



UNIVERSITÀ DI PISA

UNIVERSITA' DI PISA
DIPARTIMENTO DI INGEGNERIA CIVILE ED INDUSTRIALE

TESI DI LAUREA SPECIALISTICA IN INGEGNERIA
AEROSPAZIALE

IMPLICIT SIMULATIONS OF MORPHODYNAMIC SHALLOW WATER FLOWS

RELATORI

Prof. Ing. Maria Vittoria Salvetti

ALLIEVO

Vincenzo Sportiello

Anno accademico 2011-2012

Abstract

Implicit and explicit simulations of morphodynamical flows are considered. The physical problem is modeled through the shallow water equations coupled with the Exner equation which describes the time evolution of the bed profile. The formulae used to express the sediment transport fluxes are those of the Grass model and of the Meyer Peter Müller one. The spatial discretization of the equations is obtained using a finite-volume method and a modified Roe scheme for non conservative systems. In this context previous studies investigated the accuracy and efficiency of linearized implicit schemes, carried out with a strategy based on automatic differentiation to compute the flux Jacobians and on the defect correction approach to reach second-order accuracy. These studies were carried out for Froude number equal to 0.1, and it was shown that implicit time advancing was more efficient than the explicit one for slow and intermediate rates of interaction between the bedload and the flow.

The present study focuses on the same analysis by varying Froude numbers. In particular the relationship between the maximum CFL number allowed by result accuracy in the implicit simulations and the parameter that represents the speed of interaction between the flow and the sediment transport has been investigated at increasing Froude numbers. It is shown that implicit solutions become less convenient when the Froude number increases, but some differences exist depending on the sediment model used and the way of varying the Froude value. The A_g - Fr couples for whom implicit solutions are convenient have been identified. The last part is about similar investigations for 2D tests.

Contents

1	Literature survey and Introduction	5
1.1	Numerical method	6
1.2	Sediment transport formulae	8
1.3	Experiments and results	8
1.4	Aim of the present work	9
2	Physical Model	11
2.1	Shallow Water Equations	11
2.2	Sediment Transport Models	15
2.2.1	Grass Model	16
2.2.2	Meyer-Peter and Müller Model	17
3	Numerical Method	21
3.1	General definitions for the considered finite-volume formulation	21
3.2	A Roe-type scheme for non-conservative systems	23
3.2.1	Second-order extension	24
3.3	Explicit time advancing	25
3.4	Implicit time advancing	26
3.4.1	First-order scheme	26
3.4.2	Second-order DeC scheme	28
3.5	CFL condition	30
4	Numerical experiments	31
4.1	Presentation of the 1D Problem	31

4.1.1	Comparison between the solutions obtained with different sediment transport models	33
4.1.2	Explicit vs. implicit time-advancing schemes	33
4.2	Varying the Froude number	38
5	Variation of the Froude number by acting on u	39
5.1	Efficiency comparison	40
5.2	Froude = 0.2	42
5.3	Froude = 0.3	45
5.4	Froude = 0.4	47
5.5	Froude = 0.5	49
5.6	Froude = 0.6	51
5.7	Froude = 0.7	53
5.8	Concluding annotations	55
6	Variation of the Froude number by acting on h	57
6.1	Simulations for the Grass model	57
6.2	Efficiency comparison	58
6.3	Froude = 0.2	60
6.4	Froude = 0.3	62
6.5	Froude = 0.4	64
6.6	Concluding annotations	66
6.7	Simulations for the MPM model: efficiency accuracy	68
6.8	Froude = 0.2	70
6.9	Froude = 0.3	72
6.10	Froude = 0.4	74
6.11	Concluding annotations	75
7	Application to a contracting channel	77
7.1	The test case	77
7.2	Mesh definition	78
7.3	Results and discussion	81
7.4	Concluding remarks	85

8	Conclusions	88
A	Sediment transport formulae	90
	Bibliography	92

Chapter 1

Literature survey and Introduction

The problem of Shallow Water flows is recurring in coastal and environmental engineering since studies on this topic can have a lot of interesting implications for these matters. Thus the literature is full of articles, studies and so on about it, and it is always sensitive to new developments and tests. This chapter contains a review of some of the recent approaches to such a problem, in order to make a comparison with the models used and the results obtained in the present work.

First, the traditional Shallow Water problem focuses just on the hydrodynamical aspect because the bed-load is considered rigid, so no sediment transport is involved. In order to consider the bedload evolution an other additional equation must be added to the SWEs. Therefore, there is the morphological part of the problem to be considered in addition to the hydrodynamical one. At this point, it is possible to make a separation from those approaches that consider the two aspects to be separated to those where the two components are put together obtaining a coupled model. We focus on this category of studies, analyzing then some characteristic in particular: the numerical model used, the sediment transport formulae adopted and the results were carried out.

1.1 Numerical method

The system composed by the SWEs and by the equation that takes into account the sediment transport, most of the times expressed by the Exner equation [see for example 1, 29, 34, 37, 38], is rarely solved analytically because of its complexity, so a numerical method is almost always used. Of course, there is not a unique method to get a solution of this problem.

The well-established **Roe's** scheme has been modified for sediment transport problems [5, 11, 12, 13]. One of the drawbacks of this method could be an increase of computational costs due to its treatment of the source terms. Some authors extended the **ENO** (Essentially Non Oscillatory) and **WENO** (Weighted Essentially Non Oscillatory) schemes to sediment transport problems [15] while others used the **CWENO** (Central Weighted Essentially Non Oscillatory) scheme [8]. The central idea of these schemes is to compute the smoothness for several possible stencils which cover a given region of space and then interpolate this zone with high order of accuracy using this types of reconstruction via primitive function, developed by many authors. Unfortunately, most ENO, WENO and CWENO method, that solve real two-dimensional sediment transport problems accurately are still very computationally expensive.

Numerical methods using **discontinuous Galerkin** techniques have also been studied [30]: this method is a robust finite element method that is particularly well suited for advection dominated transport equations. It incorporates upwind numerical fluxes and slope limiters to provide sharp resolution of steep bathymetric gradients that may form in the solution, and it possesses a local conservation property that conserves sediment mass on an element level.

Numerical method based on **relaxation** approach have also been applied to sediment transport equations in [16]. Using this method, the nonlinear set of equations is transformed to a semilinear diagonalizable problem with linear characteristics variables. The relaxation stage is solved by an implicit-explicit Runge-Kutta scheme while a second order MUSCL-TVD (Total Variation Diminishing) method is used for the advection stage. The principal advantages

of this approach are that neither Riemann problem solvers nor nonlinear iterations are required during the solution process.

However, most of these methods for unstructured grids present results with an order of accuracy smaller than that expected. Besides this fact, it is known that TVD schemes have their order of accuracy reduced to the first order in the presence of shocks due to the effects of limiters. Unstructured meshes can be highly advantageous based on their ability to provide local mesh refinement near important bedload features and structures. As a consequence, the ability to provide local mesh refinement where it is needed leads to improved accuracy for a given computational cost as compared to methods that use structured meshes.

An other example of finite volume solver is the **Godunov-type** method [16, 38, 41] based on dynamically adaptive quadtree grids. This particular grid can be seen as an unstructured mesh that makes the application of SWEs for complex geometries easier. Godunov scheme with an appropriate Riemann solver can capture the steep water surface elevation gradient and for the problem of interest it can be used where a dramatic water surface change occurs, e.g. scour due to dam break flow.

The **Method of Characteristics** (MOC) [28, 29] is for example one of the well-known tools for the solution and analysis of unsteady open channel flow problems even if it has not been widely used for solving hyperbolic fluid flow problems. This is due to its relative complexity for coding when compared to finite volume methods. On the other hand, this method gives a visual representation of flow structure for 1D problems that can be used to provide valuable physical insight into the problem under consideration. Moreover MOC schemes comprise reliable quasi-analytical solutions for problems where analytical examples are rare or not present at all, like those of fluid flow over an erodible bed. Furthermore, in problems where shocks do not occur the MOC becomes a particularly attractive method, being the shock one of the principal problems for this method.

1.2 Sediment transport formulae

Most of the studies on morphodynamic used the Grass formula for modeling the sediment transport. This is due to the simplicity of this formulation: even if it does not take into account all the fundamental mechanisms of sediment transport, it is simple to be implemented. A more "realistic" model is then offered by the Meyer-Peter Müller formula where the presence of a threshold velocity helps to simulate in a more physical way the evolution of the bed profile. Other recurring formulae for sediment transport in these papers are described in Appendix.

1.3 Experiments and results

In order to validate and to test the efficiency of the modelling of the problem several numerical experiments were carried out in the literature. For the large part these experiments are elementary examples, in order to assess was the accuracy of the numerical simulations.

The sinusoidal bump that is the object of the simulations of this study is a well-known case used for many of the works present in the literature [10, 16, 28]. Many of these tests were computed using the Grass formula for the bedload sediment flux, and so they focused on the evolution of the bottom profile varying the parameter of interaction between the flow and the bedload using for all cases a Froude number of 0.1 [1, 8, 10, 15]. This is the classical situation for performing this kind of simulations, as the starting point for the investigations of this work, as it will be seen in the next chapters.

An analysis carried out different Froude numbers is [27] where one of the issues was to investigate whether the traditional, decoupled approach or whether the coupled one is preferable. For a small Froude number, in terms of computational cost, the decoupled approach is preferable, also from the point of view of reducing the amount of numerical diffusion in the modelling. Nevertheless, for high Froude numbers and a relatively mobile bed, the coupled approach is clearly necessary.

One of the most popular simulations is represented by the **dam-break**

test [10, 28, 31]: this kind of flow can be summarized as an uncontrolled release of water flow induced by partial or catastrophic failure of water retaining structures and therefore the it may be cause of serious flooding to downstream areas of the failed structure. These tests were carried out on rigid beds, (this is an example of the traditional Shallow Water problem), or mobile beds. For the purpose of the present study they are not very significant, since they are characterized by sudden variations of the flow conditions, included the Froude number. Therefore, it is anticipated that implicit time advancing is not well suited for this type of problem.

Experimental and real test cases applications represent probably the most interesting part of the literature, since from them it is possible to understand the goodness and the limits of the proposed models as well as future developments. There are many examples of such of these simulations [1, 3, 11, 18, 19, 28, 31, 39].

The last chapter of this work focuses on a comparison with a two-dimensional test case analyzed in [1] in order to appreciate the accuracy of implicit simulations for a more complicated case than the one-dimensional one.

1.4 Aim of the present work

The present work is part of the research activity on the development and on the validation of a strategy for implicit time advancing of the the shallow-water equations coupled with the Exner equation, based on a defect-correction approach and on a time linearization, in which the flux Jacobians are computed through automatic differentiation [4, 5, 6]. This time advancing strategy was coupled with two different finite-volume methods for space discretization, viz. the SRNH predictor-corrector scheme [5] and a modified Roe scheme for non-conservative systems of equations [12]. The Grass expression [22], which provides one of the most popular and simple models for solid transport discharge, was initially used. Different 1D and 2D numerical experiments showed that, if the time scales characterizing the evolution of the hydrodynamic and morphodynamic components are not too small, implicit time advancing leads to large reductions of the computational costs with

respect to those of explicit schemes, while preserving the result accuracy.

Since the Grass model does not take into account the fact that the bottom movement starts when the shear stress exceeds a given critical value, the Meyer Peter Müller model has successively been considered [14]. The results were compared to the Grass ones and strategy in terms of accuracy and efficiency was investigated also for this model. It was found out that implicit simulations obtained with the Grass model can reach the same value of the Courant Friedrichs Lewy condition (CFL condition) of those carried out with MPM model but using 3 DeC iterations instead than one, so the MPM model is preferable from this point of view in terms of computational costs.

The investigations in [14] were limited to a low value of the Froude number (0.1). The aim of this work is to investigate what happens when varying the Froude number. Different simulations were carried out rising this parameter, acting on the velocity or on the height of the flow, in order to understand how the computational cost of implicit simulations varies for both the Grass and the MPM model with the Froude number for different speeds of interaction between the bedload and the flow. As for numerical discretization in space, the present study is limited to the modified Roe scheme. This is motivated by the fact that previous studies indicated that overall performance of the implicit time advancing is practically independent of the space discretization scheme to which it is coupled.

Chapter 2

Physical Model

2.1 Shallow Water Equations

The standard formulation of Shallow Water equations is derived starting from the incompressible Navier-Stokes equations and by assuming that the velocity in one direction and the corresponding acceleration are negligible. Then, by depth averaging the governing equations the standard Shallow Water formulation is recovered.

As for the mathematical notation, considering a fixed reference level L_{ref} , H is the bathymetry function with respect to L_{ref} and the height of the bottom is denoted by Z . Therefore, we have (see also Fig. 1.1):

$$H + Z = L_{ref} \tag{2.1}$$

the height of the flow above the bottom Z is denoted by h , while Q_i is the sediment transport flux in the x_i direction. Since only 1D and 2D physical models are considered in this work, the axis x_1 and x_2 will be simply denoted by x and y . Finally the symbol reserved for the acceleration of gravity is g and the sediment porosity is denoted by p .

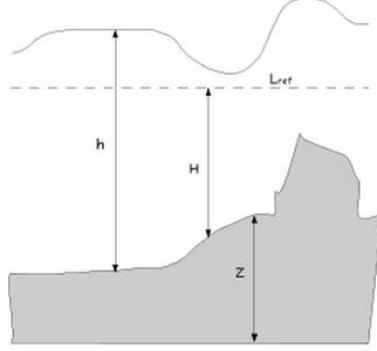


Figure 2.1: Physical variables considered for sediment transport models

The 1D Shallow Water equations can be formulated as follows:

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} = 0 \\ \frac{\partial hu}{\partial t} + \frac{\partial(hu^2 + \frac{1}{2}gh^2)}{\partial x} = -gh \frac{dZ}{dx} \end{cases} \quad (2.2)$$

while in the 2D case they assume the following form:

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \\ \frac{\partial hu}{\partial t} + \frac{\partial(hu^2 + \frac{1}{2}gh^2)}{\partial x} + \frac{\partial huv}{\partial y} = -gh \frac{dZ}{dx} \\ \frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial(hv^2 + \frac{1}{2}gh^2)}{\partial y} = -gh \frac{dZ}{dy} \end{cases} \quad (2.3)$$

Note that, in standard Shallow Water formulation the bed level Z is fixed in time, thus $Z = Z(x)$ or $Z = Z(x,y)$ for (1.2) and (1.3), respectively. The variable bottom topography introduces an additional, non-homogeneous, term in the set of equations: the discretization of this term requires particular attention. The mathematical structure of the Shallow Water equations is strictly related to the one of the Euler equations for barotropic flows. In fact, the evolution of the fluid variables can be described by the following

system of equations:

$$\begin{cases} \frac{\partial W}{\partial t} + \frac{\partial F(W)}{\partial x} = 0 \\ F(W) = (\rho u, \rho u^2 + p, \rho u \xi)^T \\ W = (\rho, \rho u, \rho \xi)^T \end{cases} \quad (2.4)$$

The first two equations are, respectively, the mass and the momentum balances for the fluid. The third equation describes the time-evolution of the substance ξ . The conservation law for ξ is decoupled from the rest of the system and, as a consequence, ξ is usually referred as a passive scalar.

Indeed, by considering only the homogeneous part of the system or, equivalently, a constant bed level, the following 1D system is recovered:

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} = 0 \\ \frac{\partial hu}{\partial t} + \frac{\partial(hu^2 + \frac{1}{2}gh^2)}{\partial x} = 0 \end{cases} \quad (2.5)$$

In order to include the effect of sediment transport, an additional equation which describes the time evolution of the bed level is required. The Exner equation, a well-known and a common choice for this kind of problems, has been used here:

$$(1 - p) \frac{\partial Z}{\partial t} + \frac{\partial Q_1}{\partial x} + \frac{\partial Q_2}{\partial y} = 0 \quad (2.6)$$

where p is the (constant) sediment porosity and Q_1 and Q_2 are the bed-load sediment transport fluxes in the x and y directions. There are many different formulae in the literature that define the sediment transport fluxes; this work focused on two of them, and they will be described in the following section.

Using (2.3) and (2.6) the following complete system is then obtained:

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial hu}{\partial x} + \frac{\partial hv}{\partial y} = 0 \\ \frac{\partial hu}{\partial t} + \frac{\partial(hu^2 + \frac{1}{2}gh^2)}{\partial x} + \frac{\partial huv}{\partial y} = -gh \frac{dZ}{dx} \\ \frac{\partial hv}{\partial t} + \frac{\partial huv}{\partial x} + \frac{\partial(hv^2 + \frac{1}{2}gh^2)}{\partial y} = -gh \frac{dZ}{dy} \\ \frac{\partial Z}{\partial t} + \xi \frac{\partial Q_1}{\partial x} + \xi \frac{\partial Q_2}{\partial y} = 0 \end{cases} \quad (2.7)$$

where

$$\xi = \frac{1}{1-p} \quad (2.8)$$

It is possible to write (1.6) as a system of conservation laws with a source term, that is:

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{\partial \mathbf{F}_1(\mathbf{W})}{\partial x} + \frac{\partial \mathbf{F}_2(\mathbf{W})}{\partial y} = \mathbf{S}(\mathbf{W}) \quad (2.9)$$

where

$$\begin{cases} \mathbf{W} = (h, hu, hv, H)^T \\ \mathbf{F}_1(\mathbf{W}) = (hu, hu^2 + \frac{1}{2}gh^2, huv, -\xi Q_1)^T \\ \mathbf{F}_2(\mathbf{W}) = (hv, hvu, hv^2 + \frac{1}{2}gh^2, -\xi Q_2)^T \\ \mathbf{S}(\mathbf{W}) = (0, gh \frac{\partial H}{\partial x}, gh \frac{\partial H}{\partial y}, 0)^T \end{cases} \quad (2.10)$$

It can be easily checked that the Jacobian matrices of $\mathbf{F}_1(\mathbf{W})$ and $\mathbf{F}_2(\mathbf{W})$ in (1.9) are singular and this may cause severe numerical difficulties in many finite-volume schemes. Furthermore, in order to obtain a well-balanced scheme, the presence of a source term requires a specific treatment. Thus, in the literature different formulations of system (2.10) have been derived to cope with these difficulties. Among the different possibilities, following the work in [12], the source term in (2.10) is rewritten as follows:

$$\mathbf{S}(\mathbf{W}) = \mathbf{B}_1(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial x} + \mathbf{B}_2(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial y} \quad (2.11)$$

where:

$$\mathbf{B}_1(\mathbf{W}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & gh \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{B}_2(\mathbf{W}) = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & gh \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

The previous system can further be recast in a non-conservative form as follows:

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{A}_1(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial x} + \mathbf{A}_2(\mathbf{W}) \frac{\partial \mathbf{W}}{\partial y} = 0 \quad (2.12)$$

where:

$$\mathbf{A}_k(\mathbf{W}) = \frac{\partial \mathbf{F}_k(\mathbf{W})}{\partial \mathbf{W}} - \mathbf{B}_k(\mathbf{W}) \quad k = 1, 2 \quad (2.13)$$

Note that the matrix \mathbf{A}_k is the sum of a conservative term, the Jacobian of $\mathbf{F}_k(\mathbf{W})$, and a non-conservative one, $\mathbf{B}_k(\mathbf{W})$ which takes into account the spatial variation of the bed. Furthermore, matrices \mathbf{A}_1 and \mathbf{A}_2 are not singular [12, 13] and it can be easily seen that this approach eliminates the source term.

2.2 Sediment Transport Models

The sediment can be defined as a fragmented material from rocks that has been formed by different physical and/or chemical process. The study of sediment transport processes includes movement of rocks in a mountain as material diffusion in water, among other processes. Transport is caused by gravity effects and by friction effects with the air or the fluid containing the sediment. Sediment transport is usually divided into three types: bedload, saltation and suspension (see Fig. 2.2). Bedload transport is defined as the type of transport where sediment grains roll or slide along the bed. Saltation transport is defined as the type of transport where single grains jump over the bed a length proportional to their diameter, losing for instants the contact with the soil. Sediment is suspended when the flux is intense enough such as the sediment grains reach height over the bed. This work focuses on the study of bedload sediment transport.

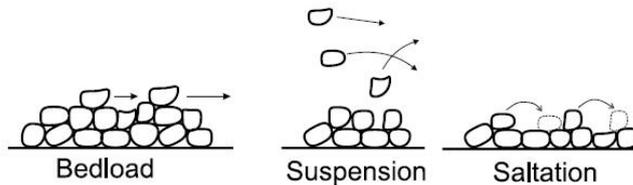


Figure 2.2: Types of sediment transport

In literature different equations to model the solid transport sediment flux could be found: Grass equation [22], Meyer-Peter & Müller's equation

[33], Van Rijn's equation, Nielsen's equation, Kalinske, Einstein's equation, etc., generally obtained by empirical methods. Among all these formulae, some are deterministic formulae and others are based on probabilistic terms. In most of them, except for Grass model, the movement of the sediment is controlled by a physical parameter called critical shear stress, usually determined experimentally. This work focuses on deterministic equations only.

The two formulae used herein for the bed-load transport fluxes are usually obtained for stationary fluxes in rivers, but they can also be applied to tidal or coastal currents, as the time of response of the sediment is very small in comparison with the period of tides or waves.

