solved by way of numerical integration in a computer spreadsheet. This line falls very close to the concentration profile calculated by DELFT3D, although DELFT3D does appear to generally overestimate the concentrations very slightly. The reason for this discrepancy is possibly numerical diffusion, however we accept it as being insignificant when compared to other uncertainties in computing sediment transport rates.

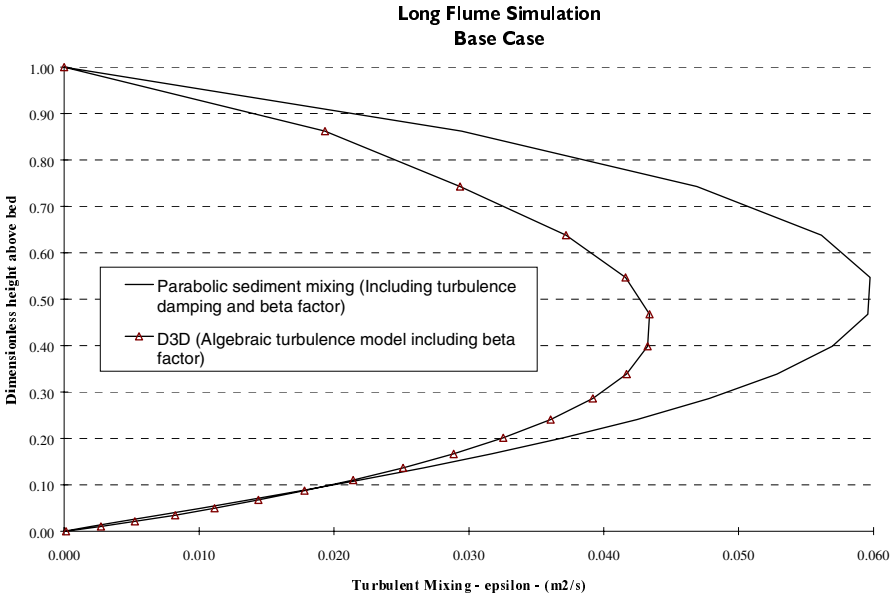


Figure 11 - Turbulent mixing present in base simulation

The turbulent mixing profile calculated by the algebraic turbulence model used in this simulation results in slightly greater mixing than predicted by van Rijn in the bottom 10% of the flume, and significantly less turbulent mixing above this level. We note that the reasonably large difference in turbulent mixing calculated higher in the water column makes little difference to the sediment concentration profile as the sediment concentration gradient high in the water column is relatively small.

To further investigate the discrepancy in the turbulent mixing profiles, the test was re-run using the k-epsilon turbulence model in place of the algebraic model. The results of this test show that, under equilibrium conditions, changing from the algebraic turbulence model to the k-epsilon turbulence model makes little difference to either the sediment concentration profile (Figure 12), or the turbulent mixing (Figure 13) computed by DELFT3D.

The most interesting point arising from this test is that the k-epsilon turbulence model appears to be somewhat less stable than the algebraic turbulence model, as we find that in order to achieve a convergent solution the simulation time-step has to be reduced by a factor of four when the k-epsilon turbulence model is used. Further, we find that this effect

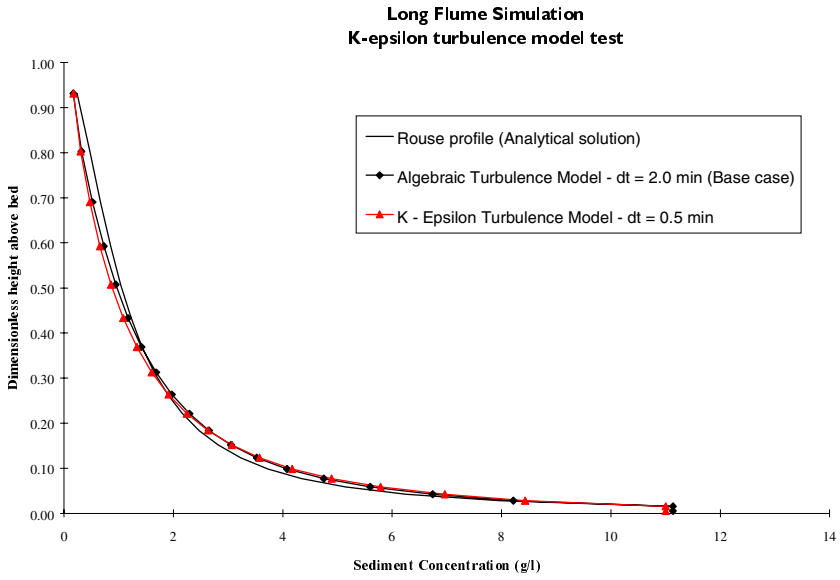


Figure 12 - Influence of turbulence model on sediment concentration profile

can not be attributed to the presence of sediment as the time-step has to be similarly reduced in simulations conducted without sediment.

An earlier version of the sediment implementation suffered from the sediment concentration profiles being dependent on the vertical grid scheme chosen (van Kessel 1999). This dependency has been completely removed, as can be seen in Figure 14. This figure shows that the present implementation is capable of producing an accurate equilibrium sediment concentration profile with as few as five layers. This is a surprisingly good result.

A further test is performed to test the sensitivity of the model to the vertical distribution of

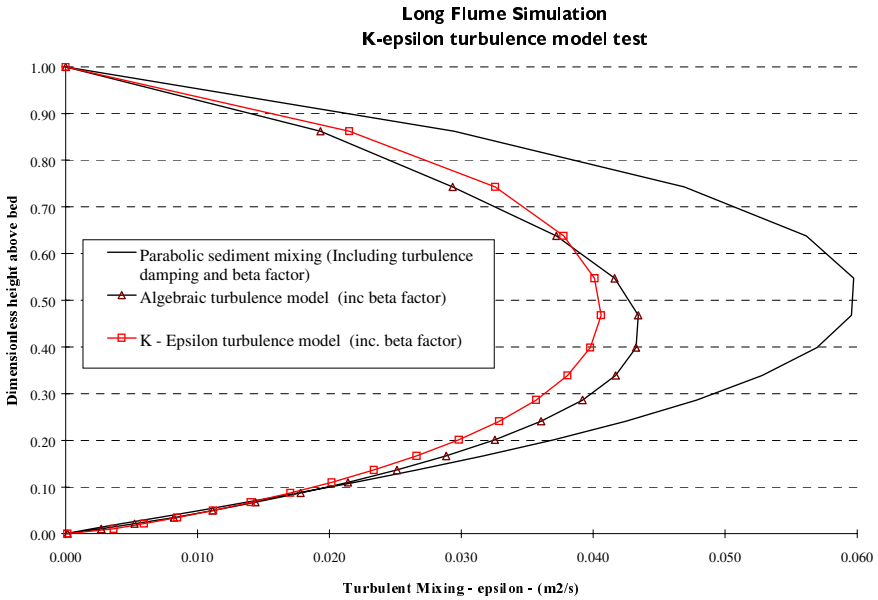


Figure 13 - Influence of turbulence model on turbulent mixing profile

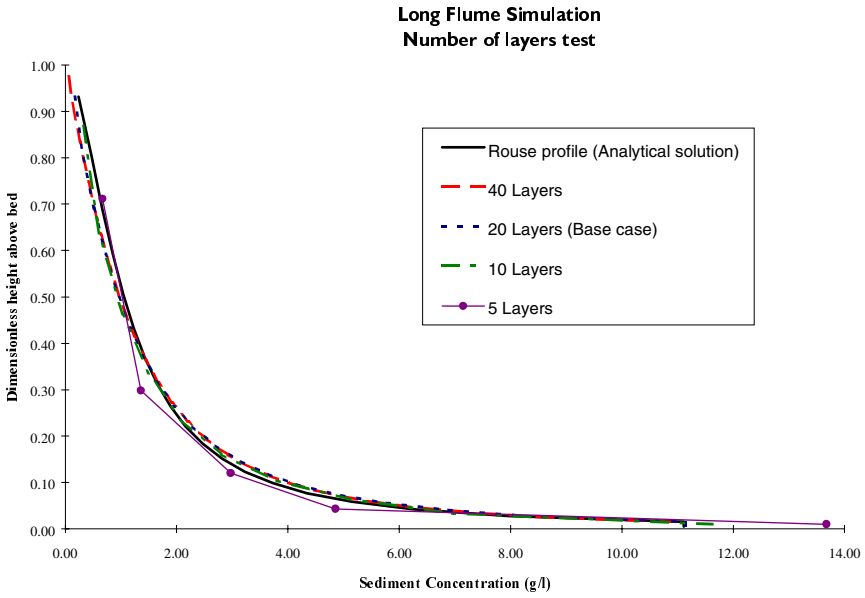


Figure 14 - Influence of number of layers on sediment concentration profile

the layers. The results of this test are presented in Figure 15 which demonstrates that, within reasonable limits, the model is not sensitive to a change in the bottom layer thickness. However the test also demonstrates that the vertical spacing of the layers is important, as 20 evenly spaced layers (5% of the water depth each) introduces a significant error into the solution. On the basis of this test it is recommended that a logarithmic distribution of layer thickness is used for simulations which include non-cohesive sediment transport.

A test is also carried out to check the sensitivity of the model to the use of the central difference or first-order upwind particle settling scheme. This test is required as we find that the upwind particle settling scheme is more stable than the central difference scheme in

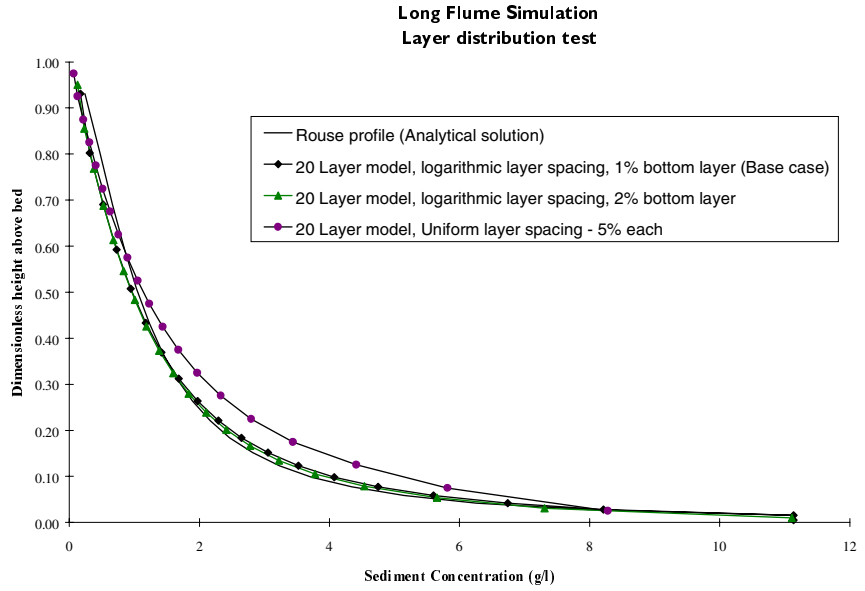


Figure 15 - Influence of distribution of layer thickness on sediment concentration profile

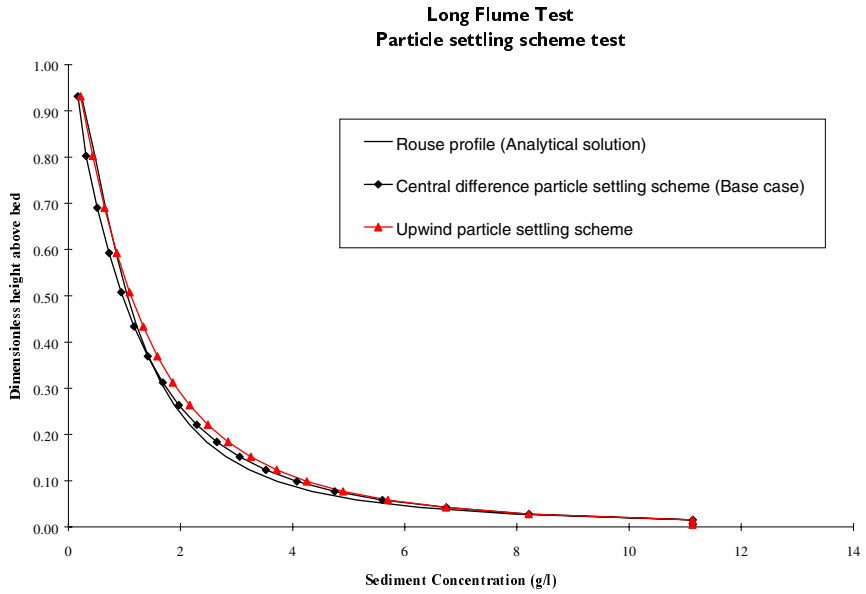


Figure 16 - Influence of particle settling scheme

situations where little upward diffusion of sediment particles exists. Figure 16 shows the result of this test. It is apparent that use of the upwind particle settling scheme introduces little error (approximately a 10% increase in the total suspended sediment transport) in the case of 20 layers; this is also confirmed in the case of fewer layers. Because of the increased stability offered by the upwind settling scheme, at the expense of relatively little error, the upwind particle settling scheme is adopted as the standard for all further simulations.

Finally, we present a test run using the concentration bottom boundary condition (version

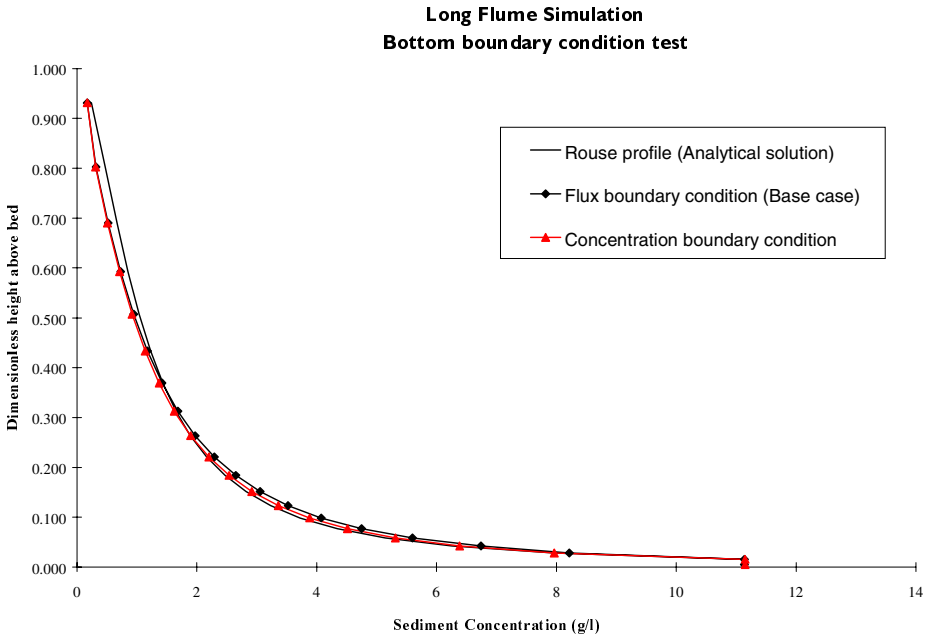


Figure 17 - Influence of flux or concentration bottom boundary condition

03.05.006) of the code which produces a virtually identical equilibrium sediment concentration profile to that produced using the flux bottom boundary condition (refer to Figure 17 below). This result is reassuring, however the adjustment distance required to reach equilibrium conditions is a more stringent test of the operation of the bottom boundary condition. The results of this test are discussed in section 4.2.3 below.

4.2.2 Test 2 - Simple sediment settling basin

Model set-up

The settling basin test is performed using exactly the same model as the long flume experiment described above except that in this case the flume is positioned exactly level, a zero discharge is specified at the left hand boundary, and a constant 5m deep water level at the right hand boundary. This arrangement results in a perfectly still basin of water. However, in this case, the initial conditions specify that the water everywhere carries and initial sediment load of 2Kg/m^3. The simulation is then run until all the suspended sediment accumulates at the bed.

Computational results

Figure 18 shows the resulting build up of sediment at a point near the middle of the flume. The results of several runs are included to show the influence of choosing different time-step values for the computation. It can be seen that the standard time-step used in the base run in the equilibrium tests above results in a reasonably rough approximation to the theoretical accumulation of sediment at the bed. This is not a surprising result as, in theory, the entire suspended sediment load should be deposited in approximately 1½ time-steps.

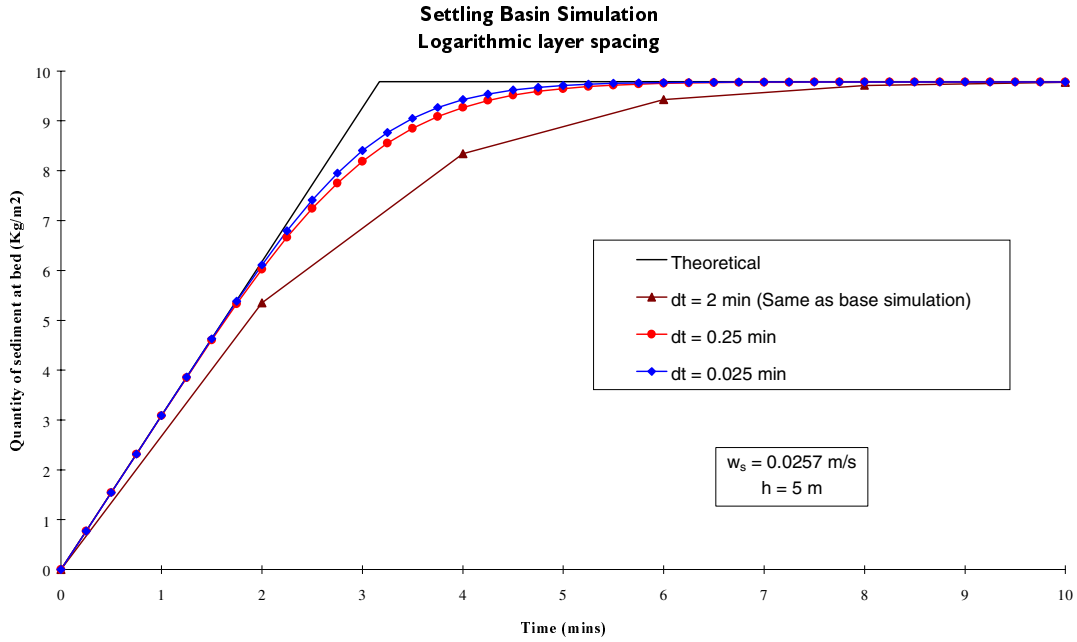


Figure 18 - Influence of time-step on settling test

Figure 18 clearly shows that (substantially) reducing the computational time-step does improve the accuracy of the solution. Another point clearly illustrated by the results is that although a smaller time-step may improve the calculation of the rate at which the sediment accumulates, the *total* amount of sediment accumulated after a long time is always accurately calculated. This is taken to be a positive indicator for the conservative nature of the computational scheme used within the DELFT3D-FLOW module.

We would like to emphasise at this point that only the sediment held in the active computational layers above van Rijn's reference height is available to exchange with the bed. Sediment located below van Rijn's reference height is regarded as bed-load transport, and neglected from the suspended sediment transport calculations. In this settling basin example the two lowest computational layers are regarded as falling in the bed-load region, and are ignored. These two layers make up 2.15% of the total water depth (0.1075m). Thus in this case, the expected sediment load accumulated at the bottom is 4.8925m x 2Kg/m^3 = 9.785 Kg /m^2. This compares very well with the 9.779 Kg/m^2 calculated by DELFT3D (0.06% error).

The results of the settling test performed using a model with a vertical grid consisting of 20 evenly spaced layers (5% of the water depth each) are also presented (Figure 19). This figure clearly indicates that changing the vertical grid spacing makes little difference to the rate of accumulation of sediment at the bed.

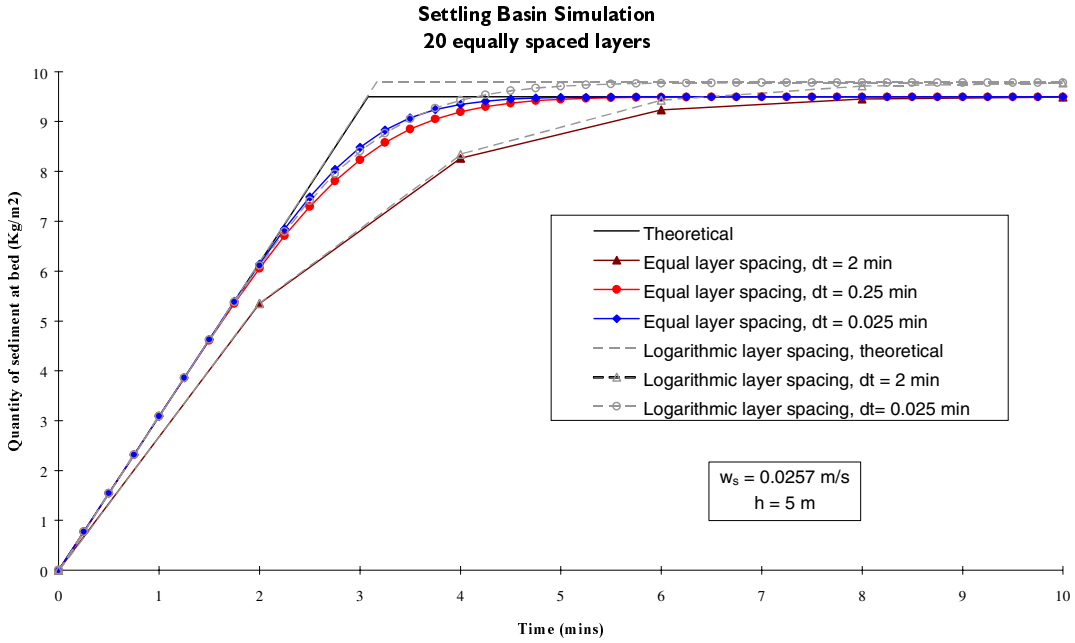


Figure 19 - Settling test with 20 evenly spaced layers

Note that the total amount of sediment accumulated in this test is slightly less than in the last. In this case the bottom layer (5% of the water depth) is regarded as containing bed load and is neglected from the suspended sediment transport calculations. In the previous case the neglected layers made up just 2.15% of the water depth.

4.2.3 Test 3 - Longitudinal development of equilibrium sediment concentration profiles

This test simulates a classic sediment transport problem, the adjustment of sediment concentration profiles in a horizontally uniform flow that is sediment free at the inlet. The general arrangement of this problem is shown in Figure 20.

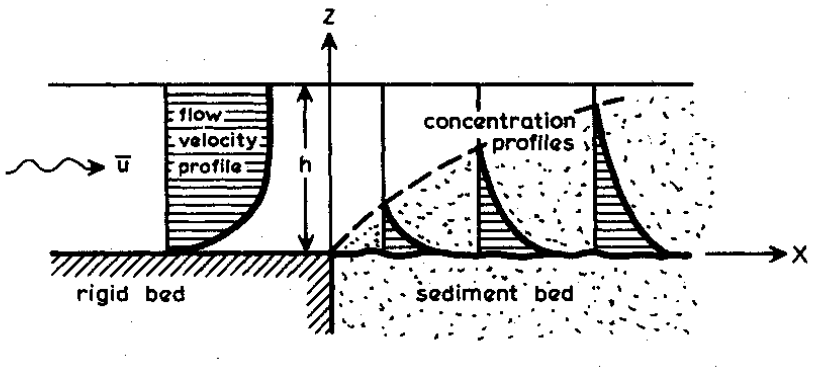


Figure 20 - Sediment concentration profile development in an initially clear flow (Source: van Rijn 1987)

In order to test the sensitivity of the DELFT3D numerical scheme to a range of numerical parameters a number of simulations are performed and compared with the results of test runs performed by van Rijn (1985) using the SUTRENCH model.

Model set-up

In order to directly compare the results of the DELFT3D simulations with van Rijn's results the simulations are carried out using the following parameters:

$\overline{u}$ =	1.5 m/s
h =	1.0 m
C =	47 $m^{0.5}$/s (implies K_s = 0.024 m)
I =	0.001
d_{50} =	143 micron (implies w_s = 0.015 m/s)
Δx =	1 m
n =	10 layers (logarithmic spacing)
Δt =	0.025 minutes
Simulated time =	50 minutes
Bottom updating	OFF (by setting [MORFAC] = 0)
[**AKSFAC**]	= 1.0 (implies $a = K_s$ = 0.024 m)

Computational results

Figure 21 presents the results of two DELFT3D simulations, the results of L.C. van Rijn's computations using the SUTRENCH 2D computer program, and the depth-averaged analytical solution of Galappatti (1983). It can be seen that changing between the flux bottom boundary condition and the concentration bottom boundary condition makes little

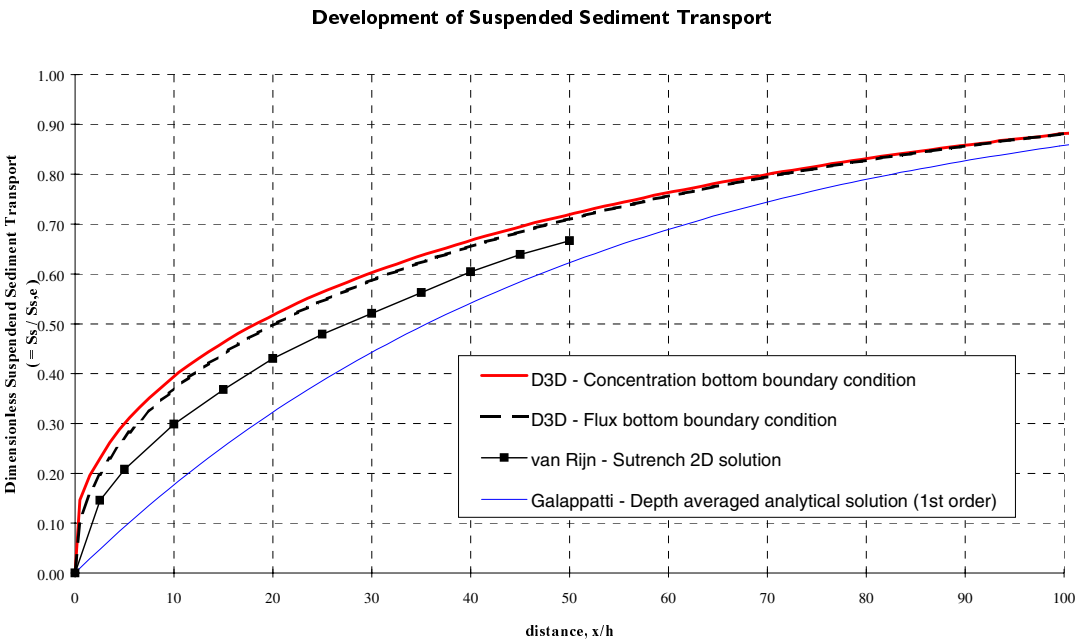


Figure 21 - Increase in suspended sediment transport with distance

difference to the solution, and that both these solutions are in reasonable agreement with the results of van Rijn (1985), although DELFT3D predicts a somewhat more rapid development of the suspended sediment transport. The depth-averaged analytical solution of Galappatti predicts a significantly slower adaptation of the sediment transport rate to the flow conditions than either the DELFT3D solutions or the SUTRENCH solution of van Rijn. This result is not unexpected, as the error in the analytical solution increases with the difference between the local mean concentration and the equilibrium mean concentration (Galappatti, 1983). The error in the analytical solution is therefore most pronounced near the inflow boundary. This is clearly consistent with the results presented in Figure 21.

Further simulations are performed to show the sensitivity of the DELFT3D solution to a number of the selected numerical parameters, the results of these simulations are presented in Figure 22. The following results are of particular interest:

1. Changing from the central difference to the upwind particle settling scheme (which is used as the standard in all other simulations as discussed in section 4.2.1 above) increases the total suspended sediment transport rate by approximately 10% near the downstream end of the flume. It makes little difference to the early development of the suspended sediment load. The reason for this effect can be clearly seen in Figure 23 which shows the higher sediment concentrations calculated high in the water column in the case of the upwind particle settling scheme. This is a numerical error caused directly by the first order upwind approximation used by this settling scheme.

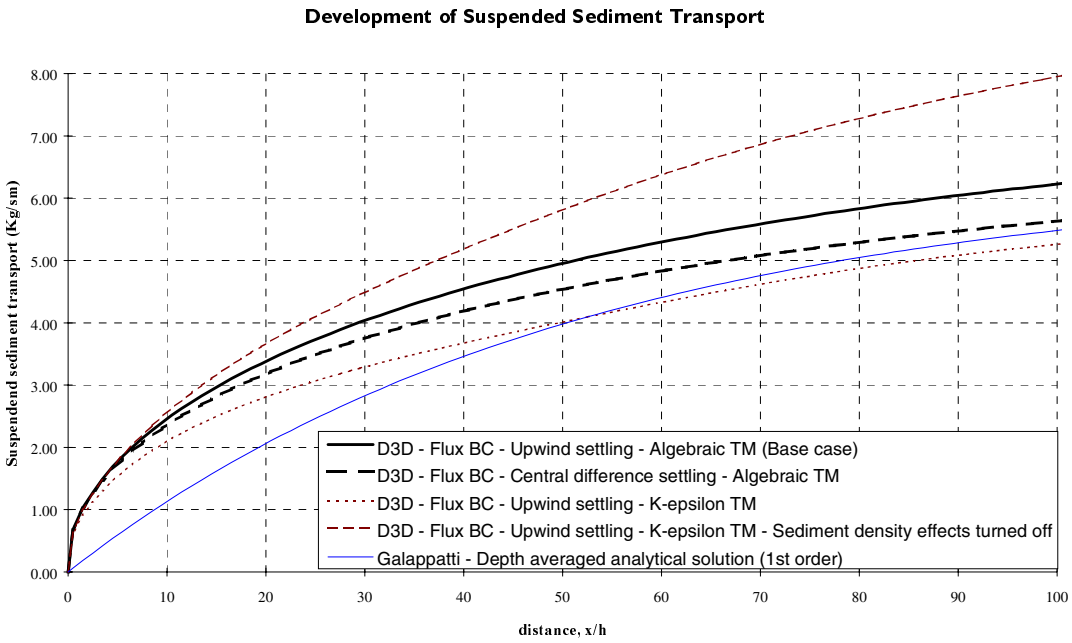


Figure 22 - Impact of changing numerical parameters

2. Changing from the algebraic to the k-epsilon turbulence model causes a significant (approximately 20%) decrease in the calculated suspended sediment transport rate, this effect is more or less constant along the length of the flume. Inspection of Figure 23 shows that this decrease is predominantly due to a decrease in the computed near-bed flow velocity (for a constant discharge). This results in a decrease of the reference concentration (from about 19 to 18 Kg/m^3) and affects the entire sediment concentration profile. The alteration of the velocity profile caused by the use of the different

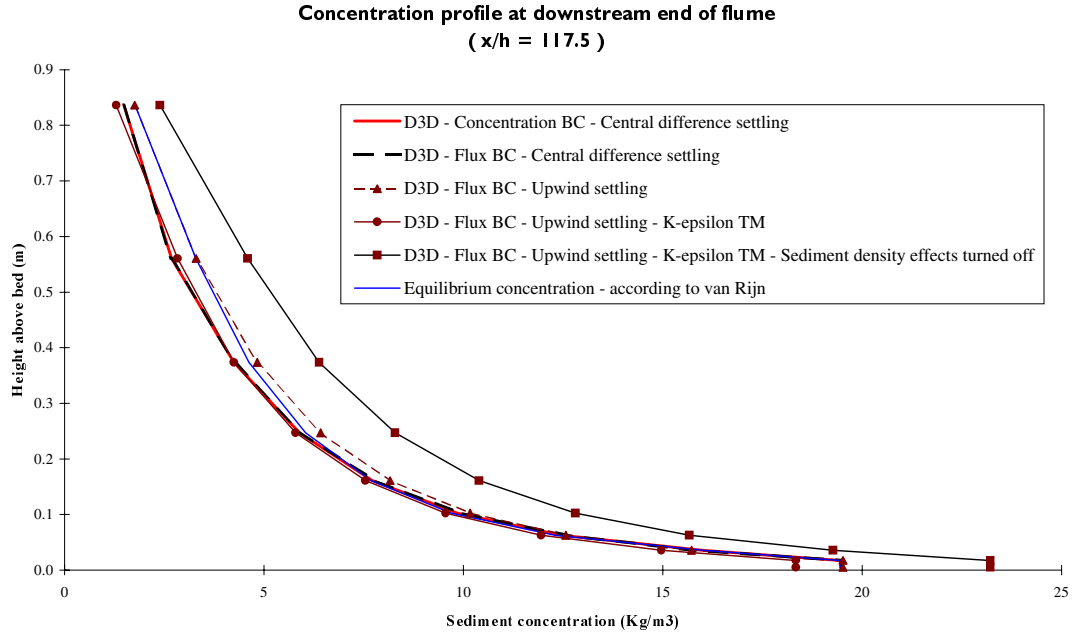


Figure 23 - Sediment concentration profiles calculated at near-equilibrium conditions

turbulence models is shown in Figure 24. In the case of the k-epsilon model the retardation of the flow velocities near the bed (where concentrations are high) and increase in flow velocities near the surface (where concentrations are low) also serves to decrease the computed suspended sediment transport rate. We believe that the effect of changing from the algebraic to k-epsilon turbulence model is more pronounced in this test than the results of the tests discussed in section 4.2.1 above because the sediment concentrations calculated in this test are significantly higher. This hypothesis is confirmed by an additional simulation which is performed using the k-epsilon turbulence model, but with the density effects of the sediment removed from the calculations.