2.2.1 Grass Model

Grass [22] proposed the following formulae for the solid transport discharge:

$$\begin{cases} Q_1 = A_g u (u^2 + v^2)^{\frac{m-1}{2}} \\ Q_2 = A_g v (u^2 + v^2)^{\frac{m-1}{2}} \end{cases} \quad (2.14)$$

where $A_g (0 \leq A_g \leq 1)$ and $m (1 \leq m \leq 4)$ are parameters depending on the particular problem under consideration. A_g takes into account the grain diameter and the kinematic viscosity and is directly related with the strength of the interaction between the water flow and the bedload; low values of A_g correspond to a weak interaction between the sediment and the fluid, while values close to 1 to a strong interaction. As for m , in the following, only the usual value $m = 3$ is considered, so that (2.14) reduces to:

$$\begin{cases} Q_1 = A_g u (u^2 + v^2) \\ Q_2 = A_g v (u^2 + v^2) \end{cases} \quad (2.15)$$

Notice that, according to Grass formula, the bedload sediment transport begins automatically when the fluid starts to move. This is the principal concern about this formula. Also the maximum of sediment mass flow is on the upper side of the water, and this is clearly unphysical. Hogg (2005) proposed a correction with water column h .

2.2.2 Meyer-Peter and Müller Model

In order to better understand Meyer-Peter & Müller [33] (MPM in what follows) expression, let us underline which are the essential elements of Shield's theory about incipient motion. According to this theory, bed-load movement begins when the shear stress exceeds a critical shear value. Above this value, the current is able to transport the granular sediment. Various physical parameters should be considered:

- shear stress at the bottom τ ;
- sediment density ρ_s and fluid density ρ or their ratio $G = \frac{\rho_s}{\rho}$;
- diameter of the sediment d ;
- characteristics of fluid dynamic motion.

The problem is affected by local phenomena. Shields studied an average depth, composed of granular and non-cohesive sediments and this allowed him to avoid the influence of these local phenomena. Thus, he imposed the balance of forces on a generic sediment particle, observing that the movement starts when the shear stress exceeds a given critical shear stress τ_c . In the literature it is possible to find different approaches to determine the incipient motion condition, based on a critical Froude number or a critical mass flow or a given mean speed of the fluid. The threshold found by Shields is obtained through several experiments, carried out on larger grains and then extrapolated for smaller cases. The shear stress is written as:

$$\tau = \gamma_s R_H |SF| \quad (2.16)$$

γ_s is the specific water weight, R_H is the hydraulic ratio, usually equal to the water column h , and the form factor is defined according to the Manning Theory:

$$SF = \frac{g\eta_0^2 u |u|}{R_H^{3/4}} \quad (2.17)$$

η_0 is the Manning coefficient and u is the fluid speed at the bottom used to evaluate the fluid-dynamic forces. After several experiments, Shields ob-

tained an incipient motion diagram (Fig. 2.3), through the non-dimensional shear:

$$\theta = \frac{\tau}{(\gamma_s - \gamma)d} \quad (2.18)$$

where γ is the specific water weight. The incipient motion condition is expressed as:

$$\theta > \theta_{cr} \quad (2.19)$$

where

$$\theta_{cr} = \frac{\tau_{cr}}{(\gamma_s - \gamma)d} \quad (2.20)$$

θ_{cr} is known as *Shields number*, and it varies between 0.03 and 0.06. The diagram in Fig. 2.3 shows that a considerable amount of data is scattered but, after a given Reynolds value, the Shields number is almost constant ($\simeq 0.04 \div 0.06$). The non dimensional shear stress can still be written following the non-dimensional form proposed by Chezy:

$$\theta = C_\tau \frac{u^2}{(G - 1)gd_{50}} \quad (2.21)$$

where C_τ is called *Chezy parameter* which is generally of the order of 10^{-2} . It can be chosen depending on the specific case and it is very often used in practice. Furthermore, θ is defined according to the mean diameter d_{50} , but it can also be expressed with respect to other characteristic parameters (e.g. the diameter of a similar particle with a higher weight).

Meyer-Peter & Müller developed one of the most known formulae for the solid transport discharge, based on median grain diameter d_{50} . The original formula can be reduced to the following general expression in the 1D formulation:

$$\frac{q_b}{\sqrt{(G - 1)gd_{50}^3}} = 8 \operatorname{sgn}(u)(\theta - \theta_{cr})^{\frac{3}{2}} \quad (2.22)$$

Typically, it describes the sediment transport for rocky rivers, rather than for sandy areas as in Grass model (2.15). It is important the choice of the model for the shear stress τ , to well define the Shields condition. Generally, the motion condition is $\theta_{cr} = 0.047$ [20], according to Bathurst theory. For

different cases this value could be changed. This formula is set for coarse sediments ($d_{50} = 0.4429mm$), useful for coarse stream-beds and for depth inclination of less than 2. For higher inclinations, MPM formula overestimates sediment discharge. Efficiency of relation (2.22) is significant, because it is based on a large experimental data set and it takes into account only the mean characteristic of the flow. An interesting correction of the MPM model is investigated:

$$\frac{q_b}{\sqrt{(G-1)gd_{50}^3}} = 8 \operatorname{sgn}(u)(\theta - \gamma\Delta\eta - \theta_{cr})^{3/2} \frac{\theta - \gamma\Delta\eta}{\|\theta - \gamma\Delta\eta\|} \quad (2.23)$$

where γ is a non-dimensional parameter, related to bottom slope effects. The slope of the depth appears by its gradient, expressing the stabilizing effect of the seabed reshaping.

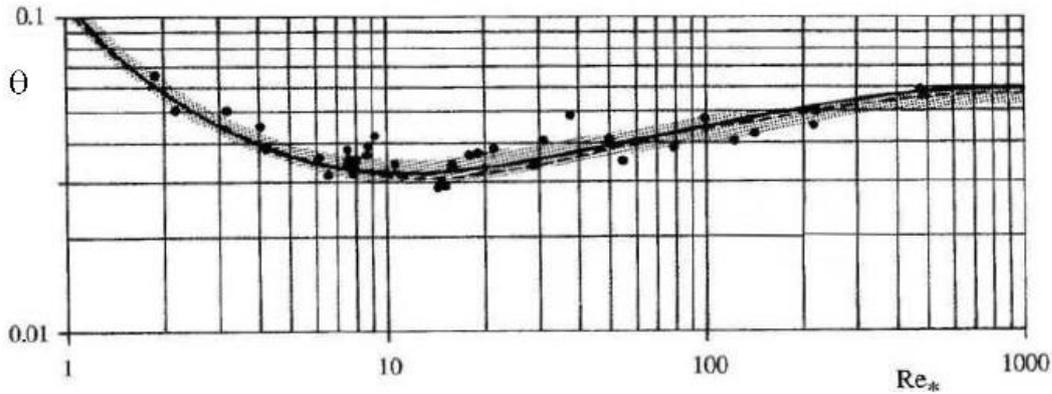


Figure 2.3: Motion condition according to Shields theory. The line represents the incipient movement condition, with respect to the Reynolds Number at the bottom ($Re_* = \frac{U_*d}{\nu}$ where U_* is the velocity at the bottom, d the particle diameter and ν the kinematic viscosity)

It is possible to rewrite the MPM formula in a form similar to the Grass one. In order to do this, it is important the choice of the model for the shear stress τ , to well define the Shields condition. To simplify the computational problem, the shear stresses are written in the Chezy form $\tau = Cu^2$ where C is a constant that can be determined by different theories. So, the non-

dimensional shear stress τ is:

$$\theta = \frac{C}{K}u^2 \quad (2.24)$$

where K can be defined as: $K = (G-1)gd_{50}$ and, substituting in the previous expression of MPM formula, a simplified equation is obtained:

$$q_b = \text{sgn}(u)\tilde{A}(u^2 - \tilde{u}^2)^{3/2} \quad (2.25)$$

where:

$$\tilde{A} = a\frac{C^{3/2}}{K}d_{50} \quad (2.26)$$

$$\tilde{u}^2 = \theta_{cr}\frac{K}{C} \quad (2.27)$$

a is usually taken equal to 8. In this way, the incipient motion condition is related to the square of velocity u . If $u^2 > \tilde{u}^2$, sediment transport is present and expressed by (2.23).

Chapter 3

Numerical Method

In this chapter a finite-volume discretization of the physical model presented in the previous one is described. In this work a Modified Roe scheme (MR) is used for space discretization. The MR explicit scheme has been developed by Castro et al. (see e.g.[12]) and it is a Roe scheme modified in order to deal with non conservative systems of equations. The implicit counterpart of the aforementioned explicit numerical scheme for the 2D case is obtained following[5, 6]. In particular the implicit numerical schemes is generated through the use of the automatic differentiation tool TAPENADE [25]. This chapter presents the 2D numerical discretization, skipping the 1D numerical case. However, once the 2D numerical method is defined, the 1D formulations can easily be recovered by simply setting to zero the velocity component along the y-axis and all derivatives in the y-direction.

3.1 General definitions for the considered finite-volume formulation

The considered space discretization is based on a finite-volume approach. At a preliminary stage, the considered 2D computational domain $\mathcal{V} \in \mathbb{R}^2$ is approximated by means of a polygonal domain \mathcal{V}^{pol} which, in turn, is divided into triangles. Then, the i_{th} finite-volume cell V_i , associated with the vertex

\mathbf{P}_i , is given by:

$$V_i = \bigcup_{h \in t(i)} V_i^{(h)} \quad (3.1)$$

where $t(i)$ is the set of indexes marking those triangles which share \mathbf{P}_i as a vertex and $V_r^{(h)}$ represents the subset of the triangle T_h which is defined by further dividing T_h into six sub-triangles by means of its medians and subsequently considering those two subtriangles which share \mathbf{P}_r as a vertex (see Fig. 3.1).

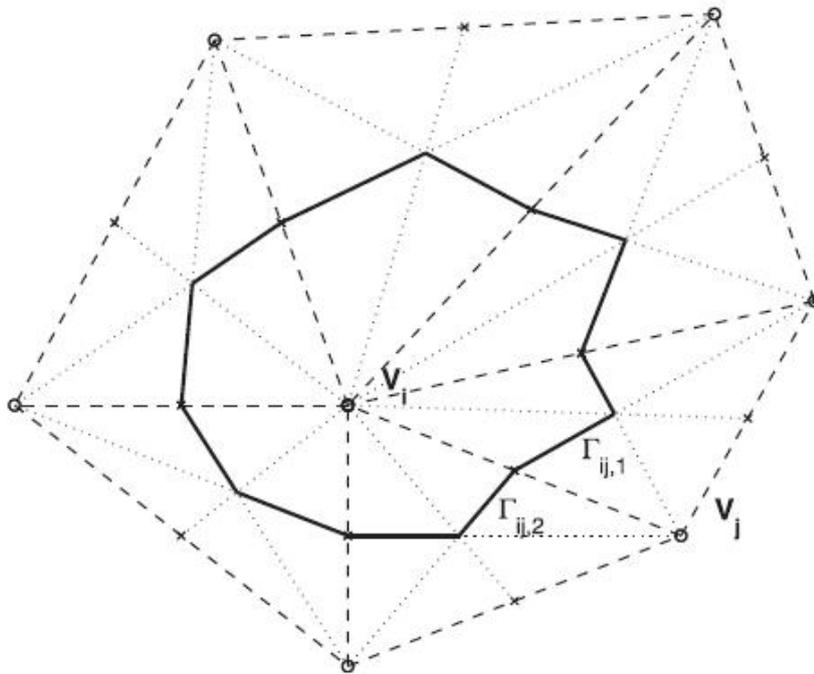


Figure 3.1: Generation of the dual mesh

Clearly, there is a finite-volume cell for each vertex. The considered finite-volume discretization is sometimes referred to as a "dual mesh" (see e.g., [20]), by virtue of the specific procedure which is adopted in order to build the cells starting from the triangles. Some properties for this choice of computational cells are given in [23], where it is shown that, on these cells, finite-volume centered schemes are equivalent to P1 finite-element mass-lumped schemes. The following notation is also considered: given a finite-volume cell V_i , $|V_i|$ is its area and \mathbf{G}_i is its center of mass. $N(i)$ is the set of indexes of the neigh-

boring cells of the i_{th} -cell and $B_i = N(i) \cup \{i\}$. Furthermore, $\bar{B}_i = \cup_{j \in B_i} B_j$ is the set of indexes marking the i_{th} -cell, its neighbors and the neighbors of the neighbors.

$\Gamma_{ij,1}$ and $\Gamma_{ij,2}$ are the two segments of the common interface between cell V_i and V_j and $|\Gamma_{ij,1}|$ and $|\Gamma_{ij,2}|$ their length (see Fig. 3.1). The normal unit vector to $\Gamma_{ij,1}$ is $\mathbf{n}_{ij,1} = (n_{x,ij,1}, n_{y,ij,1})^T$ and a similar definition holds for $\mathbf{n}_{ij,2}$. The average normal vector for the interface between the i_{th} and j_{th} -cell is defined as follows:

$$|\Gamma_{ij}|\mathbf{n}_{ij} = |\Gamma_{ij,1}|\mathbf{n}_{ij,1} + |\Gamma_{ij,2}|\mathbf{n}_{ij,2} \quad (3.2)$$

where \mathbf{n}_{ij} is the normal unit vector and $|\Gamma_{ij}|$ is the interface length. Finally, \mathbf{W}_i^n is the average value of the solution \mathbf{W} in the i_{th} cell at time t_n :

$$\mathbf{W}_i^n = \frac{1}{|V_i|} \int_{V_i} \mathbf{W}(\mathbf{x}, t_n) dV \quad (3.3)$$

3.2 A Roe-type scheme for non-conservative systems

The scheme adopted herein for the space discretization has been proposed by Castro et al. in a series of papers [11, 12, 13]. More precisely, it is possible to define a Roe-type scheme for the nonconservative system (1.11) leading to the following semi-discrete form:

$$\frac{\partial \mathbf{W}_i^n}{\partial t} = RHS_1(\{\mathbf{W}_j\}_{j \in B_i}) \quad (3.4)$$

where

$$RHS_1(\{\mathbf{W}_j\}_{j \in B_i}) = -\frac{1}{|V_i|} \sum_{j \in N(i)} |\Gamma_{ij}| \left(\mathcal{F}(\mathbf{W}_i^n, \mathbf{W}_j^n, \mathbf{n}_{ij}) - \frac{1}{2} \mathbf{B}_{ij} (\mathbf{W}_j^n - \mathbf{W}_i^n) \right) \quad (3.5)$$

and $\mathcal{F}(\mathbf{W}_i^n, \mathbf{W}_j^n, \mathbf{n}_{ij})$ is a Roe-like numerical flux function defined as:

$$\begin{aligned} \mathcal{F}(\mathbf{W}_i^n, \mathbf{W}_j^n, \mathbf{n}_{ij}) &= \frac{1}{2} \left(\mathbf{F}_{nij}(\mathbf{W}_i^n) + \mathbf{F}_{nij}(\mathbf{W}_j^n) \right) \\ &\quad - \frac{1}{2} |A_{ij}| \left(\mathbf{W}_j^n, \mathbf{W}_i^n, \mathbf{n}_{ij} \right) \left(\mathbf{W}_j^n - \mathbf{W}_i^n \right) \end{aligned} \quad (3.6)$$

where $\mathbf{F}_{nij}(\mathbf{W}) = n_{x,ij} \mathbf{F}_1(\mathbf{W}) + n_{y,ij} \mathbf{F}_2(\mathbf{W})$ while the expressions of the matrices $|A_{ij}|$ and \mathbf{B}_{ij} depend on \mathbf{W}_i^n , \mathbf{W}_j^n and \mathbf{n}_{ij} . (see [4, 6] for more details).

3.2.1 Second-order extension

To extend to second-order of accuracy the previous MR scheme, first a reconstruction operator $P_i(x)$ is defined at each cell. This reconstruction operator is a function of the values of the solution in the neighbor cells of V_i , that is:

$$P_i(\mathbf{x}) = P_i\left(\mathbf{x}; \{\mathbf{W}_j^n\}_{j \in B_i}\right) \quad (3.7)$$

In this work we consider a MUSCL-like reconstruction operator which is explicitly defined in [4, 6]. Once the reconstruction operator is defined, it is possible to extend the Modified Roe scheme to the second order accuracy. The semi-discrete formulation of (2.12) is:

$$\begin{aligned} \frac{\partial \mathbf{W}}{\partial t} &= -\frac{1}{|V_i|} \sum_{j \in N(i)} \int_{\Gamma_{ij}} \left(\mathcal{F}(\mathbf{W}_{ij}^-(\sigma), \mathbf{W}_{ij}^+(\sigma), \mathbf{n}_{ij}) \right. \\ &\quad \left. - \frac{1}{2} \mathbf{B}_{ij}(\sigma) (\mathbf{W}_{ij}^+(\sigma) - \mathbf{W}_{ij}^-(\sigma)) \right) d\sigma \\ &+ \frac{1}{|V_i|} \int_{V_i} \left(\mathbf{B}_1(P_i(\mathbf{x})) \frac{\partial P_i}{\partial x}(\mathbf{x}) + \mathbf{B}_2(P_i(\mathbf{x})) \frac{\partial P_i}{\partial y}(\mathbf{x}) \right) dV \end{aligned} \quad (3.8)$$

where σ is a point of the interface Γ_{ij} , $\mathbf{W}_{ij}^- = P_i(\sigma)$, $\mathbf{W}_{ij}^+ = P_j(\sigma)$ and $\mathbf{B}_{ij}(\sigma) = \mathbf{B}(\mathbf{W}_{ij}^-(\sigma), \mathbf{W}_{ij}^+(\sigma), \mathbf{n}_{ij})$, \mathbf{B} being defined in [4]. Even if it does not clearly appear, the right-hand-side of (3.8) is time-dependent. Indeed, due to (3.7), $P_i(\mathbf{x})$ is a function of space but also of the solution $\mathbf{W}(\mathbf{x}, t)$. Note that, due to the non-conservative formulation, the second-order scheme

is not only a function of the extrapolated values of the solution at the cell interfaces, but also function of the solution in the interior of the cells. The integrals in (3.8) are numerically approximated and, in order to preserve the second-order spatial accuracy of the scheme the order of the quadrature formula must be higher than that of the reconstruction operator. In [3] it has been shown that the third-order Gauss quadrature formula for the line integrals and the barycenter quadrature formula for the surface integrals satisfy both criteria.

The resulting semi-discrete expression of the second-order MR scheme is:

$$\frac{\partial \mathbf{W}_i}{\partial t} = RHS_2\left(\{\mathbf{W}_j\}_{j \in \bar{B}_i}\right) \quad (3.9)$$

where $RHS_2\left(\{\mathbf{W}_j\}_{j \in \bar{B}_i}\right)$ is defined as follows:

$$\begin{aligned} RHS_2\left(\{\mathbf{W}_j\}_{j \in \bar{B}_i}\right) &= \mathbf{B}_1(P_1(\mathbf{G}_i)) \frac{\partial P_i}{\partial x}(\mathbf{G}_i) \\ &\quad + \mathbf{B}_2(P_1(\mathbf{G}_i)) \frac{\partial P_i}{\partial y}(\mathbf{G}_i) \\ &- \frac{1}{|V_i|} \sum_{j \in N(i)} \sum_{l=1}^2 |\Gamma_{ij,l}| \sum_{m=1}^2 W_{lm} \left(\mathcal{F}(\mathbf{W}_{ij,lm}^+, \mathbf{W}_{ij,lm}^+, \mathbf{n}_{ij}) \right. \\ &\quad \left. - \frac{1}{2} \mathbf{B}_{ij,lm} (\mathbf{W}_{ij,lm}^+ - \mathbf{W}_{ij,lm}^-) \right) \end{aligned} \quad (3.10)$$

where $\mathbf{W}_{ij,lm}^- = P_i(\sigma_{lm})$, $\mathbf{W}_{ij,lm}^+ = P_j(\sigma_{lm})$, $\mathbf{B}_{ij,lm} = \mathbf{B}_{ij}(\sigma_{lm})$ and w_{lm} and σ_{lm} are respectively, the weights and the points of the quadrature rule. Note that the dependency from $j \in \bar{B}_i$ in (3.9) results from the reconstruction operator and, more specifically, from $\mathbf{W}_{ij,lm}^-$ and $\mathbf{W}_{ij,lm}^+$. Finally, limiters are used following the same approach as in [12, 13]; a detailed description can be found in [4, 6].

3.3 Explicit time advancing

To obtain an explicit scheme first-order accurate in time, the time-discretization can be carried out using an explicit Euler method, so that the resulting first-

order numerical scheme is:

$$\mathbf{W}_i^{n+1} = \mathbf{W}_i^n + \Delta t^n RHS_1\left(\{\mathbf{W}_j\}_{j \in B_i}\right) \quad (3.11)$$

Similarly, in order to obtain an explicit scheme second-order accurate in time, the time discretization can be carried out using a second-order TVD Runge-Kutta method [21]. The final expression of the second-order numerical scheme is:

$$\begin{cases} \mathbf{W}_i^{n+1/2} = \mathbf{W}_i^n + \Delta t^n RHS_2\left(\{\mathbf{W}_j^n\}_{j \in \bar{B}_i}\right) \\ \mathbf{W}_i^{n+1} = \frac{\mathbf{W}_i^{n+1/2} + \mathbf{W}_i^n}{2} + \frac{1}{2}\Delta t^n RHS_2\left(\{\mathbf{W}_j^{n+1/2}\}_{j \in \bar{B}_i}\right) \end{cases} \quad (3.12)$$

3.4 Implicit time advancing

In this section the issue of generating an implicit scheme, starting from its explicit counterpart, is addressed. In a first step, we will describe the first-order numerical scheme, then the second-order extension is discussed.

3.4.1 First-order scheme

Generally speaking, the implicit counterpart of a first-order explicit Euler method is obtained by considering the right hand side term as a function of the solution at time $n + 1$ instead of n . That is, a fully-implicit first-order version of the scheme in Eq. (3.11) is the following:

$$\mathbf{W}_i^{n+1} - \Delta t^n RHS_1\left(\{\mathbf{W}_j^{n+1}\}_{j \in B_i}\right) = \mathbf{W}_i^n \quad (3.13)$$

However, from a practical point of view this would require the solution of a large non-linear system of equations at each time step. The computational cost for this operation is in general not affordable in practical applications and, in general, significantly overcomes any advantage that an implicit scheme could have with respect to its explicit counterpart. A common technique to overcome this difficulty is to linearize the numerical scheme, that is

to find an approximation of $RHS_1\left(\{\mathbf{W}_j^{n+1}\}_{j \in B_i}\right)$ in the form:

$$RHS_1\left(\{\mathbf{W}_j^{n+1}\}_{j \in B_i}\right) \simeq RHS_1\left(\{\mathbf{W}_j^n\}_{j \in B_i}\right) + \sum_{j \in B_i} \mathcal{D}_{ij}^n \Delta^n \mathbf{W}_j \quad (3.14)$$

where $\Delta^n(\cdot) = (\cdot)^{n+1} - (\cdot)^n$ and \mathcal{D}_{ij}^n are matrices depending on the solution in a neighborhood of V_i at time t^n , i.e. $\mathcal{D}_{ij}^n = \mathcal{D}_{ij}\left(\{\mathbf{W}_l^n\}_{j \in B_i}\right)$.

Using this approximation, the following linear system must be solved at each time step:

$$\frac{\Delta^n \mathbf{W}_i}{\Delta^n t} - \sum_{j \in B_i} \mathcal{D}_{ij}^n \Delta^n \mathbf{W}_j = RHS_1\left(\{\mathbf{W}_j^n\}_{j \in B_i}\right) \quad (3.15)$$

The implicit linearized scheme is completely determined once a suitable definition for the matrices \mathcal{D}_{ij}^n is given. If the right hand side is differentiable, a common choice is to use the Jacobian matrices, hence:

$$\mathcal{D}_{ij}^n \simeq \frac{\partial RHS_1\left(\{\mathbf{W}_l^n\}_{j \in B_i}\right)}{\partial \mathbf{W}_j^n} \quad (3.16)$$

Nevertheless, it is not always possible nor convenient to exactly compute the Jacobian matrices. This problem has been solved herein through the use of the automatic differentiation software Tapenade [20]. The operational principle of an automatic differentiation software is as follows: given the source code of a routine which computes the function $y = F(x)$, the automatic differentiation software generates a new source code which compute the analytical derivative of the original program. In practice, each time the original program performs some operation, the differentiated program performs additional operations dealing with the differential values. For example, if the original program at some time executes the following instruction on variables a, b, c :

$$a = b \cdot c \quad (3.17)$$

then the differentiated program computes also the differentials da, db, dc of

these variables [35]:

$$da = db \cdot c + b \cdot dc \quad (3.18)$$

Through an automatic differentiation software it is possible to quickly implement an implicit linearized scheme of the form (3.15), once a routine which computes the explicit flux function is available. As a consequence using an automatic differentiation tool, starting from a first-order explicit method, it is possible to automatically compute the matrices \mathcal{D}_{ij}^n and then implement the linearized implicit method (3.15) without additional modifications.

3.4.2 Second-order DeC scheme

A second-order implicit scheme can be obtained from its explicit counterpart using the same approach described for the first-order scheme in the previous subsection. Therefore the second-order fully implicit approach could be written as follows:

$$\begin{cases} \mathbf{W}_i^{n+1/2} - \Delta t^n RHS_2\left(\{\mathbf{W}_j^{n+1/2}\}_{j \in \bar{B}_i}\right) = \mathbf{W}_i^n \\ \mathbf{W}_i^{n+1} - \frac{1}{2}\Delta t^n RHS_2\left(\{\mathbf{W}_j^{n+1}\}_{j \in \bar{B}_i}\right) = \frac{\mathbf{W}_i^{n+1/2} + \mathbf{W}_i^n}{2} \end{cases} \quad (3.19)$$

The application of this method would require the solution of two non linear systems of equations at each time step, thus dramatically increasing the computational costs with respect to the explicit version.

An alternative approach, generally more efficient in terms of computational costs, is to use a second-order backward differentiation formula in time:

$$\frac{(1 + 2\tau)\mathbf{W}_i^{n+1} - (1 + \tau)^2\mathbf{W}_i^n + \tau^2\mathbf{W}_i^{n-1}}{\Delta^n t + \tau} - RHS_2\left(\{\mathbf{W}_j^{n+1}\}_{j \in \bar{B}_i}\right) = 0 \quad (3.20)$$

where $\tau = \frac{\Delta^n t}{\Delta^{n-1} t}$. Similarly to the first-order case, a linearization of $RHS_2\left(\{\mathbf{W}_j^{n+1}\}_{j \in \bar{B}_i}\right)$ must be carried out in order to avoid the solution of a nonlinear system at each time step. Clearly, the same approach as for the first-order scheme could be considered, that is to approximate

$RHS_2(\{\mathbf{W}_j^{n+1}\}_{j \in \bar{B}_i})$ in the form:

$$RHS_2(\{\mathbf{W}_j^{n+1}\}_{j \in \bar{B}_i}) \simeq RHS_2(\{\mathbf{W}_j^n\}_{j \in \bar{B}_i}) + \sum_{j \in \bar{B}_i} \mathcal{D}_{2,ij}^n \Delta^n \mathbf{W}_j \quad (3.21)$$

However, the linearization for the second-order accurate fluxes and the solution of the resulting linear system imply significant computational costs and memory requirements. This is a consequence of the more complex expression of the second-order scheme with respect to its first-order counterpart (compare (3.5) with (3.10)) and, in particular, of the larger stencil of the second-order flux function (by considering an uniform triangular grid, 7 and 19 nodes are involved for the first- and the second-order approaches respectively). In order to reduce the computational costs, an alternative approach, considered in this work, is to use a defect-correction technique (DeC) [39]. This method consists in iteratively solving simpler problems obtained by considering the same linearization used for the first-order scheme. The DeC iterations write as follows, the unknowns being $\Delta^s \mathcal{W}$:

$$\begin{cases} \mathcal{W}^0 = \mathbf{W}^n \\ \mathcal{L}_i^s \Delta^s \mathcal{W}_i - \sum_{j \in N(i)} \mathcal{D}_{ij}^s \Delta^s \mathbf{W}_j = \mathcal{C}_i^s & s = 0, \dots, r-1 \\ \mathbf{W}^{n+1} = \mathcal{W}^r = \mathbf{W}^n + \sum_{s=0}^{r-1} \Delta^s \mathcal{W} \end{cases} \quad (3.22)$$

in which:

$$\begin{cases} \mathcal{D}_{ij}^s = \mathcal{D}_{ij}(\{\mathcal{W}_l^s\}_{l \in \mathcal{B}_i}) \\ \mathcal{L}_i^s = \left(\frac{(1+2\tau)\mathcal{W}_i^s - (1+\tau)^2 \mathbf{W}_i^n + \tau^2 \mathbf{W}_i^{n-1}}{\Delta^n t(1+\tau)} \right) + RHS_2(\{\mathcal{W}_j^s\}_{j \in \bar{B}_i}) \end{cases} \quad (3.23)$$

\mathcal{D}_{ij} being the matrices of the approximation (3.14) computed, here, through automatic differentiation; r is typically chosen between 1 and 3. Indeed, it can be shown [32] that only one DeC iteration is theoretically needed to reach a second-order accuracy, while few additional iterations (one or two) can improve the robustness.

3.5 CFL condition

The time interval Δt is defined by using the Courant Friedrichs Lewy (CFL) condition. When explicit time-marching schemes are used low values of CFL must be used to ensure numerical stability. On the other hand, implicit time-advancing is stable for much larger values of CFL; therefore, longer time steps can be used. Nevertheless, a too large Δt could deteriorate the accuracy of the results. Therefore, it must be checked that the chosen Δt is small enough to well resolve all the relevant flow time scales. For 1D case, the CFL number is defined as follows:

$$CFL = \frac{\lambda_{max}\delta t}{\delta x} \quad (3.24)$$

where λ_{max} is the maximum value of the eigenvalues of the *Roe Matrix*.

As for a two dimensional more general point of view, the CFL condition can be expressed with the following:

$$CFL = \frac{u_x\Delta t}{\Delta x} + \frac{u_y\Delta t}{\Delta x} \quad (3.25)$$

where u_x and u_y are the velocity components in the x and the y directions respectively.

Chapter 4

Numerical experiments

The purpose of the next chapters is to investigate how the accuracy limitation on the implicit time step depends on the Froude number: in particular, it will be shown how the relation between the parameter A_g , involved in the considered sediment flux models, and the CFL condition is affected by varying the Froude value into two ways. To introduce this analysis, a well-known benchmark is used first, which is described in Sec. 3.1.

4.1 Presentation of the 1D Problem

A 1D problem is considered; it is a sediment transport problem in a channel of length $l = 1000$ m with a non-constant bottom profile. The initial bottom topography, in the case of $Fr_i = 0.1$, is given by the hump shape function in the following:

$$\begin{cases} Z(0, x) = \begin{cases} 0.1 + \sin^2\left(\frac{(x-300)\pi}{200}\right) & \text{if } 300 \leq x \leq 500 \\ 0.1 & \text{elsewhere} \end{cases} \\ h(0, x) = 10 - Z(0, x) \\ u(0, x) = \frac{10}{h(0, x)} \end{cases} \quad (4.1)$$

in which all variables are in SI units.

Two different grids are used for the simulations: GRD1, composed by

100 elements and GRD2, composed by 250 elements. The Grass and the MPM models are used. Concerning the Grass model four different values of A_g (see Grass equation 2.15) are considered, viz. 0.001, 0.01, 0.1 and 1, which correspond to different speed of interaction between the flow and the bedload and, as a consequence, to different time scales of evolution of the bottom topography.

The first value correspond to a weak interaction, the last to a strong one, while the other values to intermediate conditions. Then, for the MPM model, in order to be compared to the Grass model, the values of \tilde{A} have been chosen equal to the ones of A_g , while for each \tilde{A} (see MPM equation 2.25), three different values of the critical velocity are used, namely $\tilde{u} = 1.02, 1.04$ and 1.06 m/s. Indeed, empirical values usually adopted in the literature for the different quantities involved in the computation of \tilde{u} lead to $\tilde{u} \simeq 1.04$ m/s. Note that also A_g and \tilde{A} are functions of different physical quantities, which are usually assigned empirically depending on the problem of interest, as e.g. rocky rivers, sandy bottom. Not all the values adopted herein correspond to a realistic situation, but they have been considered in order to appraise the efficiency of implicit time advancing for varying characteristic time scales for the bedload evolution. The simulations are advanced in time until an almost steady state is reached; clearly this implies longer times for problems involving lower values of A_g and \tilde{A} , as shown in table 4.1. All the results and computational times shown in the following are at the final instant of each simulation.

A_g	1	0.1	0.01	0.001
Simulation time	700	7000	50000	500000

Table 4.1: Final simulation time (seconds) for the considered values of A_g

4.1.1 Comparison between the solutions obtained with different sediment transport models

The numerical solutions obtained with the two models are reported in the figures 4.1-4.8. For brevity, not all the simulations are reported, in particular we show only these with a threshold velocity of $\tilde{u} = 1.04$ m/s for the MPM model. From these plots it clearly appears that the choice of the solid transport discharge model has a strong effect on the solution. Due to the threshold on the motion inception, the bottom shape is less regular for the MPM model than the one obtained for the Grass ones. The general behavior in the model comparison does not depend on the grid resolution or on the variation of the speed of interaction between the bedload and water flow.

The solutions computed with the implicit schemes, considering an adequate CFL value depending on the speed of interaction between the flow and the bedload are practically overlapping the ones obtained with the explicit time advancing at $CFL = 0.8$ for all morphodynamic models.

4.1.2 Explicit vs. implicit time-advancing schemes

The use of an implicit time-advancing allows to avoid the time step limitations due to CFL conditions typical of explicit schemes. Nevertheless, the accuracy of the results of implicit simulations deteriorates if the time step is too large. In a precedent study [6], a first analysis has been carried out for the Grass model showing that the CFL number up to which the implicit solution is not deteriorated depends on the speed of interaction between bedload and water flow. More precisely, the CFL limit for implicit computations was found to be roughly inversely proportional to the parameter A_g

$$CFL \simeq \frac{1}{A_g} \quad (4.2)$$

For instance, for both first and second order accuracy, an implicit solution coinciding with the explicit one is obtained up to $CFL = 1000$ for the slow speed interaction (i.e. $A_g = 0.001$). An additional gain of efficiency can be obtained considering CFL values one order of magnitude larger and 3 DeC

iterations instead of only one (see the figures below).

$$CFL \simeq \frac{10}{A_g} \quad (4.3)$$

Consequently, the implicit time-advancing appears very attractive in terms of computational costs with respect to the explicit one for weak and intermediate interactions.

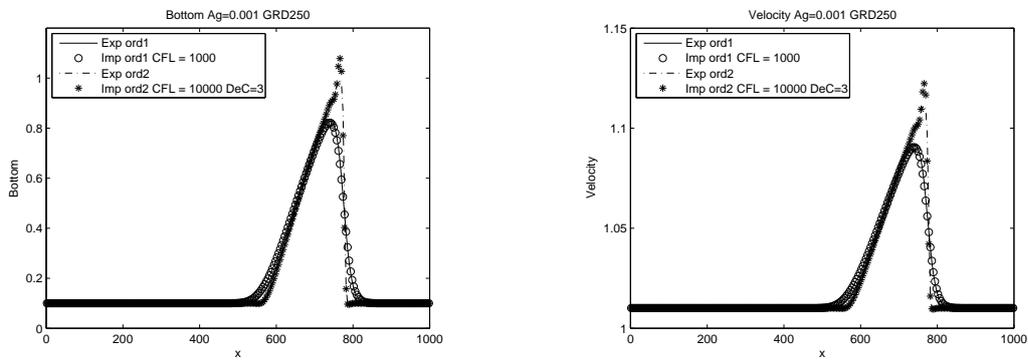


Figure 4.1: Explicit and implicit solutions for the evolution of the bottom and the velocity profiles for Grass model for $A_g = 0.001$

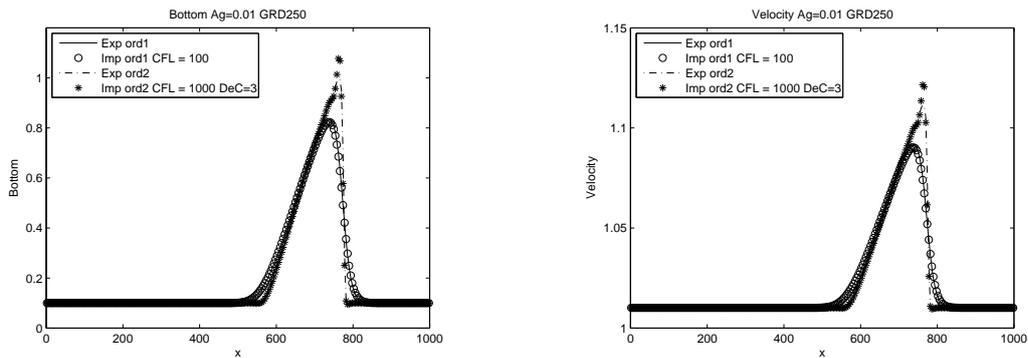


Figure 4.2: Explicit and implicit solutions for Grass model for the evolution of the bottom and the velocity profiles for $A_g = 0.01$

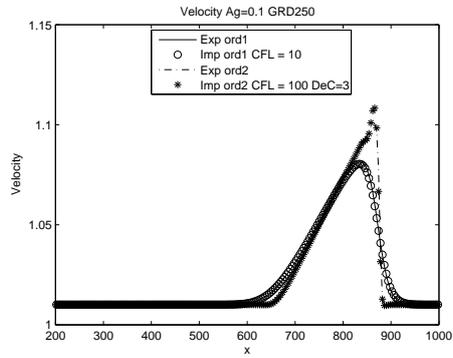
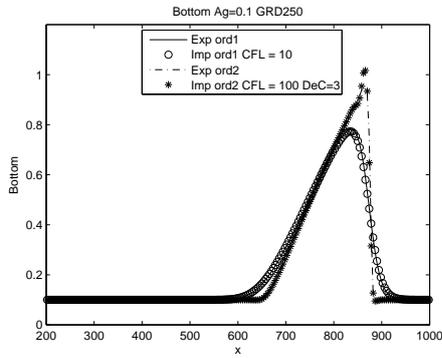


Figure 4.3: Explicit and implicit solutions for Grass mode for the evolution of the bottom and the velocity profiles for $A_g = 0.1$

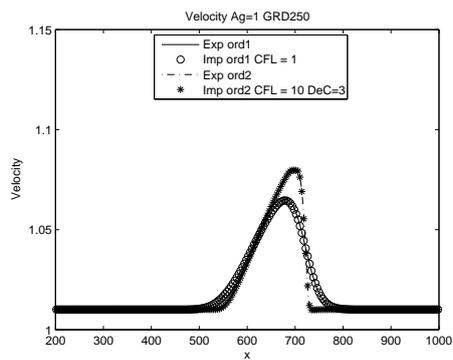
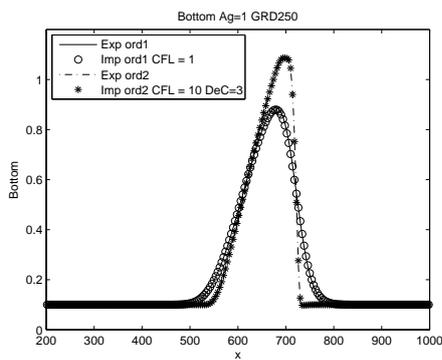


Figure 4.4: Explicit and implicit solutions for Grass model for the evolution of the bottom and the velocity profiles for $A_g = 1$

Concerning the simulations with the MPM model, the following figures report some explicit and implicit solutions. It has been found (see [14]) that for any choice of tested \tilde{u} value, as well for the less refined grid, the maximum CFL number allowed to obtain an accurate solution is one order of magnitude larger for the MPM model than for to Grass one.

$$CFL \simeq \frac{10}{A_g} \quad (4.4)$$

As previously done for the Grass model, a possible way to try to improve the solution accuracy for large values of the time step can be to consider additional DeC iterations in the implicit numerical formulation. It appears that 1 additional DeC iteration is not enough to stabilize the solution with the presence of additional fluctuations. Conversely, the use of 2 or 3 additional DeC iterations gives a rather satisfactory solution. Thus, the advantages of using an implicit approach appear even increased for this model but the use of additional DeC iterations does not further improve the performances.

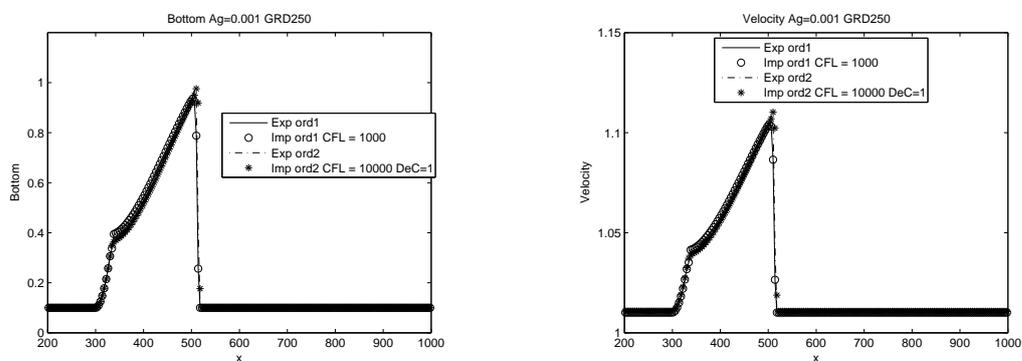


Figure 4.5: Explicit and implicit solutions for MPM model for the evolution of the bottom and the velocity profiles for $A_g = 0.001$

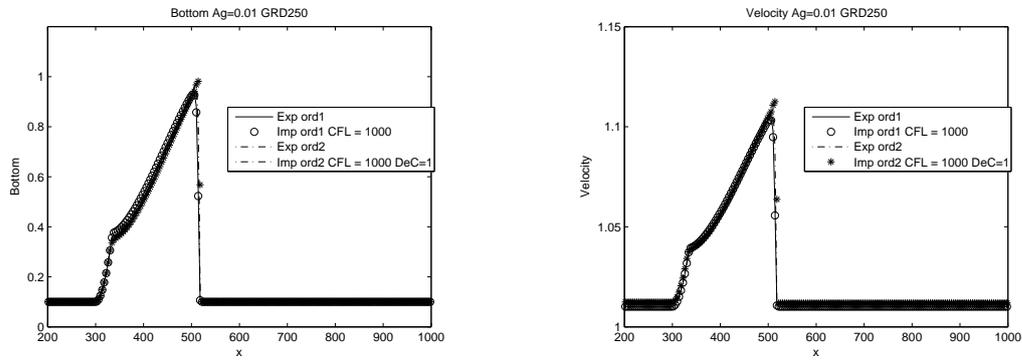


Figure 4.6: Explicit and implicit solutions for MPM model for the evolution of the bottom and the velocity profiles for $A_g = 0.01$

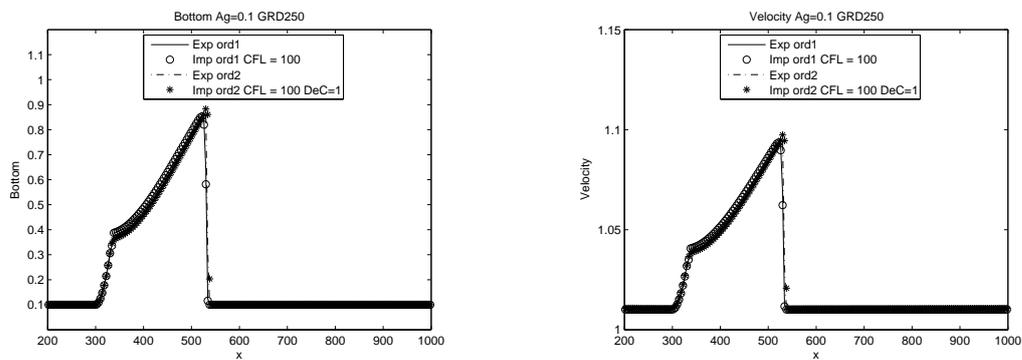


Figure 4.7: Explicit and implicit solutions for MPM model for the evolution of the bottom and the velocity profiles for $A_g = 0.1$

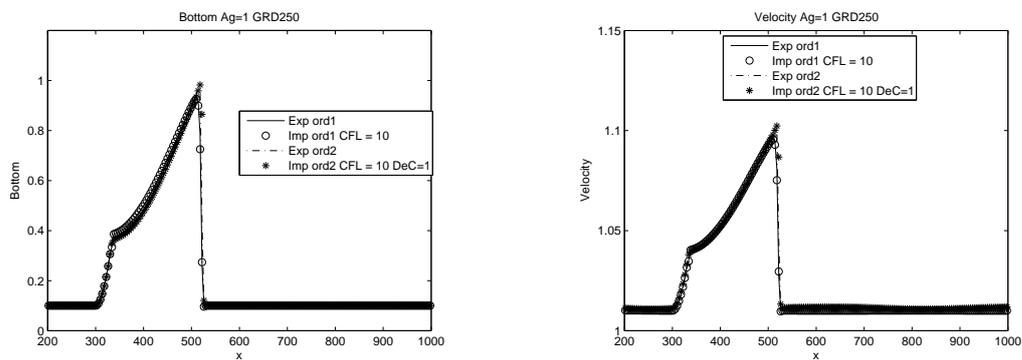


Figure 4.8: Explicit and implicit solutions for MPM model for the evolution of the bottom and the velocity profiles for $A_g = 1$

Summarizing, the efficiency of implicit time advancing seems not to be affected by the model used for solid discharge fluxes; rather, the largest gains in computational time have been obtained for the MPM model, which is the most complex and contains a threshold, as shown in the following two figures.