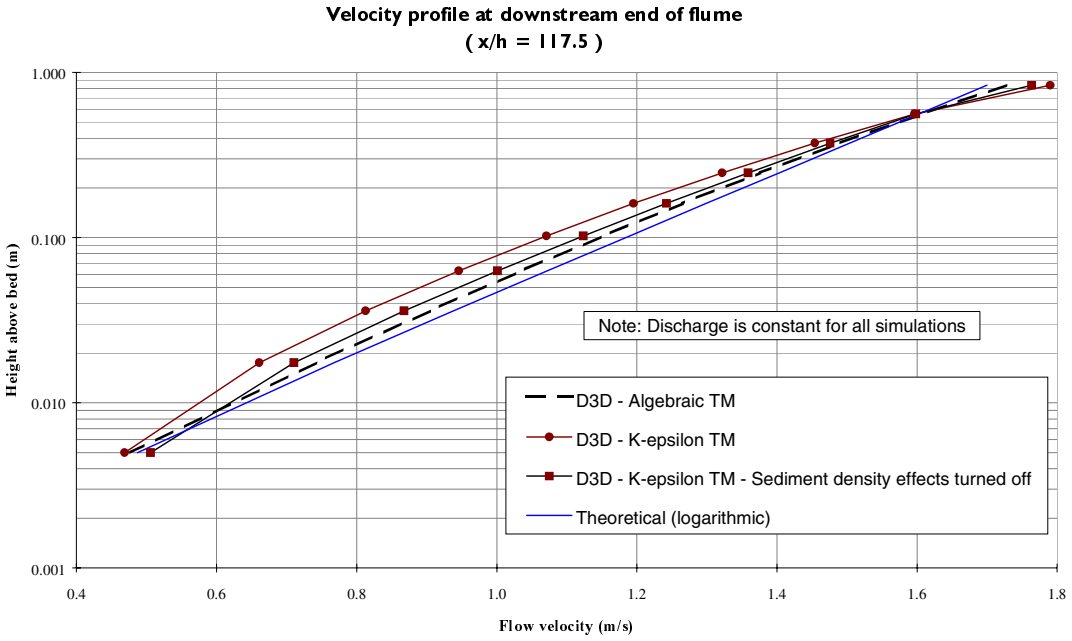


Figure 24 - Velocity profiles calculated at near-equilibrium conditions

3. Use of the k-epsilon turbulence model with the sediment density effects turned off results in a dramatic increase in the computed suspended sediment transport rate. The reasons for this increase are the exact opposite of those described in point 2 above and may also be observed in Figure 23 and Figure 24. It is clear that the k-epsilon turbulence model is very (possibly overly) sensitive to vertical density gradients, and careful re-calibration of this model may be required by future researchers.

Finally, the gradual development of the sediment concentration profile downstream of the inlet is shown in Figure 25 below. This figure confirms the gradual upward net diffusion of the sediment. It is clear that the suspended sediment load has not completely reached the expected equilibrium value by the distance x/h = 100.

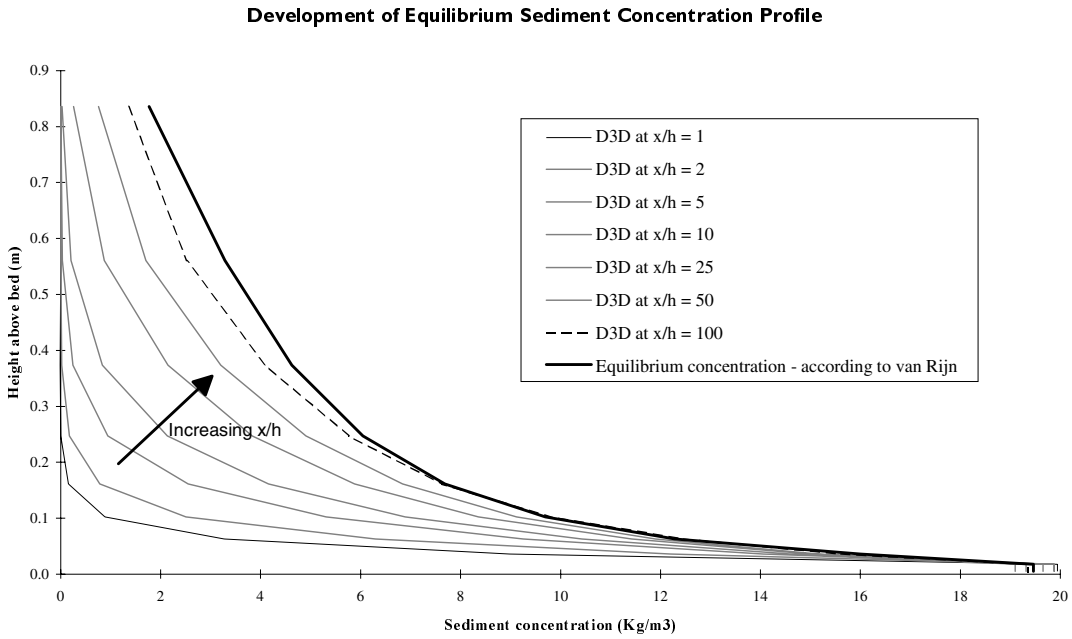


Figure 25 - Progressive development of sediment concentration profile with distance (Algebraic turb. model)

4.2.4 Test 4 - Trench migration

This test recreates an experiment conducted by L.C. van Rijn at Delft Hydraulics' laboratories as part of the validation of the SUTRENCH 2D computer model (van Rijn 1985). The experiment was conducted as follows (adapted from van Rijn 1987).

The experiment consisted of measuring flow velocity profiles, sediment concentration profiles and the bed level changes of a trench in a flume (length = 30m, width = 0.5m, depth = 0.7m). The trench was excavated in the sediment bed which had a thickness of about 0.2m. The characteristic diameters of the sediment material were d_{50} = 160µm, d_{90} = 200µm.

To maintain equilibrium conditions upstream of the trench (no scour or deposition), sediment of the same size and composition was supplied at a constant rate of 0.04 Kg/sm. The water depth and mean flow velocity upstream of the trench was held constant (h_0 = 0.39m, u_0 = 0.51m/s). The initial trench dimensions are shown in Figure 26.

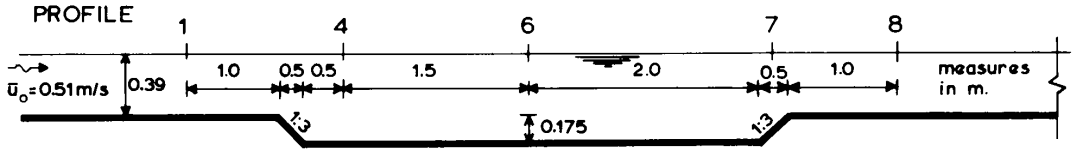


Figure 26 - Trench migration experiment (source : van Rijn 1987)

At each monitoring location the flow velocity and sediment concentration profile were measured simultaneously. Based on the measured velocities and sediment concentrations upstream of the trench, the equilibrium suspended sediment transport rate was found to be $s_{s,0}$ = 0.03 Kg/sm. Since the total-load transport rate was $s_{t,0}$ = 0.04 kg/sm (feeding rate), the bed-load transport rate was $s_{b,0}$ = 0.01 kg/sm. Based on an analysis of suspended sediment samples van Rijn determined that the size of the suspended sediment particles was 160µm near the bed and about 120µm near the water surface.

Model set-up

The flume is modelled over a length of 30m, with a mobile sand bed on a very slight slope (0.012m fall in 30m) so that the water depth and depth averaged velocity are equal at either end of the flume. The upstream boundary is simulated using a constant water level for the flow. The flow enters with an equilibrium sediment concentration profile (as described in sections 3.4.2 and 3.5 above). At the downstream boundary a constant discharge is prescribed, with a logarithmic velocity profile specified. Other important parameters used in this run are listed below:

Sediment diameter (d_{50})	= 140µm
Sediment density (ρ_s)	= 2650 Kg/m3
Initial thickness of sediment bed	= 0.5m

Longitudinal grid size (Δx)	= 0.1m
Vertical layer arrangement varies, see below	
Time-step (Δt)	= 0.001min
Simulation time before start of morphological computations	= 25min
Duration of morphological computations (flow time)	= 5min
Morphological time-scale factor (MORFAC)	= 180
Duration of morphological computations (morphological time)	= 900min (= 15 hours)
Bottom roughness (k_s)	= 0.025m
Reference height factor (AKSFAC)	= 0.5 (so a = 0.0125m)

Computational results

The velocity and sediment concentration profiles computed by version 03.05.007 (flux bottom boundary condition) of the DELFT3D-FLOW module are presented in Figure 27. This model run is performed using the algebraic turbulence model and 10 computational layers (spaced logarithmically, based on a 4% bottom layer thickness). Figure 27 shows the velocity and sediment concentration profiles calculated once the model has reached a steady state, but before the bottom elevations begin to change. This situation is equivalent to the time t = 0 in van Rijn's experiment. The concentrations and velocities measured by van Rijn at t = 0 are also presented in Figure 27.

Generally, the computed velocities and concentrations presented in Figure 27 appear rather good. Note that larger views of these profiles are located in appendix C. At location 1 (just

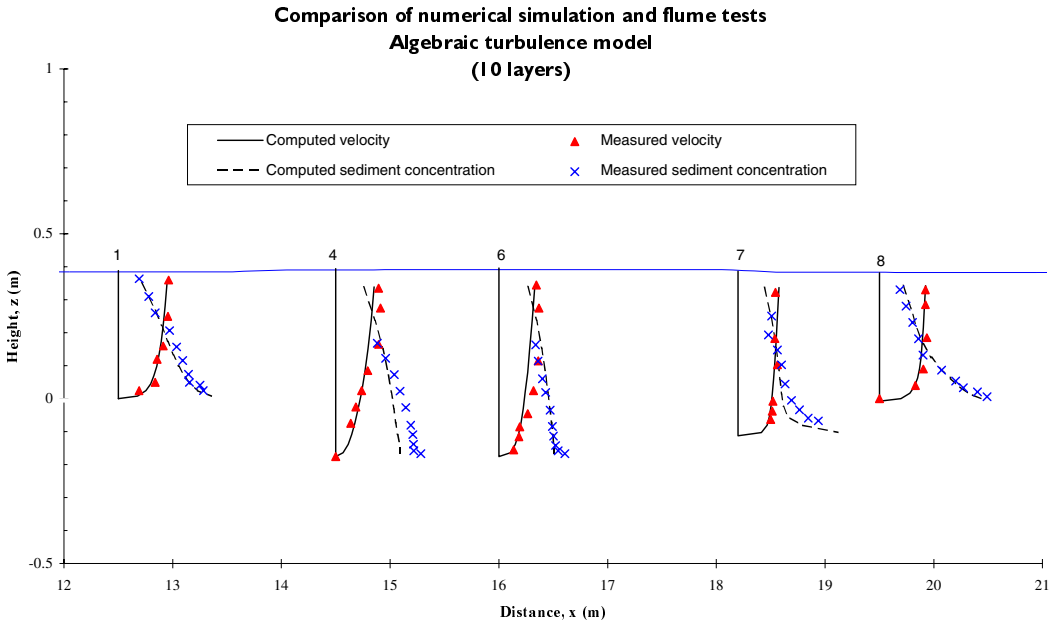


Figure 27 - Velocity and sediment concentration profiles calculated with the algebraic turbulence model

upstream of the trench) both the velocity and concentration profile appear to be very close to the measurements; this confirms the findings of section 4.2.1 above. At location 4 (in the area of decelerating flow) the computed velocity profile shows the correct behaviour, but is not altogether accurate; especially at around mid-depth. It is possible that this is due to the fact that the flow solver in DELFT3D is based on the (shallow water) assumption that the vertical momentum of the flow may be neglected. This assumption may be being violated in this area of rapidly expanding flow. It also appears that the vertical turbulent mixing of both fluid momentum and sediment is rather too high at this location. At location 6 (in the centre of the trench) the velocity profile suffers from a similar problem to that described for location 4, indeed it is likely that the cause of the discrepancy at this location is the error introduced at the point of flow expansion as it can be seen that the measured bulge in the velocity profile is spread over a greater range of depths at location 6. The computed sediment concentration profile at location 6 is good near the bottom and rather over-estimated near the top. This also tends to indicate rather too much turbulent mixing occurring in the middle of the flow. At location 7 (in the region of flow acceleration) the velocity profile agrees with the experimental results very well indeed, the concentration profile is not responding so well however. It appears that at this point there is insufficient turbulent mixing calculated near the bed to lift the required amount of sediment into suspension. At location 8 (downstream of the trench) both the computed velocities and concentrations agree very well with the experimental measurements.

In summary, it appears that the velocity and sediment concentration profiles computed using the algebraic turbulence model behave reasonably over a rather abrupt trench. The weakest link seems to be the algebraic turbulence model which appears to over-estimate the turbulent mixing around mid-flow height in the zone of decelerating flow, and underestimate the turbulent mixing near the bed in the acceleration zone.

With this in mind we now present the results of the same simulation performed using the k-epsilon turbulence model. The computer simulation performed using the k-epsilon turbulence model is identical to that described above, except that we find that the number of computational layers must be increased in order for the deposition pattern to behave sensibly in the region immediately downstream of location 7. The sensitivity of the k-epsilon model to the number of computational layers is discussed in more depth later in this section.

Figure 28 presents the computed velocity and sediment concentration profiles in a manner identical to that discussed above (again enlargements of these profiles are located in appendix C). It can be seen that once again the computed velocity and sediment concentration profiles at location 1 agree very well with the measured values, in fact the sediment concentration profile computed using the k-epsilon turbulence model fits the experimental results rather better than that computed using the algebraic turbulence model. At location 4 (deceleration area) both the velocity profile and the sediment concentration profile fit the experimental results considerably better. There are still slight indications of excessive mixing at around the mid-height of the flow, although this discrepancy could also be partially due to the shallow water assumption discussed above. Neither the velocity or the sediment concentration profile at location 6 (mid trench) seem to be much improved over those discussed above, once again the computed turbulent mixing appears to be much

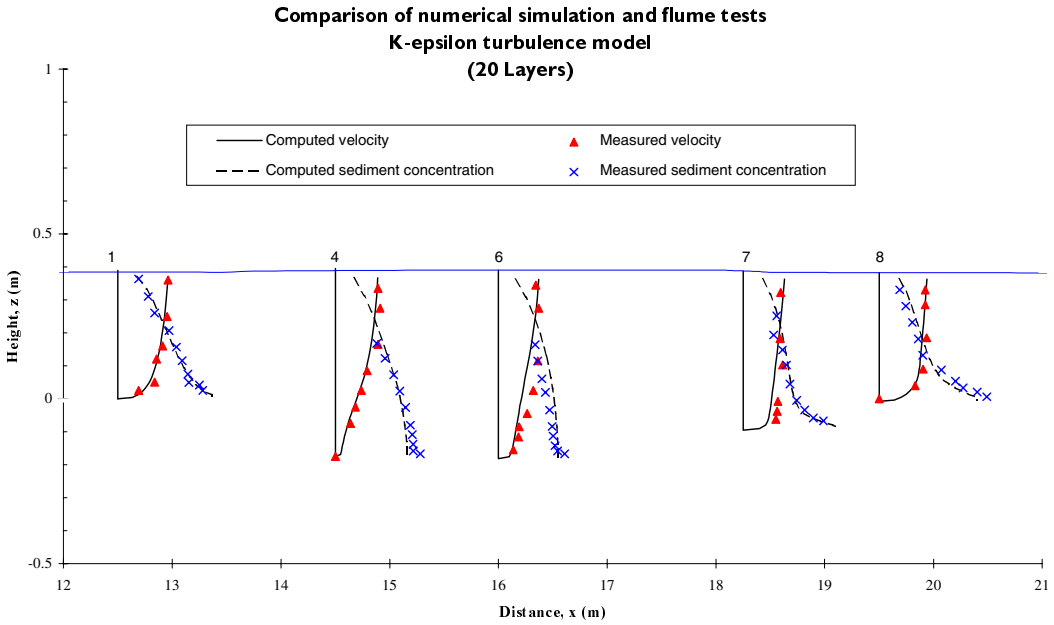


Figure 28 - Velocity and sediment concentration profiles calculated with the k-epsilon turbulence model

too high in the mid-depth region. At location 7 (acceleration) the concentration profile is greatly improved, and matches the measured values almost perfectly. The computed velocity profile is a little low near the bed. One possible explanation for this is that the graph only shows the computed *horizontal component* of the velocity vector, as this location is on a one in three slope the total velocity magnitude near the bed is approximately 5% greater than the horizontal component alone. It is not clear to what extent the experimental measurements would have included the vertical velocity component. At location 8 (downstream) the computed velocity profile is very close to the measured profile, the concentration profile is slightly low in the bottom 25% of the flow depth. Further tests show that the velocity and concentration profiles at location 7 can be significantly improved by turning on the usually neglected $\frac{\partial u}{\partial x}$, $\frac{\partial u}{\partial y}$, $\frac{\partial v}{\partial x}$, and $\frac{\partial v}{\partial y}$ production terms in the k-epsilon turbulence model (by turning on wall friction and setting the wall roughness to a very small value) see appendix C for the computed profiles. It is clear that in the region of accelerating flow the first of these terms is likely to be producing significant quantities of turbulent kinetic energy. Interested readers are referred to the DELFT3D-FLOW user manual for further details of the k-epsilon turbulence model.

Figure 29 and Figure 30 present the morphological changes calculated using the algebraic and k-epsilon turbulence models respectively, superimposed on these figures are the bed levels computed by van Rijn at a morphological time of 15 hours. In his report on the validation of the SUTRENCH-2D computer model, van Rijn showed that his computed bed profile for the case of suspended sediment transport = 75% of total sediment transport ($s_s = 0.75s_t$) was very close to the measured bed level after 15 hours. Unfortunately we are unable to make such a direct comparison, as the DELFT3D-FLOW module does not yet account for any bed-load transport. However van Rijn also provided two further computed bed profiles, these are for the cases of $s_s = 0.9s_t$ and $s_s = 0.6s_t$.

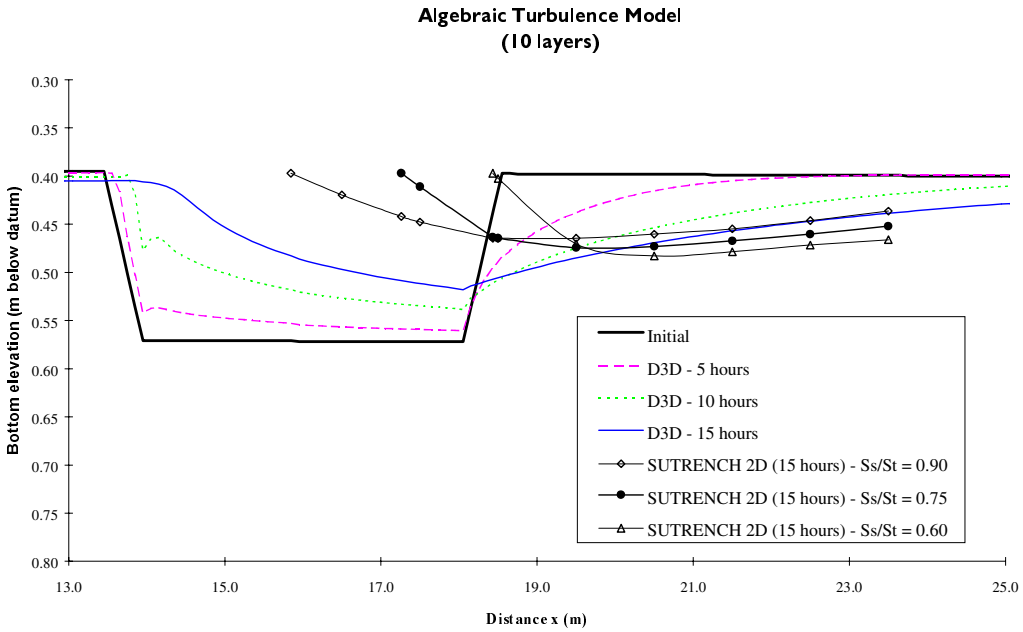


Figure 29 - Computed bed evolution with algebraic turbulence model

With this in mind, inspection of the bed profiles calculated by DELFT3D shows that both simulations produce reasonable results insofar as both models predict a trapping rate that is obviously in the correct order of magnitude and the trench slopes appear to have roughly the correct slope after 15 hours. Although the quantity of sediment deposited in the trench is significantly greater for the k-epsilon model than the algebraic model, it is difficult to say which of these simulations is more realistic as it is hard to judge what proportion of the bed-load transport will be caught by the trench. A proper validation of the computed morphological development will only be possible once bed-load transport is included.

It is worth pointing out that in the DELFT3D simulations the few available sediment transport calibration parameters have been left at their default settings, this was not the case for van Rijn's SUTRENCH-2D runs where he reports that he adjusted the sediment pickup function to match the measured sediment concentration profile.

Figure 31 shows the computed total suspended transport rates for the two DELFT3D simulations and van Rijn's SUTRENCH-2D computation. It is clear in this figure that at this initial stage (t=0) there is good agreement between all three simulations about the decrease in sediment transport in the deceleration zone. However the two DELFT3D

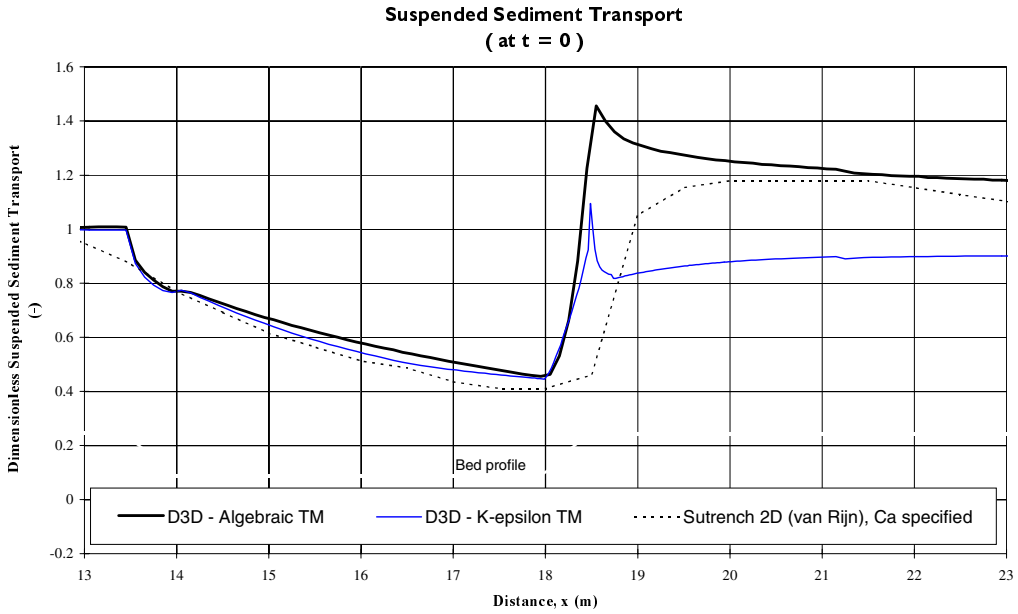


Figure 31 - Comparison of computed suspended sediment transport rates

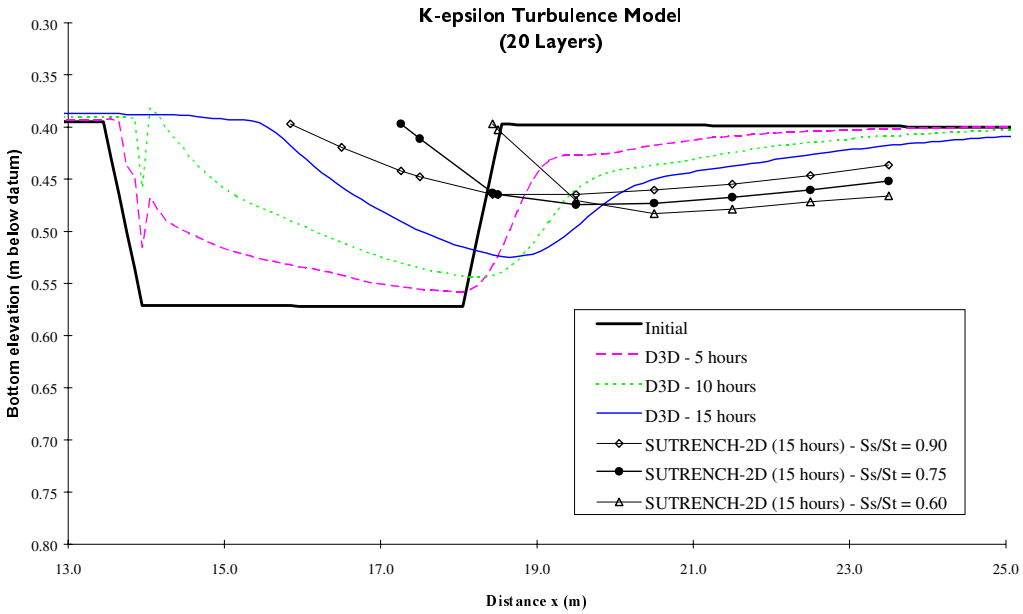


Figure 30 - Computed bed evolution with k-epsilon turbulence model

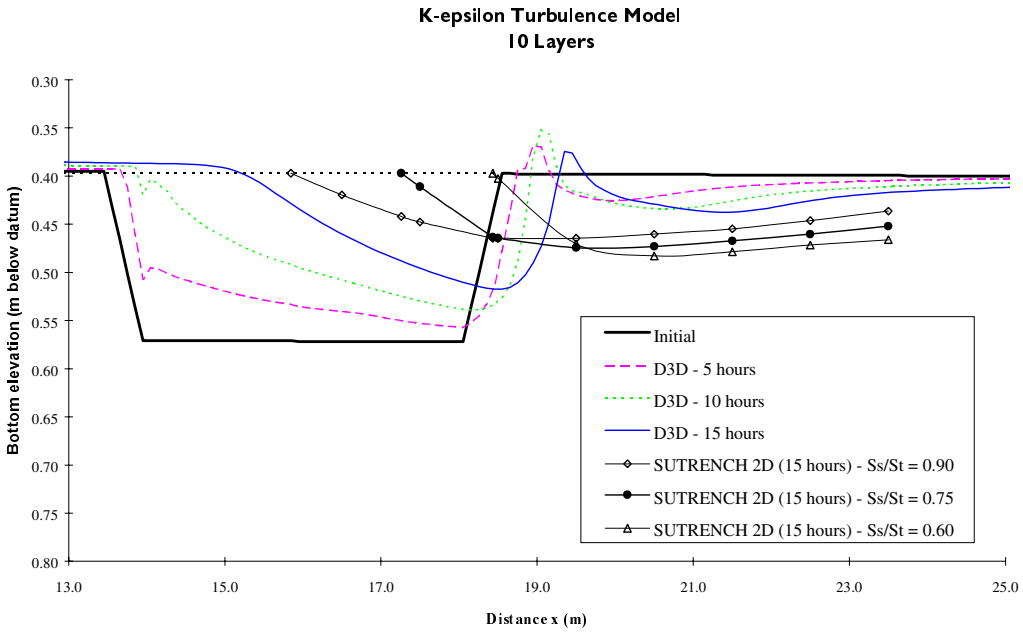


Figure 32 - Computed bed evolution with k-epsilon turbulence model and 10 computational layers

simulations show a much more rapid response of the suspended sediment transport to the accelerating flow at the downstream side of the trench, and that the two DELFT3D simulations disagree somewhat regarding the peak value of the sediment transport at the top of the trench slope. We point out, however, that in the morphological computations this situation will only exist for a rather short time, as the abrupt changes of grade are rapidly smoothed by the flow and, that as the morphology develops, all three model solutions will tend to converge.

Finally, we present the results of an investigation into the sensitivity of the k-epsilon turbulence model to the number of computational layers. Figure 32 shows the bed evolution predicted by DELFT3D using the k-epsilon turbulence model and 10 computational layers. While it would be tempting to regard the bottom undulations that appear downstream of the trench as sand-waves produced by the k-epsilon model, it seems more likely that they are actually caused by some form of numerical instability or feedback loop that is particular to the k-epsilon turbulence model. This is demonstrated further by Figure 33 which shows that the computed bed profile does reach a stable solution as long as at least 20 layers are used. A similar result has been identified by another researcher working on including three-dimensional wave-current interaction effects into the k-epsilon turbulence model.

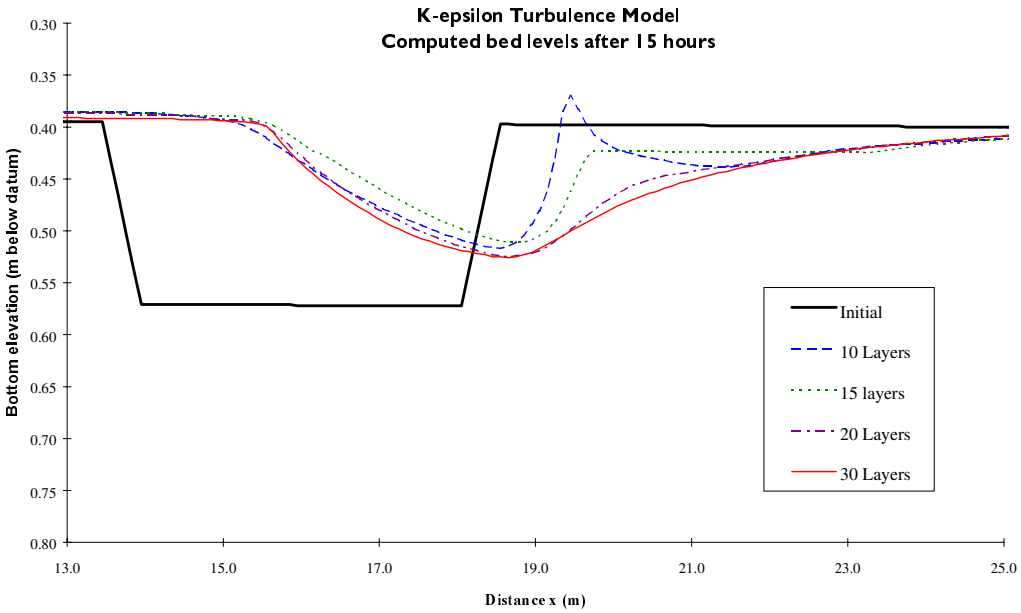


Figure 33 - Influence of number of layers on computed bed levels (k-epsilon turbulence model)

4.3 Qualitative Current-only Validation

The following sections are included to present some of the first results of applying the modified DELFT3D-FLOW module to truly three-dimensional situations. Section 4.3.1 presents a relatively simple hypothetical situation that was first used by L.C. van Rijn to test the SUTRENCH-3D computer program. Although van Rijn (1987) does present some of the results of his investigation, we have not attempted to compare the results of the DELFT3D simulation with those of van Rijn quantitatively. For this simulation we are content to observe the behaviour of the model in a qualitative manner.

Section 4.3.2 presents the first attempt at applying the modified DELFT3D-FLOW module to a real-life situation. The Westerschelde is a large estuary in the south of The Netherlands that has high recreational and natural value, as well as serving as the access channel to the Belgian port of Antwerp. As such it is an area that deserves careful analysis before any management decisions are made. Areas of the Westerschelde are rather difficult to model, however, as the twisting tidal channels produce spiral flow effects near the bends and the fresh water, delivered into the estuary by the Schelde river, produces significant salinity stratification effects. In March 2000 WL | DELFT HYDRAULICS completed a study on the ability to model the secondary flow (spiral flow) and three-dimensional sediment transport in such a complicated area. The study included the use of the modified DELFT3D-FLOW module, and a small selection of the results are presented below.

4.3.1 Long groyne simulation

Model set-up

As mentioned above, this test was first performed by L.C. van Rijn (1987) and full details of the model dimensions and settings may be found in his thesis. Our simulation closely follows that of van Rijn and the overall geometry is as per van Rijn's diagram, see Figure 34.

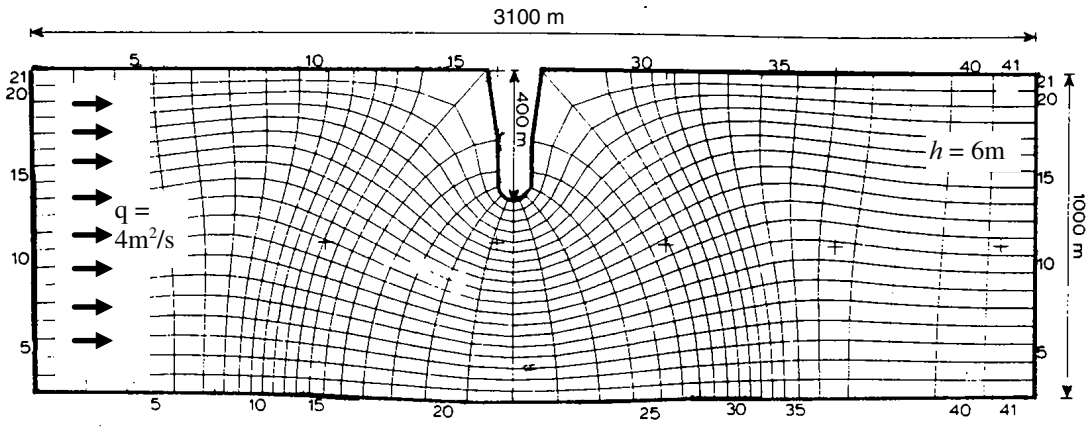


Figure 34 - Lay out of long groyne simulation (Source: van Rijn 1987)

Other key parameters required to set up our model are:

Water depth at outlet		= 6m
Discharge per unit width	q	= $4m^2/s$
Bed roughness height	k_s	= 0.25m
Sediment diameter	d_{50}	= 200µm
Initial bed thickness		= 20m
No. layers (logarithmic)		= 10
Bottom layer thickness		= 5%
time-step	Δt	= 0.5min
Turbulence model		= Algebraic

The simulation was run for a period of two hours to allow the hydrodynamic computations to stabilise before starting the updating of the bottom. The simulation then continued for another eight hours, using a morphological time-scale factor (MORFAC) of 1000. This is equivalent to simulating the morphological developments taking place over 333 days.

Computational results

Figure 35 shows the initial morphological developments that take place near the head of the groyne. It is clear that the flow is being forced to accelerate through the restriction, and this is causing erosion upstream and adjacent to the head of the groyne. Downstream of the groyne the eroded material is deposited once again.

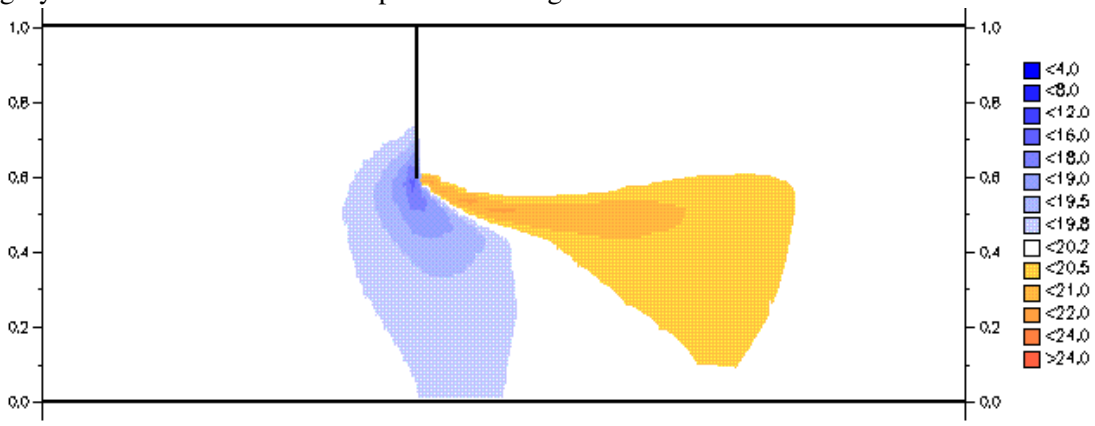


Figure 35 - Initial developments near the tip of the groyne

Figure 36 shows the morphological developments approximately half way through the simulation, after approximately 150 morphological days. It is clear that a considerable scour hole (already more than 12m deep) is developing at the head of the groyne, and a considerable ridge of deposited material (up to 4m high) is accumulating downstream. Of great interest (and more clearly seen in an animation) is the manner in which both the erosion and deposition areas are steadily spreading outwards, and in particular sideways. This indicates that the changing bathymetry is having a significant feedback effect on the flow. The sideways (upwards in this figure) movement of the deposition area is of particular interest as it is not usually seen in two-dimensional (depth averaged) simulations. It is likely that this movement is due to a spiral motion of the flow crossing the deposition area, as

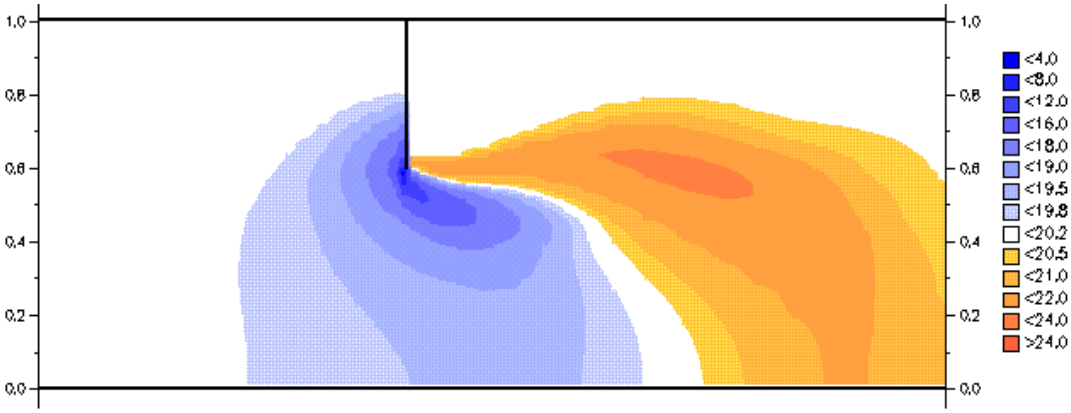


Figure 36 - Morphological changes computed near the mid point of the simulation duration

careful inspection of velocity vectors in this vicinity shows a significant deviation in direction between the top and bottom of the flow.

Figure 37 shows the morphological changes predicted at the end of the simulation. The rate of change in the bathymetry is very slow by this point. It can be seen that the scour hole at the tip of the groyne has deepened to more than 16m, and also extended a reasonable distance downstream of the tip. The deposition ridge has also lengthened in the direction of the flow and moved further sideways, but has not got any higher than four metres. It is interesting to note the secondary scour hole that has developed shoreward of (above) the deposition ridge; this appears to be caused by the flow being accelerated over the deposition ridge and attempting to form a second "channel" between this ridge and the shore.

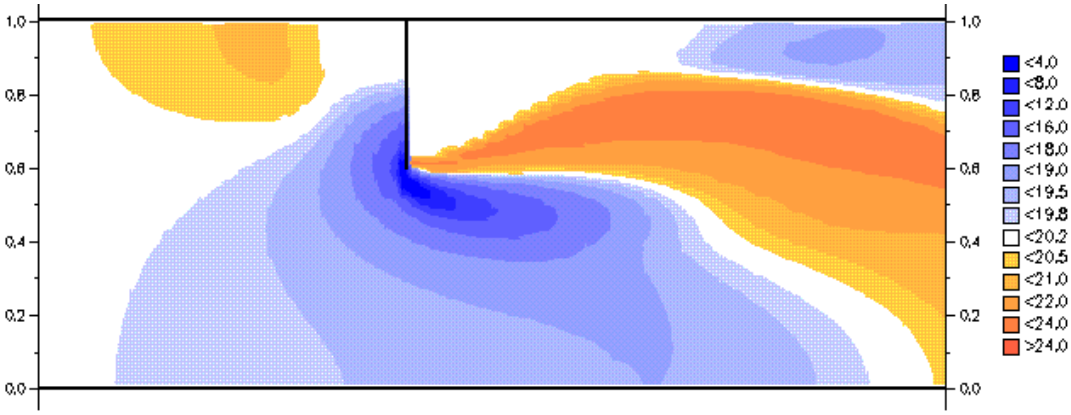


Figure 37 - Near equilibrium conditions reached at the end of the simulation

In general, the results of this simulation are very positive. The resulting morphological changes are in line with intuition and van Rijn's earlier solutions. Furthermore, the changes seem to be of a reasonable magnitude. Of particular comfort are the observations that the bathymetrical changes appear to remain smooth and computationally stable, even in the vicinity of the deep scour hole, and with the relatively high morphological time scale factor used. It is also reassuring to discover that the solution approaches a new equilibrium, as would be expected in nature, rather than oscillating to and fro between two extremes. Although not visible in the above figures, it is pleasing to note that very little deposition or erosion occurs adjacent to the model boundaries.

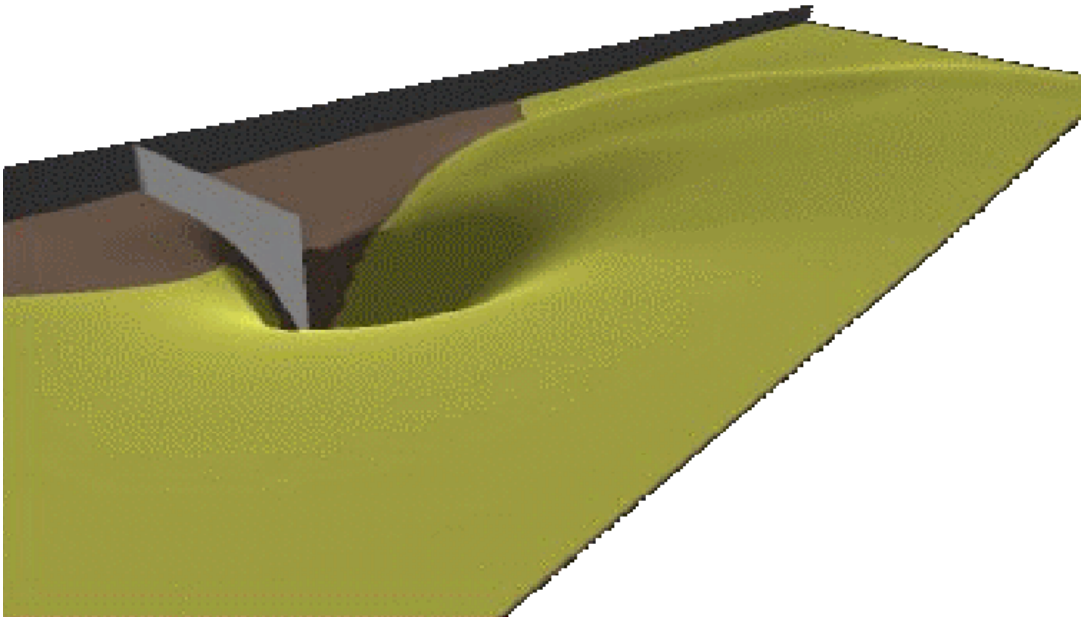


Figure 38 - Three-dimensional view of morphological changes

4.3.2 Westerschelde 3D sediment transport investigation

As mentioned in the introduction to this section, the Westerschelde is an extremely complicated hydrodynamic area to model successfully. Significant complications include:

- A large area with a complicated bathymetry and flow patterns. This requires a large, high resolution grid in order to resolve the details of the flow properly (See Figure 39, which shows about half of the required computational model).
- A large tidal range (in the order of 5m) which causes significant areas of the computational grid to repeatedly dry and flood during the simulation period.
- Significant secondary (spiral) flow occurring near the bends in the main tidal channels. This requires special modifications to convert a two-dimensional model into a "quasi three-dimensional" model if realistic results are to be obtained.
- Significant stratification effects caused by density differences between the areas of relatively fresh and salt water found in this vicinity.

In early 2000 WL | DELFT HYDRAULICS carried out an extensive investigation into the impact of the above three-dimensional phenomena on sediment transport calculations in the vicinity of the "Sill of Hansweert", as reported by Thoolen (2000, in Dutch). As part of her investigation Thoolen applied the standard DELFT3D-MOR morphodynamic simulation package (which actually carries out two-dimensional depth-averaged calculations) and included the secondary flow expressions available in that package in order to attempt to simulate the spiral flow effects. Thoolen refers to this model as a "quasi three-dimensional" model. She also carried out a simulation using the standard flow module in three-dimensional mode in order to assess the impact of salinity gradients on the computed flow field. She concluded that these effects are significant. The quasi three-dimensional model is incapable of allowing for vertical stratification effects. Thoolen then carried out three simulation runs with the modified DELFT3D-FLOW module. The first simulation had five computational layers and did not have salt included, the second had five layers and salt included, the third had ten layers and salt included. The following figures are derived from this third simulation.

A few interesting statistics regarding this computer model are:

Model area:	Approximately 25km x 5km
Grid dimensions:	265 x 236 (31,838 active points)
Time taken for 2DH simulation:	5.5 hours (workstation)
Time taken for 3D flow simulation: (7 layers)	28.5 hours (workstation)
Time taken for 3D simulation: (5 layers, including sediment)	39 hours (PC)
Time taken for 3D simulation: (10 layers, including sediment)	93 hours (PC)

Note that at WL | DELFT HYDRAULICS the UNIX workstations carry out computations significantly faster than stand-alone PCs. The modified DELFT3D-FLOW model was only

run on a PC because it was still under development. All three-dimensional simulations used the k-epsilon turbulence model.

Figure 39 shows the bathymetry of the area of interest, this is only approximately half of the area included in the complete model. Note the large shallow areas, and the meandering tidal channels; the tide floods towards the top of this figure. The two black lines indicate the cross sections along which salinity and sediment concentration profiles were extracted.

The above mentioned profiles are shown in Figure 40. This gives a good insight into the complexity of the flow situation at the time of maximum ebb flow (this flow is down the page in Figure 39).

Figure 41 shows the bathymetrical changes calculated over one morphological month using the quasi 3D model (top), and the modified DELFT3D-FLOW module (bottom). Note that the vertical stratification effect of the salinity gradients cannot be taken into account in the quasi 3D model. It is interesting to note that although the deposition (orange) and erosion (blue) patterns are similar for the two models, there are some significant differences.

Figure 42 is a plot of the difference between the two results shown in Figure 41. In this figure orange indicates that that the full 3D computation (using the modified DELFT3D-FLOW module) predicts more sedimentation (or less erosion) than the quasi 3D solution. Thoolen concludes that this difference is significant, and identifies the quasi 3D computation's inability to compute density driven flows caused by saline stratification and the fact that it can only approximate the secondary flow component in the longitudinal direction as the likely causes of this difference. Thoolen also concludes that there is a significant, although smaller, difference between the sedimentation patterns calculated using 5 and 10 computational layers in the full 3D computations. Given our experience with the k-epsilon turbulence model in complicated flow conditions this result is not surprising.

We also note that a number of "spikes" appear in the bed during the three-dimensional simulations. These spikes can be either upwards (out of the water) or downwards, however the spikes only seem to occur in shallow areas, or around the boundaries of the model. The number of spikes increases when the number of computational layers is increased to 10. We believe that the creation of spikes may be related to the drying and flooding of computational cells. This requires further investigation.

Generally we are very pleased with the results of this first attempt at real-life three-dimensional sediment transport modelling. The results produced by the modified DELFT3D-FLOW module are similar enough to the results of a state-of-the-art two-dimensional morphological model to give us confidence that they are sensible. However they are different enough, especially in complex flow areas, to give us hope that modelling sediment transport in three dimensions may well lead to more accurate and reliable morphological predictions. It is clear however that a lot more water will have to go under the bridge, or rather down the flume perhaps, before we can be sure that this, or any, three-dimensional sediment transport model is thoroughly validated.

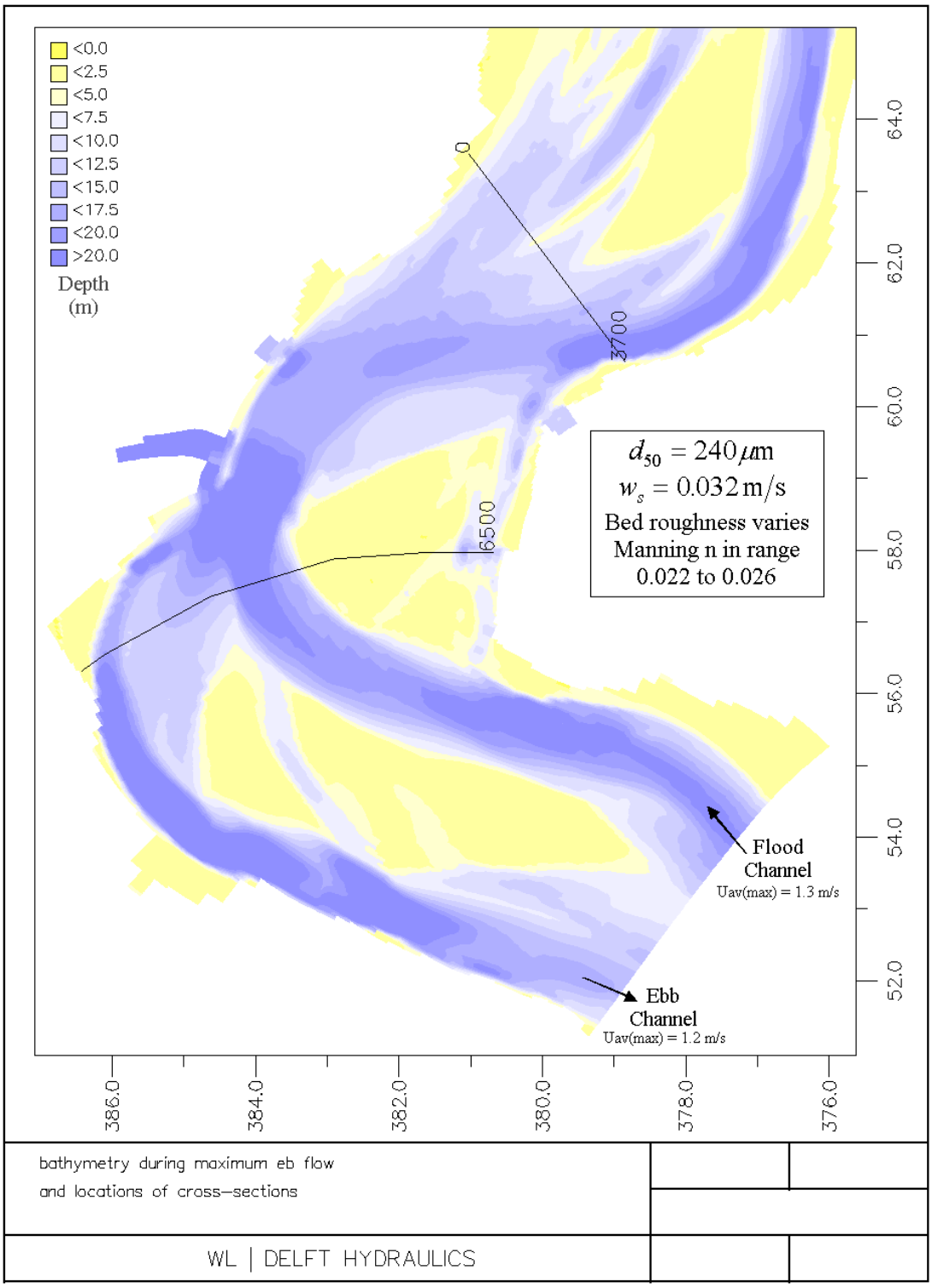


Figure 39 - Bathymetry in the vicinity of the "Sill of Hansweert" (Source: Thoolen 2000)

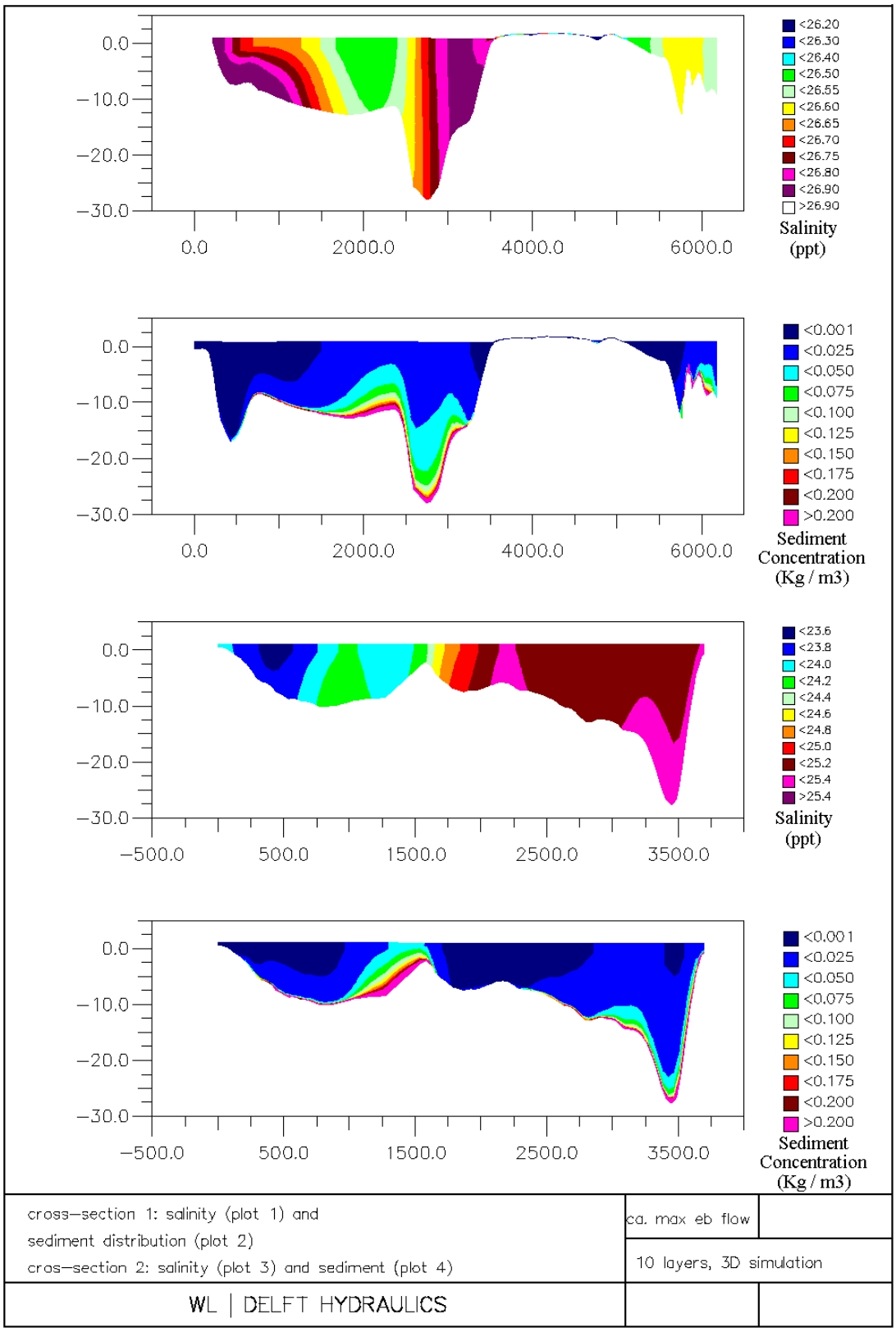


Figure 40 - Salinity and sediment concentration cross sections at time of maximum ebb flow (Source: Thoolen 2000)

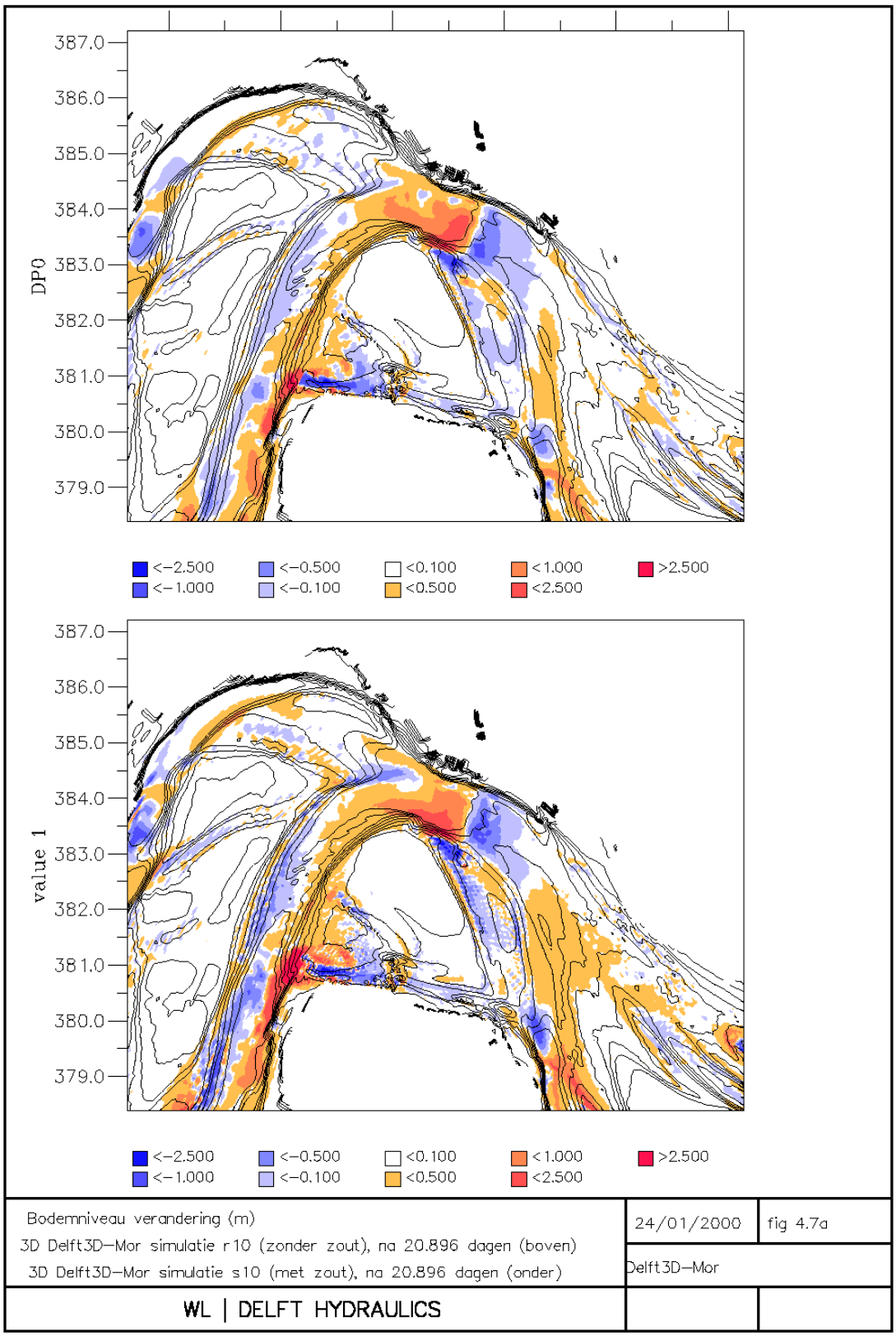


Figure 41 - Depth changes computed in one morphological month by quasi 3D (top) and full 3D (bottom) computations (Source: Thoolen 2000)

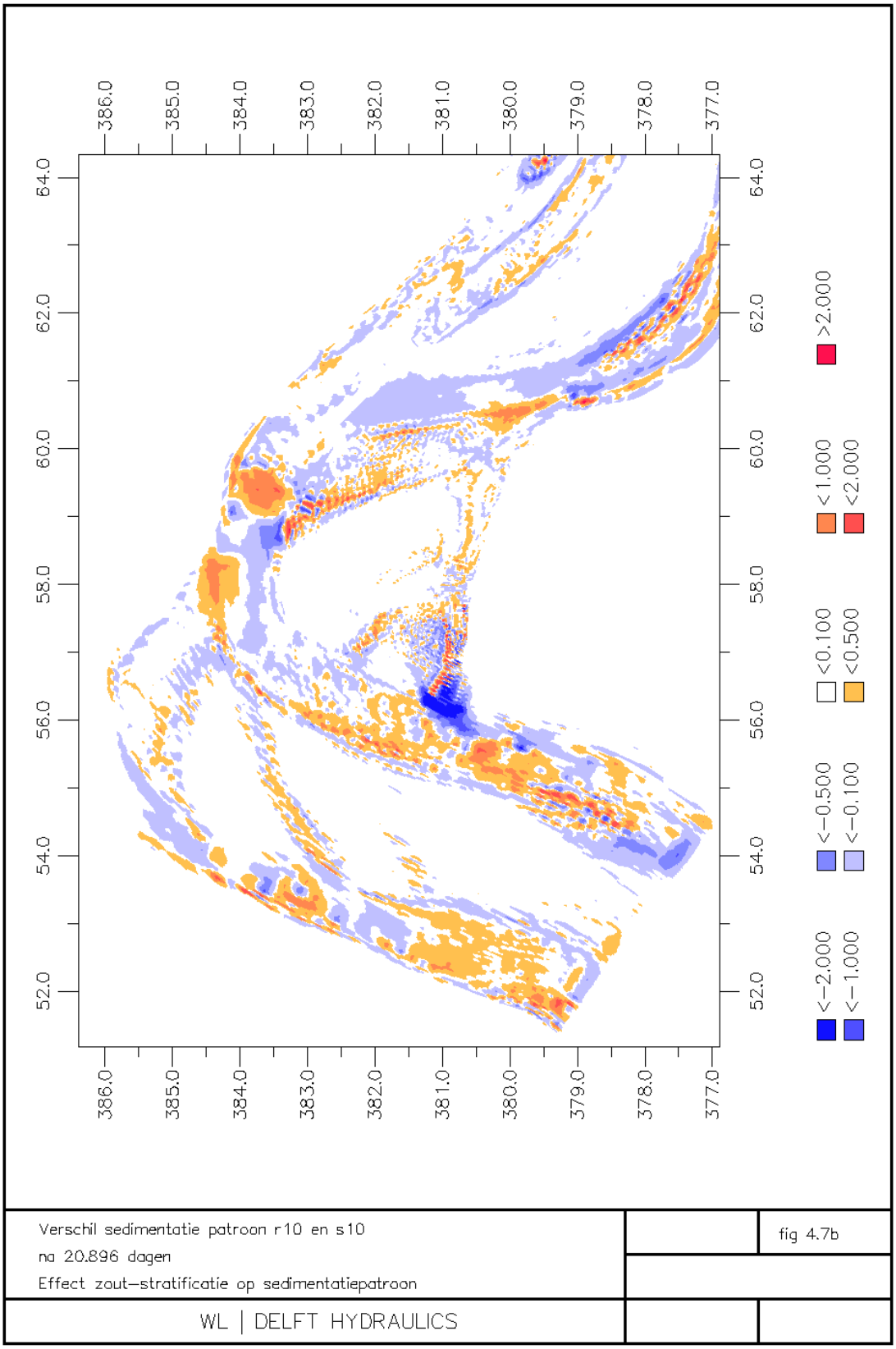


Figure 42 - Differences between depth changes calculated by Q3D and 3D models over one month (Source: Thoolen 2000)

4.4 Waves and Current Validation

The following sub-sections present the results of a brief validation of the modified DELFT3D-FLOW module used in situations where the stirring effects of waves are significant. The three experiments simulated have been selected from a series of tests conducted by Dekker and Jacobs (2000) at the TU Delft laboratory. They have been selected because they cover the situation of waves with a range of magnitudes of following current. The tests selected are:

1. 0.15 m waves only (zero net current)
2. 0.15 m waves with 0.2 m/s following current
3. 0.15 m waves with 0.4 m/s following current

We appreciate that this range of conditions only represents a fraction of the tests required to fully validate the model for a full range of wave and current interactions. We feel that these tests are appropriate, however, bearing in mind the limited number of data-sets available to validate the model against, and the rather preliminary nature of the three-dimensional wave-current interaction model incorporated in the DELFT3D-FLOW module.

4.4.1 Experimental set-up

Dekker and Jacobs conducted the physical experiments in the "Grote Speurwerk" flume in the fluid mechanics laboratory of the Delft University of Technology. The flume has a total length of 45m, a width of 0.8m and a depth of 1.0m. This allowed a moveable sand bed length of 31m, once space was allowed for the wave generator and wave damping structures. The layout of the experiment is shown in Figure 43.

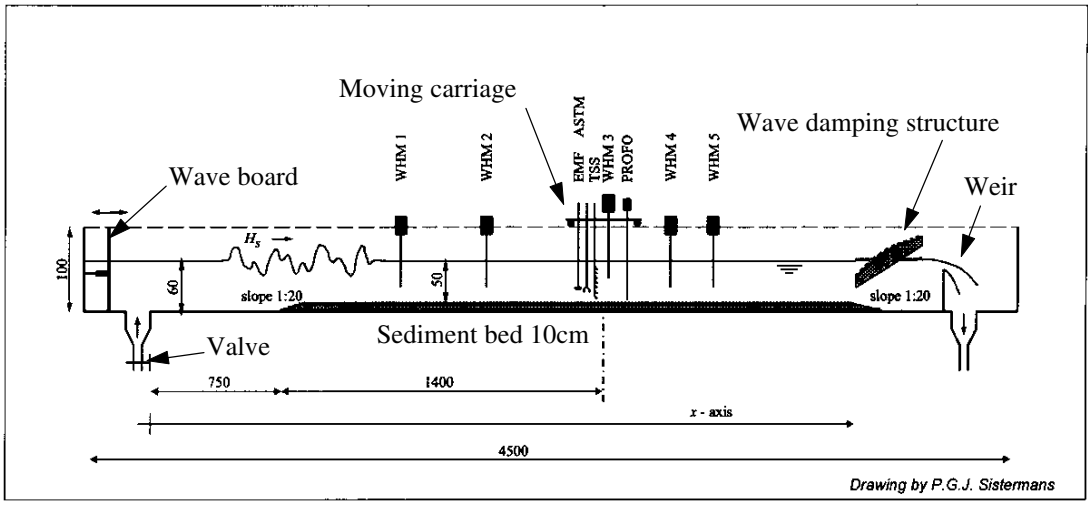


Figure 43 - Arrangement of wave and current experiments (Source: Dekker & Jacobs)

Dekker and Jacobs conducted a number of experiments with a range of wave heights and current velocities. They also repeated all experiments with both uniform and widely graded sediment. Random wave distributions were used in all experiments. We have selected three experiments using the "uniform" sediment (d_{50} = 165μm, d_{90} = 230μm), the experiments

chosen have a medium wave height (nominally H_s = 15cm) and a range of current velocities (Nominally u_{av} = 0, 0.2, and 0.4m/s). In all cases the current is flowing in the same direction as the wave propagation. Details of the key parameters for each experiment can be found in the sections that follow.

Dekker and Jacobs measured both velocities and concentrations by two methods. For measuring velocities they used both an acoustical sediment transport meter (ASTM) and an electromagnetic velocity meter (EMS). When reporting their results they point out that there is frequently a significant difference between the results recorded by the two instruments, and briefly discuss possible reasons for the disagreement. In the sections that follow we present only the velocity measurements recorded by the ASTM, as these results appear, to us, to be more consistent and reliable. We still have some reservations regarding some of the reported velocity measurements however; these are identified in the sections that follow. To measure sediment concentrations Dekker and Jacobs used the ASTM and a transverse suction system (TSS), once again they report significant differences between the results recorded by the two methods. In the following sections we only present the results recorded by the TSS, as these appear to be substantially more reliable than the results from the ASTM.