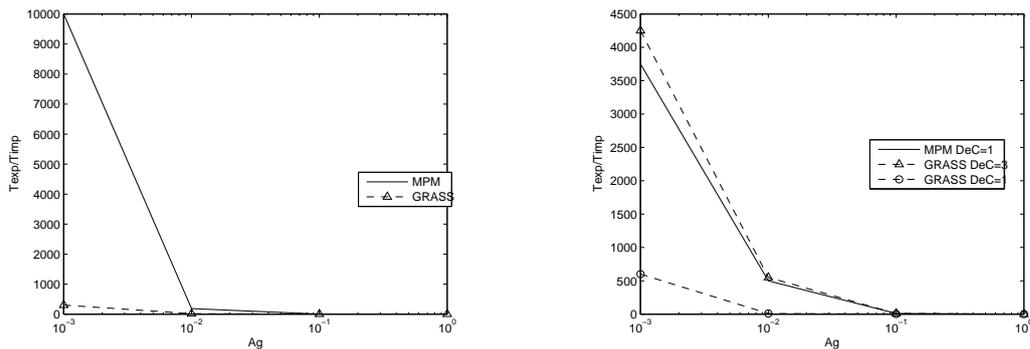


Figure 4.9: Time computational ratios between the explicit and implicit solutions for the Grass and the MPM models at first and second order

4.2 Varying the Froude number

In order to provide a quantification of how the accuracy limitation on the implicit time step depends on another scale characterizing morphodynamic problems, i.e. that related to the shallow water dynamics, different values of the Froude number at the inlet have been analyzed in this way. Two different approaches are used for changing the Froude number, since it is given by the following expression:

$$Fr = \frac{u}{\sqrt{gh}}$$

The two parameters we focused on are the velocity u and the height h of the flow, so we modified the Froude number varying these one of these two quantities, and then we compared the following results. The results are shown and discussed in chapters 5 and 6.

Chapter 5

Variation of the Froude number by acting on u

First, the inlet velocity was modified for varying the Froude number from $Fr_i = 0.1$ to $Fr_i = 0.7$, increasing by steps of 0.1. The computational domain has been enlarged to $-1000 \leq x \leq 3000$ in order to well describe the evolution of the bottom. Because of this, the previously refined grid made by 250 cells has become a coarse one: the Δx of each cell has changed from 4 to 16. Therefore a more refined grid of 500 cells has been used.

At increasing values of the Froude number at the inlet, the simulation time for each simulation was reduced in order to observe significant variations of the bed profile, being far enough from the edge of the computational domain. An empirical law was used to have a guess of the duration of the simulation: since in the Grass model the sediment transport flux is given by $A_g u^3$, we can expect that the variation of the simulation time is proportional to such an order. For example, in the case of Froude number equal to 0.3, the corresponding simulation times were obtained from those at $Fr_i = 0.1$ and then divided by 27, since $(0.3/0.1)^3 = 27$. Tab.5.1 contains all the simulations times for varying Fr_i and A_g .

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	500000	50000	7000	700
$Froude_i = 0.2$	62500	6250	875	87.5
$Froude_i = 0.3$	18518	1852	260	50
$Froude_i = 0.4$	7812.5	782	120	30
$Froude_i = 0.5$	4000	400	80	25
$Froude_i = 0.6$	2500	250	70	20
$Froude_i = 0.7$	1460	146	45	15

Table 5.1: Final simulation times (seconds) at varying Froude numbers

5.1 Efficiency comparison

As said before for the $Fr_i = 0.1$ case, for all the values of A_g , implicit and explicit solutions give practically identical results, provided that the implicit time step remains smaller than a given threshold. The dependency on A_g of the accuracy limitation on the implicit CFL at $Fr_i = 0.1$ was found to be:

$$CFL = \frac{1}{A_g} \quad \text{for DeC=1} \quad \text{while} \quad CFL = \frac{10}{A_g} \quad \text{for DeC=3} \quad (5.1)$$

At increasing values of the Froude number, the relation between CFL and A_g is no more simply inversely linear, since for the accuracy of the implicit solutions lower values of the CFL number have to be used. In order to establish the maximum value of CFL for the implicit solution giving reasonable results, we distinguished the solutions evaluating the maximum of the absolute value of the error and the mean-square error with respect to the explicit solution. The errors are made nondimensional by using the maximum value of the initial bed load profile and the inlet velocity. We accepted solutions with a mean-square error lower than the 1% and a maximum of the absolute value of the error smaller than the 10% for the evolution of the bottom profile, and values lower than 0.3% and 3% respectively for the evolution of the velocity one. At the end of these simulations, other empirical relations between the CFL number and A_g were found at each Fr_i , and they are shown in Tab.A. Obviously, lower values than 1 of the CFL are not convenient for an implicit solution, so once it is reached the efficiency of the implicit solution can be

no longer improved and therefore the errors with respect to the explicit one could exceed the boundaries defined.

	$CFL = \frac{10}{A_g}$	$CFL = \frac{1}{A_g}$	$CFL = \frac{0.1}{A_g}$	$CFL = \frac{0.01}{A_g}$	$CFL = \frac{0.001}{A_g}$
$Fr_i = 0.1$ 1st ord		X			
$Fr_i = 0.1$ DeC = 1 2nd ord	X				
$Fr_i = 0.1$ DeC = 3 2nd ord	X				
$Fr_i = 0.2$ 1st ord			X		
$Fr_i = 0.2$ DeC = 1 2nd ord			X		
$Fr_i = 0.2$ DeC = 3 2nd ord		X			
$Fr_i = 0.3$ 1st ord			X		
$Fr_i = 0.3$ DeC = 1 2nd ord			X		
$Fr_i = 0.3$ DeC = 3 2nd ord			X		
$Fr_i = 0.4$ 1st ord				X	
$Fr_i = 0.4$ DeC = 1 2nd ord				X	
$Fr_i = 0.4$ DeC = 3 2nd ord			X		
$Fr_i = 0.5$ 1st ord				X	
$Fr_i = 0.5$ DeC = 1 2nd ord				X	
$Fr_i = 0.5$ DeC = 3 2nd ord			X		
$Fr_i = 0.6$ 1st ord				X	
$Fr_i = 0.6$ DeC = 1 2nd ord				X	
$Fr_i = 0.6$ DeC = 3 2nd ord				X	
$Fr_i = 0.7$ 1st ord					X
$Fr_i = 0.7$ DeC = 1 2nd ord					X
$Fr_i = 0.7$ DeC = 3 2nd ord				X	

Table A: CFL dependency on A_g at varying Froude number

5.2 Froude = 0.2

Figures 5.1-5.4 show the simulations computed at Froude = 0.2 for the bottom and the velocity profiles. The matching of the implicit solutions with the correspondent explicit ones is really good provided that the CFL values for the implicit time advancing remains under some limits, which depend on A_g , as it is reported in the figures and it will be discussed in the following. The form of the profiles is very similar to that obtained at Froude = 0.1, in particular it is possible to notice how the variation of velocity become smaller as the interaction between the bedload and the flow is faster. Simulations obtained with the more refined grid of 500 cells are also shown (Figs 5.5-5.8): there is of course a slight modification of the profiles, but no significant differences in terms of CFL values and time computational ratios are obtained, so for the other values of the Froude number these simulations will be omitted for brevity.

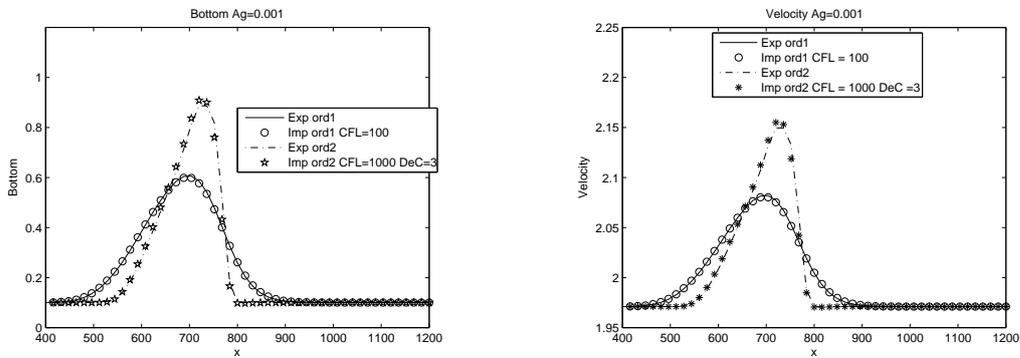


Figure 5.1: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.2

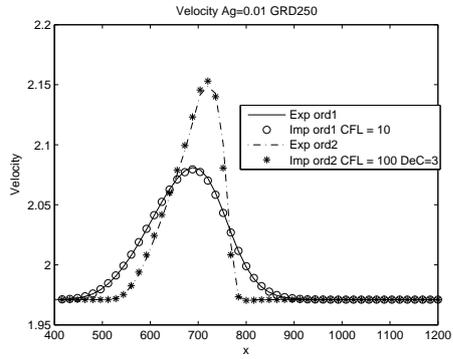
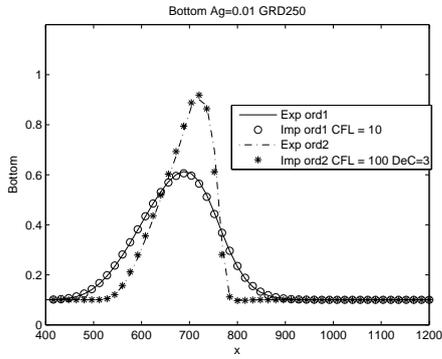


Figure 5.2: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.2

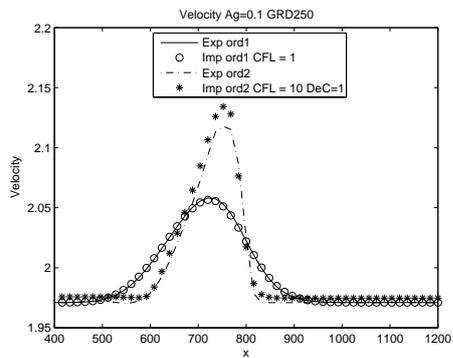
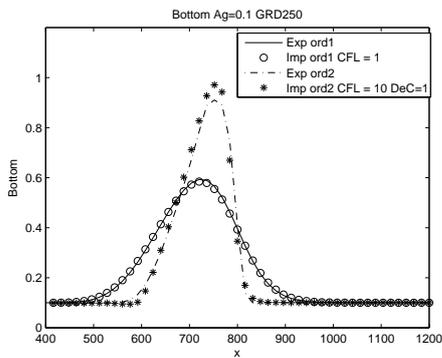


Figure 5.3: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.2

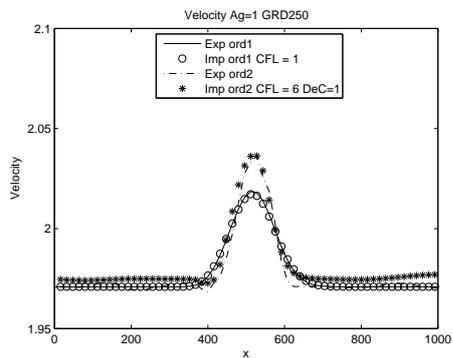
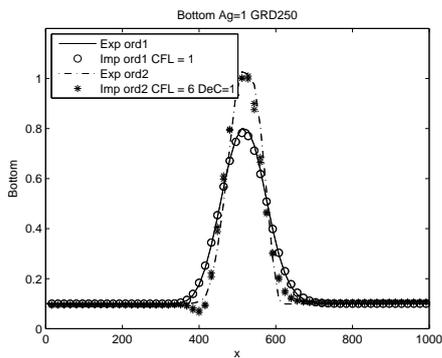


Figure 5.4: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.2

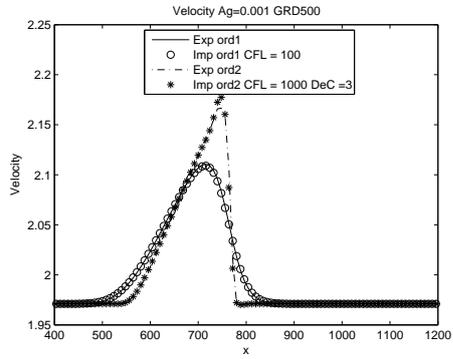
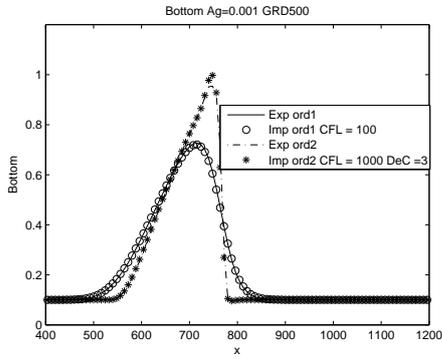


Figure 5.5: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 500 cells at Froude = 0.2

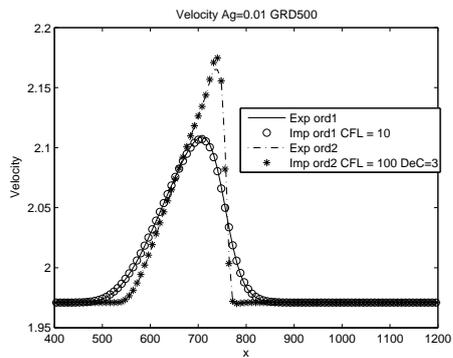
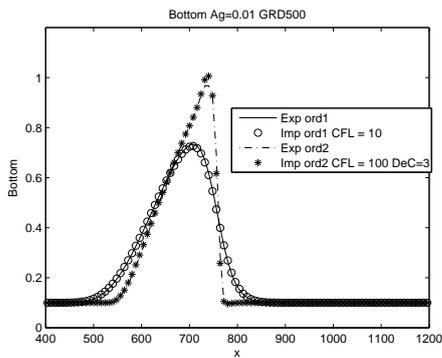


Figure 5.6: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 500 cells at Froude = 0.2

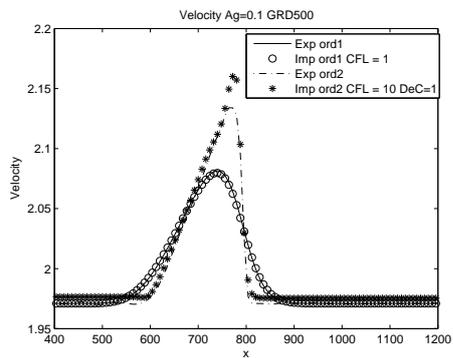
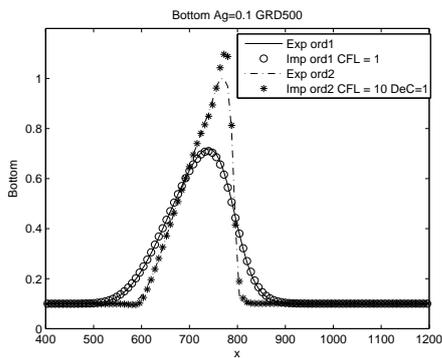


Figure 5.7: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 500 cells at Froude = 0.2

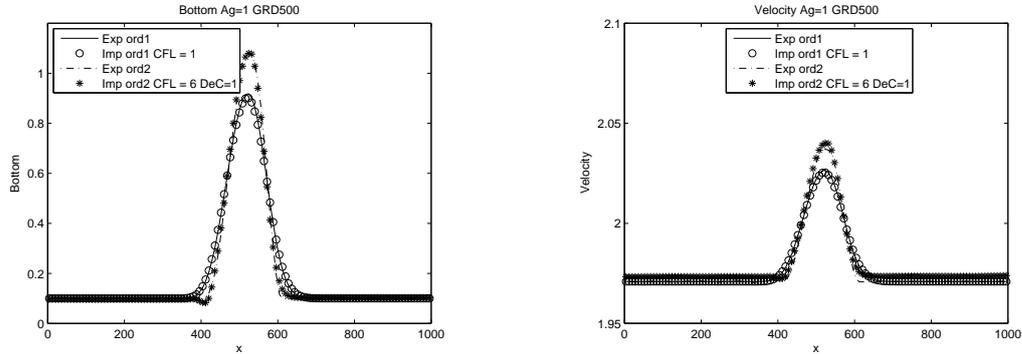


Figure 5.8: Explicit and implicit solutions for the evolution of the bottom and the velocity profiles for Grass model for $A_g = 1$ using a grid made by 500 cells at Froude = 0.2

5.3 Froude = 0.3

Figures 5.9-5.12 show the simulations computed at Froude = 0.3 for the bottom and the velocity profiles. The matching of the implicit solutions with the correspondent explicit ones is really good provided that the CFL values for the implicit time advancing remains under some limits, which depend on A_g . In particular, the roughly inversely proportional dependency of CFL on A_g is no longer valid, but it has been decreased of one order of magnitude, and no significant improvement is achieved by using more DeC iterations. Besides, clear fluctuations are visible at $A_g = 1$: as said before, lower values than 1 of CFL are not taken into consideration so at this point differences with the explicit solution starts to be noticeable, especially for the velocity profiles. Simulations obtained with the more refined grid of 500 cells are not reported for brevity.

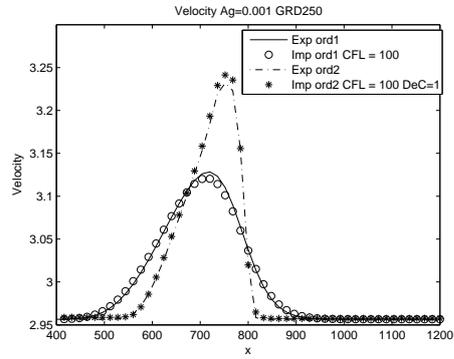
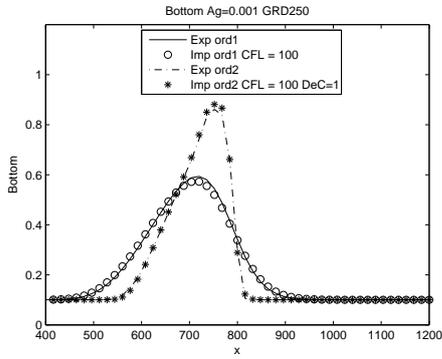


Figure 5.9: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.3

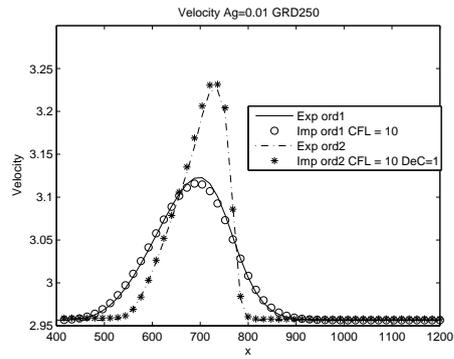
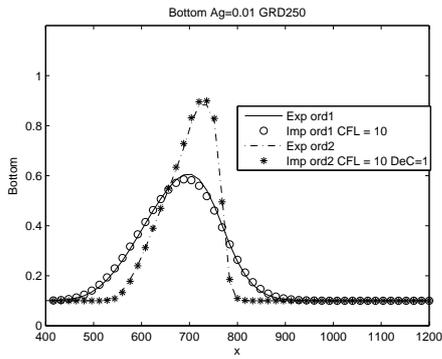


Figure 5.10: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.3

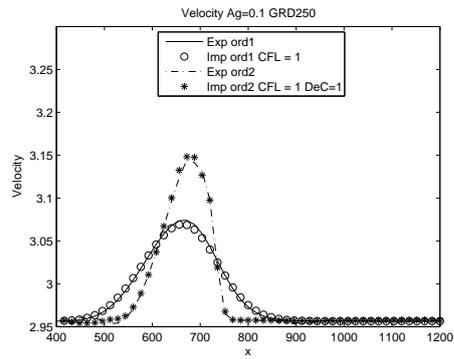
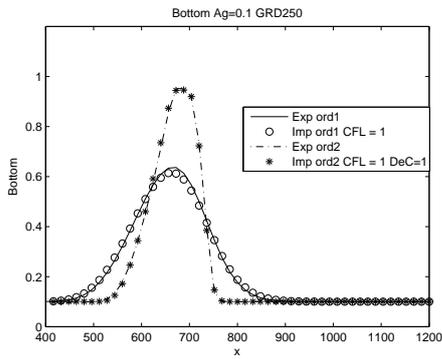


Figure 5.11: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.3

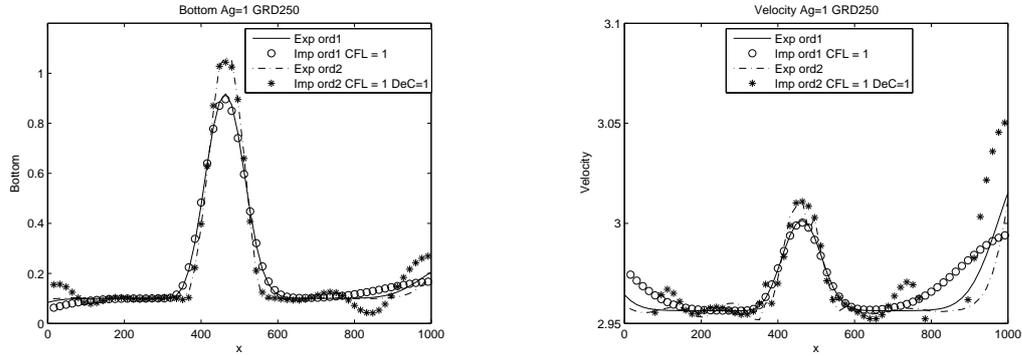


Figure 5.12: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.3

5.4 Froude = 0.4

Figures 5.13-5.16 show the simulations computed at Froude = 0.4 for the bottom and the velocity profiles. Again, the matching of the implicit solutions with the correspondent explicit ones is really good but in this case the maximum CFL allowed for the implicit solution is 10 for $A_g = 0.001$, while one order of magnitude is gained when using 3 DeC iterations. Besides, clear fluctuations are visible at $A_g = 1$: as said before, lower values than 1 of CFL are not taken into consideration so at this point differences with the explicit solution starts to be noticeable, especially for the velocity profiles. Simulations obtained with the more refined grid of 500 cells are not reported for brevity

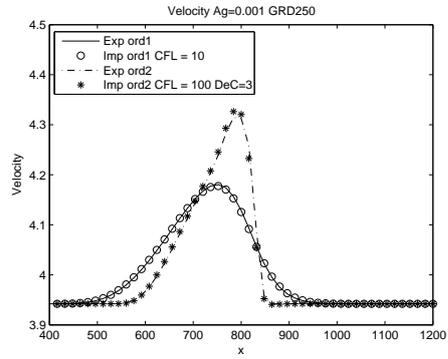
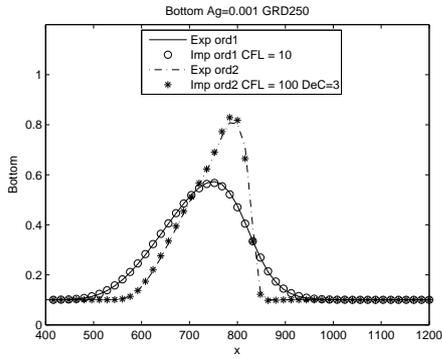


Figure 5.13: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.4

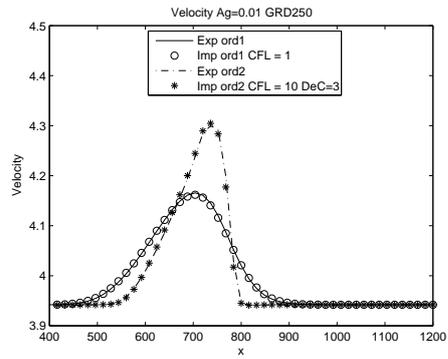
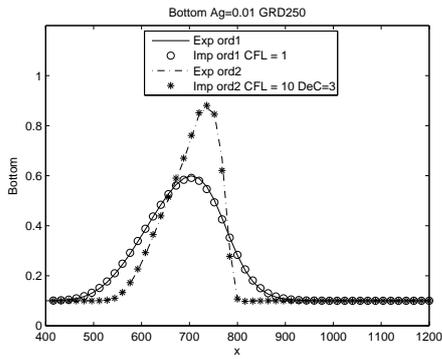


Figure 5.14: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.4

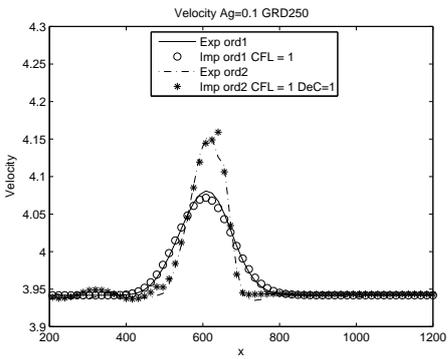
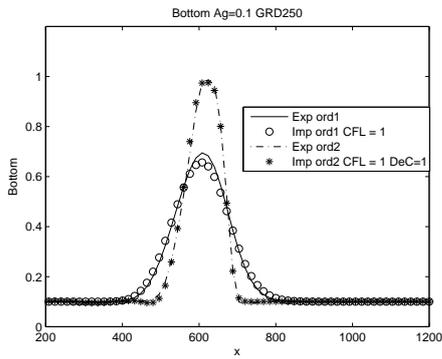


Figure 5.15: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.4

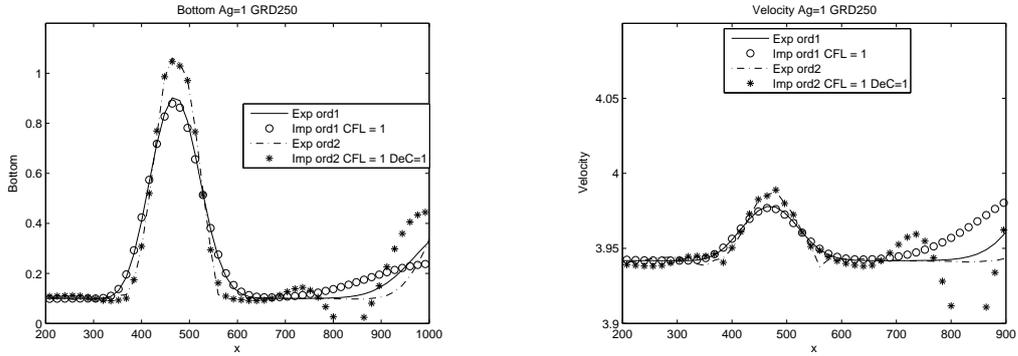


Figure 5.16: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.4

5.5 Froude = 0.5

Figures 5.17-5.20 show the simulations computed at Froude = 0.5 for the bottom and the velocity profiles. The behavior shown by the implicit solutions is very similar to that noticeable for Froude = 0.4, since the maximum CFL allowable is equal to 10 for $A_g = 0.001$ at first and second order, but it is possible to reach a value of 100 for the CFL condition by using 3 DeC iterations. Besides, clear fluctuations are still visible at $A_g = 1$ but there is no further deterioration of that seen for the same simulations carried out for the previous value of the Froude number. Simulations obtained with the more refined grid of 500 cells are not reported for brevity.

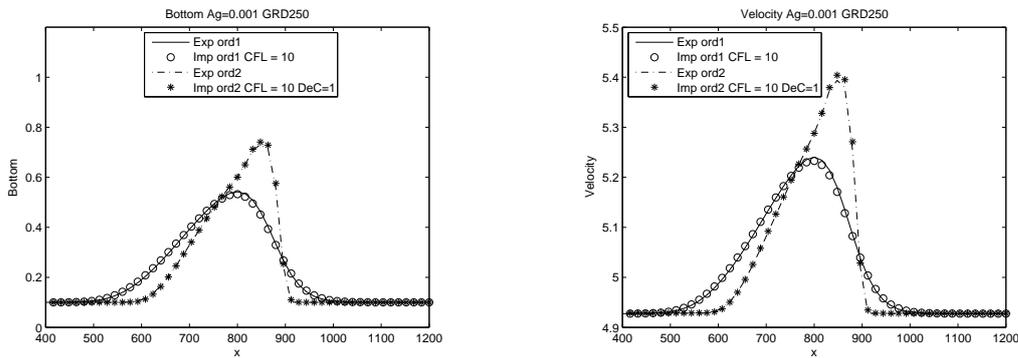


Figure 5.17: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.5

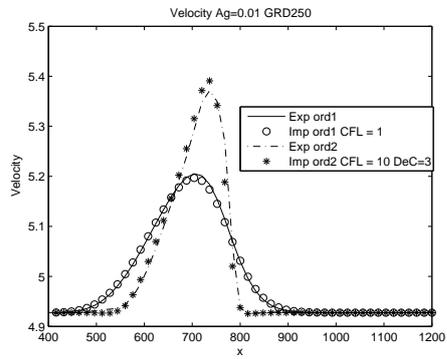
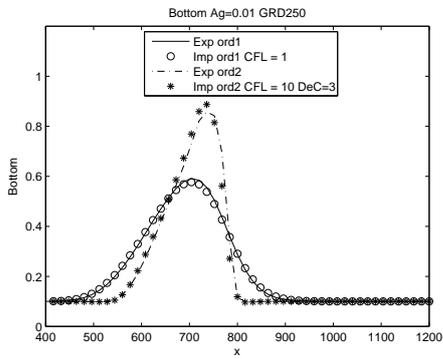


Figure 5.18: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.5

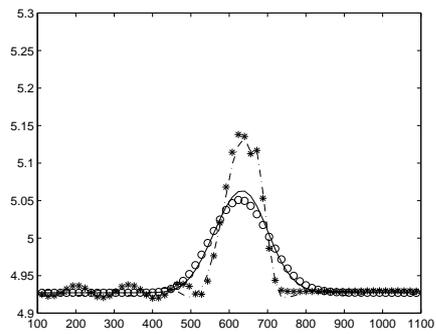
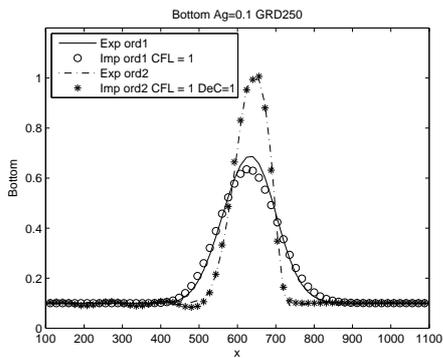


Figure 5.19: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.5

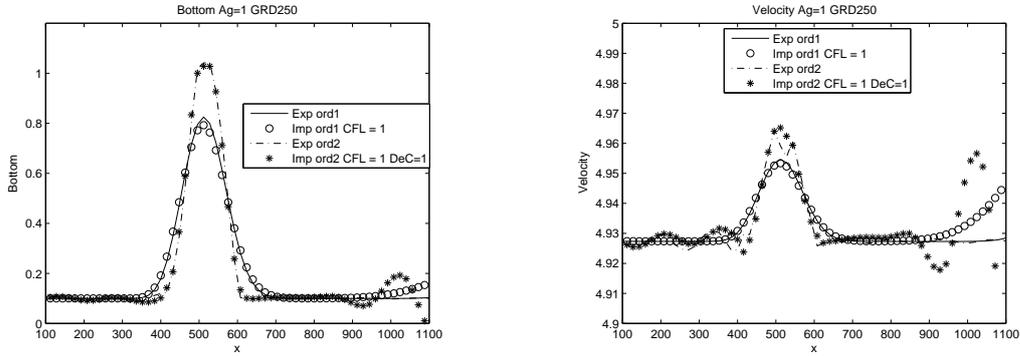


Figure 5.20: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.5

5.6 Froude = 0.6

Figures 5.21-5.24 show the simulations computed at Froude = 0.6 for the bottom and the velocity profiles. The matching of the implicit solutions with the correspondent explicit ones is good provided that the CFL values for the implicit time advancing remains under some limits: in this case, the use of 1 or 3 DeC iterations does not affect the maximum CFL allowable, so a maximum value of 10 is used for implicit solutions at $A_g = 0.001$, then the limited value of 1 has been set. The fluctuations are still visible at $A_g = 1$ and, again, especially for the velocity profiles. Simulations obtained with the more refined grid of 500 cells are not reported for brevity.

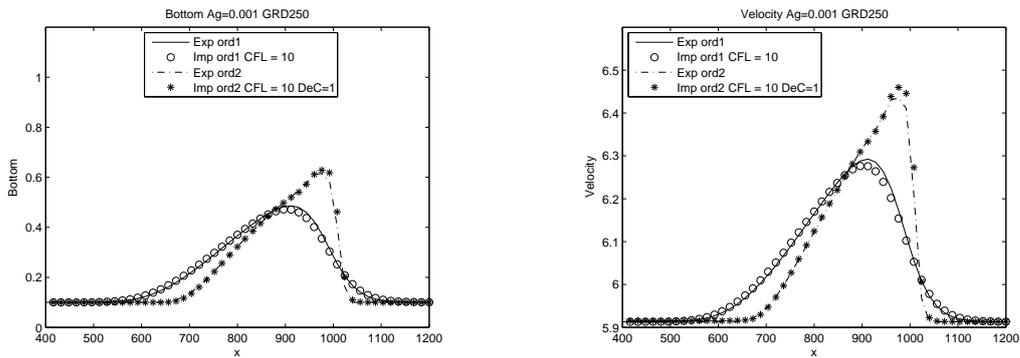


Figure 5.21: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.6

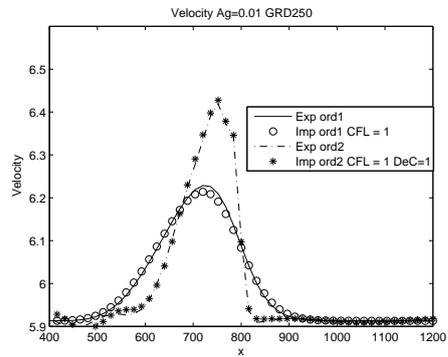
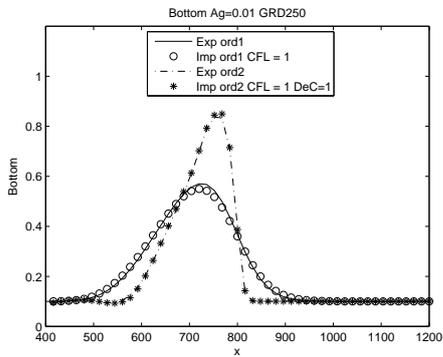


Figure 5.22: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.6

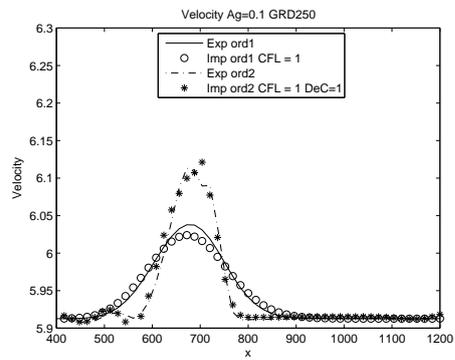
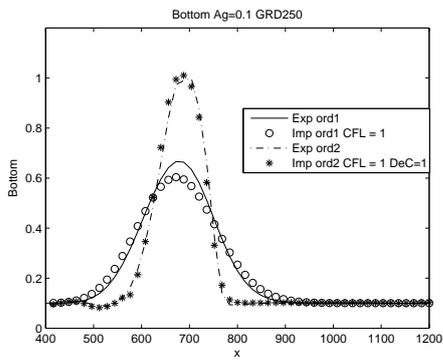


Figure 5.23: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.6

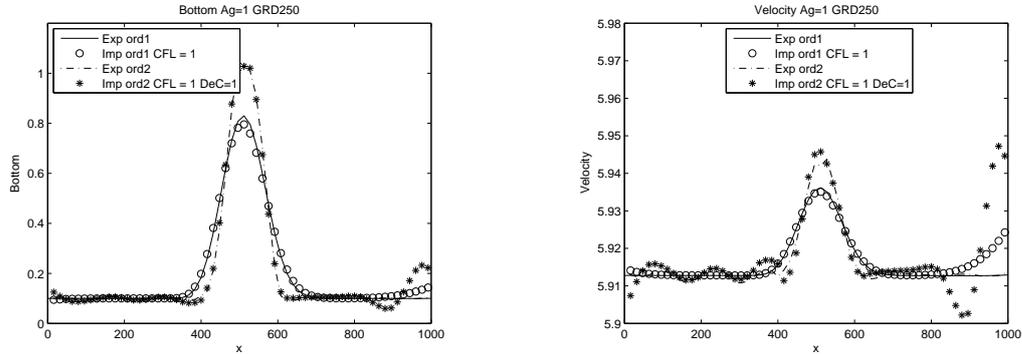


Figure 5.24: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.6

5.7 Froude = 0.7

Finally, figures 5.25-5.28 show the simulations computed at Froude = 0.7 for the bottom and the velocity profiles. In this case, the maximum CFL allowable for implicit solutions is reduced to 1 at first and at second order when using 1 DeC iteration, and therefore the computational costs for the implicit solutions exceed those of the corresponding explicit ones for every value of A_g . Just by using 3 DeC iterations it is possible to gain one order of magnitude for $A_g = 1$ and this allows an equality of the computational costs of the explicit and the implicit solutions for this value of the speed of interaction. Fluctuations are still visible at $A_g = 1$ and, again, especially for the velocity profiles. Simulations obtained with the more refined grid of 500 cells are not reported for brevity.

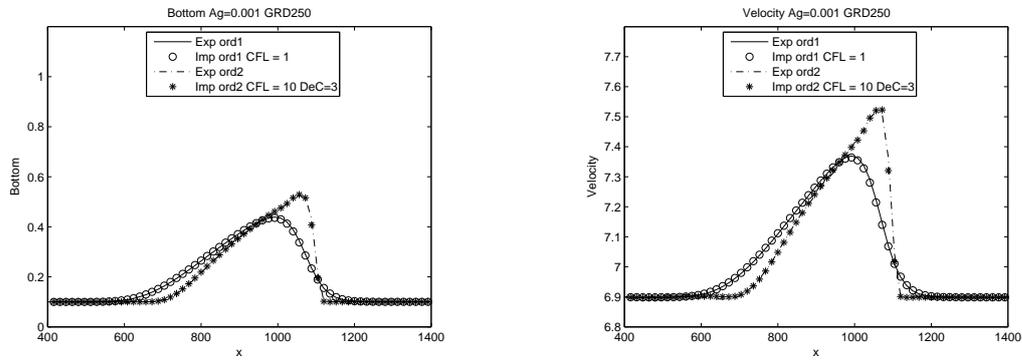


Figure 5.25: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.7

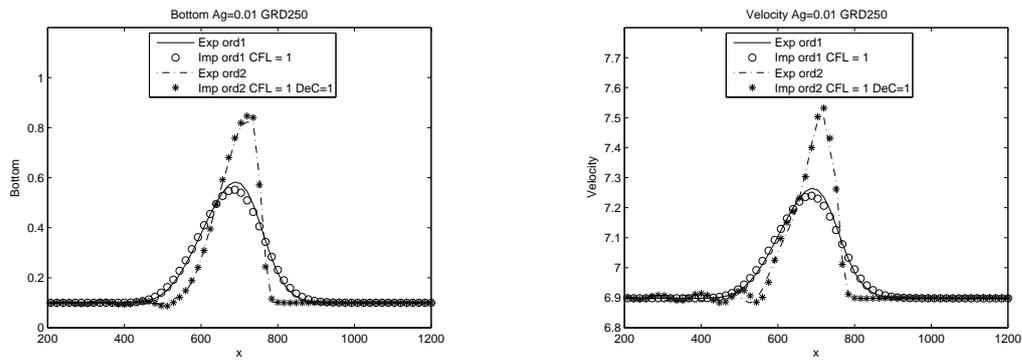


Figure 5.26: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.7

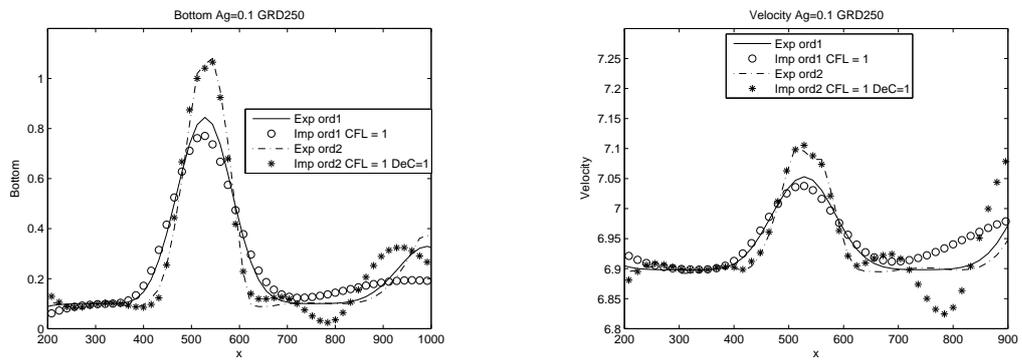


Figure 5.27: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.7

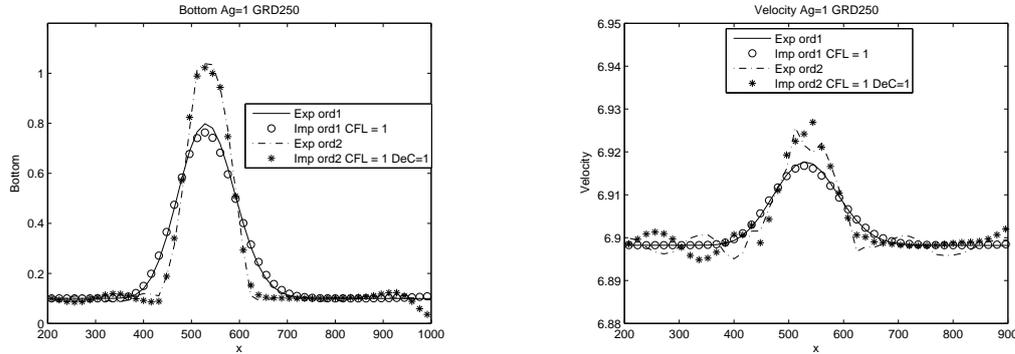


Figure 5.28: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.7

5.8 Concluding annotations

In a first analysis of the maximum CFL allowed by the solution accuracy for implicit time advancing, the CFL has been decreased of steps equals to 10. Succedely, a more refined investigation on the CFL number was made for those simulations where the explicit solution was found to be more convenient. Given the computational time requested by a specific explicit simulation, the correspondent implicit solution with the similar computational cost was analyzed in order to understand if this one was acceptable and therefore if the CFL number could be raised to higher values. In other words, the CFL has been decreased of steps equals to 1 than to 10 in order to better appreciate the convenience of implicit solutions. However, this more refined research have not led results in this case.