The computer simulations were performed using the modified (v03.05.007) DELFT3D-FLOW module, incorporating the (preliminary) modifications made by D.J. Walstra to incorporate the effects of three-dimensional wave-current interaction on the computed flow velocities. The modifications made by Walstra include the effects of the non-uniform vertical distribution of the wave-driven mass flux, the addition of wave-induced turbulence, and the effects of streaming in the wave boundary layer. A detailed description of the modifications made to the DELFT3D-FLOW module in order to account for these wave effects is currently in press (Walstra and Roelvink, 2000). Because of the developmental status of the wave-current interaction formulations in this research version of DELFT3D-FLOW the following results should also be regarded as preliminary.

The standard parameters used for the computer simulations are as follows:

Sediment diameter d_{50}	= 165µm
Sediment density ρ_s	= 2650 Kg/m3
Longitudinal grid size Δx	= 0.5m
Length of grid	= 29m
Vertical layers (logarithmic)	= 20 layers
Time-step Δt	= 0.01min
Simulation time	= 10min
Time to reach steady solution	= 7min (approx.)
Morphological time-scale factor (MORFAC)	= 0 (bottom updating off)
Bottom roughness varies, see below	
Reference height factor (AKSFAC)	= 1.0
Wave roughness factor (RWAVE)	= 2.0 (default)
Sediment inflow at upstream boundary (EQMBC)	= .false.

The input for, and results of, the individual simulations are presented below.

4.4.2 Wave-only test

This simulation recreates experiment *m015u* of Dekker and Jacobs. The important parameters particular to this experiment are:

Significant wave height	H_s	= 15.77cm
Peak wave period	T_p	= 2.55s
Depth averaged current velocity	u_{av}	= 0.0m/s
Water depth at test section	h	= 54.06cm
Average ripple height	Δ_r	= 0.7cm
Average ripple length	λ_r	= 10.7cm
Equivalent bed roughness (see below)	k_s	= 0.01m

Note that the effective bed roughness is manually calculated from the reported ripple dimensions using the approach of van Rijn (1993). This approach assumes

1. $k_s = k_s' + k_s''$
2. $k_s' = 3d_{90}$
3. $k_s'' = k_r'' = 20\gamma_r\Delta_r\left(\frac{\Delta_r}{\lambda_r}\right)$

Where: k_s' = grain related roughness
k_s'' = bed-form related roughness
k_r'' = roughness due to ripples
γ_r = ripple presence factor (=1 for ripples only)
Δ_r = ripple height (m)
λ_r = ripple length (m)

The simulations are performed with the above settings, and all other parameters at their default values. Figure 44 shows the resulting computed velocity profile. The effect of the streaming in the wave boundary layer can be clearly seen in the computed result, and is hinted at in the experimental measurements. Generally it appears that the computed streaming is rather greater (or extends higher in the water column) than indicated by the measurements. However the calculated fluid viscosity appears to be reasonable, as the gradient of the computed velocity profile is in good agreement with the measurements.

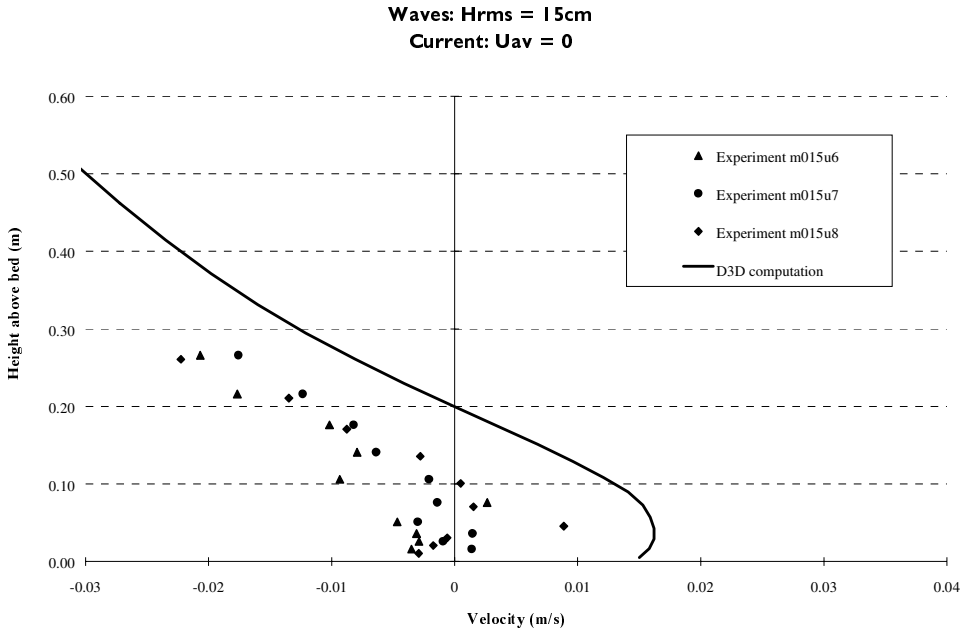


Figure 44 - Measured and computed velocity profiles

Figure 45 presents the computed sediment concentration and turbulent mixing profiles for this experiment. It can be clearly seen that the calculated reference concentration is directly in line with the measured concentrations, this indicates that van Rijn's pickup function for waves and currents is well suited to this situation. Furthermore, the calculated turbulent mixing appears to be very accurate. The equilibrium sediment concentration profile, calculated using the Runga Kutta method, passes through the measured concentrations almost perfectly. The main, computed, solution climbs slightly above the equilibrium profile. It is possible that this may be due to numerical diffusion in the DELFT3D

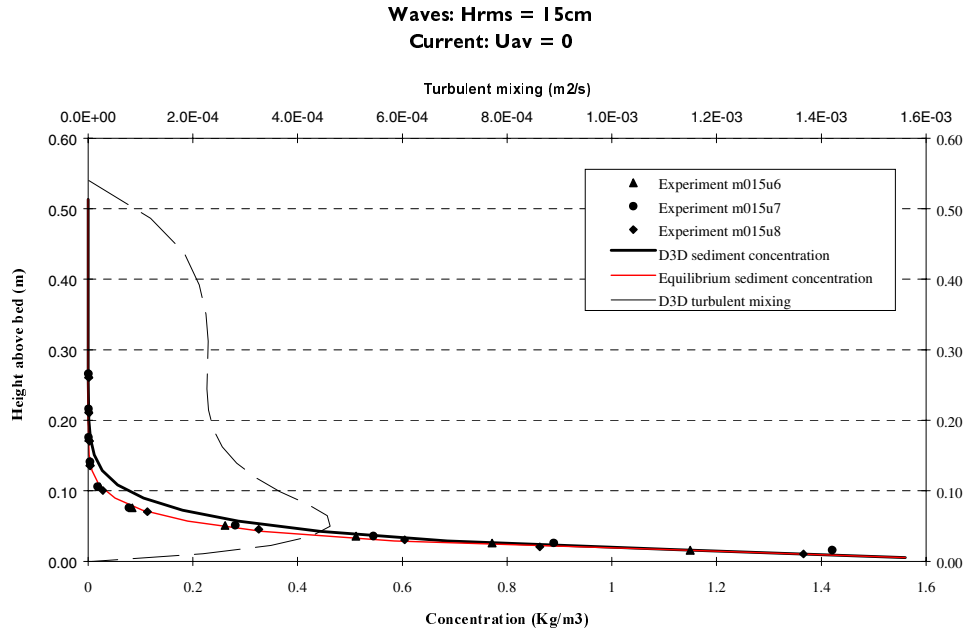


Figure 45 - Measured and computed sediment concentration profiles

computational scheme.

The calculated turbulent mixing profile shows a pronounced local peak near the bed, presumably due to the extra turbulence produced in the wave boundary layer.

4.4.3 Waves with 0.2m/s following current

This simulation recreates experiment *m218u* of Dekker and Jacobs. This experiment is virtually identical to experiment *m015u* described above, apart for the addition of an approximately 0.2m/s current following the waves. The important parameters particular to this experiment are:

Significant wave height	H_s	= 15.24cm
Peak wave period	T_p	= 2.64s
Depth averaged current velocity	u_{av}	= 0.26m/s (see note [5])
Water depth at test section	h	= 50.55cm
Average ripple height	Δ_r	= 1.8cm
Average ripple length	λ_r	= 15.7cm
Equivalent bed roughness (calculated as above)	k_s	= 0.041m

The simulations are performed with the above settings, and all other parameters at their default values. Figure 46 shows the computed velocity profile and the experimental measurements made with and without waves. In this case the velocities are somewhat over-

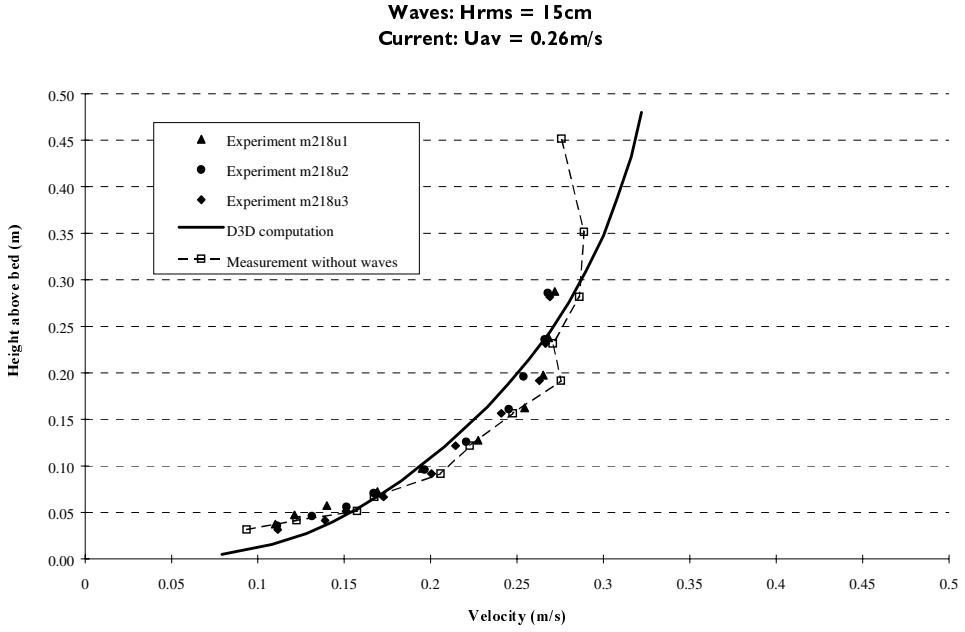


Figure 46 - Measured and computed velocity profiles

[5] *Note that Dekker and Jacobs report that the depth-averaged velocity for this experiment is 0.21m/s. We have re-computed this figure by fitting a logarithmic profile to the velocity measurements taken without waves (disregarding the top four readings which appear to suffer from a lot of scatter), and integrating this from the bed to the water surface.*

estimated near the bed, under estimated from about 10% to 60% of the water height, and probably over-estimated above this level.

The over-estimation of the velocities near the bed make it appear that the bed roughness has been significantly underestimated. We are not entirely confident in the reported experimental results, however, as the velocities measured near the top of the water column, in the absence of waves, appear rather improbable. This casts some doubt upon the accuracy of the other velocity measurements, especially those high in the flow. We suspect that this may be due to the bed roughness changing while the measurements were recorded. It is also surprising that a somewhat larger difference is not observed between the velocities measured with and without waves.

Figure 47 presents the computed and measured sediment concentration profiles for this experiment. Once again the computed reference concentration is in good agreement with the experimental results, even though the reference height has been substantially increased by the increased bed roughness. Again the shape of the computed sediment concentration profile appears to be in very good agreement with the experimental results. This indicates that the modified K-epsilon turbulence model is performing well in this situation.

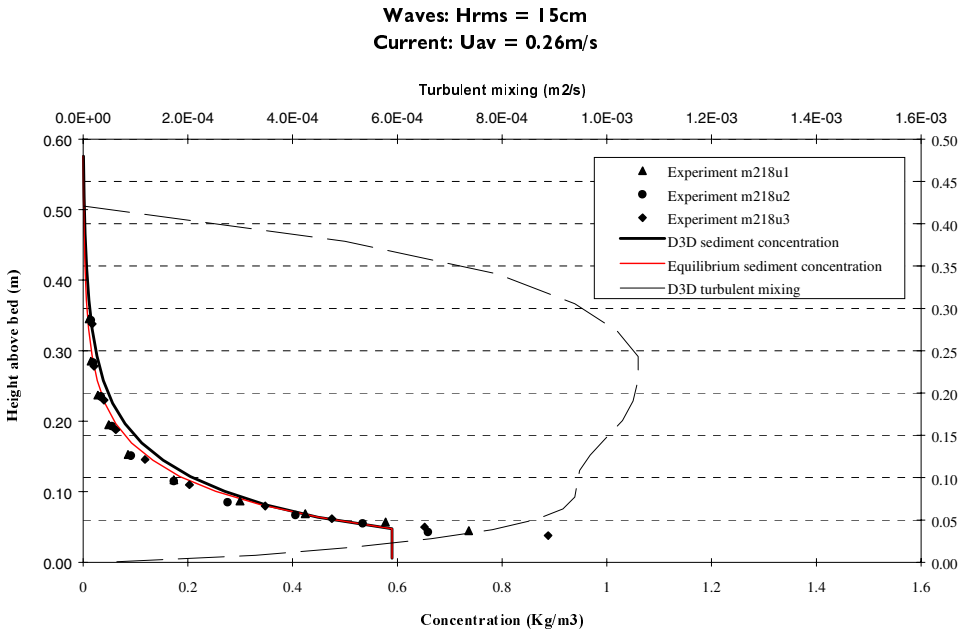


Figure 47 - Measured and computed sediment concentration profiles

4.4.4 Waves with 0.4m/s following current

This simulation recreates experiment *m418u* of Dekker and Jacobs. Again this experiment is virtually identical to the two experiments described above, except that in this case the imposed current is rather stronger. The important parameters particular to this experiment are:

Significant wave height H_s = 15.33cm

Peak wave period	T_p	= 2.61s
Depth averaged current velocity	u_{av}	= 0.40m/s (see note [6])
Water depth at test section	h	= 51.82cm
Average ripple height	Δ_r	= 2.0cm
Average ripple length	λ_r	= 20.4cm
Equivalent bed roughness (calculated as above)	k_s	= 0.039m

The simulations are performed with the above settings, and all other parameters at their default values. Figure 48 shows the computed velocity profile and the experimental measurements made with and without waves. In this case there is no significant difference between the experimental results recorded in the presence of waves, and those measured with current alone. The computed velocity profile is also closer to the standard logarithmic profile than in the previous experiment (as expected), however the computed velocity profile is somewhat less curved than the measurements. This could be due to excessive turbulent mixing in the middle of the flow.

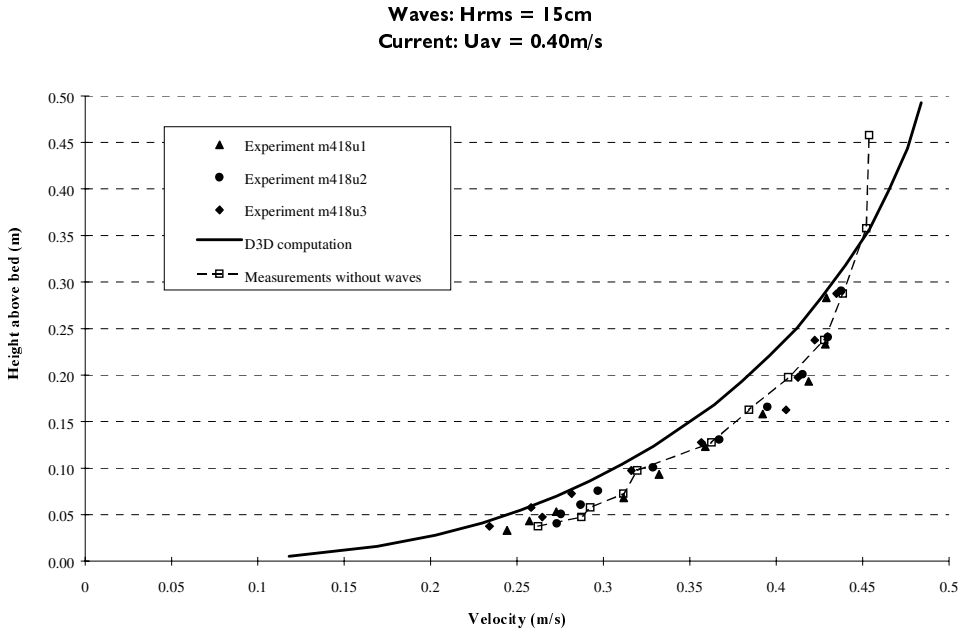


Figure 48 - Measured and computed velocity profiles

Figure 49 presents the measured and computed sediment concentration profiles for this experiment. It can be seen that in this situation the reference concentration is under predicted by approximately a factor of two. It is likely that this is at least partially due to the 10% under prediction of the flow velocity at the edge of the wave boundary mixing layer (height approximately 10-15cm) that can be observed in Figure 48 above. Notwithstanding the above comment, even a factor of two error in the reference concentration (in a simulation performed with all parameters set to their default values) is hardly an unexpected or unacceptable error.

[6] *Again Dekker and Jacobs report a different velocity (0.34m/s). We have re-computed this figure based on the ASTM velocity measurements (including all data points this time).*

Once again the turbulent mixing appears to be very accurately computed by the K-epsilon model. The computed concentrations follow the shape of the measured concentrations very closely. The computed turbulent mixing profile shows that in this situation the effect of the turbulence production in the wave boundary layer is rather insignificant compared to the turbulent mixing produced by the flow itself. The error between the Runga Kutta solution and the main DELFT3D solution is considerably reduced in this simulation. It is possible that this is because the level of numerical diffusion is now insignificant compared to the mixing generated by the flow.

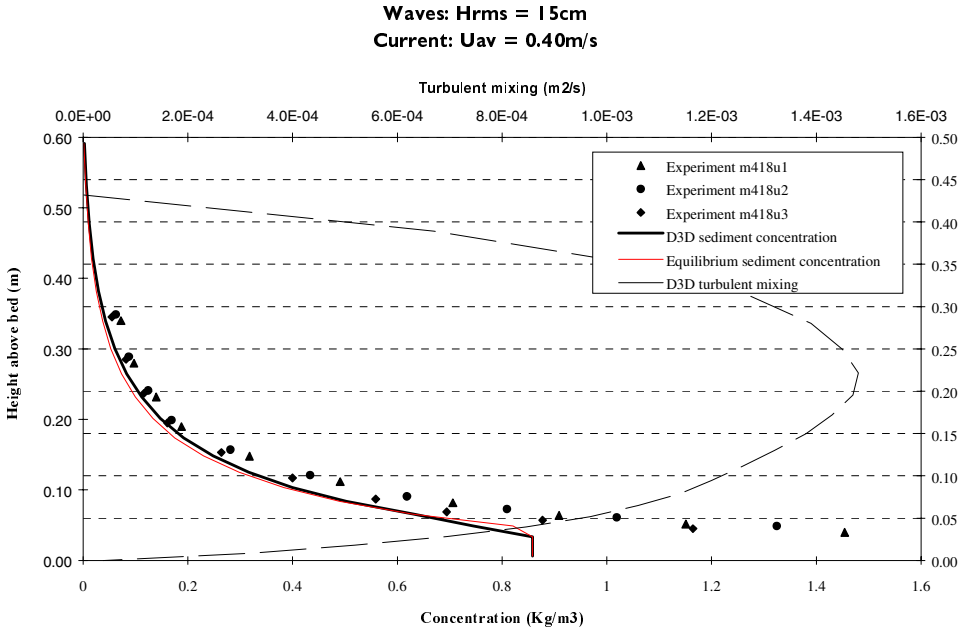


Figure 49 - Measured and computed sediment concentration profiles

5 Conclusions

5.1 Regarding the Current-only Validation

1.1. Van Rijn's pick-up function for current alone is implemented correctly and produces a reasonable reference concentration in all the situations tested.

1.2. The modified DELFT3D-FLOW module accurately reproduces the Rouse sediment concentration profiles under equilibrium conditions.

1.3. Two options exist for the bottom sediment boundary condition, flux or concentration. Both options have been tested and found to give very similar results. The flux boundary condition has proved to be stable, accurate, and flexible enough to include fixed layers, multiple sediment fractions, and a morphological time-scale factor. The concentration boundary condition has not been developed as far as it appears to be less flexible.

1.4. Changing between the k-epsilon and algebraic turbulence models produces a small (ca. 10%) change in the computed equilibrium suspended sediment transport rate. The size of this effect is dependent on the sediment load and is mainly due to a slight change in the computed velocity profile. This results in a somewhat reduced bed shear stress and reference concentration in the case of the k-epsilon turbulence model.

1.5. At equilibrium conditions both the k-epsilon and algebraic turbulence models predict significantly less turbulent mixing than is predicted by the standard parabolic profile using van Rijn's expression for turbulence damping effects of sediment. However the differences in the computed turbulent mixing are mainly high in the water column and do not effect the computed sediment concentration profile significantly.

1.6. The computed suspended sediment concentration profile is virtually independent of the number of layers used. However the choice of layer spacing can have a significant effect. A logarithmic distribution of layer thickness is necessary to achieve accurate results. In a simulation with few layers, the thickness of the bottom layer should be chosen carefully to avoid two of the layers being neglected from the suspended sediment transport calculations. The k-epsilon turbulence model imposes a restriction on the minimum number of layers if complex flow conditions exist.

1.7. The upwind particle settling scheme is more stable than the central difference scheme in situations with little upward diffusion. The upwind scheme introduces negligible error in most practical cases.

1.8. The DELFT3D-FLOW numerical scheme shows excellent conservative properties: The total mass of sediment available in the bed and the flow is well preserved. The initial volume of sediment located in layers below the sediment reference layer (*kmx*) is not included in the mass calculations however.

1.9. The suspended sediment transport load responds somewhat more quickly to changing flow conditions than the SUTRENCH-2D computer model. The adaptation rate is not strongly influenced by the choice of bottom boundary condition or turbulence model.

1.10. Both the algebraic and k-epsilon turbulence models do a good job of predicting the measured velocity and concentration profiles in the situation of a flow crossing a steep-sided trench. The results of the k-epsilon model are generally rather better than those of the algebraic turbulence model however. Turning on the extra production terms in the k-epsilon turbulence model can have a significant (and beneficial) effect on the computed sediment concentration profiles in regions of flow acceleration.

1.11. The predicted morphological changes for the migrating trench experiment appear to be reasonable for both the k-epsilon and algebraic turbulence models. It is likely that the k-epsilon model results are closer to those measured in reality, although it is impossible to confirm this until a bed-load formulation is included in the modified FLOW module.

1.12. The three-dimensional simulation of the long groyne shows that the modified DELFT3D-FLOW module tends to predict the smooth and progressive development of a logical and stable bathymetry. The impact of three-dimensional flow effects are clearly visible even in this relatively simple simulation, and this supports the decision to introduce sediment transport formulations into a three-dimensional flow model.

1.13. The implementation of fixed layers is working correctly; erosion is prevented but deposition is not hindered when a fixed layer is reached.

1.14. The morphological time-scale factor is implemented correctly, and provides a useful method to simulate morphological developments over the medium term. The choice of morphological factor depends on the situation being modelled, and is a matter of judgement. Tests have shown that the bathymetry can remain stable with morphological factors as high as 1000.

1.15. The user-specified option to require equilibrium sediment concentration profiles at inflow boundaries appears to work well. Little change in bed elevations near inflow boundaries is observed when this option is selected. This conclusion is questioned slightly when the k-epsilon turbulence model is used. Refer to section 5.3 below for further discussion of this observation.

1.16. The modified DELFT3D-FLOW module has been successfully applied to the extremely complex Westerschelde morphological model. With the exception of a number of spikes that appeared in shallow areas, the computed morphological changes appear to be logical, and in reasonable agreement with other simulations. Unfortunately insufficient measurement data is available, at this point in time, to determine whether the results of the three-dimensional computation are more, or less, accurate than the quasi three-dimensional approach.

1.17. The adapted FLOW module can successfully model the simultaneous density effects of salinity and sediment, in three dimensions.

5.2 Regarding the Wave and Current Validation

2.1. In the limited number of tests performed, the modified DELFT3D-FLOW module provided excellent results for suspended sediment transport in waves, with or without a following current.

2.2. The van Rijn pickup formulation for waves and currents appears to accurately predict the reference concentration. The predicted reference concentrations are within a factor of 2 for all cases tested, this is achieved using the standard parameter settings recommended by van Rijn in all cases.

2.3. Use of the van Rijn formulation for ripple dimensions is a reasonable method to estimate the wave-related roughness in a complicated (2D) situation and produces reasonable results in the limited number of cases tested. However, the formulation used does not include the effect of the current on the formation of ripples. This could be significant in areas of strong current and, for this reason, the implemented formulation should be improved by extending its validity to the case of waves and current.

2.4. Obtaining reliable data-sets for the velocities and concentrations measured above a moveable sand bed under the combined influence of random waves and current is difficult, and the number of data-sets available in the literature is strictly limited. Further effort should be invested in obtaining accurate data-sets under a range of wave and current situations so that these can be used for further validation of sediment transport and flow models.

2.5. A Runga Kutta solution scheme has been implemented in order to calculate equilibrium sediment concentration profiles in the case of current and waves.

5.3 Regarding the Hydrodynamic Computations

3.1. The k-epsilon turbulence model is not as stable as the algebraic turbulence model. This instability is not caused by the sediment. We find that a Courant number of approximately 1 is required to achieve a stable computation using the k-epsilon model whereas a Courant number of around 4 is possible when using the algebraic turbulence model.

3.2. The k-epsilon turbulence model requires at least 20 layers to function properly in challenging flow situations (e.g. when there are abrupt changes in bottom gradient, or when waves are present). The algebraic turbulence model produces reasonable results in simple situations with as few as 5 layers.

3.3. The minimum number of layers required for a simulation is governed by the hydrodynamic calculations, rather than by the sediment calculations, see above for rough guidelines as to the number of layers required. Thinner layers may require smaller time-steps to avoid “mass closure error” warnings.

3.4. The k-epsilon turbulence model tends to produce slightly non-logarithmic velocity profiles under equilibrium conditions, even without sediment being present. The main area of concern is the very lowest computational layer, where the computed velocity is

reduced significantly. This reduction of the bottom layer velocity has several flow-on effects such as the reduction of the computed bottom shear stress and sediment reference concentration.

3.5. The k-epsilon turbulence model is very sensitive to vertical density gradients (much more so than the algebraic turbulence model). The calibration of this sensitivity is important as it has a significant impact on the computed sediment concentration profile.

3.6. The inflow of K and epsilon at inflow boundaries does not seem well adjusted to the downstream flow. This may cause minor sedimentation problems adjacent to the model boundaries when the k-epsilon model and equilibrium sediment concentration profiles at inflow boundaries are used together.

3.7. The absence of van Rijn's expression for the wave-current interaction factor means that it is impossible to exactly re-create van-Rijn's calculations using the current implementation in a wave and current situation. We believe that van Rijn's formulation should be added to the list of wave-current interactions that the user may select.

3.8. The limits of the shallow water (hydrostatic pressure) assumption inherent in the DELFT3D hydrodynamic computations may start to have significant effects on the computed velocity profiles when steep (1:3) gradients are present. It is not clear whether these inaccuracies in the velocity profiles cause significant errors in the computed bottom evolution. We believe that the effects are likely to be minor however as steep bottom gradients tend to be smoothed rather rapidly in areas with significant sediment transport rates.

6 Recommendations

6.1 Primary Recommendations

1. That research version 03.05.007 of the DELFT3D-FLOW module be adopted as the basis for the future direction of three-dimensional morphological modelling within the DELFT3D modelling framework.
2. That the following recommendations be adopted, in order that a version of DELFT3D capable of morphological computations in three dimensions is available for public release within a reasonable time frame.

6.2 Supplementary Recommendations

6.2.1 That the following features be added before public release is considered

1.1. A suitable bed-load transport formulation must be implemented and tested.

1.2. The communication, history, and map file output must be updated to include several new parameters. Most importantly they must reflect the fact that the depth to the bed is now a function of time in the FLOW module.

1.3. All other modules (such as GPP) must be modified to read the new values from the communication, history, and map files.

1.4. The morphological start time [**MORSTT**] should be changed to read a user-specified *time* rather than a time-step number.

1.5. A simple expression should be implemented in the FLOW module for a depth-averaged sediment transport rate so that the flow module will run sensibly if the user specifies a one layer (two-dimensional) model grid.

6.2.2 That the following testing be performed

2.1. Testing of cohesive sediment transport in current-only and current and wave situations must be carried out.

2.2. Testing if the inflow of K and epsilon at inflow boundaries is correct.

2.3. Determine the cause of the reduced bed shear stress in the case of the k-epsilon turbulence model, and remedy if possible.

2.4. Test, and if necessary re-calibrate, the sensitivity of the k-epsilon turbulence model to vertical gradients in fluid density.

2.5. Determine the reason for the minimum number of layers (approximately 20) required for the k-epsilon turbulence model to operate correctly.

2.6. Test the sensitivity of a number of real-life simulations to the choice of k-epsilon or algebraic turbulence model, and also to the number of computational layers used.

2.7. Repeat the trench migration simulation once an expression for bed-load transport is included.

2.8. Further testing of the interaction of waves and currents in DELFT3D should be carried out when appropriate data-sets can be identified.

2.9. More advanced tests of waves, currents, and sediments should be carried out.

2.10. Check computed velocity profiles against those produced by a full three-dimensional flow solver (one that includes momentum in the vertical direction) to see if this simplification has any significant impact on the trench migration experiment.

2.11. Carry out a number of tests on models that include drying, flooding, and very shallow areas. The goal of this testing is to attempt to identify the cause of the occasional spikes that occasionally appear in shallow areas of complicated models.

2.12. Considerable effort should be invested in obtaining reliable data-sets for measured velocity and concentration profiles over a moveable sand bed under the action of waves and current.

6.2.3 That the following guidelines be followed by users

3.1. That models including non-cohesive sediment transport be constructed using a logarithmic distribution of layer thickness. The thinnest layers should be at the bottom.

3.2. If a model is run with a small number of layers (ca. less than 10), then the thickness of the bottom layer should be chosen so that, in most locations, it includes van Rijn's reference height. This will prevent two layers from being disregarded in the suspended sediment transport calculations.

3.3. If the k-epsilon turbulence model is used, then the results of simulations containing fewer than 20 layers should be treated with caution. Until the testing recommended above has been carried out it is recommended that simulations using fewer than 20 layers are conducted using the algebraic turbulence model.

6.2.4 That the following features be considered for future inclusion

4.1. The prediction of the wave-related bottom roughness by way of estimating the ripple height using van Rijn's expressions should be improved so that it logically (and preferably even accurately) includes the effect of current on ripple formation.

4.2. Van Rijn's formulation for the enhanced bottom roughness felt by a current in the presence of waves should be added to the list of formulations available to the user.

4.3. Consideration should be given to introducing a factor, either specified or computed, that adjusts the fall velocity of the material in suspension to account for the fact that the d_{50} of the material in suspension is generally finer than that at the bed

4.4. Thorough implementation of the behaviour of multiple sediment fractions is required. This predominantly consists of developing an advanced bottom interaction model, and rigorous testing. Our vision for the future operation of this model is located in appendix D.

4.5. The algebraic turbulence model should be extended to take account of the turbulent mixing due to waves. This could be easily achieved by adding van Rijn's expressions for the wave-related mixing to the existing formulations.

7 References

de Vantier B.A., Meijer K. and van Rijn L.C. 1992, Analytical solutions comparison and sensitivity analysis SUSTIM-3D model.

Dekker S and Jacobs C.E.J. 2000, Sediment Concentration Due To Irregular Waves And Currents. Master's thesis (in print), Delft University of Technology

Eckart C. 1958, Properties of water, Part II. The equation of state of water and sea water at low temperatures and pressures. American Journal of Science, 256, 225-240.

Galappatti R. 1983, A depth integrated model for suspended transport. Report 83-7, Communications on Hydraulics, Department of Civil Engineering, Delft University of Technology.

Partheniades E. 1965, Erosion and Deposition of Cohesive Soils. Journal of the Hydraulic Division, ASCE, Vol 91, No. HY1

Thoolen P.M.C 2000, Modellering secundaire stroming en 3D sedimenttransport Westerschelde. WL | DELFT HYDRAULICS report for RIKZ Middelburg.

Uittenbogaard R.E., Winterwerp J.C., van Kester J.A.Th.M., Leepel H. 1996a, 3D cohesive sediment transport Part 1: Text.

Uittenbogaard R.E., Winterwerp J.C., van Kester J.A.Th.M., Leepel H. 1996b, 3D cohesive sediment transport Part 2: Figures and Appendices.

Uittenbogaard R.E., Status of implementation of cohesive sediment transport in DELFT3D-FLOW, version 2.47.

van Kessel T. 1999, 3D sediment transport in DELFT3D-FLOW Test calculations with sediment release. Report No. Z2534.10, WL | DELFT HYDRAULICS.

van Rijn L.C. 1984, Sediment transport, Part II: Suspended Load Transport. Journal of Hydraulic Engineering, No 12.

van Rijn L.C. 1985, Two-dimensional vertical mathematical model for suspended sediment transport by currents and waves. SUTRENCH model. Report S 488 part IV, WL | DELFT HYDRAULICS.

van Rijn L.C. 1987, Mathematical Modelling of Morphological Processes in the case of Suspended Sediment Transport. Doctoral thesis, Delft University of Technology.

van Rijn L.C. 1993, Principles of Sediment Transport in Rivers, Estuaries and Coastal Seas. Aqua Publications, Amsterdam.

Walstra D.J.R and Roelvink J.A. 2000, 3D Calculation of Wave Driven Cross-shore Currents. International Conference on Coastal Engineering July 2000 (In print).

Vatvani D. June 1999, DELFT3D-FLOW sediment version. R&D report no. M3090.05 / Z222534.00, WL | DELFT HYDRAULICS.

WL | DELFT HYDRAULICS. August 1996, DELFT3D-FLOW User Manual, release 2.48.

WL | DELFT HYDRAULICS. September 1999, DELFT3D-FLOW version 3.05 User Manual.

A Changes to the Computer Code

A.1 General

The changes described in this appendix relate to the implementation of suspended sediment transport in wave and current situations. The base case for these changes is version 03.05.005 of the DELFT3D-FLOW module, as found on Deepak Vatvani's CDROM. The changes described are those required to create sediment version 03.05.007; this version uses a flux bottom boundary condition.

Changes made between the original public release version 03.05 and version 03.05.005 are additional to the changes described here. They are discussed in more depth by Vatvani 1999 and van Kessel 1999.

A.2 Overview of the new functions

Version 007 of the FLOW module allows the computation of suspended sediment transport in three dimensions. This is achieved by specifying a constituent of type "Sedimentx" and supplying the required sediment and morphological input files. The implementation allows the simultaneous presence of multiple sediment fractions, each of these fractions must be specified as consisting of "sand", "floc", or "mud". The modified FLOW module calculates the transport, erosion, and deposition of each sediment fraction, accounting for the density and hindered settling effects caused by the total sediment load. Morphological changes are computed and introduced into the flow calculations as they occur.

Several additional features are also included, these include:

- Fixed layers. An initial quantity of sediment at the bed is specified. Erosion will not exceed this specified sediment depth.
- Morphological time-scale factor. This factor allows the acceleration of morphological developments relative to the simulated flow period.
- The onset of morphological developments may be delayed to a user-specified time.
- Equilibrium sediment concentration profiles may be specified at inflow boundaries for "sand" sediment fractions.
- Three-dimensional wave effects are included. These changes have been implemented by Dirk Jan Walstra of WL | DELFT HYDRAULICS and are included in the changes described below. Wave effects included are the non-uniform vertical distribution of the wave-driven mass flux, additional wave induced turbulence, and streaming in the wave boundary layer.

A.3 Overview of the changes to the code

The following changes have been included (the name(s) of the subroutine(s) are denoted in uppercase):

- An additional input file (**morph.inp**) is required for morphological parameters. This file is read in TRISOL
- Van Rijn's reference height and reference concentration are calculated, in situations with both waves and current, and are converted to appropriate sediment source and sink (flux) terms near the bed (EROSED)
- The equilibrium sediment concentration is calculated for every computational cell (EROSED)
- Van Rijn's beta factor (sediment diffusion) is calculated for every location (EROSED)
- Sand sediment diffuses at a different rate to other constituents (DIFU)
- Equilibrium sediment concentration profiles may be applied at inflow boundaries (DIFU)
- The density effects of sediments are included (DENS)
- Sediment settling is calculated (DIF_WS)
- The forester filter may be applied to sediment (FORFIL)
- The quantity of available bottom sediments, and resulting flow depth is updated (BOTTOM)
- The vertical distribution of the 2^{nd} order Stokes drift is computed (EULER, STOKES, STOKTB)
- The turbulent effects of waves are included in the k-epsilon turbulence model (TRATUR)
- A shear stress due to streaming in the wave boundary layer is added (CUCNP, CUCNP2, UZD, TAUBOT)

A.4 Changes to the code

In order to implement the desired changes 30 existing routines are modified and 4 new routines are created. The changes are described below. Only substantive changes are described, full details can be found in the source code itself. The programmer's initials are as follows:

Initial	Programmer
DJW	Dirk Jan Walstra
GL	Giles Lesser
HL	Heleen Leepel

INICHK files

Changed routines:

Routine name	Changed by	Changes
CHKDRY	HL?	– Setting of KFV=0 fixed
INCHKR	GL GL DJW GL & DJW GL & DJW	– Call to INISED changed - extra parameters DPSED, CDRYB – Call to DENS changed - sends value .FALSE. in place of parameter DENSIN – Call to EULER changed - extra parameters TP, HRMS, THICK – 1st call to TAUBOT changed - extra parameters DFU, DELTAU, HRMS, WRKA10 – 2nd call to TAUBOT changed - extra parameters DFV, DELTAV, HRMS, WRKA11
INISED	GL GL	– Header changed - extra parameters DPSED, CDRYB – Loops 80 and 100 added to set initial depth of sediment
TKECOF	GL	– Comment added - warning SIGMA undefined

MAIN files

Changed routines:

Routine name	Changed by	Changes
IIDIM	GL	– Added definition of array KMXSED
RJDIM	GL HL	– Added definition of array BETA – Added definition of arrays HRMS, DFU, DFV, DELTAU, DELTAV
TRICOM	GL HL DJW	– Added pointers to arrays KMXSED, BETA – Added pointers to arrays HRMS, DFU, DFV, DELTAU, DELTAV – Call to SETWAV changed (x2) - extra parameter HRMS

OUTPUT files

Changed routines:

Routine name	Changed by	Changes
POSTPR	DJW	– Call to EULER changed - extra parameters TP, HRMS, THICK
SETWAV	DJW	– Header changed - extra parameter HRMS – Call to FRDINT changed - parameter HRMS replaces UORB – Line 430 changed - parameter HRMS replaces UORB

REK3D files

Changed routines:

Routine name	Changed by	Changes
ADI	DJW	– Header changed - extra parameters DFU, DFV, DELTAU, DELTAV, TP, RLABDA – Call to UZD changed - extra parameters DFV, DELTAV, TP, RLABDA – Call to SUD changed - extra parameters DFU, DELTAU, TP, RLABDA – Call to UZD changed - extra parameters DFU, DELTAU, TP, RLABDA – Call to SUD changed - extra parameters DFV, DELTAV, TP, RLABDA

AOI	DJW	– Header changed - extra parameters DFU, DFV, DELTAU, DELTAV, TP, RLABDA – 1st call to UZD changed - extra parameters DFU, DELTAU, TP, RLABDA – 2nd call to UZD changed - extra parameters DFV, DELTAV, TP, RLABDA – 1st call to CUCNP2 changed - extra parameters DFU, DELTAU, TP, RLABDA – 2nd call to CUCNP2 changed - extra parameters DFV, DELTAV, TP, RLABDA
CUCNP	DJW	– Header changed - extra parameters DFU, DELTAU, TP, RLABDA – Include PROCS.INC added – New lines 343 - 348, 638 – New lines 694 - 722. Add streaming shear stress in WBL
CUCNP2	DJW	– Header changed - extra parameters DFU, DELTAU, TP, RLABDA – Include PROCS.INC added – New lines 347, 351, 739, 746 – New lines 778 - 806. Add streaming shear stress in WBL
SUD	DJW	– Header changed - extra parameters DFU, DELTAU, TP, RLABDA – Call to CUCNP changed - extra parameters DFU, DELTAU, TP, RLABDA
UZD	DJW	– Header changed - extra parameters DFU, DELTAU, TP, RLABDA – Include PROCS.INC added – New lines 416, 420, 422 - 424, 463, 477 – New lines 527 - 554. Add streaming shear stress in WBL

REKEN files

New routines:

Routine name	Created by	Functions
BOTTOM	GL	– Computes change in BODSED based on source and sink terms calculated in EROSED and morphological acceleration factor MORFAC. Applies new concentrations to implicit sink terms – Calculates new sediment mixing layer thickness (DPSED), based on change in BODSED values – Calculates change in depth at zeta points based on change in DPSED values – Sets depths at U and V points by upwind method from nearest zeta point.
STOKES	DJW	– Calculates vertical distribution of Stokes 2nd order drift velocity
STOKTB	DJW	– Calculates the 2nd order stokes drift at the bed
WAVENR2	DJW	– Calculates the wave number – May be better to use existing routine WAVENR

Changed routines:

Routine name	Changed by	Changes
DENS	GL	– Header changed - extra parameter DENSIN – Loop 190 modified. Test if DENSIN is true before adding sediment to density calculations
DIF_WS	GL	– Header changed extra parameters S1, DPS, SIG, AKS, KMXSED, DIFFAC removed parameters BDDDX, BDDX, BDX, BUX, BUUX, BUUUX – Loop 630 changed. Settling sediment runs from layer K=1 to K=MAXLAY. – Concentration bottom boundary condition removed – Loop 725 added. Sets concentrations in cells below K=MAXLAY, depending on state of DIFFAC.

DIFU	GL	– Header changed - extra parameters AKS, BETA, BODSED, KMXSED, DIFFAC, EQMBC – Loop 600 modified. – Loop 550 added to remove vertical advection across bottom of KMXSED layer – Loop 580 renumbered 680 – Loop 580 added to calculate vertical diffusion of sediment fractions using different diffusion coefficient – Call to dif_ws changed extra parameters S1, DPS, SIG, AKS, KMXSED, DIFFAC removed parameters BDDDX, BDDX, BDX, BUX, BUUX, BUUUX – Loop 825 added to insert equilibrium sediment concentration profiles at open boundaries if EQMBC=true – Lines 1019 - 1043 added. Removes horizontal advection of sand sediment below layer K=KMXSED (odd NM values) – Lines 1095 - 1119 added. Removes horizontal advection of sand sediment below layer K=KMXSED (even NM values)
DIFUEX	GL	– Some changes made. Requires re-programming to include similar changes as DIFU routine. Not presently operational.
DIFUIM	GL	– Some changes made. Requires re-programming to include similar changes as DIFU routine. Not presently operational.
DIFUQQ	GL	– Some changes made. Requires re-programming to include similar changes as DIFU routine. Not presently operational.
DIFUVL	GL	– Some changes made. Requires re-programming to include similar changes as DIFU routine. Not presently operational.
EROSED	GL	– Header changed extra parameters Z0UCUR, Z0VCUR, SIGMOL, S1, UORB, TP, AKS, SIGDIF, LSTSCI, BETA, DICWW, KMXSED, DIFFAC, THRESH, MORFAC, AKSFAC, UMEAN, VMEAN, RWAVE, ROUSE; removed parameters CREF, CSOIL, FLSDBD, RHO, AFLC, BFLC, WS0, WSM, SALMAX, POWSED, SDBUNI, CDRYB – Include PROCS.INC added – New line 233 – New loop 80. Calculate total available bottom sediments – Old loop 30 removed. Resetting of sink and sour terms unnecessary – Lines 252 - 761. Complete rewrite of pick-up from bed for sand sediment. Present functionality is as follows: – Line 261. Reset Prandtl-Schmidt number for sand fractions – Lines 263 - 491. Set van Rijn's reference height and concentration, valid for waves and current. Bottom shear stress due to current is based on a near-bed velocity. Includes an estimate of the wave-generated ripple height. Determination of TAUCR follows Shields, retained from earlier implementation. – Lines 495 - 500. Adjust reference concentration for multiple sediment fractions. – Lines 504 - 516. Calculate van Rijn's beta factor. – Lines 525 - 616. Calculate equilibrium sediment concentration profiles. Use either a standard Rouse profile incorporating van Rijn's beta and psi factors, or calculated diffusion profile and Runga Kutta numerical integration. Depends on setting of ROUSE. – Lines 627 - 744. Calculate SINK and SOURce terms for KMXSED layer using split explicit/implicit approach. Includes limitation of terms near a fixed layer. Also sets DIFFAC parameter to control setting of concentrations below KMXSED level. – Following changes are for COHESIVE sediment – Line 772 added. set van Rijn's beta=1 – Lines 775 - 779 added. Warning comments. – Lines 786, 787 added. Calculation of thickness of bottom cell – Lines 792 - 798 modified. SOURce and SINK terms corrected for thickness of bottom layer – Updating of BODSED variable removed.

EULER	DJW	– Header modified - extra parameters TP, HRMS, THICK – Include PROCS.INC added – Loop 200 modified. Includes vertical distribution of wave-induced mass flux. Includes call to new routine STOKES – Old (non-wave influenced) code copied to loops 300 and 400 for non-wave situations – New subroutine STOKES included (see above) – New subroutine WAVENR2 included (see above)
FORFIL	GL	– Header modified - extra parameters LSTS, LSED – Loop 4300 modified to include sediment – CAUTION: assumes dc/dz<0 if vertical Forester filter used.
TAUBOT	DJW, GL DJW DJW DJW GL GL GL DJW DJW DJW DJW DJW DJW	– Header modified - extra parameters DFU, DELTAU, HRMS, Z0UCUR – Include PROCS.INC added – New real variable KS defined – Parameter ALFAW = 20.0 – Loop 270 added to store Z0UCUR values – Line 507 removed – Line 508 modified to use stored Z0UCUR values – Lines 581 - 604 added. Sets wave dissipation due to bottom friction (DFU) and boundary layer thickness (DELTAU) values. – Lines 611 - 616 added. Include effect of streaming on bed shear stress – Line 645 removed. (Setting TAUBSU(NM)=0.0) – Loop 700 modified. Bed shear stress is corrected for mass fluxes, modified for case of waves – Loop 800 added. Original loop 700 for case of no waves. – New subroutine STOKTB included (see above).
TRATUR	DJW	– Header modified - extra parameters DELTAU, DFU, DIS, HRMS, UORB, TP, UMEA, VMEA, UBND, KDISMX, HSURFT, PKWAV – Include PROCS.INC added – Lines 706 - 709 added. Reset pkwav and pkwbt – Loop 727 added. Determine vertical length of distribution of turbulent energy – Loop 729 added. Determine turbulence across vertical – Loop 731 added. Add turbulent production due to wave dissipation in wave boundary layer – Loop numbers 733 and 732 changed to 734 and 736 respectively – Loop 749 added. Source of energy dissipation (epsilon) in breaking waves and in bottom boundary layer. – Lines 1055 - 1057 moved from above line 1047 (no effect)

TRISOL	GL	– Define extra parameters MORSTT, ILUN,L,MORFAC, THRESH, AKSFAC, RWAVE, MORUPD, EXIST, EQMBC, DENSIN, ROUSE, TXTPUT
	GL	– Lines 507 - 582 added. Read user-defined parameters MORFAC, MORSTT, THRESH, MORUPD, EQMBC, DENSIN, AKSFAC, RWAVE, ROUSE from **morph.inp** file
	DJW	– Call to ADI modified. Extra parameters DFU, DFV, DELTAU, DELTAV, TP, RLABDA
	DJW	– Call to AOI modified. Extra parameters DFU, DFV, DELTAU, DELTAV, TP, RLABDA
	DJW	– Call to EULER modified. Extra parameters TP, HRMS, THICK
	DJW, GL	– 1st call to TAUBOT modified. Extra parameters DFU, DELTAU, HRMS, WRKA10
	DJW, GL	– 2nd call to TAUBOT modified. Extra parameters DFV, DELTAV, HRMS, WRKA11
	GL	– Call to CALTMX modified. Extra parameters WRKA10, WRKA11, WRKA6
	GL	– Call to EROSED modified.
	GL	– extra parameters WRKA10, WRKA11, SIGMOL, S1, UORB, TP, WRKA7, SIGDIF, LSTSCI, BETA, DICWW, KMXSED, WRKA8, THRESH, MORFAC, AKSFAC, UMEAN, VMEAN, RWAVE, ROUSE;
	GL	– removed parameters CREF, CSOIL, FLSDBD, RHO, AFLC, BFLC, WS0, WSM, SALMAX, POWSED, SDBUNI, CDRYB
	GL	– Call to TRITRA modified. Extra parameters WRKA7, BETA, BODSED, KMXSED, WRKA8, EQMBC
	DJW	– Call to TRATUR modified. Extra parameters DELTAU, DFU, DIS, HRMS, UORB, TP, IBUFF, WRKA1, WRKB11
	GL	– Call to FORFIL modified. Extra parameters LSTS, LSED
	DJW	– Call to EULER modified. Extra parameters TP, HRMS, THICK
	GL	– Call to DENS modified. Condition extended to include sediment. Extra parameter DENSIN
	GL	– Call to BOTTOM added. Conditional on presence of sediment and time-step > MORSTT
		– Above call changes repeated for 2nd half time-step
TRITRA	GL	– Header modified - extra parameters AKS, BETA, BODSED, KMXSED, DIFFAC, EQMBC
		– Call to DIFU modified. Extra parameters AKS, BETA, BODSED, KMXSED, DIFFAC, EQMBC
		– Calls to DIFUVL, DIFUEX, DIFUQQ slightly modified and temporarily removed.
		– *In 2nd half time-step*
		– Call to DIFU modified as above
		– Calls to DIFUVL, DIFUIM, DIFUQQ slightly modified and temporarily removed.

INCLUDE files

Changed file:

Routine name	Changed by	Changes
R-I-CH	GL	– Definition of BETA added
	HL	– Definitions of DELTAU, DELTAV, DFU, DFV added
	HL	– Definition of HRMS added
	HL?	– Lines 920, 921 reformatted (no effect)
	GL	– Definition of KMXSED added
	GL, HL	– Common real pointers for BETA, DELTAU, DELTAV, DFU, DFV, HRMS added
	GL	– Common integer pointer for KMXSED added

B Calculation of Bed Shear Stress in Currents and Waves

B.1 Background

Bed shear stress is one of the primary parameters required to calculate the pick-up of sediment into the flow. In wave and current situations van Rijn (1993) calculates the time-averaged absolute bed shear stress using the following equation

$$\left|\tau_{b,cw}\right| = \alpha_{cw}\tau_{b,c} + \left|\tau_{b,w}\right| \tag{B.1}$$

where:
$\tau_{b,w}$ is the time-averaged value of the bed shear stress due to waves and is given by the expression

$$\tau_{b,w} = \frac{1}{4}\rho f_w \hat{u}_\delta^2 \tag{B.2}$$

where $\hat{u}_\delta$ = peak orbital velocity at edge of wave boundary layer.
f_w = total wave-related friction factor.

$\tau_{b,c}$ is the time-averaged bed shear stress due to current alone, and is given by the expression

$$\tau_{b,c} = \rho u_*^2 \tag{B.3}$$

where u_* is the bed shear velocity, calculated as

$$u_* = \kappa u\left(z = e\, z_0\right) \tag{B.4}$$

where $z_0 = \frac{k_s}{30}$.

α_{cw} is a bed shear stress reduction factor which takes into account the reduction in near-bed current velocities due to wave-current interaction. This effect is shown diagramatically in Figure 50.

Figure 50 illustrates that, in the presence of waves, the main body of the flow feels an increased, or apparent, bed roughness (k_a). Close to the bed, inside the wave boundary mixing layer (WBML), the flow feels the physical bed roughness (k_s).

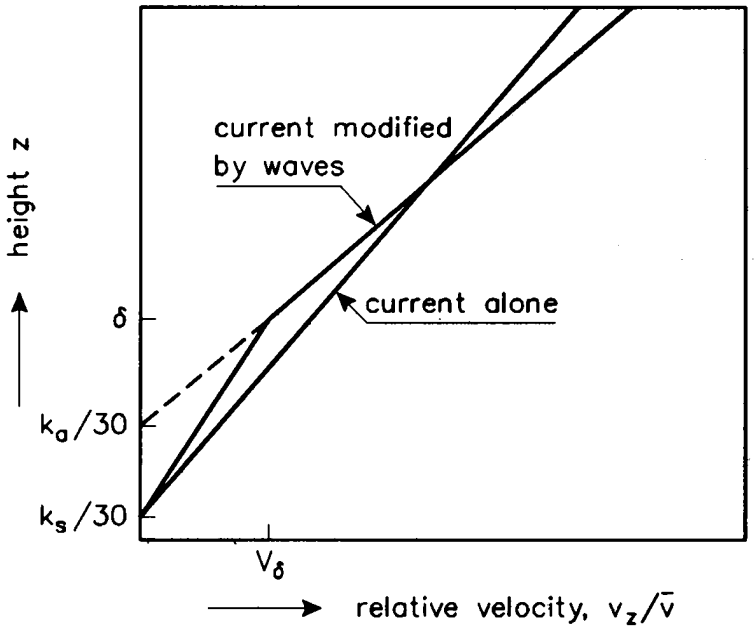


Figure 50 - Influence of waves on current velocity profile. (Source: van Rijn 1993)

B.2 Implemented Approach

B.2.1 Current-only situations

In this case we simply use the flow velocity computed by DELFT3D in the bottom computational layer to calculate the bed shear velocity (u_*) using the expression

$$u(z) = \frac{u_*}{\kappa} \ln\left(1 + \frac{z}{z_0}\right) \tag{B.5}$$

rearranged to solve for u_* . The values $u(z)$ and z are the velocity and centre height of the bottom computational layer respectively. The computed u_* value is then used in equation (B.3) to calculate the bed shear stress due to current.

Note: We use $\left(1 + \frac{z}{z_0}\right)$ in equation (B.5) in place of the more conventional $\left(\frac{z}{z_0}\right)$ because this is consistent with the assumption made by DELFT3D that the bottom ($z = 0$) is at z_0.

B.2.2 Wave and current situations

In wave and current situations the definition level for u_* ($e z_0$) will be *inside* the wave boundary mixing layer (WBML). We follow van Rijn and assume that within this region the velocities consist of a logarithmic velocity profile based on the physical roughness (k_s). However, in the presence of waves, DELFT3D calculates the velocity profile on the basis of an apparent roughness (k_a) which depends on the wave-current interaction model chosen by the user. DELFT3D *does not* explicitly consider a change in the apparent roughness when computing flow velocities within the WBML. So, if van Rijn's schematisation of the velocity profile in wave and current situations is accepted, DELFT3D correctly calculates velocities outside the WBML, but we need a velocity at a height ($e z_0$) inside the WBML.

Van Rijn adjusts the bed shear stress calculated using the depth-averaged flow velocity (calculated using the enhanced bed roughness) by introducing a bed shear stress reduction factor α_{cw} to account for the change in apparent bed roughness at the edge of the WBML. We follow a slightly different, two step, approach based on the same schematisation of the velocity profile in a wave and current situation.

Step 1 - Calculate the flow velocity at the edge of the wave boundary layer

Given a velocity calculated by DELFT3D on the basis of the enhanced bed roughness $z_{0a} = \frac{k_a}{30}$, we calculate the velocity at the edge of the wave boundary mixing layer (WBML) at the height $z = \delta_m$. Note: $\delta_m = 3\delta_w$, δ_w = thickness of the wave boundary layer.

For this purpose we use the velocity, taken from the DELFT3D flow calculation, that is *closest* to the height $z = \delta_m$ (this will not necessarily be the velocity in layer k = kmax). We will call this velocity u_k at height z_k.

If we assume a logarithmic velocity profile, based on the *enhanced* bed roughness, then

$$u_k = \frac{u_*}{\kappa} \ln\left(1 + \frac{z_k}{z_{0a}}\right) \tag{B.6}$$

and, at the top of the WBML ($z = \delta_m$),

$$u_{\delta_m} = \frac{u_*}{\kappa} \ln\left(1 + \frac{z_{\delta_m}}{z_{0a}}\right) \tag{B.7}$$

Combining these two equations we find

$$u_{\delta_m} = u_k \frac{\ln\left(1+\frac{z_{\delta_m}}{z_{0a}}\right)}{\ln\left(1+\frac{z_k}{z_{0a}}\right)} \tag{B.8}$$

Step 2 - Calculate bed shear stress

The bed shear velocity u_* (= the velocity at height $e z_0$) is calculated based on the velocity at the edge of the WBML (u_{δ_m}) assuming a logarithmic velocity distribution between $z = \delta_m$ and $z = e z_0$ based on the *unenhanced* bed roughness $z_0 = \frac{k_s}{30}$. In this case

$$u_{\delta_m} = \frac{u_*}{\kappa} \ln\left(1+\frac{z_{\delta_m}}{z_0}\right) \tag{B.9}$$

and rearranging for u_* gives

$$u_* = \frac{u_{\delta_m} \kappa}{\ln\left(1+\frac{z_{\delta_m}}{z_0}\right)} \tag{B.10}$$

We then calculate the bed shear stress due to currents in the presence of waves ($\tau_{b,cw}$) using the standard expression

$$\tau_{b,cw} = \rho u_*^2 \tag{B.11}$$

As demonstrated above, this two step approach is consistent with van Rijn's assumed velocity profile (Figure 50) however it eliminates the need for van Rijn's bed shear stress reduction factor α_{cw}.

C Detailed Velocity and Concentration Profiles for the Trench Migration Experiment