As expected, at increasing Froude number the allowable value of CFL becomes smaller; using 3 DeC instead of 1 it is possible to increase the value of CFL of one order of magnitude, but not in all cases, since, as shown in the table, for $Fr_i = 0.3$ and $Fr_i = 0.6$ there is no significant improvement when using more DeC iterations.

These simulations show another important feature: at increasing values of the Froude number, the computational time ratio between explicit and implicit time advancing moves quickly towards values smaller than 1, i.e. explicit solutions are less expensive than implicit ones. This trend is more

evident at first order, where, except for $Fr = 0.1$, explicit solutions are faster for the large part of the types of interaction between bed load and water flow. At second order this evidence is mitigated, as shown in the following images.

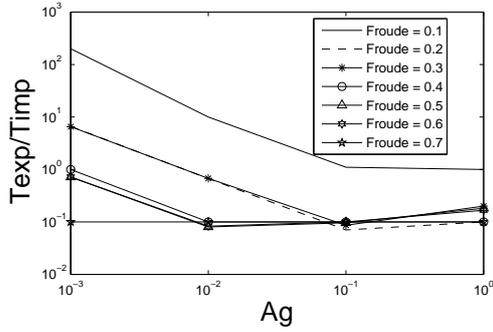


Figure 5.29: 1st-order

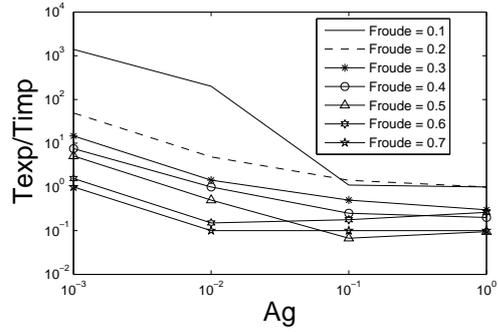


Figure 5.30: 2nd-order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	200	10	≥ 1	1
$Froude_i = 0.2$	6.63	0.672	0.07	0.1
$Froude_i = 0.3$	6.51	0.675	0.086	0.2
$Froude_i = 0.4$	1	0.1	0.1	0.1
$Froude_i = 0.5$	0.716	0.08	0.096	0.182
$Froude_i = 0.6$	0.718	0.083	0.1	0.167
$Froude_i = 0.7$	0.1	0.1	0.1	0.1

Table 5.2: Time computational ratios between explicit and implicit time advancing at first order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	1400	200	≥ 1	1
$Froude_i = 0.2$	48.9	4.85	1.42	1
$Froude_i = 0.3$	14.64	1.43	0.5	0.3
$Froude_i = 0.4$	7.5	1	0.25	0.2
$Froude_i = 0.5$	5.1	0.5	0.067	0.095
$Froude_i = 0.6$	1.56	0.15	0.178	0.262
$Froude_i = 0.7$	1	0.1	0.1	0.1

Table 5.3: Time computational ratios between explicit and implicit time advancing at second order

Chapter 6

Variation of the Froude number by acting on h

6.1 Simulations for the Grass model

With this second approach, the Froude number has been changed by modifying the height of the flow. However, it has not been reduced too much, since a strong reduction of h entails a big decrease of the dimensions of the bed profile as well and then the test case could not be of physical interest. That is why the initial Froude number has been increased up to the value of 0.4, by steps of 0.1 as before. The computational domain has been enlarged as in the previous case. The total simulation time of every simulation has been modified even in this case, but since the inlet velocity has not been changed, a linear way has been used for decreasing it, as shown in the following table.

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	500000	50000	7000	700
$Froude_i = 0.2$	250000	25000	3500	350
$Froude_i = 0.3$	167000	16700	2340	234
$Froude_i = 0.4$	125000	12500	1750	175

Table 6.1: Final simulation times (seconds) at varying Froude numbers

6.2 Efficiency comparison

As before, at increasing values of the Froude number, the relation between CFL and A_g is no more simply inversely linear, since for the accuracy of the implicit solutions lower values of the CFL number have to be used. In order to establish the maximum value of CFL for the implicit solution giving reasonable results, we distinguished the solutions evaluating the maximum of the absolute value of the error and the mean-square error with respect to the explicit solution. The errors are made nondimensional by using the maximum value of the initial bed load profile and the inlet velocity. We accepted solutions with a mean-square error lower than the 1% and a maximum of the absolute value of the error smaller than the 10% for the evolution of the bottom profile, and values lower than 0.3% and 3% respectively for the evolution of the velocity one. At the end of these simulations, other empirical relations between the CFL number and A_g were found at each Fr_i , and they are shown in the following table. Obviously, lower values than 1 of the CFL are not convenient for an implicit solution, so once it is reached the efficiency of the implicit solution can be no longer improved and therefore the errors with respect to the explicit one could exceed the boundaries defined. This procedure has been applied for both of the models used with this approach; the new empirical relations found from the simulations for the Grass model are contained in table B.

	$CFL = \frac{10}{A_g}$	$CFL = \frac{1}{A_g}$	$CFL = \frac{0.1}{A_g}$	$CFL = \frac{0.01}{A_g}$	$CFL = \frac{0.001}{A_g}$
$Fr_i = 0.1$ 1st ord		X			
$Fr_i = 0.1$ $DeC = 1$ 2nd ord		X			
$Fr_i = 0.1$ $DeC = 3$ 2nd ord	X				
$Fr_i = 0.2$ 1st ord		X			
$Fr_i = 0.2$ $DeC = 1$ 2nd ord			X		
$Fr_i = 0.2$ $DeC = 3$ 2nd ord		X			
$Fr_i = 0.3$ 1st ord			X		
$Fr_i = 0.3$ $DeC = 1$ 2nd ord			X		
$Fr_i = 0.3$ $DeC = 3$ 2nd ord			X		
$Fr_i = 0.4$ 1st ord			X		
$Fr_i = 0.4$ $DeC = 1$ 2nd ord				X	
$Fr_i = 0.4$ $DeC = 3$ 2nd ord			X		

Table B: CFL dependency on A_g at varying Froude number

6.3 Froude = 0.2

Figures 6.1-6.4 show the simulations computed at Froude = 0.2 for the bottom and the velocity profiles. Making a comparison with the simulations carried out with the same Froude number in the chapter before, a good matching of the implicit solutions with the correspondent explicit ones is obtained using one order of magnitude larger for CFL values at first order, and this enables greater time computational ratios for this case, as it can be seen in the table for the first order at the end of the section for the Grass model. Instead, for the second order the situation has not been changed. Besides, even at high values of A_g no significant fluctuations are noticeable. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

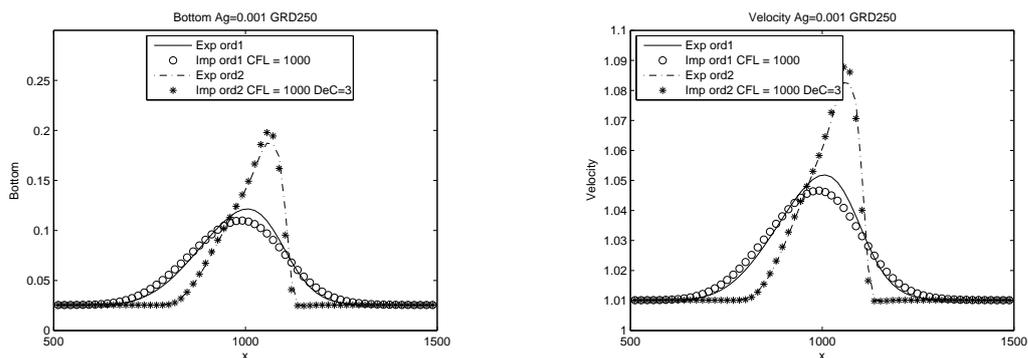


Figure 6.1: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.2

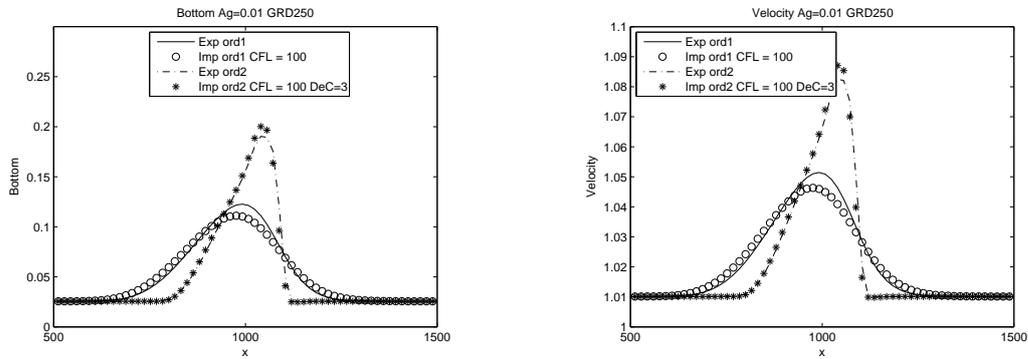


Figure 6.2: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.2

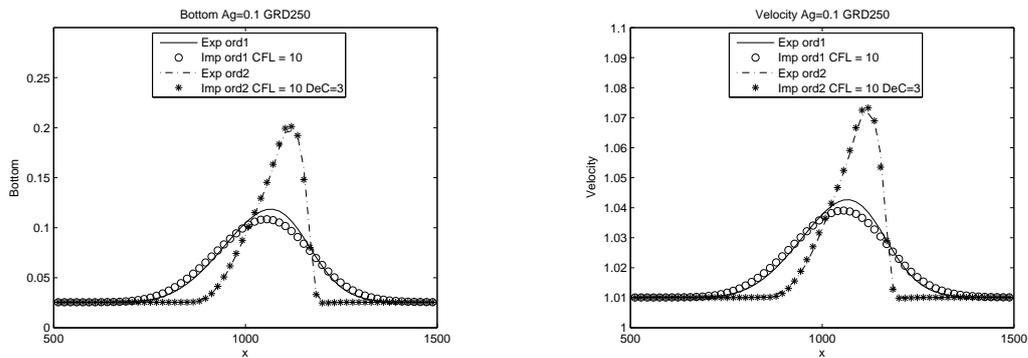


Figure 6.3: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.2

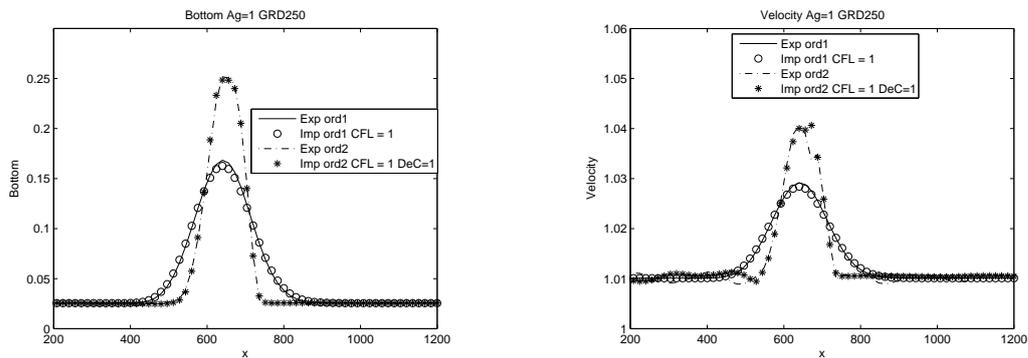


Figure 6.4: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.2

6.4 Froude = 0.3

Figures 6.5-6.8 show the simulations computed at Froude = 0.3 for the bottom and the velocity profiles. Looking at what has been found for the same Froude number in the chapter before, the accuracy of the implicit solutions with respect to the correspondent explicit ones is obtained using the same values of CFL both at the first order and at the second one, so it can be seen how also in this case the maximum CFL allowable is one order of magnitude lower than to the one used for Froude = 0.1. As before, even at high values of A_g no significant fluctuations are noticeable, instead of what has been seen for the corresponding case in the previous chapter. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

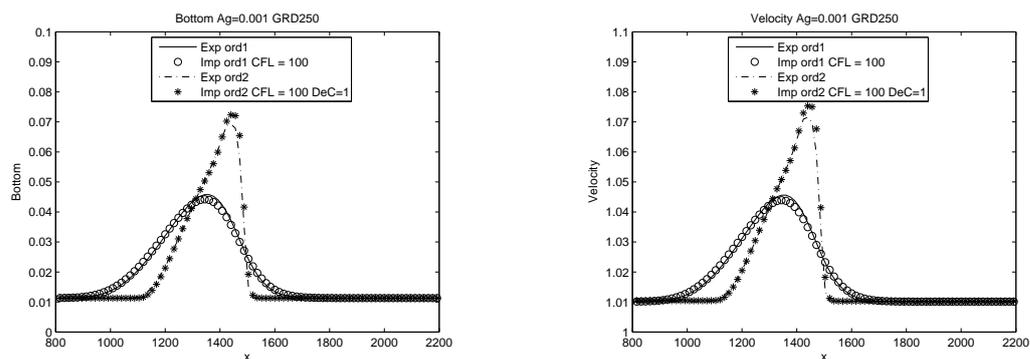


Figure 6.5: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.3

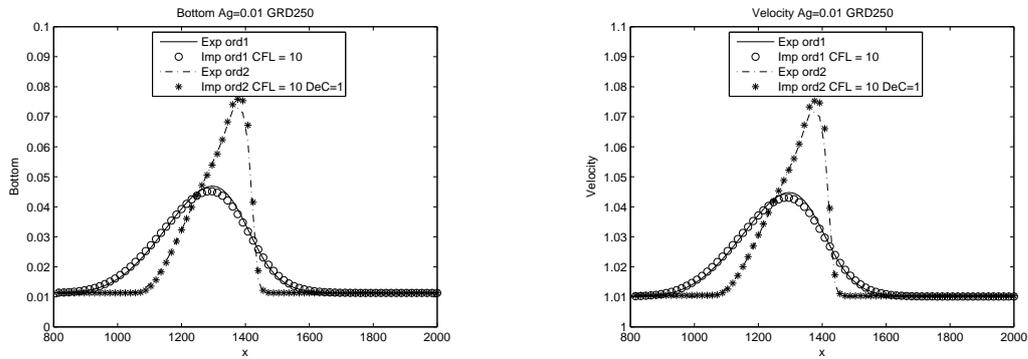


Figure 6.6: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.3

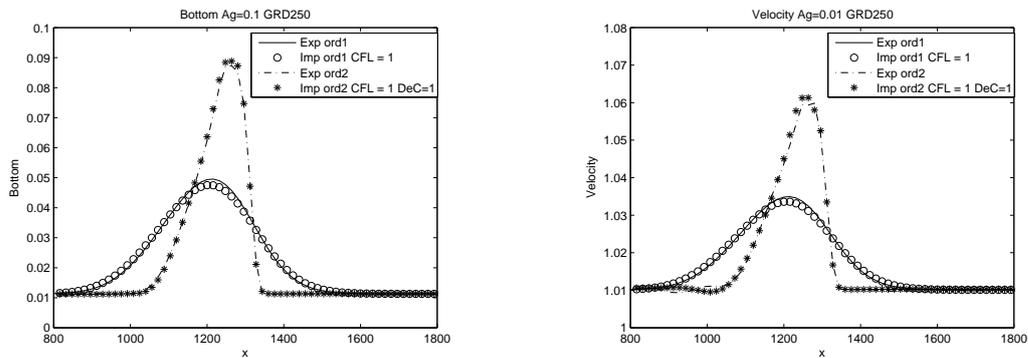


Figure 6.7: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.3

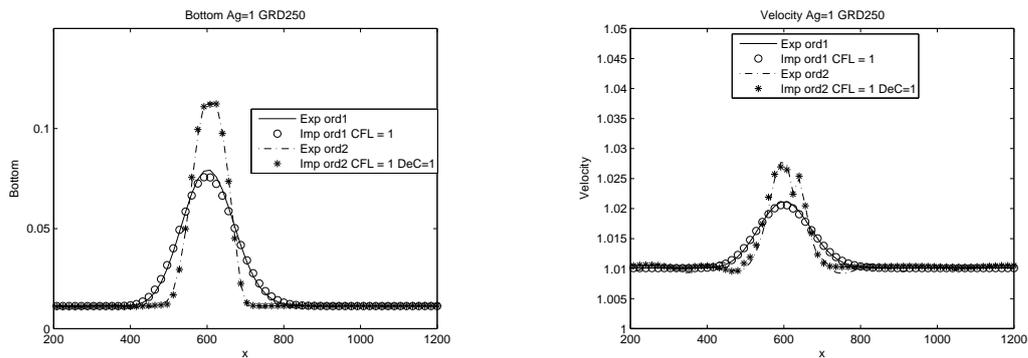


Figure 6.8: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.3

6.5 Froude = 0.4

Figures 6.9-6.12 show the simulations computed at Froude = 0.4 for the bottom and the velocity profiles. As for the case of Froude = 0.2, if we make a comparison with the corresponding case in the previous chapter, one order of magnitude larger for the CFL value has been used for implicit simulations at first order, while the second one shows the same behavior. Also, even for this Froude number the bottom and, especially, the velocity profiles looks much more stable at high values of A_g than the corresponding case in the previous chapter. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

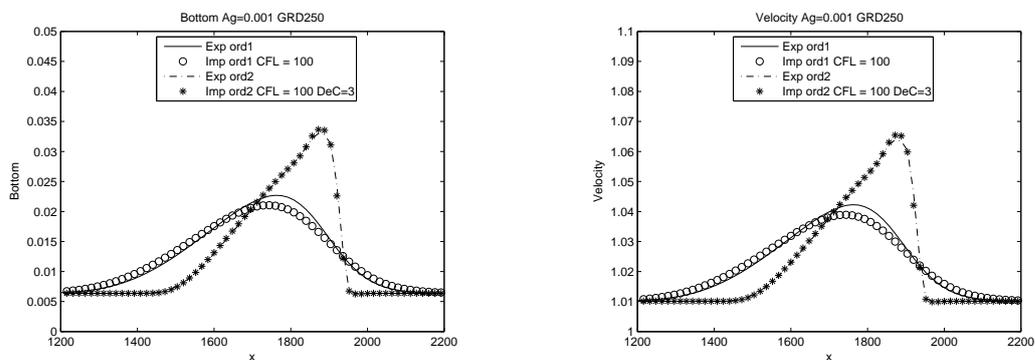


Figure 6.9: Bottom and velocity profiles for Grass model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.4

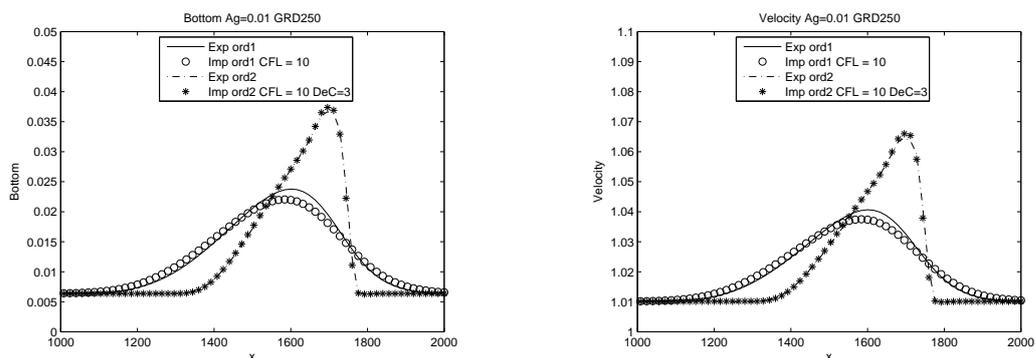


Figure 6.10: Bottom and velocity profiles for Grass model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.4

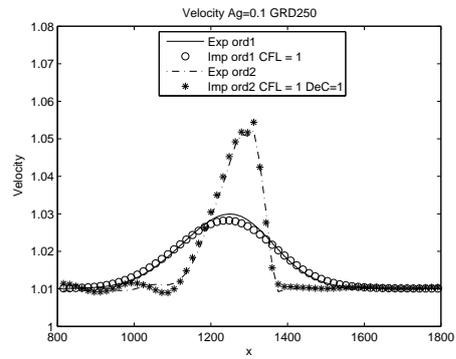
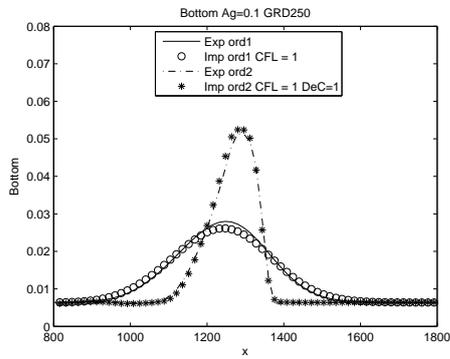


Figure 6.11: Bottom and velocity profiles for Grass model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.4

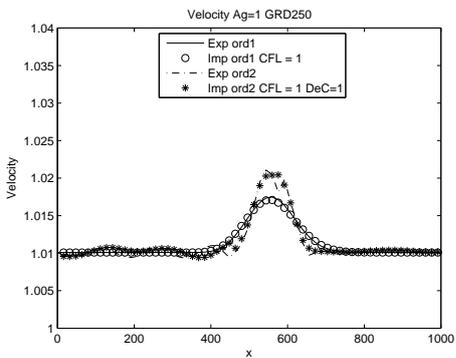
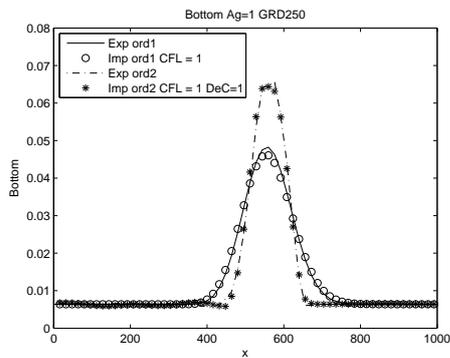


Figure 6.12: Bottom and velocity profiles for Grass model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.4

6.6 Concluding annotations

As done in the previous chapter, after a first analysis of the maximum CFL allowed where it has been decreased of steps equals to 10, a more refined investigation on the CFL number was made for those simulations where the explicit solution was found to be more convenient, by decreasing the CFL values of steps equals to 1 instead of 10. However, also in this case this more refined research have not led results since an unaccetable loss of accuracy has been found each time the CFL number has been increased even slightly.