C.1 Profiles with Algebraic Turbulence Model

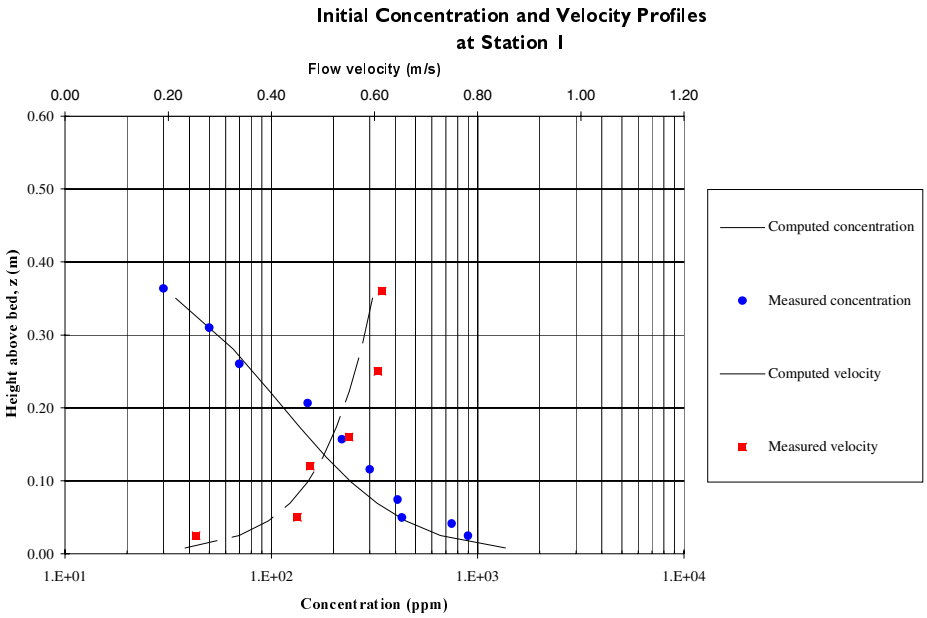


Figure 51 - Station 1, algebraic turbulence model

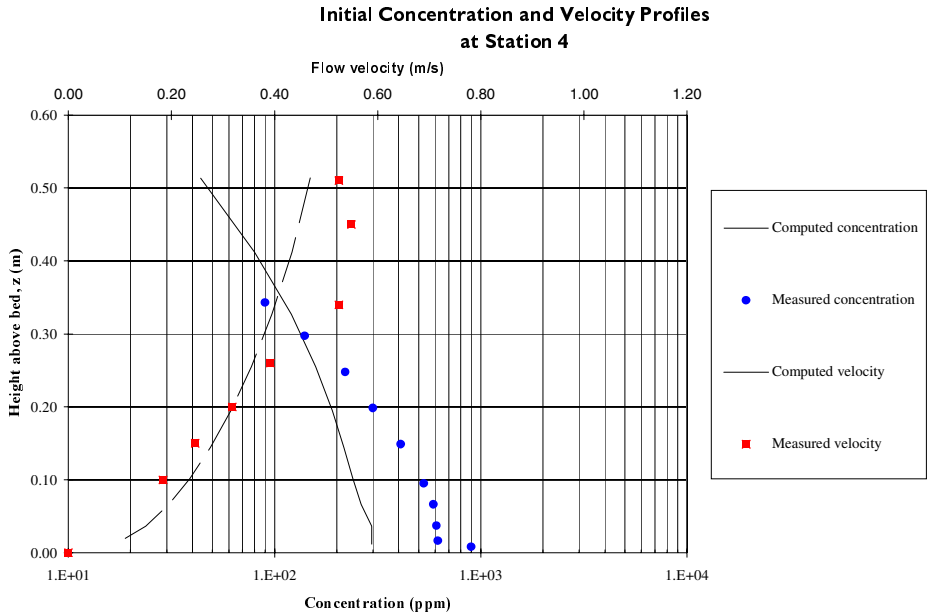


Figure 52 - Station 4, algebraic turbulence model

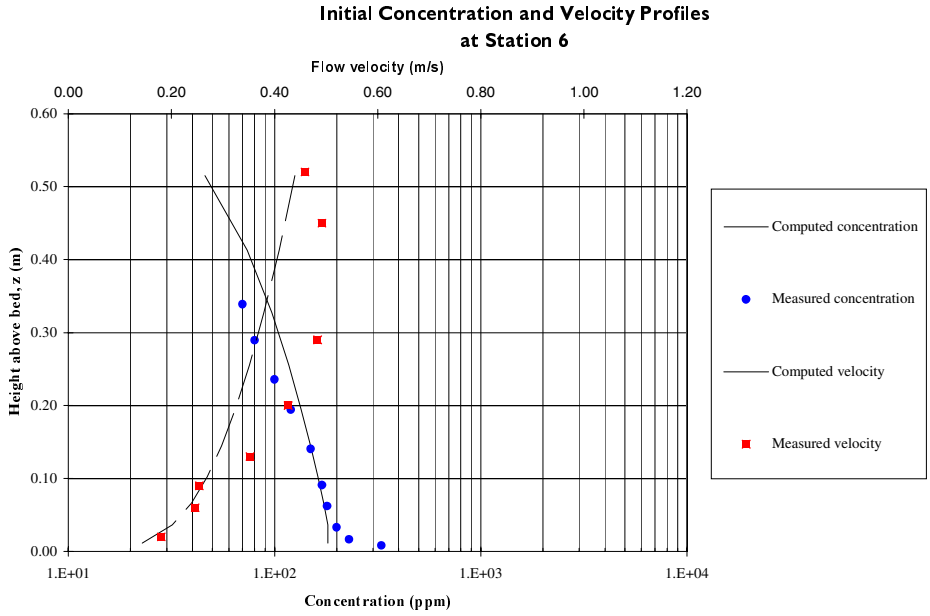


Figure 53 - Station 6, algebraic turbulence model

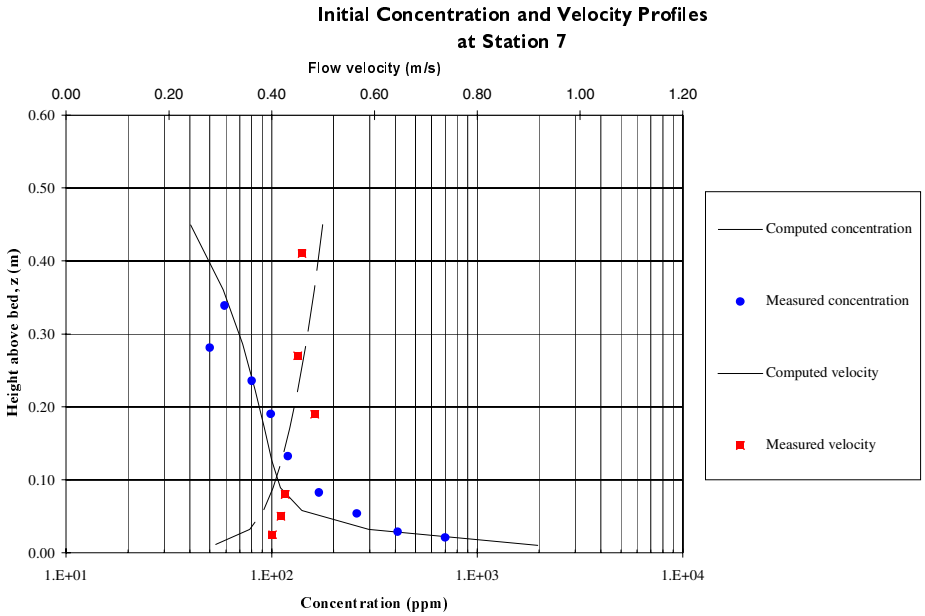


Figure 54 - Station 7, algebraic turbulence model

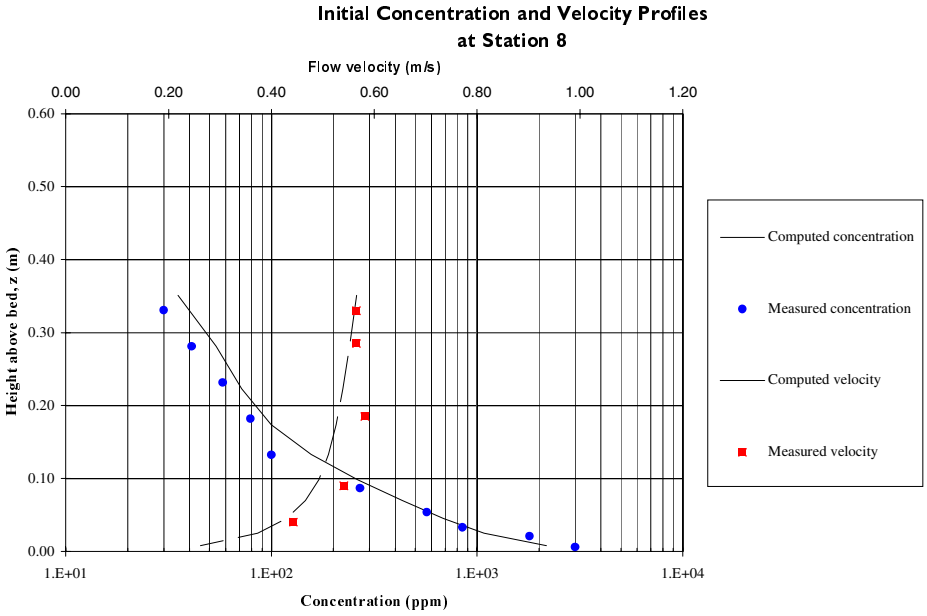


Figure 55 - Station 8, algebraic turbulence model

C.2 Profiles with k-epsilon Turbulence Model

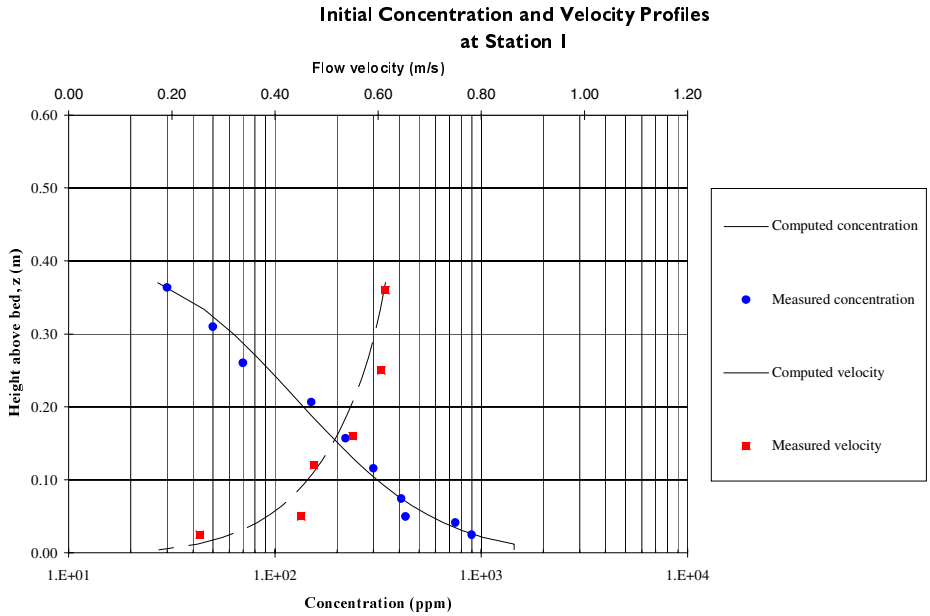


Figure 56 - Station 1, k-epsilon turbulence model

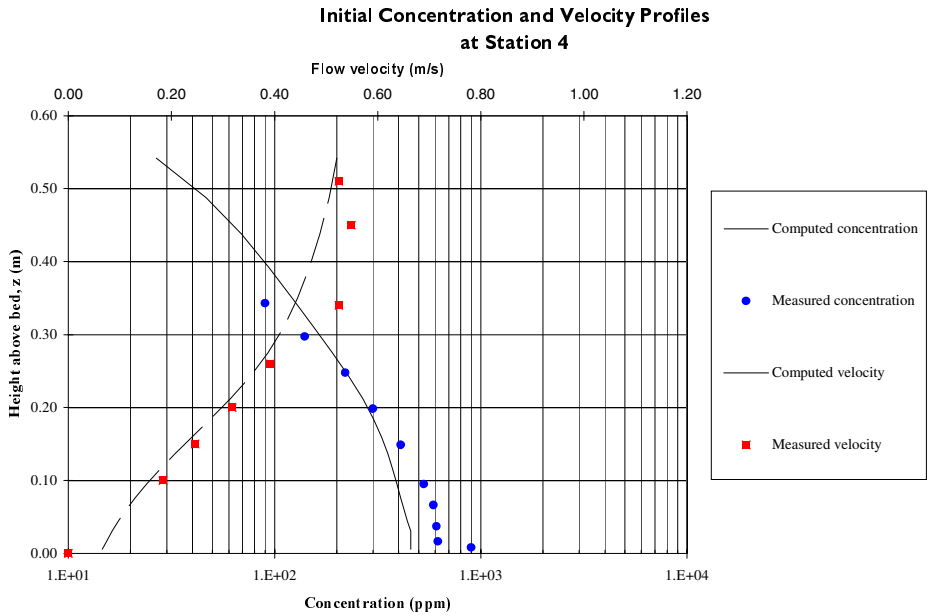


Figure 57 - Station 4, k-epsilon turbulence model

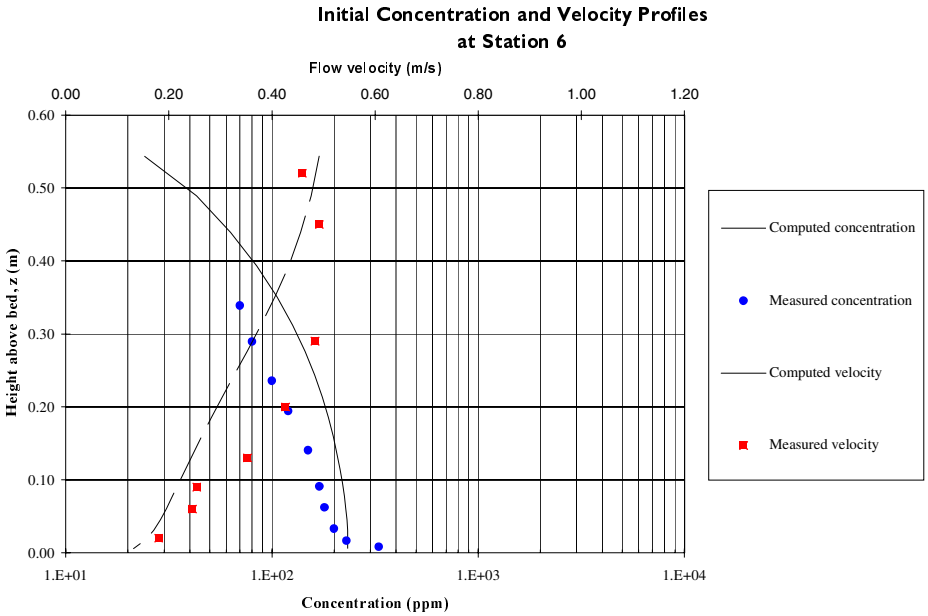


Figure 58 - Station 6, k-epsilon turbulence model

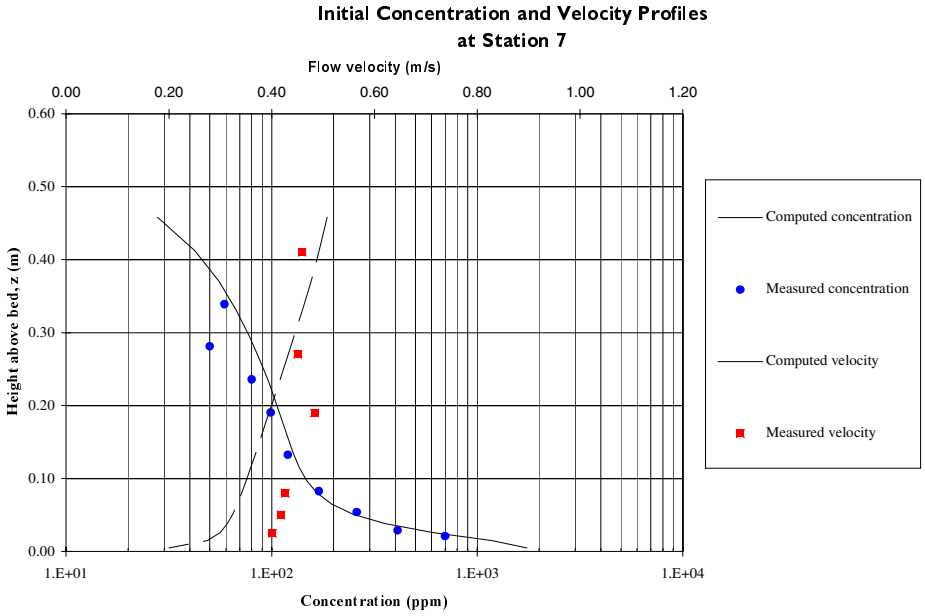


Figure 59 - Station 7, k-epsilon turbulence model

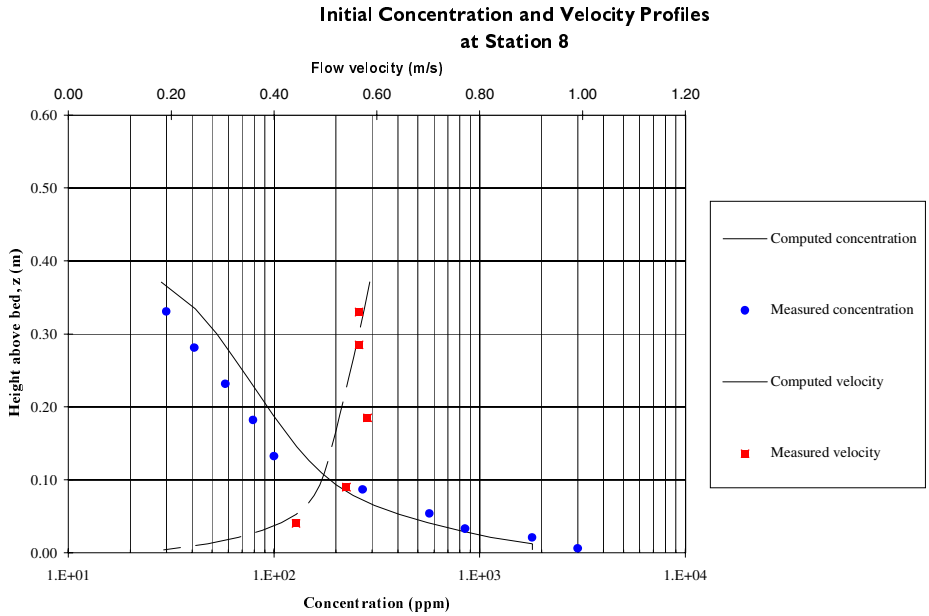


Figure 60 - Station 8, k-epsilon turbulence model

C.3 Influence of the Extra Terms in the k-epsilon Turbulence Model

Turning on the extra production terms in the k-epsilon turbulence model has a noticeable effect on the computed velocity and sediment concentration profiles in the region of rapid flow acceleration. The profiles at station 7 are shown for simulations with and without the extra production terms.

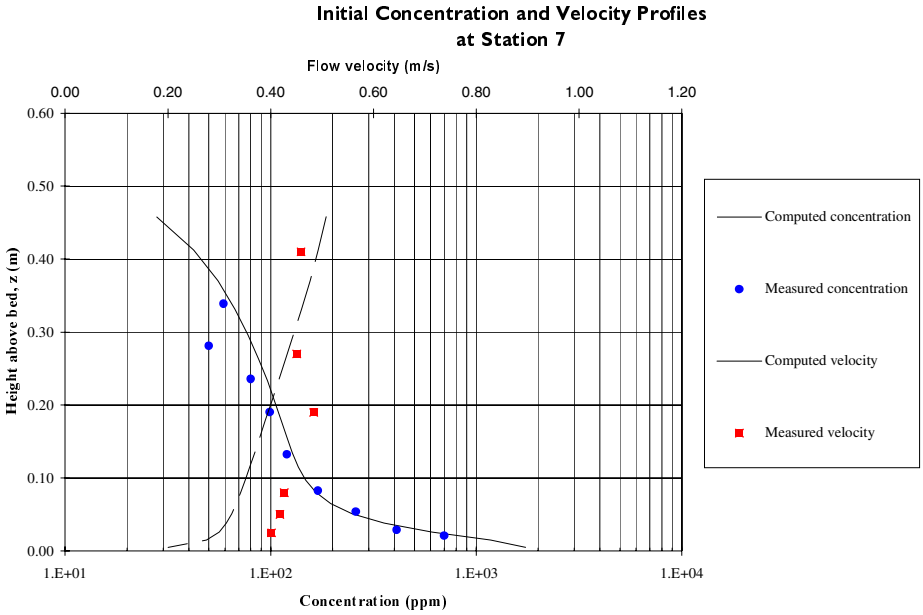


Figure 61 - Station 7, k-epsilon turbulence model, standard production terms

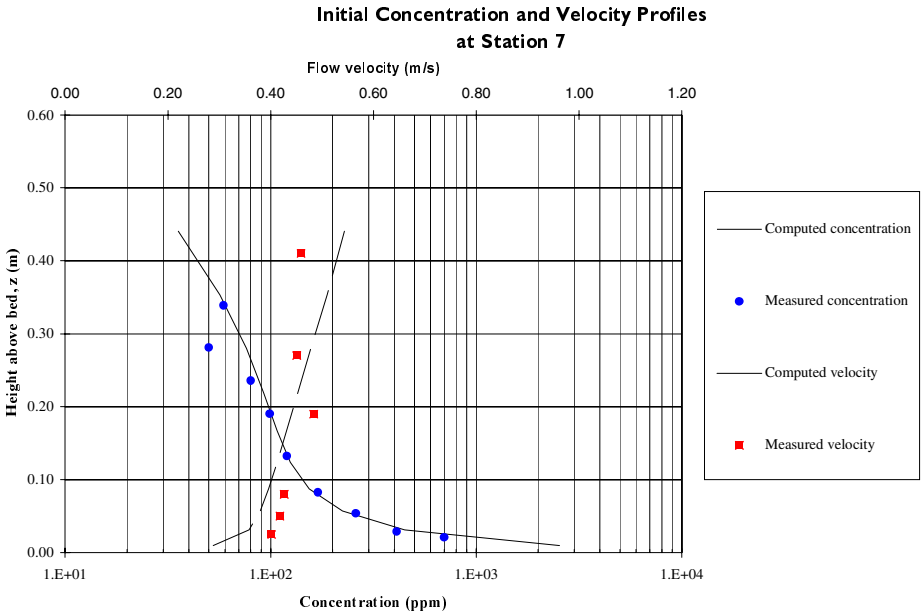
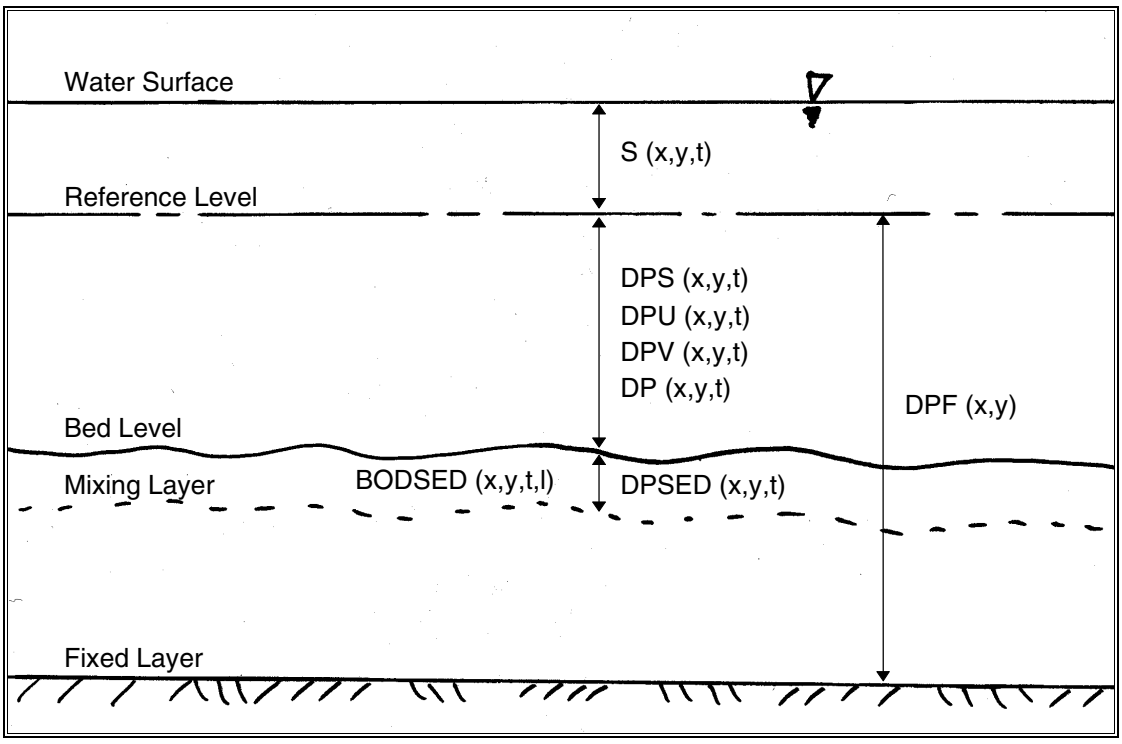


Figure 62 - Station 7, k-epsilon turbulence model, all production terms included

D Vision for the Future Development of a Bed Interaction Model

D.1 Definition Sketch for Bathymetry-related Variables



Note: S, DPS, DPSED, DPF, BODSED are all calculated at zeta (water-level) points
DPU, DPV, and DP are calculated at u, v, and depth points respectively.
Additional variable(s) will be required to specify the composition of the bed materials. This will need to be initially specified by the user, and will vary with time. The precise manner of achieving this has not been considered, but it will need to be some function of x,y,t,l and, to some extent, z.

D.2 Vision for the Future

- Initial values for DPS(x,y,t=0) are obtained by interpolating between values read from a user input file (as is currently the case). The user input file specifies depths at depth points (i.e. DP(x,y,t=0)).

- DPU and DPV points are set and updated by an upwind scheme from adjacent DPS points (This has been implemented).
- DP points are updated based on surrounding DPS, DPU, or DPV points. This has yet to be implemented.
- DPF(x,y) values will be obtained by interpolating between values read from a user input file. The user input file specifies depth to the fixed layer at depth points.
- If user-specified DPF(x,y) values are less than the specified initial depth values then the DPF(x,y) value should be set equal to the initial depth, DPS(x,y,t=0). This provides the user with an easy method of specifying areas with a fixed bottom.
- The DPF(x,y) values are not functions of time.
- DPSED(x,y,t) (the thickness of the mixing layer) will be calculated on the basis of local flow conditions and/or bed roughness and may be limited by the depth to the fixed layer.
- The quantity of each sediment fraction available to be picked up by the flow, measured in kg/m^2, is assigned to the variable BODSED(x,y,t,l). In this expression 'l' is the sediment fraction number.
- BODSED(x,y,t,l) will be linked to DPSED(x,y,t) by calculating the dry density of the mixing layer and the proportion of each BODSED fraction.
- The dry density of the mixing layer will be estimated giving due consideration to the presence of multiple sediment fractions.
- BODSED(x,y,t,l) will be updated to reflect the net flux of sediment from the mixing layer to the suspended load (multiplied by a user-specified morphological time-scale factor), and the movement of the bottom of the mixing layer up or down through the user defined bottom sediments.
- Movement of the bottom of the mixing layer should update the bed composition variable(s).
- Time dependent depth output to the map and history files should be at the depth points, to avoid confusing users.

D.3 Simplifying Assumptions for the Existing Implementation

1. The mixing layer extends all the way from the bed to the fixed layer. In other words, the available bottom sediments are always evenly mixed.
2. The bottom material, whether being eroded or deposited, is assumed to have a constant in-situ density regardless of the relative concentration of the different sediment fractions.

These assumptions are certainly rather crude when a situation with multiple sediment fractions is considered. However they are sufficient to illustrate whether the transport of one or more sand fractions can be successfully modelled by DELFT3D-FLOW. Furthermore, the manner in which the variables are used is consistent with the envisaged future development of sediment transport in DELFT3D-FLOW. This means that the existing implementation can be simply interfaced with a more sophisticated bottom change routine if this is implemented in the future.

The above assumptions do not have any impact on the accuracy of a model involving only one sediment fraction.

D.4 Consequences of the Simplifying Assumptions

- DPF and the composition of the bed material is not yet required.
- BODSED(x,y,t=0,l) can be specified directly by the user.
- DPSED(x,y,t) will be back-calculated from the BODSED(x,y,t,l) values using the relationship

$$DPSED(x,y,t) = \frac{\sum_{1}^{lsed} BODSED(x,y,t,l)}{\rho_{mixing-layer}(x,y,t)}$$

- The dry density of the mixing layer will be given by the very simple (constant) expression

$$\rho_{mixing-layer}(x,y,t) = \rho_{soil}(\#1)\,.$$

- BODSED(x,y,t,l), and thus DPSED(x,y,t), will vary with time only according to the net erosion/deposition flux from the bed to the suspended sediment.

Part II

Bed-load Transport, Inclusion in MORSYS, & Other Improvements

Giles Lesser
Jan van Kester
J.A. (Dano) Roelvink

August 2000

WL | delft hydraulics

Contents

List of Figures

List of Figures

1 Calculation of Bed-Load Sediment Transport

1.1 Overview

The calculation of bed-load sediment transport of 'sand' sediment fractions is included in the modified DELFT3D-FLOW module, generally following the method described by van Rijn (1993). Activation of the bed-load transport option causes the bed-load transport vector to be calculated at each computational point at each time step. Spatial gradients in bed-load transport will cause erosion and accretion of the bed which may also be taken into account when performing on-line morphological computations, as described in section 2 of this report. The bed-load sediment transport vector includes an adjustment for the effect of longitudinal and transverse bed slopes.

The present implementation uses a three-step approach to calculate the spatial distribution of the bed-load transport. First, the magnitude and direction of the bed-load transport vector are calculated (including bed slope effects and consideration of fixed layers) at each internal (non-boundary) water level point. Second, the transport vectors at open boundary points are then copied from the adjacent internal points. Finally, the bed-load transport components are transferred to the U and V velocity points using an upwind numerical scheme. This approach is used as it avoids having to recalculate a large number of flow parameters at the velocity points, and also avoids the necessity for implementing a complicated (and computationally expensive) bed updating scheme. The following sections describe each of these steps in more detail.

1.2 Calculation of the bed-load transport vector

1.2.1 Calculate bed-load transport magnitude assuming a horizontal bed

Initially the magnitude of the bed-load transport vector is calculated assuming a horizontal bed. This is achieved using a bed-load transport formula advised by van Rijn (personal communication, June 2000).

$$\left|S_b''\right| = 0.25\,\eta\,\rho_s\,d_{50}\,u_*'\,D_*^{-0.3}T \tag{1.1}$$

Where:

$\left\|S_b''\right\|$	=	mass bed-load transport rate (Kg/m/s)
η	=	relative availability of the sediment fraction in the mixing layer (as calculated for suspended sediment transport)
ρ_s	=	sediment density (Kg/m^3)
d_{50}	=	sediment diameter (m)

u_*'	=	effective bed shear velocity = $u_* \sqrt{\mu_c}$
u_*	=	bed shear velocity (as calculated for suspended sediment transport)
μ_c	=	efficiency factor current (as calculated for suspended sediment transport)
D_*	=	dimensionless particle diameter (as calculated for sus. sediment transport)
T	=	dimensionless bed-shear stress (see below)

The dimensionless bed-shear stress T is calculated in a similar manner to the T_a parameter required for the suspended sediment transport calculations, using the formula

$$\begin{aligned} T &= \frac{(\mu_c \tau_{b,cw} + \mu_w \tau_{b,w}) - \tau_{cr}}{\tau_{cr}} \\ T_{\min} &= 0 \end{aligned} \tag{1.2}$$

Where all parameters are identical to those used in the suspended sediment calculations except for μ_w (the efficiency factor waves) which is recalculated as discussed below. Readers should note that this calculation of the T parameter includes the stirring effect of waves, but neglects the effect of bed slopes on the critical bed shear stress τ_{cr}. This latter effect is neglected as we find that if it is included the model formulations become somewhat too sensitive to bed gradients, and the bed may become unstable during morphological computations. The effect of bed slope on the critical bed shear stress is more thoroughly described by van Rijn (1993).

The efficiency factor waves is calculated differently for bed-load transport than for suspended load transport. For bed-load transport we use the relationship

$$\mu_w = \frac{f_w'}{f_w} \tag{1.3}$$

where:

f_w'	=	grain-related wave friction factor = $\exp\left[-6 + 5.2\left(\frac{\hat{A}_\delta}{3d_{90}}\right)^{-0.19}\right]$
$\hat{A}_\delta$	=	near-bed peak orbital excursion (as for suspended sediment calculations)
d_{90}	=	sediment diameter (m)
f_w	=	total wave friction factor (as for suspended sediment calculations)

This method follows that suggested by van Rijn (1993), as used in the TRANSPOR 1993 model.

1.2.2 Calculation of vector components (transport direction)

The bed-load transport vector components are calculated under the assumption that the bed-load transport occurs in the same direction as the velocity vector in the bottom computational cell. Thus the two bed-load transport components are calculated as follows:

$$S''_{b,x} = \frac{u_{b,x}}{|u_b|}|S''_b|$$
$$S''_{b,y} = \frac{u_{b,y}}{|u_b|}|S''_b| \qquad (1.4)$$

Where $u_{b,x}, u_{b,y}$, and $|u_b|$ are the same local bottom-layer flow velocity components and magnitude as used to calculate the bed shear stress. Readers should note that we envisage that in the future the two components $S''_{b,x}$ and $S''_{b,y}$ will be calculated directly by an intra-wave bed-load transport model which is capable of accounting for the net effects of wave asymmetry. When implemented this intra-wave model will replace equations (1.1) to (1.4) above.

1.3 Adjust bed-load transport for bed-slope effects

1.3.1 Longitudinal bed slope

We adjust the *magnitude* of the bed-load transport vector if a bed slope exists *in the direction of the bed-load transport vector*. This bed slope is calculated as

$$\frac{\partial z_b}{\partial s} = \frac{\partial z_{(u)}}{\partial x}\frac{S''_{b,x}}{|S''_b|} + \frac{\partial z_{(v)}}{\partial y}\frac{S''_{b,y}}{|S''_b|}$$
$$\left(\frac{\partial z_b}{\partial s}\right)_{\max} = 0.9\tan^{-1}(\phi) \qquad (1.5)$$

where:

$\frac{\partial z_b}{\partial s}$	= bed slope in the direction of bed-load transport (Note that as z_b is the depth down to the bed from a reference height (positive down), a downward bed slope returns a positive value.)
$\frac{\partial z_{(u)}}{\partial x}$	= bed slope in the positive x direction evaluated at the downwind U point.
$\frac{\partial z_{(v)}}{\partial y}$	= bed slope in the positive y direction evaluated at the downwind V point.
ϕ	= internal angle of friction of bed material (assumed to be 30°)

Note that the bed slope is calculated at the downwind U and V points as these are the locations at which the bed-load transport vector components will finally be applied.

The size of the adjustment is calculated following a modified form of the expression suggested by Bagnold (1966)

$$\left|S_b'\right| = \alpha_s \left|S_b''\right| \tag{1.6}$$

or, in vector component form

$$\begin{aligned} S_{b,x}' &= \alpha_s S_{b,x}'' \\ S_{b,y}' &= \alpha_s S_{b,y}'' \end{aligned} \tag{1.7}$$

where

$$\alpha_s = 1 + \alpha_{bs}\left[\frac{\tan(\phi)}{\cos\left(\tan^{-1}\left(\frac{\partial z}{\partial s}\right)\right)\left(\tan(\phi) - \frac{\partial z}{\partial s}\right)} - 1\right] \tag{1.8}$$

where α_{bs} is a user specified tuning parameter [**ALFABS**] (default = 1.0).

1.3.2 Transverse bed slope

We adjust the *direction* of the bed-load transport vector if a bed slope exists *in the direction normal to the bed-load transport vector*. This bed slope is calculated as

$$\frac{\partial z_b}{\partial n} = -\frac{\partial z_{(u)}}{\partial x}\frac{S_{b,y}''}{\left|S_b''\right|} + \frac{\partial z_{(v)}}{\partial y}\frac{S_{b,x}''}{\left|S_b''\right|} \tag{1.9}$$

An additional bed-load transport vector is then calculated, perpendicular to the main bed-load transport vector. The magnitude of this vector is calculated using a formulation based on the work of Ikeda (1982, 1988) as presented by van Rijn (1993). Van Rijn's equation (7.2.52) is modified to equation (1.10) below by setting the reference co-ordinates *s* and *n* aligned with and perpendicular to the local flow direction respectively. This implies that there is no flow in the *n* direction: i.e. $u_{b,n} = 0$.

$$S_{b,n} = \left|S_b'\right| \alpha_{bn}\left(\frac{u_{b,cr}}{u_b}\right)\frac{\partial z_b}{\partial n} \tag{1.10}$$

where:

$S_{b,n}$	=	Additional bed-load transport vector. The direction of this vector is normal to the unadjusted bed-load transport vector, in the down slope direction.
$\left\|S_b'\right\|$	=	Magnitude of the unadjusted bed-load transport vector (adjusted for longitudinal bed slope only).

	= $\sqrt{\left(S'_{b,x}\right)^2 + \left(S'_{b,y}\right)^2}$
α_{bn}	= User specified coefficient [**ALFABN**] (default = 1.5)
$u_{b,cr}$	= Critical (threshold) near-bed fluid velocity.
u_b	= Near-bed fluid velocity vector.
$\frac{\partial z_b}{\partial n}$	= Bed slope in the direction normal to the unadjusted bed-load transport vector.

To evaluate equation (1.10) we substitute:

$$\left(\frac{u_{b,cr}}{u_b}\right) = \left(\frac{\tau_{b,cr}}{\tau_b}\right)^{0.5} \tag{1.11}$$

where:

$\tau_{b,cr}$	= Critical bed shear stress
τ_b	= Bed shear stress due to current and waves = $\mu_c \tau_{b,cw} + \mu_w \tau_{b,w}$

resulting in

$$S_{b,n} = \left|S'_b\right| \alpha_{bn} \left(\frac{\tau_{b,cr}}{\tau_{b,cw}}\right)^{0.5} \frac{\partial z_b}{\partial n} \tag{1.12}$$

or

$$S_{b,n} = \left|S'_b\right| f_{norm} \tag{1.13}$$

where

$$f_{norm} = \alpha_{bn} \left(\frac{\tau_{b,cr}}{\tau_{b,cw}}\right)^{0.5} \frac{\partial z_b}{\partial n} \tag{1.14}$$

The two components of this vector are then added to the two components of the bed-load transport vector as follows:

$$\begin{aligned} S_{b,x} &= S'_{b,x} + \left|S'_b\right| f_{norm} \left(-\frac{S'_{b,y}}{\left|S'_b\right|}\right) \\ S_{b,y} &= S'_{b,y} + \left|S'_b\right| f_{norm} \left(\frac{S'_{b,x}}{\left|S'_b\right|}\right) \end{aligned} \tag{1.15}$$

Resulting in

$$\begin{aligned} S_{b,x} &= S'_{b,x} - S'_{b,y} f_{norm} \\ S_{b,y} &= S'_{b,y} + S'_{b,x} f_{norm} \end{aligned} \tag{1.16}$$

Where $S_{b,x}$ and $S_{b,y}$ are the components of the required bed-load transport vector, calculated at the water level points. The manner in which these components are used to calculate the morphological changes due to gradients in the bed-load transport is discussed in the following sections.

1.4 Inclusion of a fixed layer

Inclusion of a fixed layer in a DELFT3D simulation implies that the quantity of sediment at the bed is finite and may, if excessive erosion occurs, become exhausted and be unavailable to supply sediment to suspended and bed-load transport modes. This is taken into account in the bed-load formulations by comparing the thickness of sediment available at the bed with a user-specified threshold value. If the quantity of sediment available is less than the threshold then the magnitude of the calculated bed-load transport vector is reduced as follows

$$S''_b = f_{FIXFAC}\, S''_b \tag{1.17}$$

Where:

S''_b	= magnitude of the bed-load transport vector (before correction for bed slope effects)
f_{FIXFAC}	= fixed layer proximity factor = DPSED / [**THRESH**], limited to the range $0 \le f_{FIXFAC} \le 1$
DPSED	= depth of sediment available at the bed
[**THRESH**]	= user-specified erosion threshold

In effect, because of the upwind approach used to transfer the bed-load transport components to the U and V velocity points, this method limits the sediment that can leave a cell, if the quantity of the sediment at the bed is limited. One implication of the use of this rather simple approach is that a finite (although always less than the user-specified threshold) thickness of sediment is required at the bed if a non-zero magnitude of the bed-load transport vector is required.

Users should note that areas may be initially specified as containing zero bottom sediment if non-erodible areas are required. It is likely that these areas will accrete a little sediment in order to allow an equilibrium bed-load transport pattern to develop.

1.5 Calculation of bed-load transport at open boundaries

The bed-load transport vector at open boundaries is set perpendicular to the boundary and equal in magnitude to the component in this direction at the adjacent internal water level point. This approach is used as the bed slope cannot be properly evaluated at open boundaries. A consequence of using this approach combined with the upwind transfer of bed-load vector components to the U and V velocity points (described in the following

section) is that no gradient in bed-load transport can occur across the computational cell adjacent to inflow boundaries. We do not believe that this limitation presents any problem in practical situations.

1.6 Calculation of bed-load transport components at U and V velocity points

As the control volume for bed level change calculations is centred on the water level points (see Figure 2) the bed-load transport vector components are actually required at the U and V velocity points, rather than at the water level points where $S_{b,x}$ and $S_{b,y}$ are calculated (as described in section 1.2 above). We use a simple "upwind" numerical scheme to set the bed-load transport components at the U and V points as this ensures that the bed will remain stable and does not require a complex (and computationally expensive) bed updating algorithm. For each active velocity point the upwind direction is determined by summing the bed-load transport components at the water level points on either side of the velocity point and taking the upwind *direction* relative to the resulting net transport direction. The bed-load transport component at the velocity point is then set equal to the component computed at the water level point immediately "upwind" (Figure 1).