As said before, at increasing Froude number the allowable value of CFL becomes smaller but making a comparison with the same Froude numbers of the last chapter, at first order a better behavior, in terms of maximum CFL allowed, is shown, while in the second order no particular differences are noticeable with respect to the corresponding cases of the previous analysis.

So these simulations confirm the trend shown before: at increasing values of the Froude number, the computational time ratio between explicit and implicit time advancing moves quickly towards values smaller than 1, i.e. explicit solutions are less expensive than implicit ones. This is more evident at first order, where, except for $Fr = 0.1$, explicit solutions are faster for the large part of the types of interaction between bed load and water flow. At second order this evidence is mitigated, as shown in the following images.

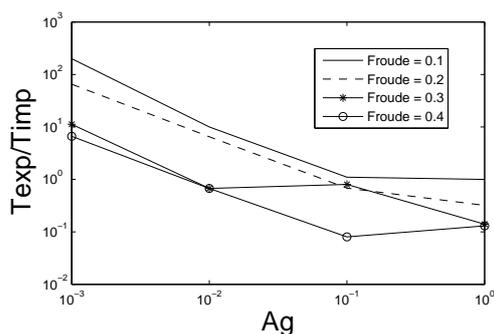


Figure 6.13: 1st-order

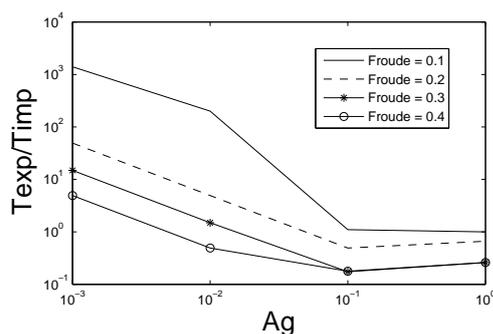


Figure 6.14: 2nd-order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	200	10	≥ 1	1
$Froude_i = 0.2$	65.3	6.5	0.68	0.32
$Froude_i = 0.3$	11.2	0.67	0.08	0.14
$Froude_i = 0.4$	6.6	0.67	0.08	0.13

Table 6.2: Time computational ratios between explicit and implicit time advancing at first order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	1400	200	≥ 1	1
$Froude_i = 0.2$	49.2	4.9	0.49	0.66
$Froude_i = 0.3$	14.82	1.49	0.173	0.26
$Froude_i = 0.4$	4.9	0.49	0.178	0.26

Table 6.3: Time computational ratios between explicit and implicit time advancing at second order

6.7 Simulations for the MPM model: efficiency accuracy

What has been done for the Grass model is now applied to the MPM one, since by varying h for increasing the Froude number the threshold condition on the velocity is preserved. At increasing values of the Froude number, for the accuracy of the implicit solutions lower values of the CFL number have to be used and so the previous dependency of CFL on A_g found for this model at Froude = 0.1 is not valid anymore. As before, in order to establish the maximum value of CFL for the implicit solution giving reasonable results, we distinguished the solutions evaluating the maximum of the absolute value of the error and the mean-square error with respect to the explicit solution. The errors are made nondimensional by using the maximum value of the initial bed load profile and the inlet velocity. We accepted solutions with a mean-square error lower than the 1% and a maximum of the absolute value of the error smaller than the 10% for the evolution of the bottom profile, and values lower than 0.3% and 3% respectively for the evolution of the velocity one. At the end of these simulations, other empirical relations between the CFL number and A_g were found at each Fr_i , and they are shown in the table C. Once the CFL reaches a value of 1, it is useless to use implicit simulations with lower values for this condition from the point of view of computational costs.

	$CFL = \frac{10}{A_g}$	$CFL = \frac{1}{A_g}$	$CFL = \frac{0.1}{A_g}$	$CFL = \frac{0.01}{A_g}$	$CFL = \frac{0.001}{A_g}$
$Fr_i = 0.1$ 1st ord		X			
$Fr_i = 0.1$ $DeC = 1$ 2nd ord	X				
$Fr_i = 0.1$ $DeC = 3$ 2nd ord	X				
$Fr_i = 0.2$ 1st ord		X			
$Fr_i = 0.2$ $DeC = 1$ 2nd ord		X			
$Fr_i = 0.2$ $DeC = 3$ 2nd ord		X			
$Fr_i = 0.3$ 1st ord		X			
$Fr_i = 0.3$ $DeC = 1$ 2nd ord			X		
$Fr_i = 0.3$ $DeC = 3$ 2nd ord		X			
$Fr_i = 0.4$ 1st ord			X		
$Fr_i = 0.4$ $DeC = 1$ 2nd ord			X		
$Fr_i = 0.4$ $DeC = 3$ 2nd ord		X			

Table C: CFL dependency on A_g at varying Froude number

6.8 Froude = 0.2

Figures 6.15-6.18 show the simulations computed at Froude = 0.2 for the bottom and the velocity profiles. The matching of the implicit solutions with the correspondent explicit ones is really good provided that the CFL values for the implicit time advancing remains under some limits, which depend on A_g , as it is reported in the figures and it will be discussed in the following. As happened for the Grass model, lower values for the maximum CFL allowed have to be used in order to reach a satisfying accuracy for the implicit solutions: in this particular case, both at the first and at the second order, an inversely proportional dependency of the CFL on A_g is recovered, with no changes when using one or more DeC iterations. The form of the profiles is very similar to that obtained at Froude = 0.1, in particular it is possible to notice how the variation of velocity become smaller as the interaction between the bedload and the flow is faster. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

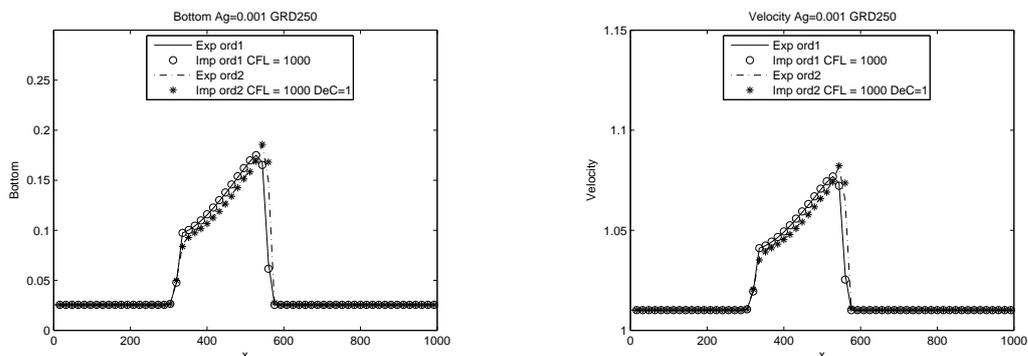


Figure 6.15: Bottom and velocity profiles for MPM model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.2

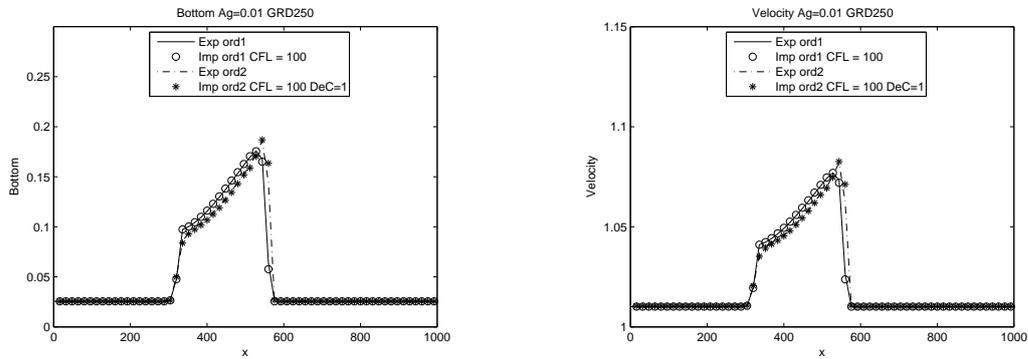


Figure 6.16: Bottom and velocity profiles for MPM model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.2

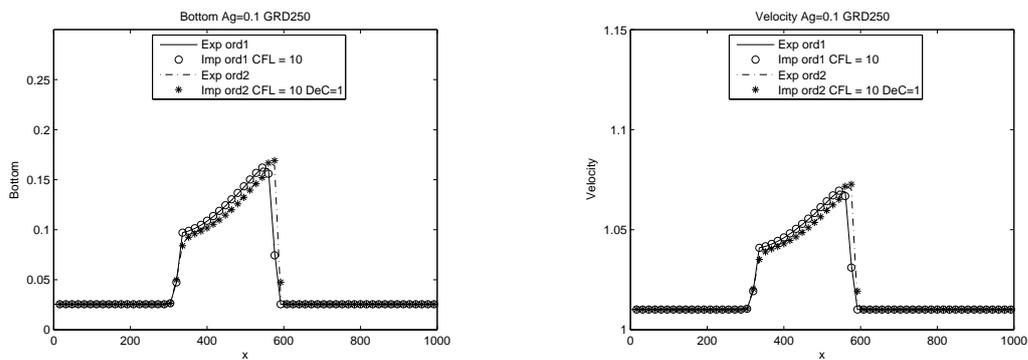


Figure 6.17: Bottom and velocity profiles for MPM model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.2

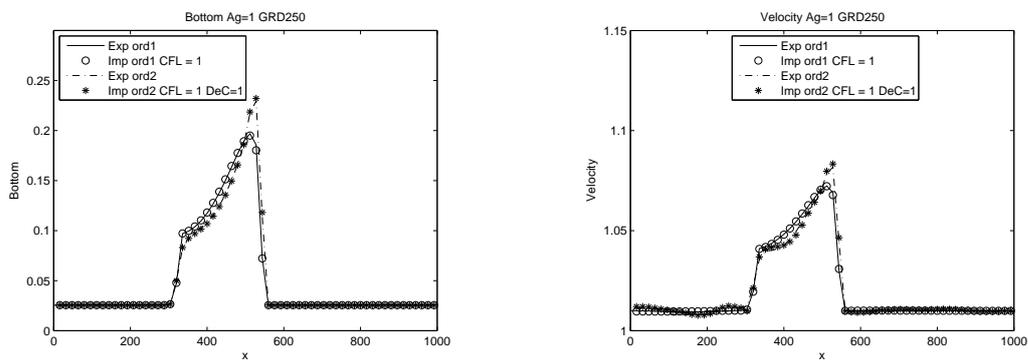


Figure 6.18: Bottom and velocity profiles for MPM model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.2

6.9 Froude = 0.3

Figures 6.19-6.22 show the simulations computed at Froude = 0.3 for the bottom and the velocity profiles. In this case, an important feature can be noticed. For Froude numbers equal to 0.1 and 0.2, the use of more DeC iterations was basically useless in achieving higher values of the CFL allowed for the implicit simulations. Now it appears from these figures than the use of 3 DeC iterations enables to maintain the same order of magnitude for the CFL values of the previous subsection, while using just 1 DeC iteration lower values have to be used. The first order, instead, shows the same behavior of that seen for the previous Froude number. As for the corresponding case for the Grass model but in the previous chapter, at high values of A_g significant fluctuations of the simulations start to be noticeable. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

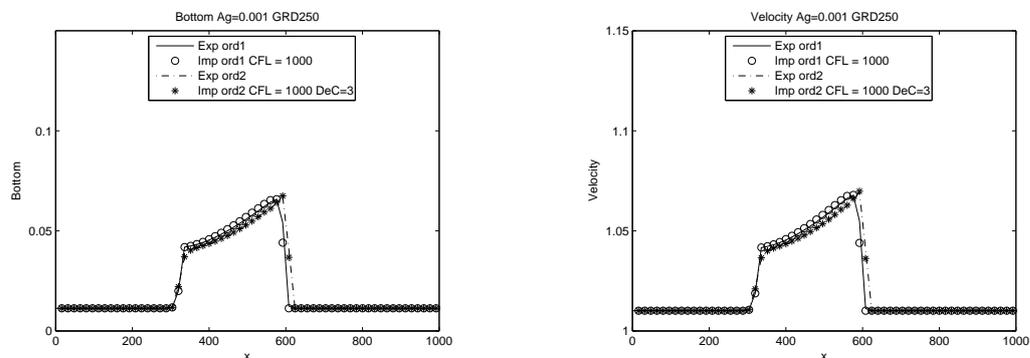


Figure 6.19: Bottom and velocity profiles for MPM model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.3

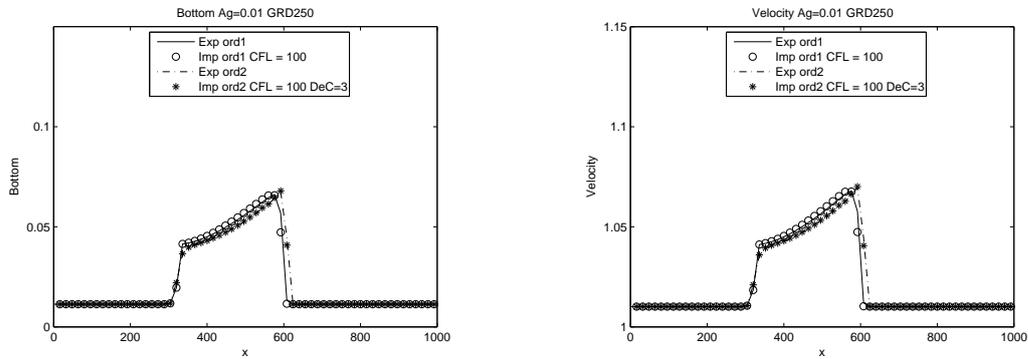


Figure 6.20: Bottom and velocity profiles for MPM model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.3

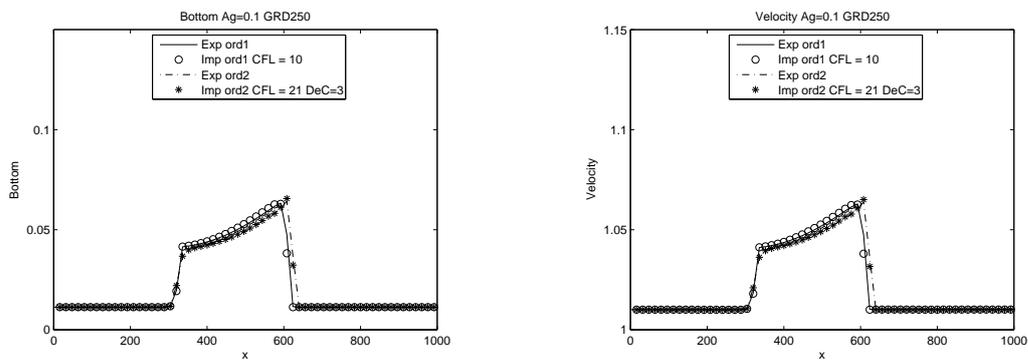


Figure 6.21: Bottom and velocity profiles for MPM model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.3

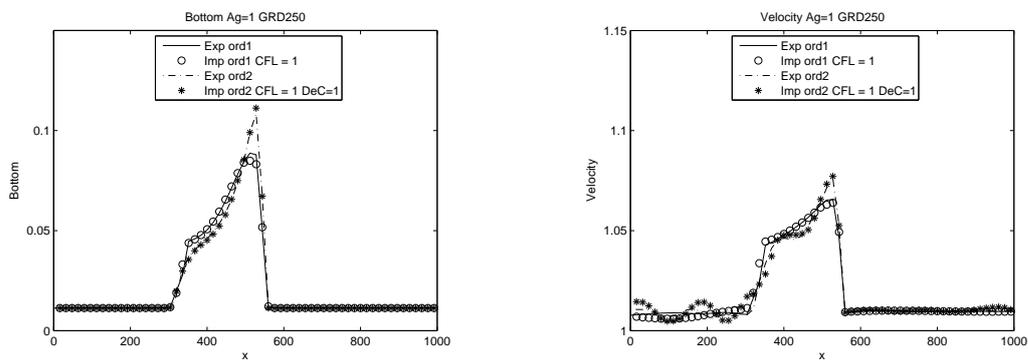


Figure 6.22: Bottom and velocity profiles for MPM model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.3

6.10 Froude = 0.4

Figures 6.23-6.26 show the simulations computed at Froude = 0.4 for the bottom and the velocity profiles. This case shows the same behavior for the second order of the previous Froude number, while at the first order the maximum CFL value has to be decreased to lower values. The fluctuations at high values of A_g now are very evident, especially for the velocity profiles. Simulations obtained with the more refined grid of 500 cells are omitted for brevity.

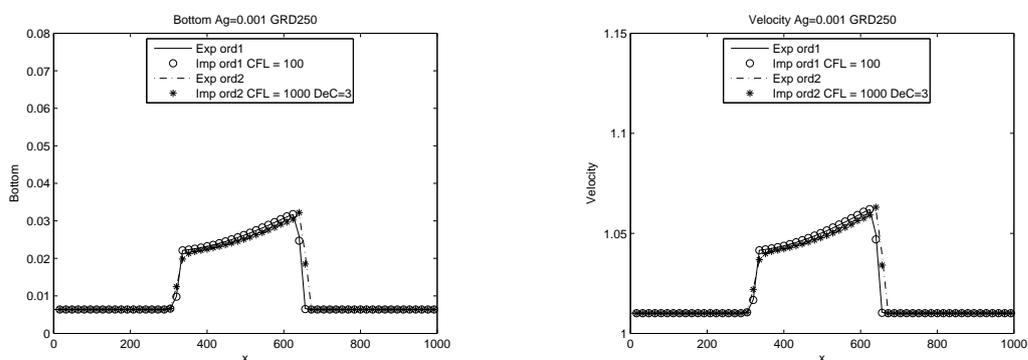


Figure 6.23: Bottom and velocity profiles for MPM model for $A_g = 0.001$ using a grid made by 250 cells at Froude = 0.4

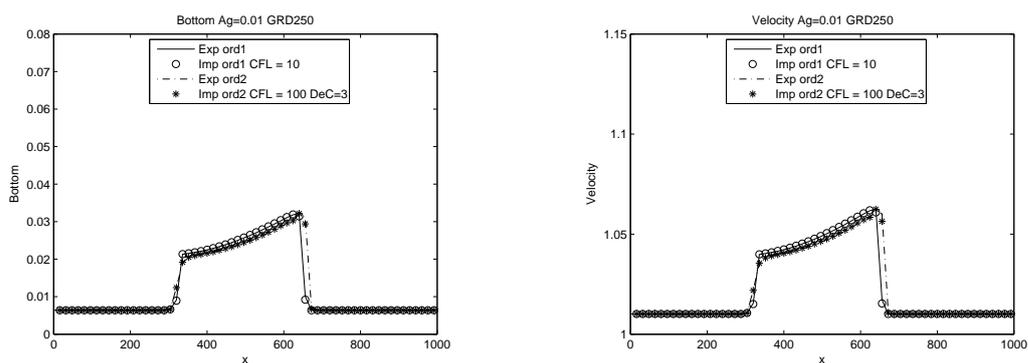


Figure 6.24: Bottom and velocity profiles for MPM model for $A_g = 0.01$ using a grid made by 250 cells at Froude = 0.4

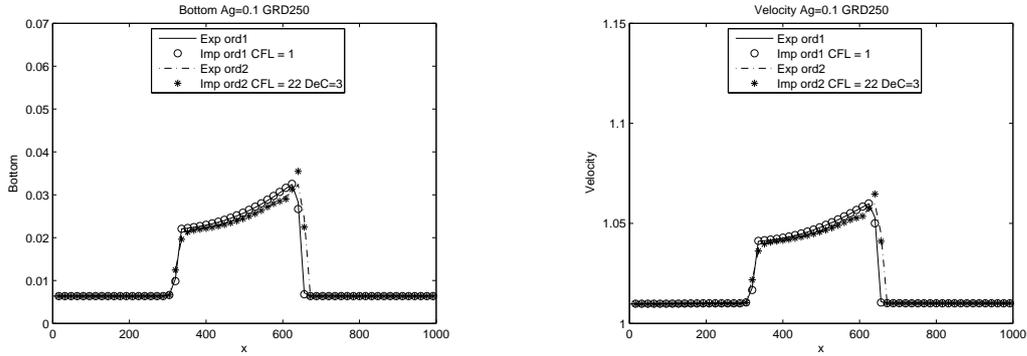


Figure 6.25: Bottom and velocity profiles for MPM model for $A_g = 0.1$ using a grid made by 250 cells at Froude = 0.4

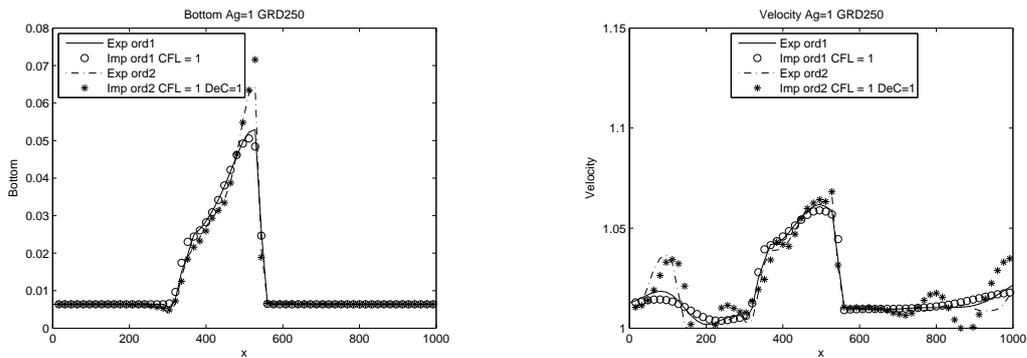


Figure 6.26: Bottom and velocity profiles for MPM model for $A_g = 1$ using a grid made by 250 cells at Froude = 0.4

6.11 Concluding annotations

The more refined investigation on the CFL number has led some results in this case. In fact, for Froude numbers equal to 0.3 and 0.4, at values of A_g of 0.1, values of CFL of 21 and 22 respectively have been used. For this cases, the computational cost of the implicit solutions is of the same type of the corresponding explicit ones, and this explains the values equal to 1 in the time computational ratios tables shown in the following tables for those cases.