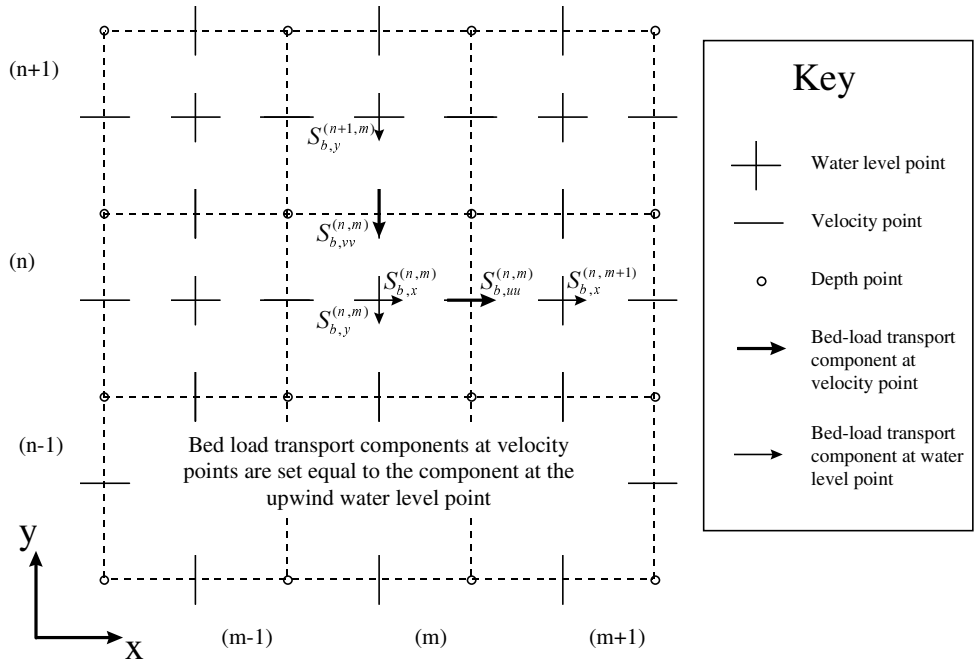


Figure 1 - Setting of bed-load transport components at velocity points

In the example shown in Figure 1 the bed-load transport component $S_{b,uu}^{(n,m)}$ is set equal to $S_{b,x}^{(n,m)}$ and the component $S_{b,vv}^{(n,m)}$ is set equal to $S_{b,y}^{(n+1,m)}$. This completes the calculation of the bed-load transport field. The transports at the U and V velocity points are then stored for use in the computation of bed level changes, as described in section 2 of this report.

2 Bathymetry Updating Including Bed-load Transport

Spatial gradients in the bed-load transport rate have an effect on morphological developments. These effects are now included in the modified version of DELFT3D-FLOW (version 009). First the bed-load transport vector is calculated and stored at the velocity (U and V) points, as described in section 1 above (Refer Figure 2).

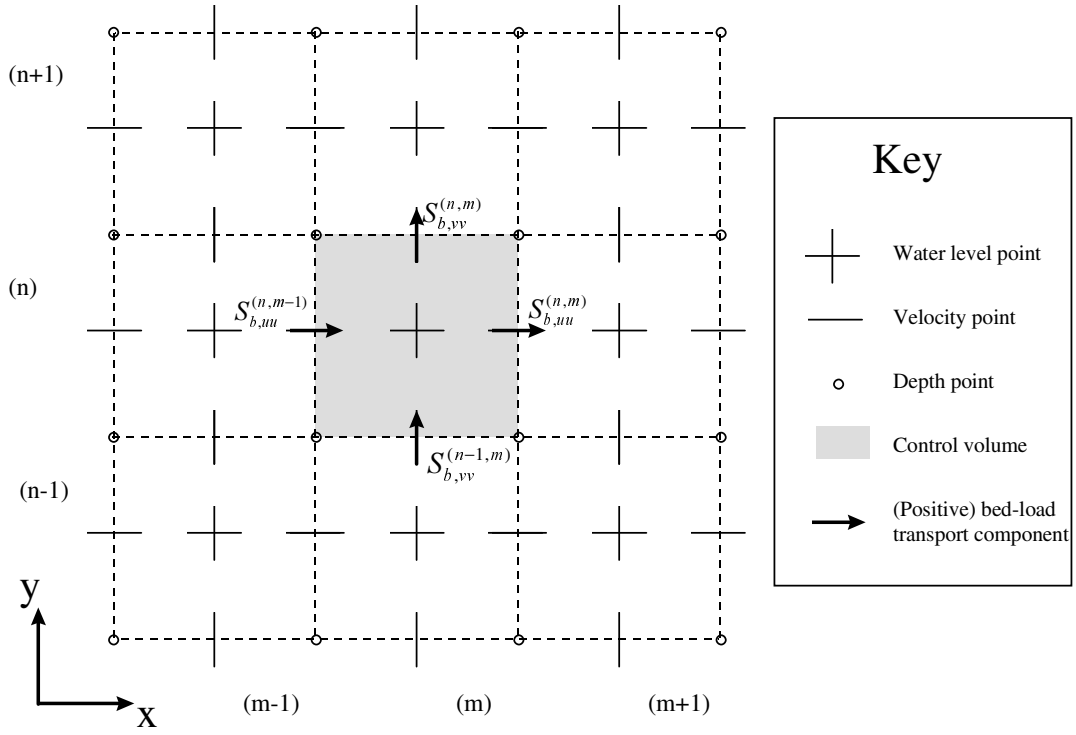


Figure 2 - Morphological control volume and bed-load transport components

The change in the quantity of bottom sediments caused by the bed-load transport is then calculated using the expression

$$\Delta_{SED}^{(n,m)} = \frac{\Delta t\, f_{MORFAC}}{A^{(n,m)}} \left(\begin{array}{l} S_{b,uu}^{(n,m-1)}\, \Delta y^{(n,m-1)} - S_{b,uu}^{(n,m)}\, \Delta y^{(n,m)} + \\ S_{b,vv}^{(n-1,m)}\, \Delta x^{(n-1,m)} - S_{b,vv}^{(n,m)}\, \Delta x^{(n,m)} \end{array} \right) \tag{2.1}$$

where:

$\Delta_{SED}^{(n,m)}$	=	change in quantity of bottom sediment at location (n,m) (Kg/m^2)
Δt	=	computational time-step (s)
f_{MORFAC}	=	user-specified morphological acceleration factor [**MORFAC**] (-)
$A^{(n,m)}$	=	area of computational cell at location (n,m) (m^2)
$S_{b,uu}^{(n,m)}$	=	computed bed-load sediment transport vector in *u* direction, held at the *u* point of the computational cell at location (n,m) (Kg/s/m)

$\Delta x^{(n,m)}$	=	cell width in the *x* direction, held at the V point of cell (n,m) (m)
$\Delta y^{(n,m)}$	=	cell width in the *x* direction, held at the U point of cell (n,m) (m)

This calculation is repeated for each 'sand' sediment fraction, if more than one is present, and the resulting change in the bottom sediment is added to the change due to the suspended sediment sources and sinks and included in the bottom updating scheme. The bottom is updated at every computational time-step.

3 Calculation of the Vertical Sediment Diffusion Coefficient

In version 009 of the modified DELFT3D-FLOW module the calculation of the vertical sediment diffusion coefficient has been significantly improved in situations including waves. In order to affect these changes the coefficient for the vertical diffusion of sediment particles is now stored separately from the vertical diffusion coefficient for momentum and all other constituents. The calculation of the vertical sediment diffusion coefficient now depends on the type of sediment, the choice of turbulence closure model, and whether waves are present. The various methods of calculating the vertical sediment diffusion coefficient are summarised below.

3.1 Using the algebraic or K-L turbulence model

3.1.1 Without waves

If the algebraic or K-L turbulence model is selected and waves are inactive then the vertical mixing coefficient for sediment is set directly from the vertical fluid mixing coefficient calculated by the selected turbulence closure model. In the case of 'sand' type sediment the fluid mixing coefficient is multiplied by van Rijn's 'beta factor' which is intended to describe the different diffusivity of a fluid 'particle' and a sand grain. Expressed mathematically:

$$\varepsilon_s = \beta \varepsilon_f \tag{3.1}$$

where:

ε_s	= vertical sediment mixing coefficient for the sediment fraction
β	= van Rijn's 'beta' factor for the sediment fraction. = 1.0 for cohesive sediment fractions.
ε_f	= vertical fluid mixing coefficient calculated by the selected TCM.

3.1.2 Including waves

If waves are included in a simulation using the algebraic or K-L turbulence closure model then the sediment mixing coefficient for 'sand' sediment fractions is calculated entirely separately from the turbulence closure model, using expressions given by van Rijn (1993) for both the current-related and wave-related vertical turbulent mixing of sediment.

The current-related mixing is calculated using the 'parabolic-constant' distribution recommended by van Rijn, namely:

$$\begin{aligned} z < 0.5h \qquad & \varepsilon_{s,c} = \kappa \beta u_{*,c} z(1 - z/h) \\ z \geq 0.5h \qquad & \varepsilon_{s,c} = 0.25 \kappa \beta u_{*,c} h \end{aligned} \tag{3.2}$$

where:

$\varepsilon_{s,c}$	=	vertical sediment mixing coefficient due to currents (for this sediment fraction)
$u_{*,c}$	=	current-related bed shear velocity

In the lower half of the water column this expression should produce similar turbulent mixing values to those produced by the algebraic turbulence closure model. The turbulent mixing in the upper half of the water column is generally of little importance to the transport of 'sand' sediment fractions as sediment concentrations in the upper half of the water column are low.

The wave- related mixing is also calculated following van Rijn (1993). In this case van Rijn recommends a step type distribution over the vertical, with a linear transition between the two steps (refer Figure 3).

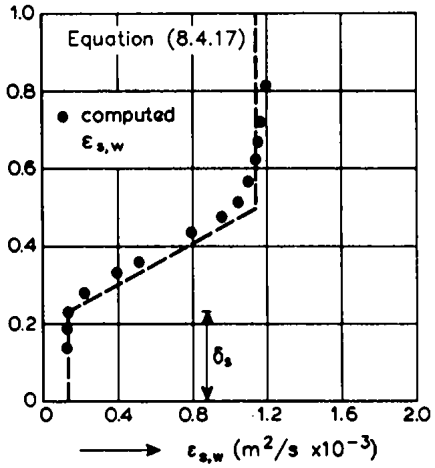


Figure 3 - Sediment mixing coefficient in non-breaking waves (Source: van Rijn 1993)

The expressions used to set this distribution are:

$$\begin{aligned} z \leq \delta_s \qquad & \varepsilon_{s,w} = \varepsilon_{s,bed} = 0.004 D_* \delta_s \hat{U}_\delta \\ z \geq 0.5h \qquad & \varepsilon_{s,w} = \varepsilon_{s,\max} = 0.035 h H_s / T_p \\ \delta_s < z < 0.5h \qquad & \varepsilon_{s,w} = \varepsilon_{s,bed} + \left[\varepsilon_{s,\max} - \varepsilon_{s,bed} \right] \left[\frac{z - \delta_s}{0.5h - \delta_s} \right] \end{aligned} \tag{3.3}$$

where δ_s (the thickness of the near-bed sediment mixing layer) is estimated using van Rijn's formulation:

$$\begin{aligned} \Psi < 250 \qquad & \delta_s = 3\Delta_r \\ \Psi \geq 250 \qquad & \delta_s = 3\delta_w \end{aligned} \tag{3.4}$$

where:

Ψ	=	mobility parameter ($\Psi = 250$ is the approximate transition from the ripple regime to the sheet-flow regime). = $\dfrac{(\hat{U}_\delta)^2}{(s-1)\,g\,d_{50}}$
Δ_r	=	estimated ripple height (as calculated for the wave-related roughness)
δ_w	=	thickness of the wave boundary layer = $0.072\,\hat{A}_\delta\left(\hat{A}_\delta / k_{s,w}\right)^{0.25}$
$\hat{U}_\delta$	=	near-bed peak orbital velocity
$\hat{A}_\delta$	=	near-bed peak orbital excursion
$k_{s,w}$	=	wave-related bed roughness (as calculated for suspended sediment transport).

We calculate the total vertical sediment mixing coefficient by following van Rijn and taking the sum of the squares:

$$\varepsilon_s = \sqrt{\varepsilon_{s,c}^2 + \varepsilon_{s,w}^2} \tag{3.5}$$

where ε_s is the vertical sediment diffusion coefficient used in the suspended sediment transport calculations for this sediment fraction. The above calculation of the sediment mixing coefficient is repeated for each sediment fraction.

We note that for cohesive sediment fractions the sediment mixing coefficient will still be set following equation (3.1) above. This implies that the extra turbulent mixing due to waves will not be included in the suspended sediment transport calculations (for these sediment fractions) except by way of the enhancement of the bed shear stress caused by wave-current interaction (refer to the DELFT3D-FLOW user manual for further details). This is a limitation of the present implementation.

3.2 Using the K-epsilon turbulence model

In the case of the K-epsilon turbulence closure model the vertical sediment mixing coefficient can be calculated directly from the vertical fluid mixing coefficient calculated by the turbulence closure model, using the following expression:

$$\varepsilon_s = \beta_{eff}\,\varepsilon_f \tag{3.6}$$

where:

ε_s	=	vertical sediment mixing coefficient for this sediment fraction.
β_{eff}	=	the effective van Rijn's 'beta' factor for the sediment fraction. As the beta factor should only be applied to the current-related mixing this is estimated as $1+(\beta-1)\frac{\tau_c}{\tau_w+\tau_c}$ 1.0 for cohesive sediment fractions.
β	=	van Rijn's 'beta' factor for the sediment fraction.
τ_c	=	bed shear stress due to currents
τ_w	=	bed shear stress due to waves
ε_f	=	vertical fluid mixing coefficient calculated by the K-epsilon turbulence closure model.

We note that this turbulence closure model now contains the improvements made by Walstra and Roelvink (2000) to include the three-dimensional effects of waves. However the effect of wave asymmetry on the bed-load transport is not yet included.

4 Bug fixes, simplifications and improvements

4.1 Presence of sediment made consistent with the constituent flag

A small bug has been fixed. If the constituent flag in the **.mdf** file is not set, then sediment properties are not read from the **sedinp** file.

4.2 Drying and flooding of cells improved

1. On initialisation, velocity points are set dry if the depth at the velocity point is less than the user-specified drying and flooding threshold [**DRYFLC**], rather than half this depth. This change is already made in the present standard release of DELFT3D.
2. The computational cell is set dry if the control volume is *less than or equal to* 0.0, rather than only if it was *less than* 0.0.

4.3 User selection of bed-load and suspended load

The user now has to select whether bed-load transport and/or suspended load transport should be taken into account when computing morphological changes. This is carried out by setting the two flags [**BED**] and [**SUS**] to either **.true.** or **.false.** in the **morph.inp** file.

We note that this option could be further improved by:

1. Allowing the user to specify a tuning parameter (real variable) instead of a simple **.true.** or **.false.** for each of the two transport modes. We suggest that this tuning parameter be used to multiply the constant in each of van Rijn's sediment expressions (one for the quantity of bed-load transport, and one for the reference concentration for the suspended-load transport calculations). Setting a value of 0.0 would effectively turn that transport mode off.
2. By placing much more of the code inside checks for these flags. There is currently a lot of wasted computational effort if, for example, a computation without bed-load transport is required.

4.4 Negative water depth check

In rare situations (with high morphological acceleration factors) it is possible that, in one time-step, the bed accretes more than the water depth. If this occurs the water depth will become negative (water surface level is below the bed level). This situation is now checked for and, if it occurs, the water surface level for the cell is set equal to the new bed level. The cell will then be set dry.

4.5 Error trapped in anti-creep routine

A slight restructuring of the code, and a minimum water depth of 0.1m have been included in the anti-creep routines (DIFHOR and DIFACR) to prevent 'divide by zero' errors from occurring in very shallow water.

4.6 Threshold depth for sediment calculations introduced

If the water depth in a cell is less than twice the critical depth for cell flooding or 0.1m (i.e. if $h < \max(2 f_{DRYFLC}, 0.1)$) then the sediment source and sink terms, and bed-load transport are not calculated for 'sand' sediment fractions. This has been included in order to prevent numerical problems during the computation of the reference concentration, in situations where van Rijn's formulations become invalid. The limitation of twice the critical depth for flooding is included to prevent sudden bursts of sediment from occurring when computational cells are flooded. Users simulating areas with very shallow water depths in which they would like sediment sources and sinks to occur should ensure that the user-specified parameter [**DRYFLC**] is not set too large.

4.7 Calculation of bed shear in wave and current situations altered

The calculation of the bed shear velocity (u_*) has been simplified in situations with waves and currents. The bed shear is now always calculated using the velocities computed in the bottom computational layer, rather than using the computational layer closest to the top of the sediment mixing layer (δ_m). This is a modification to appendix B of Part I of this report. The reference velocity in the bottom computational layer is now adjusted to the top of the sediment mixing layer using the apparent bed roughness (k_a) before being used to compute the bed shear velocity using the physical bed roughness (k_s). Refer Part I, appendix B for further details.

4.8 Improvements to 3D wave effects code

The implementation of 3D wave effects by Walstra and Roelvink 2000 has been improved as follows:

1. The effect of the wave angle relative to the computational grid has been properly implemented in a number of places
2. The wave angle calculation now uses the mass flux components to calculate the wave angle, rather than the wave dissipation components, as the wave dissipation should approach zero when wave breaking is not occurring.

4.9 Depth upwinding at velocity points improved

During a morphological simulation the depth stored at the U and V velocity points must be updated to reflect the bed level changes calculated in the water level points. This is performed by setting the new depth for the velocity point by copying the new depth held at

the water level point, using a simple upwind numerical scheme. A recent improvement involves the setting of the depth at velocity points where there is a zero depth-averaged flow velocity (usually these points are dry). The depth at such a point is now set to the *minimum* of the depths at the two adjacent water level points, rather than the maximum of these two values as used previously. This change appears to significantly improve the smoothness of flooding dry cells.

Users should note that the setting of depths at velocity points by upwind from the adjacent water level points only comes into effect if sediment is present and the user-specified flag [**MORUPD**] is **.true.** (ie bathymetrical changes are expected to occur at some point during the simulation period). If this condition is not met then the depths at the velocity points are not updated during the course of the simulation.

4.10 Initialisation of depths at velocity points

Initialisation of the depth held at velocity points is now consistent with the depth updating scheme described in section 4.9 above. If sediment is present and the user-specified flag [**MORUPD**] is **.true.** then the initial depths at velocity points are set from the depths at adjacent water level points, using the same upwind numerical scheme as is used for updating these depths later. If a simulation will not include bathymetrical changes then the depths at velocity points are set in the traditional manner (by interpolation between adjacent *depth* points) and are not updated during the course of the simulation.

4.11 Modifications related to time frame

In the case where the flow module is repeatedly called by MORSYS, the previous implementation did not produce continuous map and history files. This was due to the fact that the start and end time of writing to these files were adapted to the computation time interval each time FLOW was called. This has been modified, in the sense that a global time interval and increment is specified for writing to these files, which is kept unchanged during repeated calls from within MORSYS. Also, in the case of restarts, if a map and/or history file exists, the output can be appended to that file.

The following routines were modified to that end:

· TRICOM
· WRTHIS
· WRTMAP
· GETCEL (NEW)

The communication-file time frame still moves with the end time received from morsys. The rationale behind this is, that only the last or the last few flow fields have to be put on the communication file for use by other modules (i.e. the wave module).

5 Changes to MORSYS to allow 3D wave-current interaction and morphology

5.1 Introduction

In the FLOW model discussed up till now, morphological simulations can be carried out within the module, as long as wave effects are not included. When waves information is present on the communication file, this can be read and accounted for in FLOW, in the following ways:

1. enhancing bed shear stress
2. driving a wave-driven current by wave breaking and streaming
3. modifying turbulence
4. increasing stirring of sediment and bed load transport.

However, the computed flow field and the bottom changes have an effect on the waves, which must be accounted for. In order to allow an interaction between the FLOW and the WAVE module, two options are open:

1. to call WAVE from FLOW, or
2. to call WAVE and FLOW from MORSYS

The latter option allows more flexibility in the frequency of interaction and feedback to WAVE, and is the most in line with the present Delft3D-MOR approach. Therefore, this option was chosen. Some changes were still necessary, which are discussed below.

5.2 Updating MORSYS FLOW module

The changes in the FLOW routines related to sediment and 3D wave-driven currents were first implemented in the latest MORSYS module, viz. 4.02.003 of 19-06-00. This required careful comparison of the codes in order to preserve some recent changes made directly in the MORSYS FLOW routines. No major problems were encountered.

5.3 Interface from 3D FLOW to WAVE

For the time being, a relatively simple solution has been chosen. For each selected time point the full 3D flow field is read from the communication file, but only the upper layer is passed on to the WAVE module. This is done in the routine GETCUR. In later stages it will be quite easy to implement a more sophisticated interface, where some weighted average of the flow field over depth is chosen, depending on the wave parameters.

Various routines in the WAVE module have been adapted to check whether the flow field is 3D and to allocate memory accordingly.

5.4 Updating and passing on to com-file of depth DP

The depth in bottom points, DP, is the parameter which is passed on to the wave module. Since the morphological computation within FLOW is based on the depth in zeta-points, DPS, an additional action is required to update DP. This is done by interpolating the depth *changes* from DPS to DP. At the end of each call to FLOW, the updated DP is written to the communication file, hereby overwriting the old DP values. The WAVE module will pick up these DP values; also, the next time the FLOW module is entered, these DP values are read instead of the original bathymetry.

6 Validation

6.1 Validation of turbulent mixing formulations

To test the correct operation of the algebraic turbulence closure model in wave and current situations the experiments of Dekker and Jacobs (2000) are repeated, using identical settings to the runs performed earlier with the K-epsilon turbulence model, although this time the algebraic turbulence model is used. We note that, when waves are selected, use of the 'algebraic' turbulence model actually results in the application of the formulations of van Rijn (1993) for the vertical distribution of turbulent mixing. Refer to section 3 of this report.

The set-up of this experiment is fully described in Part I of this report. All three experiments that were simulated with the K-epsilon turbulence model are repeated with the algebraic turbulence model. The following sections show comparisons between the results achieved using the two turbulence models.

6.1.1 Wave-only test

Figure 4 shows the computed and measured velocity profiles for this experiment. The effect of the wave streaming stress exerted at the bed is more pronounced in the computation with the algebraic turbulence model, this tends to imply that, near the bottom, the turbulent mixing predicted by this model is substantially less than that predicted by the K-epsilon turbulence model. Both results are reasonably close to the measurements however.

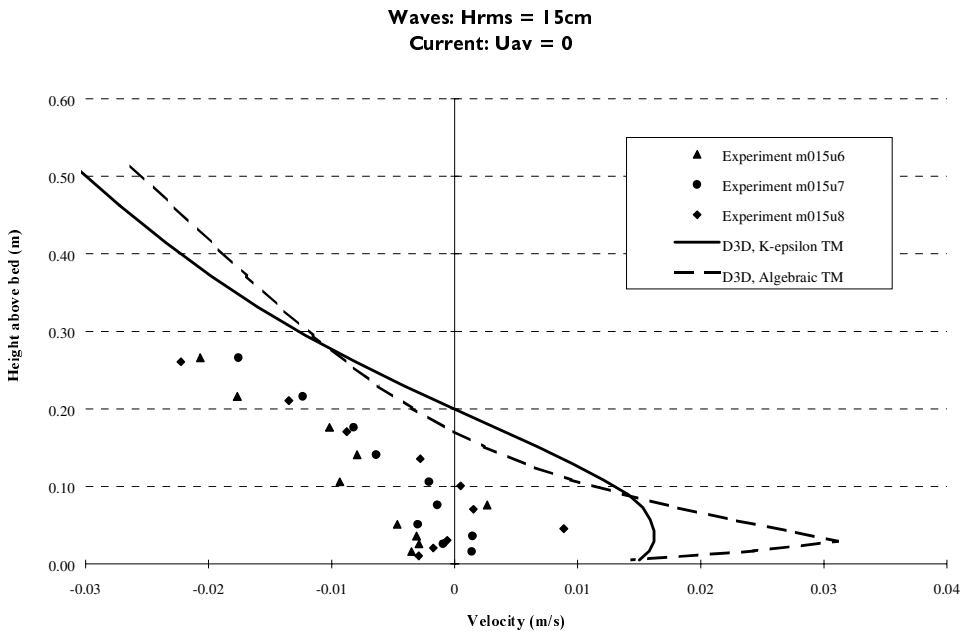


Figure 4 - Computed and measured velocity profiles

Figure 5 presents the computed and measured sediment concentration profiles for this experiment, the turbulent mixing profile produced by the K-epsilon turbulence model is also included. It can be seen that the results of both computations are very close to the measurements, the results of the computation using the algebraic turbulence model are actually somewhat better than those using the K-epsilon turbulence model in this situation.

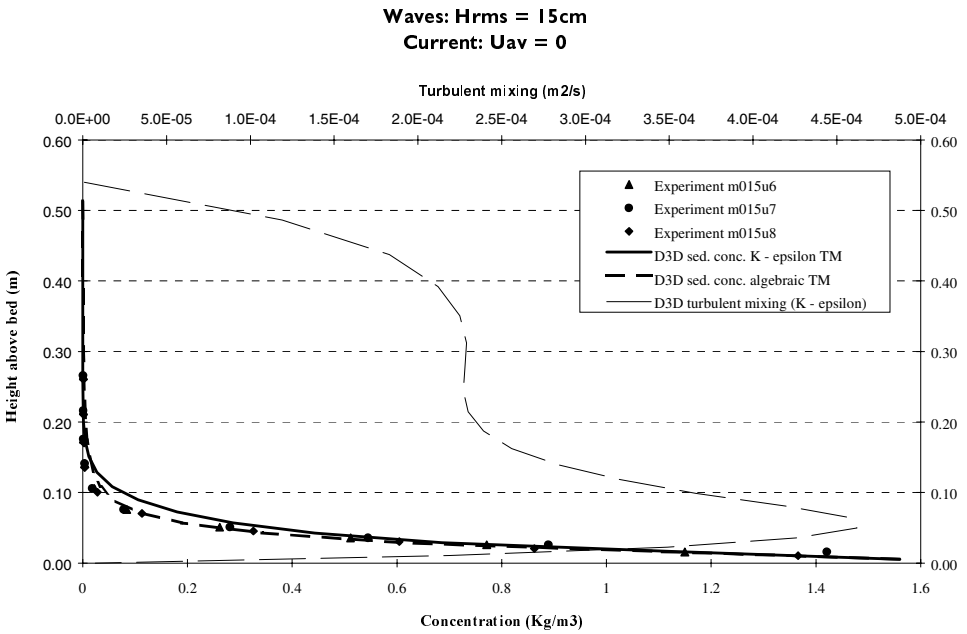


Figure 5 - Computed and measured concentration profiles

6.1.2 Waves with 0.2m/s following current

Figure 6 shows the measured and computed velocity profiles for the second experiment, which includes a small following current. Also shown in the figure is the computed velocity profile for the same experiment performed without the waves being present. The difference between this line and the 'algebraic' turbulence model including waves shows the effect of the Stokes distribution of the wave-induced mass flux - a relatively large retarding of the velocity near the surface reducing to zero at the bed.

Compared with the measurements the bed roughness appears to be rather under estimated for all the simulations.

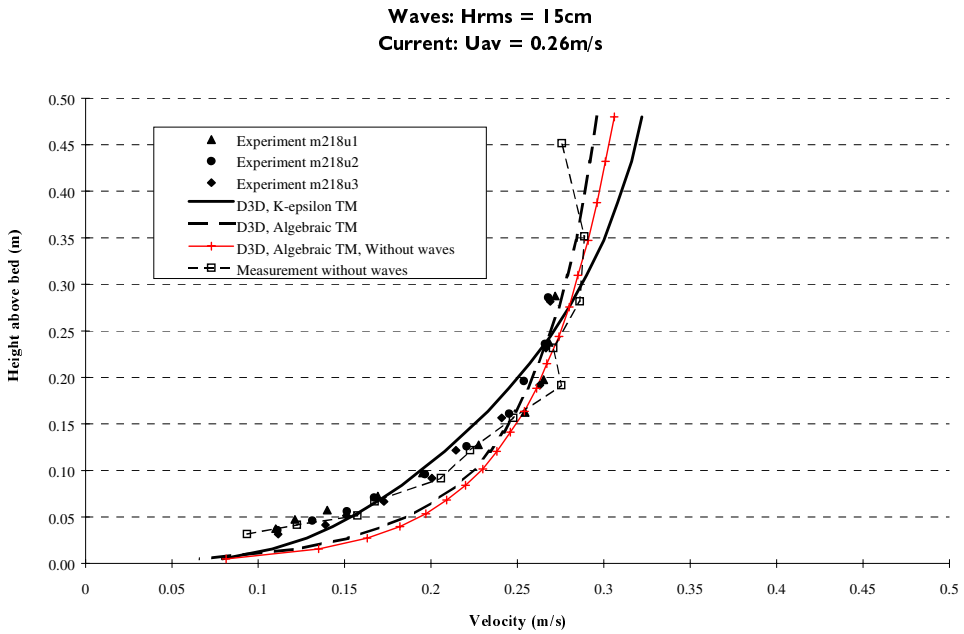


Figure 6 - Computed and measured velocity profiles

Figure 7 shows the computed and measured sediment concentration profiles for this experiment. The two computed curves are reasonably close together, although the 'algebraic' turbulence model appears to predict somewhat more turbulent mixing than the K-epsilon turbulence model near the mid-height of the water column. The effect of the increased near-bed velocities visible in Figure 6 above is clearly visible as an increased reference concentration in the case of the 'algebraic' turbulence model.

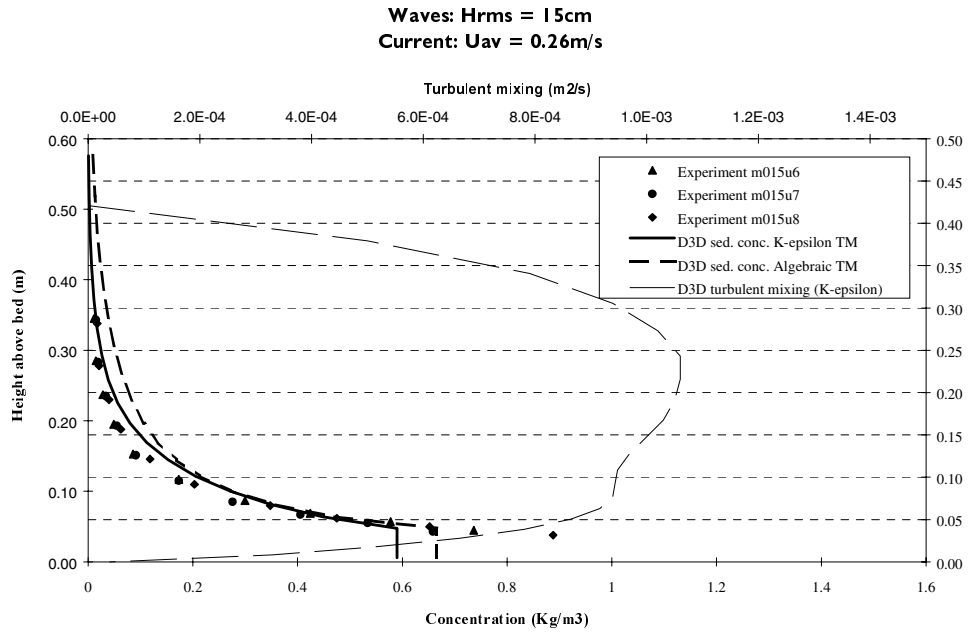


Figure 7 - Computed and measured concentration profiles

Figure 8 shows the influence of the number of computational layers on the computed sediment concentration profile, to produce this figure the computation is repeated with 10

and 5 computational layers in place of the 20 layers used in the base case. It can be seen that in this case, using the 'algebraic' turbulence model, there is no discernible difference between the simulation performed with 20 layers and that performed with 10 layers. A further reduction to 5 layers produces a fairly rough representation of the concentration profile, however we feel that the result is as accurate as can be expected. We believe that in general a minimum of about 7 layers should be used for a three-dimensional simulation using the algebraic turbulence model.

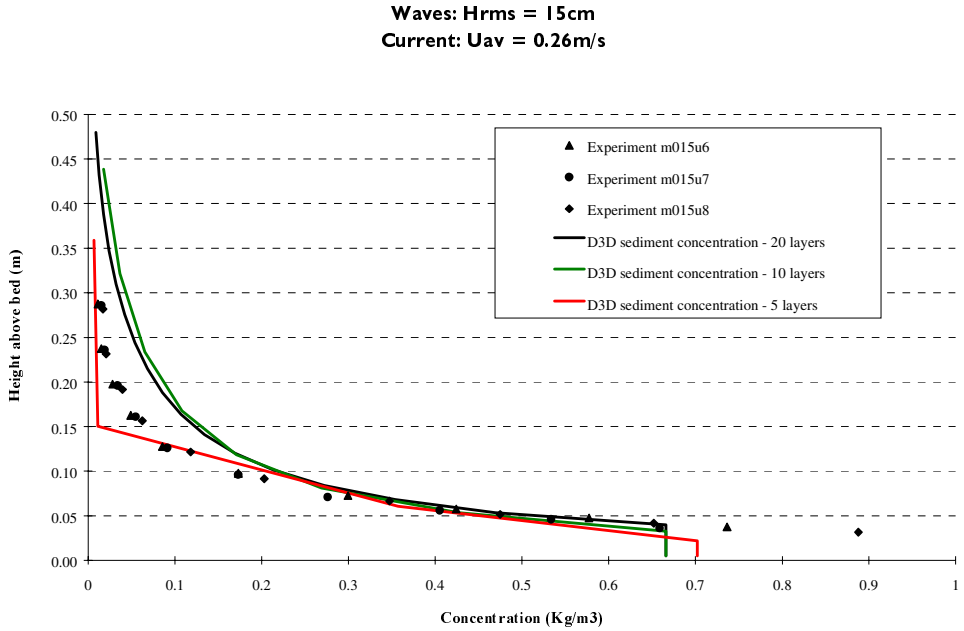


Figure 8 - Influence of the number of computational layers on the concentration profile

6.1.3 Waves with 0.4m/s following current

Figure 9 shows the measured and computed velocity profiles for the third experiment, which includes a larger following current. It is clear that the velocity profile for the 'algebraic' turbulence closure model is somewhat steeper than for the K-epsilon turbulence model, this indicates that a somewhat higher level of turbulent mixing is predicted by the 'algebraic' turbulence model. Again it appears that the bed roughness may be under-estimated somewhat in both simulations.

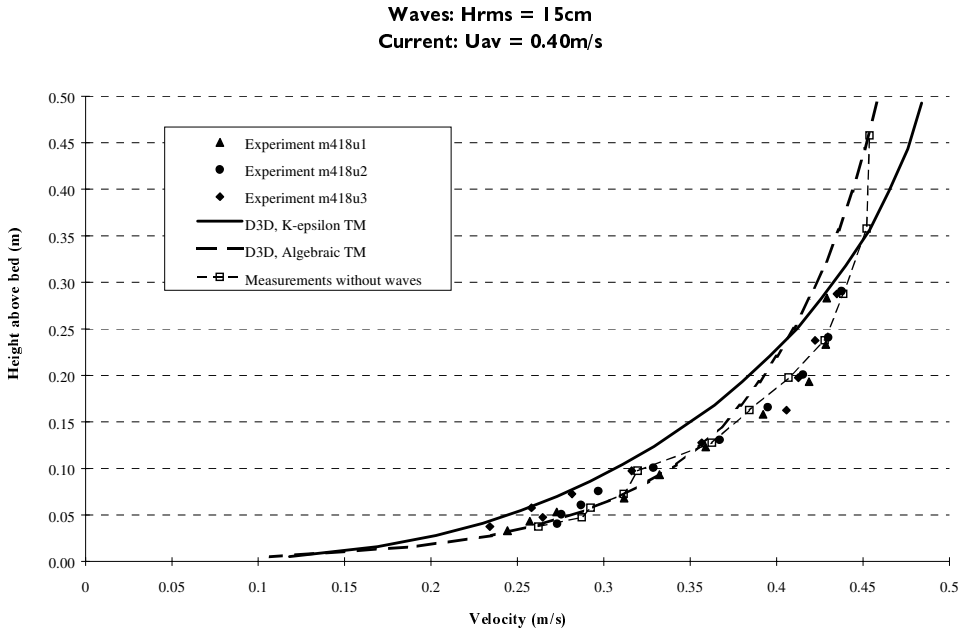


Figure 9 - Computed and measured velocity profiles

Figure 10 shows the computed and measured sediment concentration profiles for this experiment. The higher turbulent mixing produced by the 'algebraic' turbulence model at mid-depth is visible when comparing the gradients of the two concentration profiles. More dominant, however, is the effect of the higher near-bed velocity in the case of the 'algebraic' turbulence model run. This has the flow-on effect of producing a significantly higher bed shear stress and reference concentration.

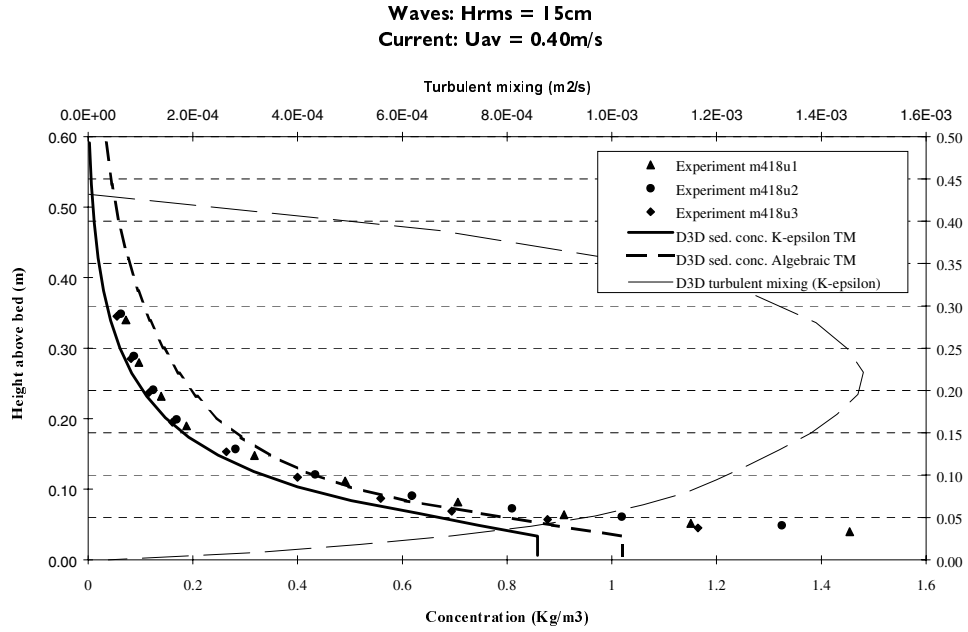


Figure 10 - Computed and measured concentration profiles

6.1.4 Conclusions

- Overall, the inclusion of van Rijn's expressions for the vertical turbulent mixing coefficient for 'sand' sediment in wave and current situations in place of the algebraic turbulence model when waves and currents are active appears to produce very good results in the situations tested.
- In general van Rijn's expressions appear to produce slightly too much turbulent mixing near mid-depth.
- The difference in turbulent mixing can have a relatively large indirect effect on the sediment concentration profile by changing the velocity profile and, therefore, the calculated bed shear stress and reference concentration.
- The modified 'algebraic' turbulence model appears to produce reasonable concentration profiles even with as few as 5 layers. This may be very useful for simulations of extensive coastal zones where the 20 layers required to produce a meaningful result with the (more accurate) K-epsilon turbulence model would require a prohibitive level of computational effort.
- A minimum of 7 layers is recommended when using the 'algebraic' turbulence model in wave and current situations.

6.2 Validation of bed-load transport

The trench migration experiment presented in Part I of this report is repeated, this time with bed-load transport included. The two simulations are repeated with all parameters left at their previous settings, except for the morphological acceleration factor MORFAC which is reduced to 90 (and the simulation period extended accordingly). We find that it is necessary to reduce MORFAC because in this case there is a very high bed-load transport rate and the bed-load transport does not behave as smoothly as the suspended load. We find that a reduced MORFAC is required to prevent bottom wiggles from developing.

As with the previous trench migration simulations the simulation using the 'algebraic' turbulence closure model is performed with 10 computational layers. The simulation using the K-epsilon turbulence model uses 20 layers.

Figure 11 presents the morphological changes computed after 5, 10, and 15 morphological hours, van Rijn's measurements of the bed elevation after 15 hours are also presented. Comparing the results with the suspended-sediment-only results in Part I of this report we see that, as expected, the inclusion of bed-load transport does add significantly to the quantity of sediment that accumulates in the trench.

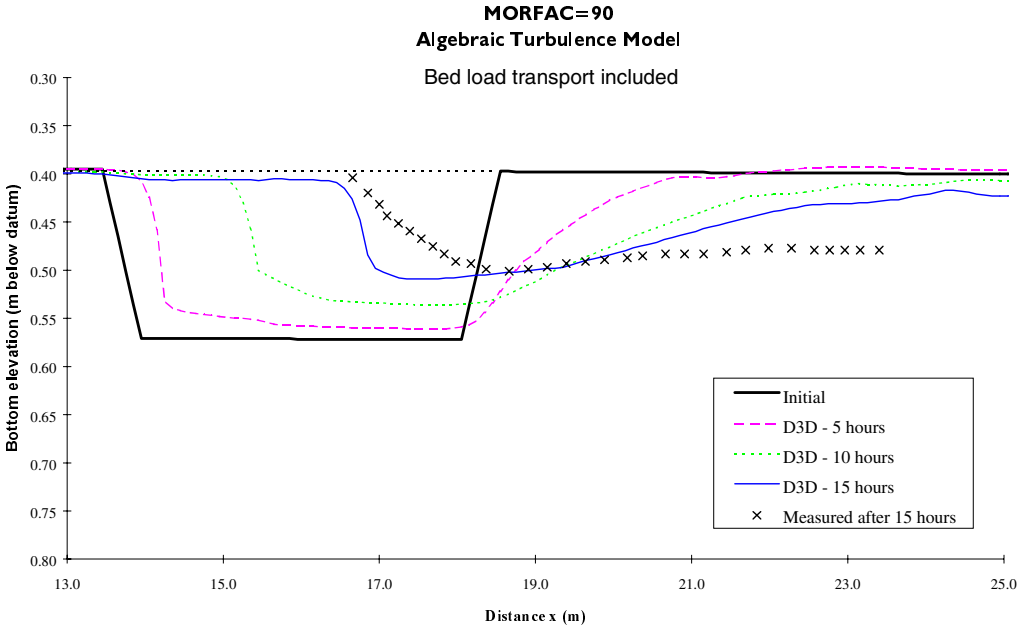


Figure 11 - Computed morphological changes using the algebraic turbulence model

Overall the computed morphological development is in good agreement with the measurements. After 15 hours the trench appears to have moved very nearly the correct distance, and is filled to the correct level. Less well simulated are the entry and exit bed slopes. We believe that the overly steep entry slope is due to the fact that deterministic (rather than stochastic) pick-up and bed-load formulations have been implemented, these do not take into account the significant turbulent eddies that would be present in this location. We believe that it is the action of these turbulent eddies that are largely responsible for

smoothing the lower portion of this entry slope. The difference in the exit slope is more puzzling. This may be partially caused by the steep entry slope causing flow separation at a point where it should in fact be following a much smoother entry.

Figure 12 presents the results of a similar run using the K-epsilon turbulence model. These results show very similar trends to the algebraic simulation. After 15 hours the overall position and depth of the migrated trench are in good agreement with the measurements, again the entry slope is rather too steep, and again the exit slope is visible whereas there is no sign of it appearing in the measurements. The spike that appears in the bed at the top of the entry slope after 5 hours is due to the high morphological acceleration factor combined with the high gradient in bed-load transport rate which occurs at this point. This spike is gradually damped out over time

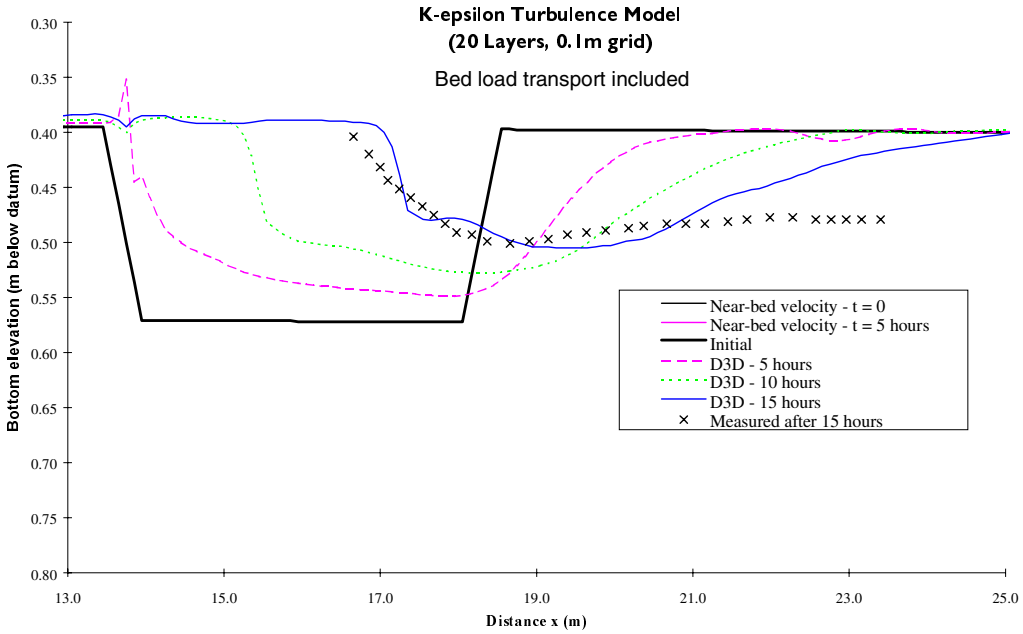


Figure 12 - Computed morphological changes using the K-epsilon turbulence model

6.3 River bend cut-off simulation

The modified flow module has been applied to a river bend cut-off simulation performed by Ir. Bert Jagers of WL | DELFT HYDRAULICS. The initial bathymetry of the simulation is shown in the upper pictures of Figure 13, during the course of the simulation the discharge at the upstream (far) boundary is steadily increased until the flat area in the middle of the river bend is inundated and the river attempts to cut a new channel.

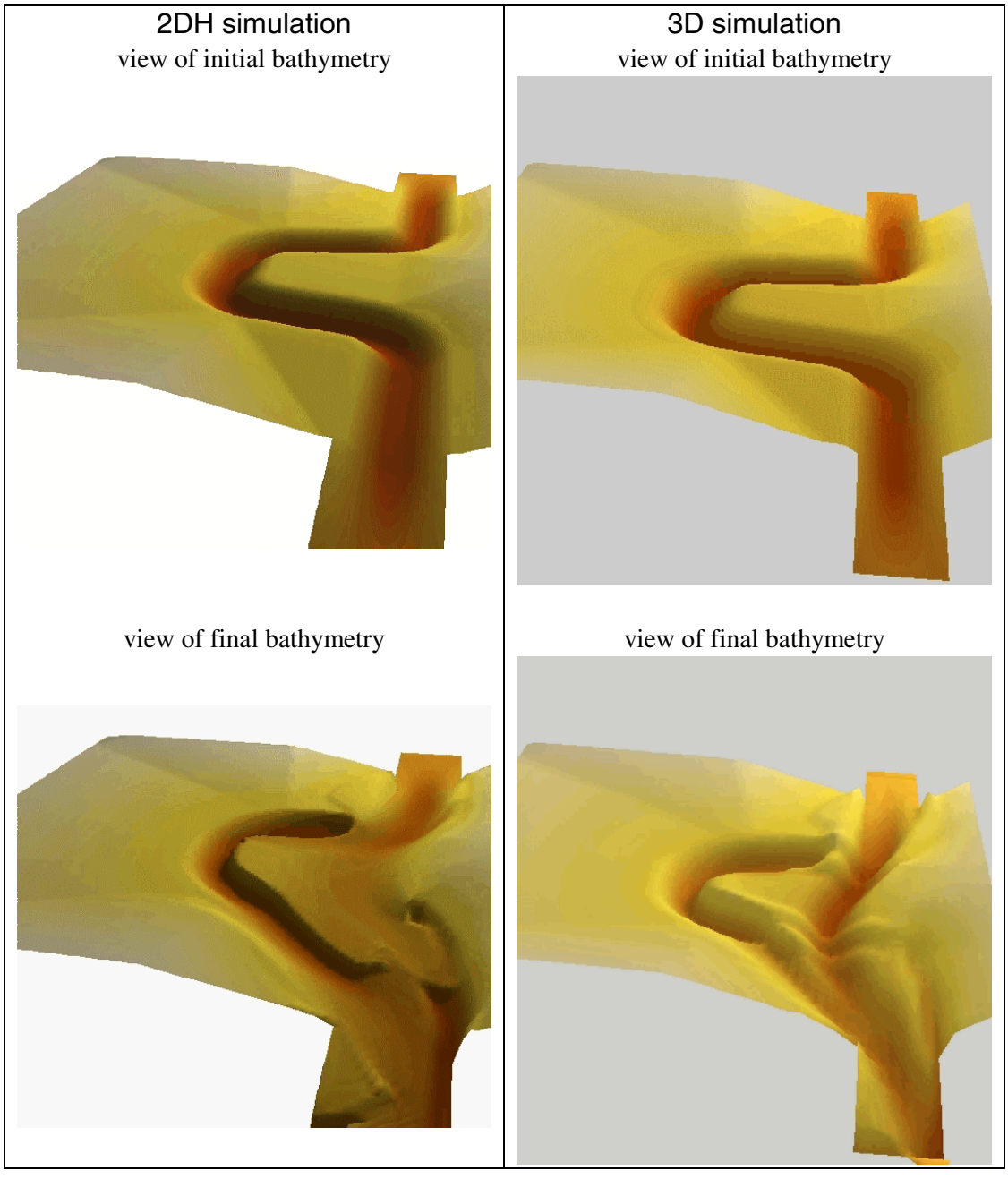


Figure 13 - Comparison of 2DH and 3D results of river bend cut-off simulation

The simulation was originally performed with the DELFT3D-MOR module which performs depth averaged sediment transport calculations, the resulting final bathymetry is shown on

the left. The simulation was also performed using the modified DELFT3D-FLOW module, using 10 computational layers and the algebraic turbulence model. The results of this simulation are shown in the right hand figures. This simulation provides a useful test of the flooding and drying procedures in the modified FLOW module. It also provides an interesting comparison of the morphological developments predicted by two different morphological modelling systems, using different sediment transport formulations. Obviously there are significant differences between the results predicted by the two systems. As there is no 'correct' solution to this problem we feel that the reasons for these differences should be investigated more thoroughly, in order to gain more confidence in the results provided by either system.

6.4 Offshore breakwater case (Nicholson et al., 1997)

6.4.1 Introduction

As a first test of the combined modelling of waves, currents and morphological changes in 3D, a relatively simple test case from literature was taken, viz. the offshore breakwater test reported in Nicholson et al. (1997). The main characteristics of this test case are:

Beach slope	1:50
Breakwater length	300 m
Breakwater axis-to-shore distance	220 m
Sedimenty grain size	250 μm
Peak energy wave period	8.0 s
Incident rms wave height	2.0 m

In Nicholson et al., this test case was run using five different 2DH morphodynamic models, among which Delft3D-MOR. The only driving force in the tests are the waves, which enter perpendicular to the coast. The waves drive a double circulation pattern which tends to bring sand to the area behind the breakwater, leading to the formation of a tombolo or salient.

6.4.2 Model set-up

A 20 m (longshore) by 10 m (cross-shore) rectangular grid was used for the flow model, and a 20 m (longshore) by 5 m (cross-shore) computational grid for the wave model. The model domain was chosen 1300 m (longshore) by 700 m (cross-shore). Since the waves were perpendicular to the beach, all flow model boundaries could be closed boundaries, and for the wave model the side boundaries were chosen to be reflecting, which gives minimal disturbances.

The time step for the flow model was set at 6 s. A morphological time-scale factor of 24 was used, so that 1 hour in the flow model actually represents 1 day of morphological changes.

The algebraic turbulence model was chosen, with a vertical grid distributed as follows:

Layer	relative thickness (%)
1 (top)	10
2	20
3	35
4	20
5	10
6 (bottom)	5

The waves were updated every 10 time steps (1 minute in flow, 24 minutes morhological time).

6.4.3 Results

The initial flow field is shown in Figure 14. It shows the overall pattern of two circulation cells in opposite directions, which are driven by set-up gradients. On top of that, we see a clear difference in flow pattern between the top and bottom layers, due to:

1. helical flow, which pushes the upper-layer velocities outward;
2. undertow, which gives an seaward component near the bed in the surf zone.

In Figure 15 the sedimentation/erosion pattern after one day of morphological change is shown, together with the final flow pattern. There is a clear tendency for accretion behind the breakwater, with the main part relatively close to the breakwater. Significant erosion occurs near the beach on either side of the breakwater. Since the dry beach cannot be eroded, the underwater profile near the beach a the erosion locations becomes very steep; after approx. one day, the steep profiles lead to unrealistically high velocities and the simulation becomes unrealistic.

The final bathymetry is shown in Figure 16. At first sight, the results are quite similar to the results from 2DH models, as given in Nicholson et al. (1997). A big difference however is that the accretion behind the breakwater goes at the expense of erosion at the beach at the sides of the lee zone, whereas in 2DH models the erosion takes place mainly near the ends of the breakwaters. In the 3D case, some erosion takes place there, but not nearly as much as in the 2DH case.

The main reason for this different behaviour is the undertow, which is not included in the 2DH computations. As wave asymmetry, a ‘beach-building’ process, is not included yet, the effect will be somewhat exaggerated.

We may conclude that these results are quite promising in that it is the first application of a fully 3D model which combines waves, flow, suspended and bed load transport and morphological changes, in a smooth and realistic simulation.

We recommend that a computationally efficient wave asymmetry transport component be implemented, as well as an algorithm which will allow the dry beach to follow the developments underwater in the case of erosion.

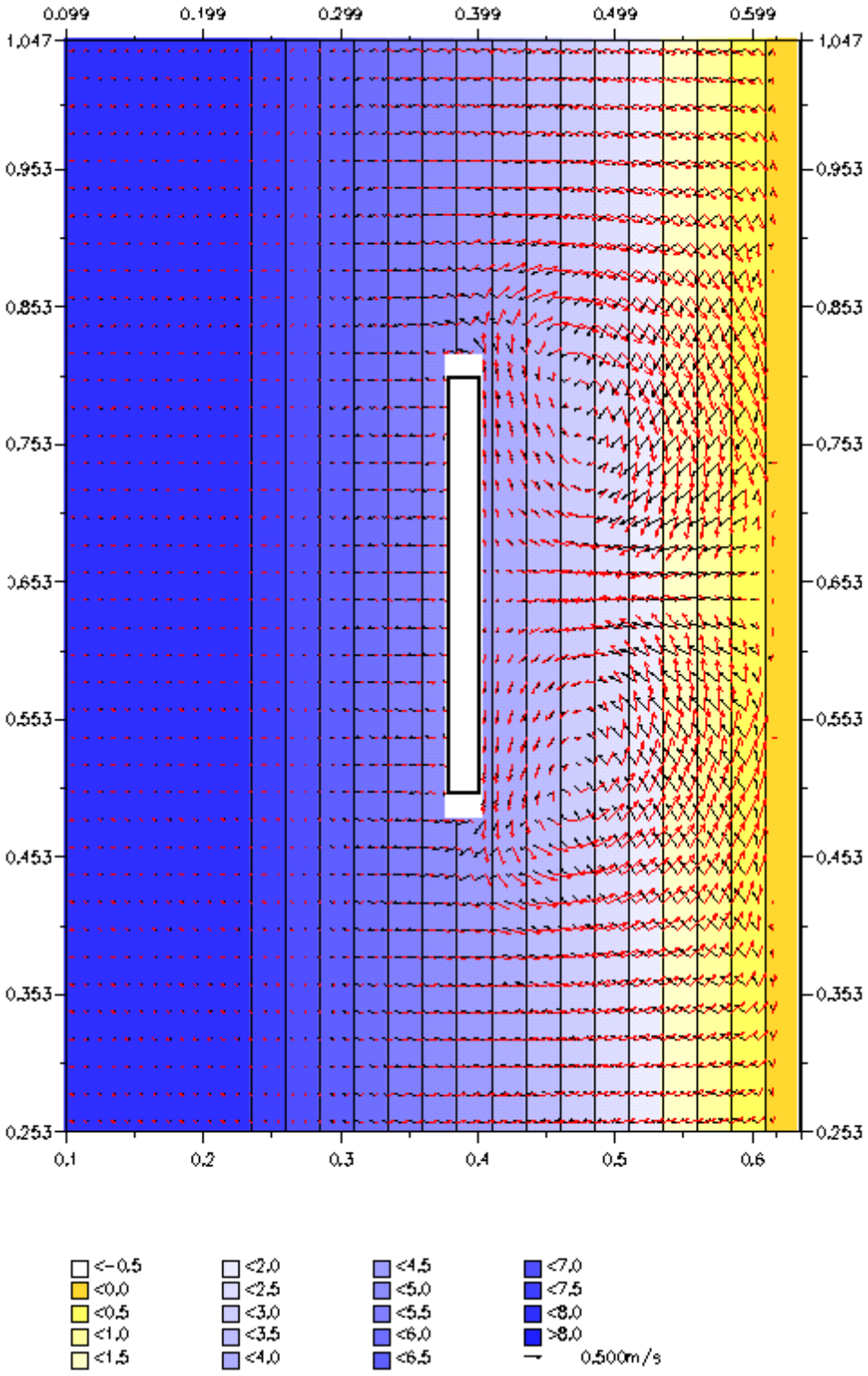


Figure 14 - Offshore breakwater case. Initial bathymetry and flow pattern in top layer (red) and bottom layer (black).

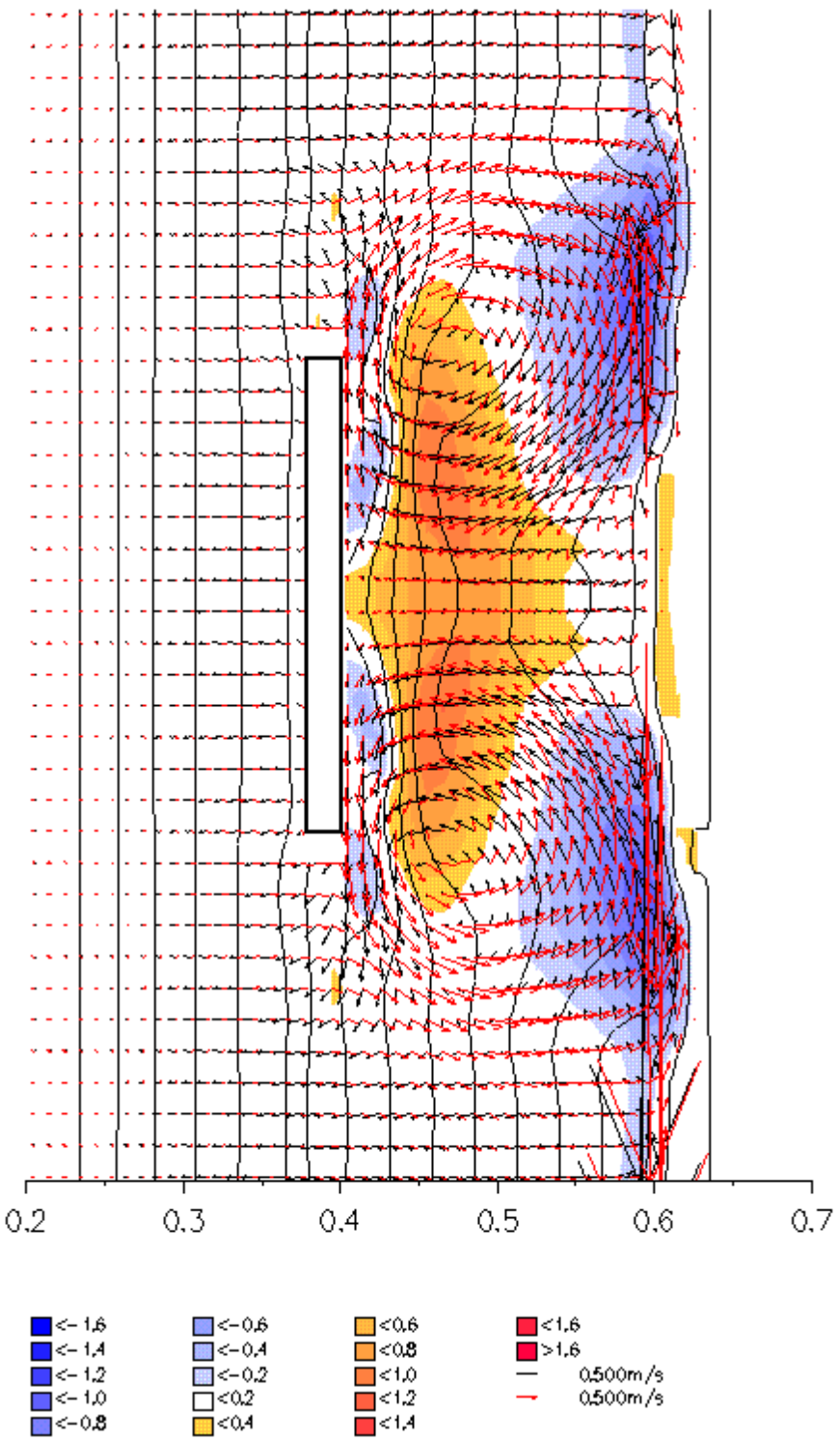


Figure 15 - Final velocity pattern (top: red; bottom: black); bathymetry after 1 day (isolines) and sedimentation / erosion pattern (orange/blue contours)

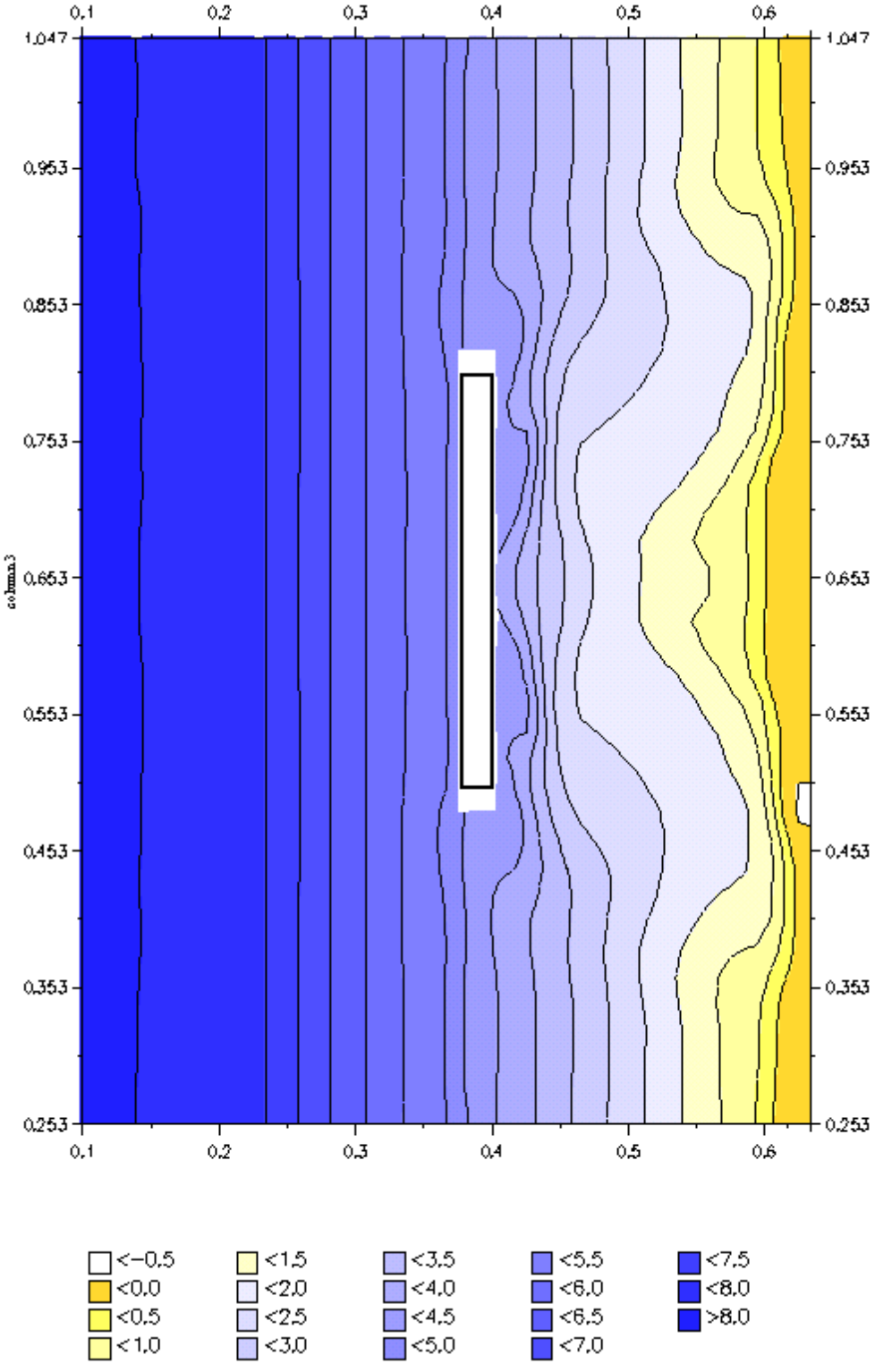


Figure 16 - Bathymetry after one day of morphological changes.

Conclusions

In this part of the study, Van Rijn's (1993) bed load transport formulations, including bed slope effects, have been added to the transport of sand. The bed load transport reacts locally to the near-bed current and orbital velocity. A robust upwind numerical scheme has been implemented for the morphological changes due to gradients in the bed load transport. Comparison with the morphological development in the trench test shows considerable improvement compared to the situation when only suspended transport is considered.

The wave effects on the vertical diffusivity profile have been implemented in the algebraic and k-L model, according to the formulations of Van Rijn. The formulations of Walstra et al. (2000) have been used in the case of k-ε turbulence model. Excellent agreement is found for the concentration in the Dekker and Jacobs (2000) tests, both for algebraic and k-ε model.

The effect of fixed layers has been implemented and tested for both suspended load and bed load transport.

The flow module in its present form can act as part of the morphological modelling system Delft3D-MOR. This is useful in case of waves, when the flow/sed model and the wave model are called alternately. This has been tested successfully for the offshore breakwater case.

Recommendations

The model in its present state of development is a very useful addition to the depth-averaged morphological tools. However, the testing against real data so far has been only for 2DV cases. For more complex. 3D situations some intercomparisons have been made between 2DH or Q3D tests with Delft3D-MOR and the present version:

- river bend cutoff (3D vs Q3D)
- Western Scheldt (3D vs Q3D)
- Offshore breakwater (3D vs 2DH)

Significantly different results are found, e.g. in the river bend case, where the 3D model predicts much more pronounced channel formation, and in the offshore breakwater case, where the 3D model predicts more serious erosional hotspots. At this stage, neither model can be proven wrong for lack of data. A systematic series of 3D laboratory tests should be carried out, with and without waves, to generate a set of validation data sets under completely ontrolled circumstances.

Comparison with field datasets should be carried out where possible, to assess the overall performance of he model in real cases.

The cohesive sediment formulations and implementation should be tested in the same rigourous fashion as has been carried out for the sand transport and morphology.

At present, wave effects on the vertical mixing of *cohesive* sediment have been implemented for k-ε only.

Wave asymmetry effects should be included in the bed load transport to obtain more balanced cross-shore transport.

A roller model should be implemented to improve the cross-shore distribution of driving forces.

Formulations for the erosion of dry banks should be studied and implemented.

The effect of turbulence on sediment moblisation can be improved.

For quick computations with compatible formulations, depth-averaged suspended transport should be implemented.

Some practical recommendations are the following:

- The input and output of the model, including User Interface and GPP, should be adapted for the sediment version. This includes writing of suspended and bed load transport vector fields.
- instead of having on/off switches for bed load and suspended load, these may be replaced by calibration factors.