Anyway, the MPM model shows the same trend of the Grass one, i.e. at increasing Froude number the allowable value of CFL becomes smaller and making a comparison between the first and the second order it is shown how implicit solutions are still good for a large number of cases for the second order. Overall, this model shows a better behavior in terms of maximum CFL allowed with respect to the Grass one.

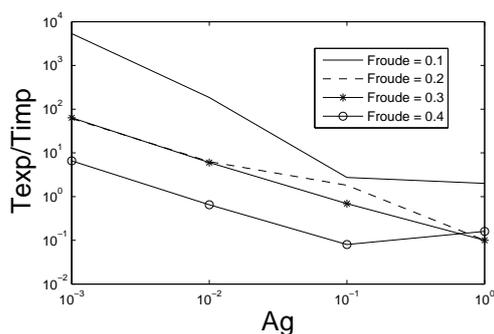


Figure 6.27: 1st-order

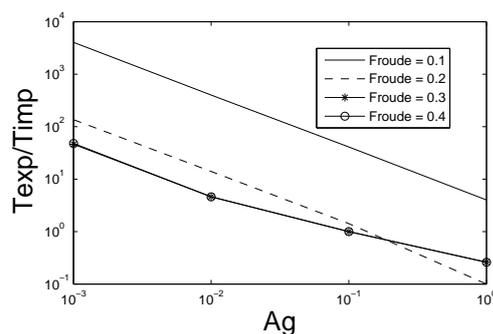


Figure 6.28: 2nd-order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	5360	185	2.73	2
$Froude_i = 0.2$	60.98	6.3	1.81	0.1
$Froude_i = 0.3$	63.64	6	0.69	0.1
$Froude_i = 0.4$	6.56	0.65	0.08	0.16

Table 6.4: Time computational ratios between explicit and implicit time advancing at first order

	$A_g = 0.001$	$A_g = 0.01$	$A_g = 0.1$	$A_g = 1$
$Froude_i = 0.1$	4030	403	40.9	4
$Froude_i = 0.2$	135.5	14	1.42	0.1
$Froude_i = 0.3$	46.14	4.61	1	0.26
$Froude_i = 0.4$	48	4.58	1	0.26

Table 6.5: Time computational ratios between explicit and implicit time advancing at second order

Chapter 7

Application to a contracting channel

The last part of this work focuses on checking the accuracy of implicit solutions by a comparison with the results of [1] over a 2D contracting channel.

7.1 The test case

The channel is 2000 m long and 500 m wide with a contraction zone in the centre of the channel as depicted in fig.7.1 . The upper curvature y_t and the low curvature y_b for the contracting channel are defined by:

$$y_t(x) = 500 - 125 \exp\left(-\frac{(x - 1000)^2}{250^2}\right), \quad y_b(x) = 125 \exp\left(-\frac{(x - 1000)^2}{250^2}\right) \quad (7.1)$$

Initially the bed is flat and the water level and velocity field are as follows:

$$h(0, x, y) = 10m, \quad u(0, x, y) = \frac{Q}{h(0, x, y)}, \quad v(0, x, y) = 0, \quad (7.2)$$

where Q is a constant discharge. The water density $\rho_w = 1000 \text{ kg/m}^3$, the porosity $p = 0.4$ and the water discharge $Q = 10 \text{ m}^2/\text{s}$. As boundary conditions we fixed the discharge in x -direction to $10 \text{ m}^2/\text{s}$ at the upstream boundary and on all remaining boundaries we used the free flow conditions.

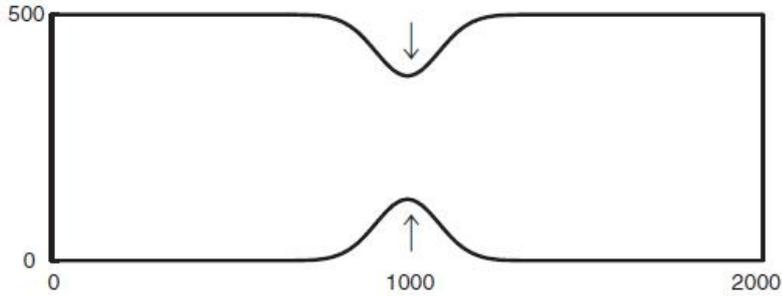


Figure 7.1: Domain of the contracting channel flow.

In all the simulations carried out in [1] explicit time advancing was used with a CFL equal to 0.8, while three kinds of meshes were considered: an adaptive one, a fixed coarse one and a fixed fine one, with the last two depicted in fig. 7.2. The study in [1] used both the Grass model and the MPM one to model

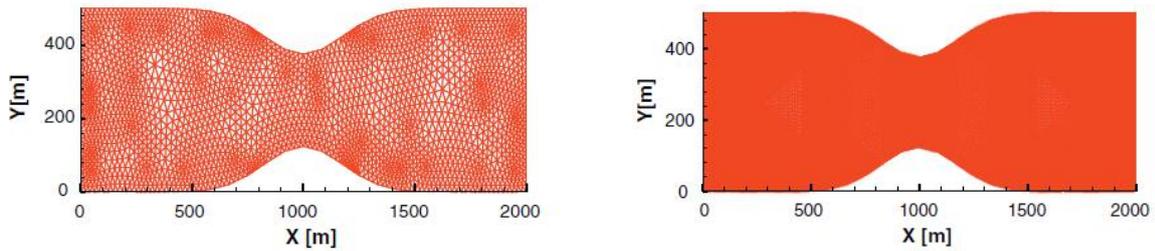


Figure 7.2: Coarse mesh (left) and fine mesh (right) used in [1]

discharge, while in this work only implicit solutions for the Grass formula have been carried out. The A_g parameter is set to 0.001 resulting in a slow interaction between the bedload and the water flow, while, as it is possible to obtain from the data given before, the Froude number $Fr = \sqrt{u^2 + v^2}/\sqrt{gh}$ is about 0.1.

7.2 Mesh definition

In order to apply the numerical method of this work to the previous test case, the first step has been building the grid. The code developed in [4] uses the Grass model for sediment transport and discretizes the system with the

MR scheme. The mesh file request is an *.amdba* file, which is composed as follow:

- **first line** $a b$ (where a is the number of the nodes and b the number of the triangles)
- **a lines** node table (number of the node - x coordinate - y coordinate - label)
- **b lines** connectivity table (number of the triangle - node 1 - node 2 - node 3 - label)

The label indicates where the vertex is located:

- 0 for a point in the flow field;
- 1 for a point on the wall;
- 2 for a point on the outlet line;
- 3 for a point on the inlet line.

Thus, it is necessary to create the mesh and to assign to each node the corresponding label. The mesh file is obtained with MATLAB, using the function *pdepoly*. Three kinds of mesh have been considered: a coarse one and two fine meshes. In the following, these meshes will be addressed as *Mesh 1*, *Mesh 2* and *Mesh 3* respectively. The more refined meshes show a larger number of nodes and cells in the central zone, especially for Mesh 3 and near the domain boundaries where, according to the reference study, the higher variations of the bed profile take place. In order to do this, a particular setting of the *pdepoly* toolbox has been modified: the growth of the cell dimensions has been changed from the standard value of 1.3 to closer values to 1, obtaining also cells with very similar dimensions. Using a graphical interface the *.amdba* file is created. Table 7.1 resumes the characteristics for each mesh, which are also shown in the figures 7.3-7-5.

As for the bathymetry, it has been set constant and equal to 10 m.

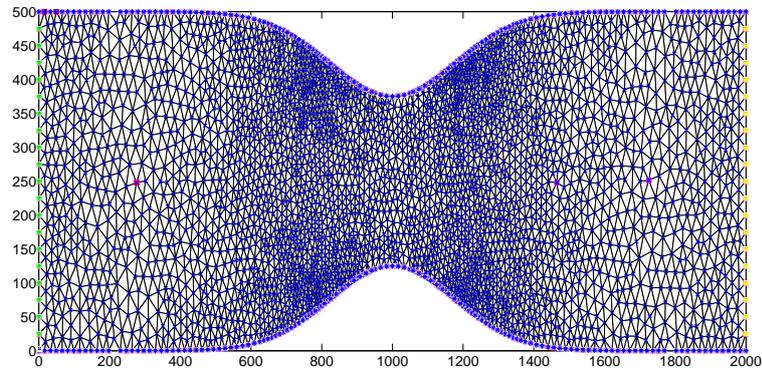


Figure 7.3: Mesh 1 with the highlighted contour, inlet, outlet and fluid points

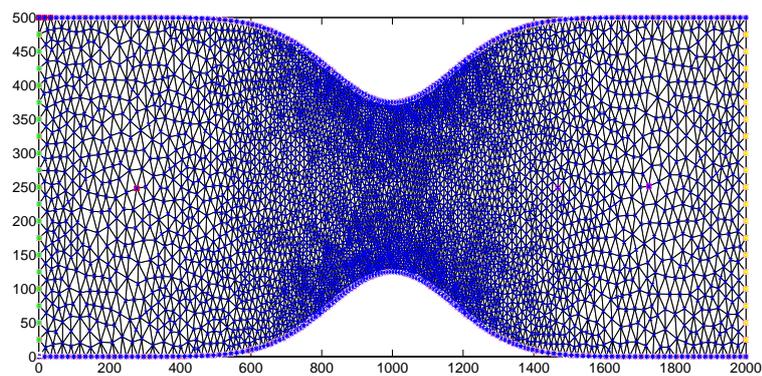


Figure 7.4: Mesh 2 with the highlighted contour, inlet, outlet and fluid points

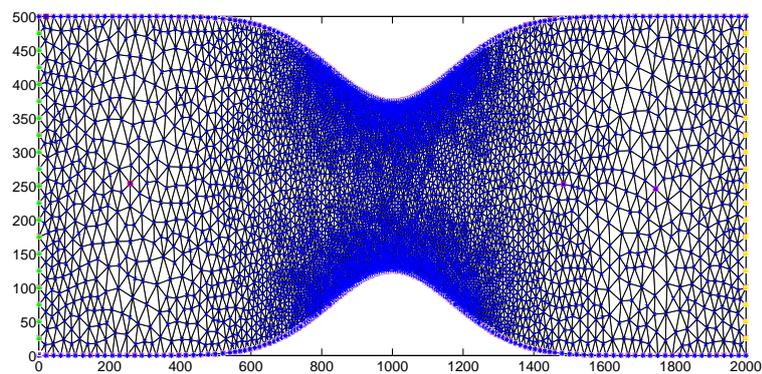


Figure 7.5: Mesh 3 with the highlighted contour, inlet, outlet and fluid points

Mesh	Contour points	Nodes	Triangles
Mesh 1	312	3479	6616
Mesh 2	340	4792	9208
Mesh 3	368	5877	11388

Table 7.1: Mesh characteristics

7.3 Results and discussion

Once the meshes have been set, two dimensional implicit 2nd order simulations were carried out setting the same parameters of the study of interest. As for the CFL number, two values were used: a value of 1000, according to the linear inversely relation with the A_g parameter for $Fr = 0.1$, and one of 100. Second-order accurate simulations were carried out, in which that independence of the time step was reached. The number of DeC iterations was set to 1. Then to have an overview of the results and compare them to those of PARAVIEW, an open source multiple-platform application for interactive, scientific visualization, has been used.



Figure 7.6: Height of the flow (right) and modulus of the velocity (right) at $t = 10$ h for implicit solutions with $CFL = 100$ using Mesh 1.

Figure 7.6 shows what happens in the channel when the fluid flows in it: an acceleration of the flow takes place in the centre of the channel, while the height of the flow increases immediately before the contraction and then decreases after the middle section of the channel. Figures 7.7-7.12 display the comparison of the snapshots for the bedload at times $t = 10, 25$ and 50

h between the implicit simulations carried out here and those obtained in the reference study. Figures 7.7-7.9 show the bedload obtained at different simulation times on Mesh 1 with both $CFL = 100$ and $CFL = 1000$. Figures 7.10-7.12 show the same plots for Mesh 2. These figures can be compared with Fig 4 of [1]. Since the results carried out with Mesh 3 are very similar to those of Mesh 2, these are not reported in the following for the sake of brevity. The time computational ratios are reported in the following tables: for Mesh 3 only simulations with $CFL = 10^3$ were carried out since no differences have been noticed when using a smaller value of CFL, as shown in the figures 7.7-7.12. Tables 7.5 shows the CPU times of the explicit simulations carried out in [comparison]: the computational gain obtained with implicit simulations is considerable by looking at these tables.

	10h	25h	50h
IMP CFL= 10^3	209,76	528,68	1070,64
IMP CFL= 10^2	528,13	1442,09	3075,75

Table 7.2: CPU times (s) for the Mesh 1

	10h	25h	50h
IMP CFL= 10^3	433,17	1097,44	2358,3
IMP CFL= 10^2	1226,98	3220,02	6774,72

Table 7.3: CPU times (s) for the Mesh 2

	10h	25h	50h
IMP CFL= 10^3	474.342	1267,14	2296.63

Table 7.4: CPU (s) for the Mesh 3

	10h	25h	50h
Coarse fixed mesh	5100	7560	11280
Fine fixed mesh	42540	74520	124980
Adaptive mesh	18540	38760	52200

Table 7.5: CPU times (s) for the three meshes used in [1]

Figure 7.7 shows the evolution for the bottom profile at $t = 10$ h; we observe that the bed reveals erosion in the converging part of the channel and develops a mound in the diverging part of it. The erosion effects are due to the increase of the water velocity in the contraction area, as shown in figure 7.6, and in regions with small velocity field no sediment transport is produced. Indeed, this can be interpreted by the fact that the velocity field is not entirely uniform across the channel with high-amplitude velocities occurring in the contraction area. Next figures show the evolution at the following times steps for different meshes.

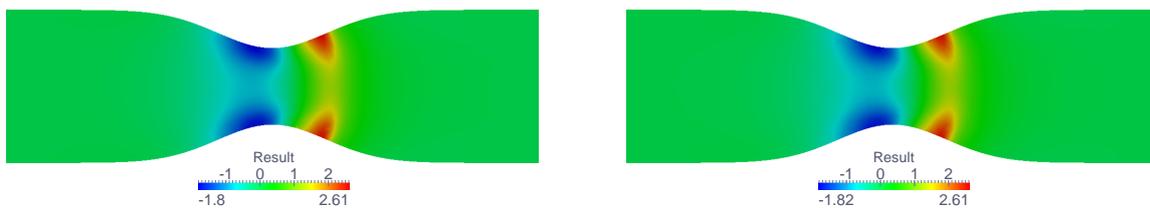


Figure 7.7: Bedload evolution at $t = 10$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 1.

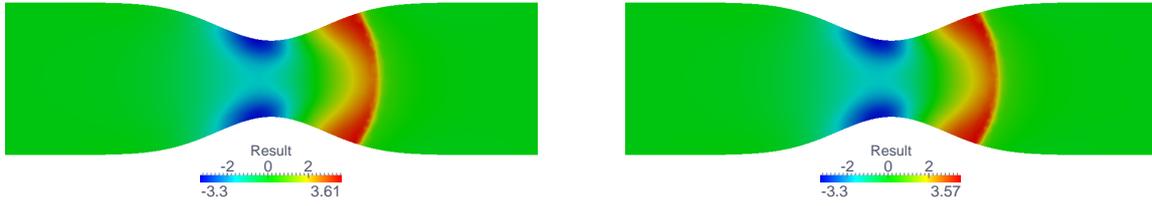


Figure 7.8: Bedload evolution at $t = 25$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 1.

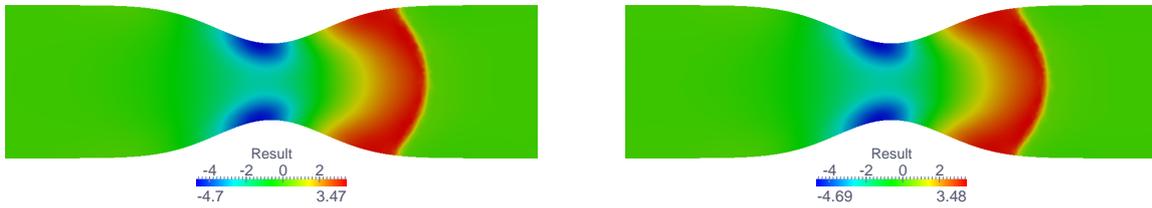


Figure 7.9: Bedload evolution at $t = 50$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 1.



Figure 7.10: Bedload evolution at $t = 10$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 2.

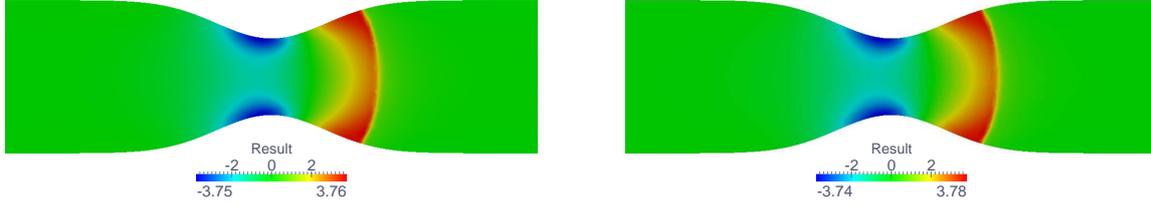


Figure 7.11: Bedload evolution at $t = 25$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 2.

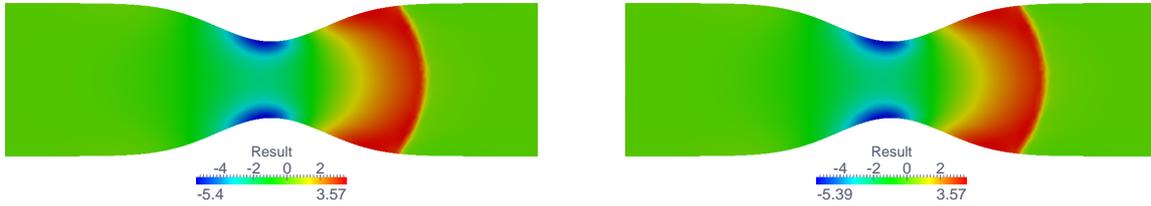


Figure 7.12: Bedload evolution at $t = 50$ h for implicit solutions with $CFL = 100$ (left) and $CFL = 1000$ (right) using Mesh 2.

Then, in order to appreciate better the evolution of the bedload, the bottom profiles at the center of the channel are reported in the following figures: the profiles from the fine fixed mesh and the adaptive one of [1] and those obtained with the three meshes studied here are compared. For the sake of brevity, just the bottom profiles achieved using the maximum allowable value of CFL are reported since negligible differences are observed for the lower CFL.

7.4 Concluding remarks

From the results shown in the previous pages some considerations can be made. First, the differences between the implicit simulations using various CFL numbers are very small, so these tests validate the use of an inversely proportional value of CFL with respect to that of A_g as provided from the 1D studies. However, by making a comparison with the simulations carried out in [1], some dissimilarities can be noticed. The evolution of the bottom

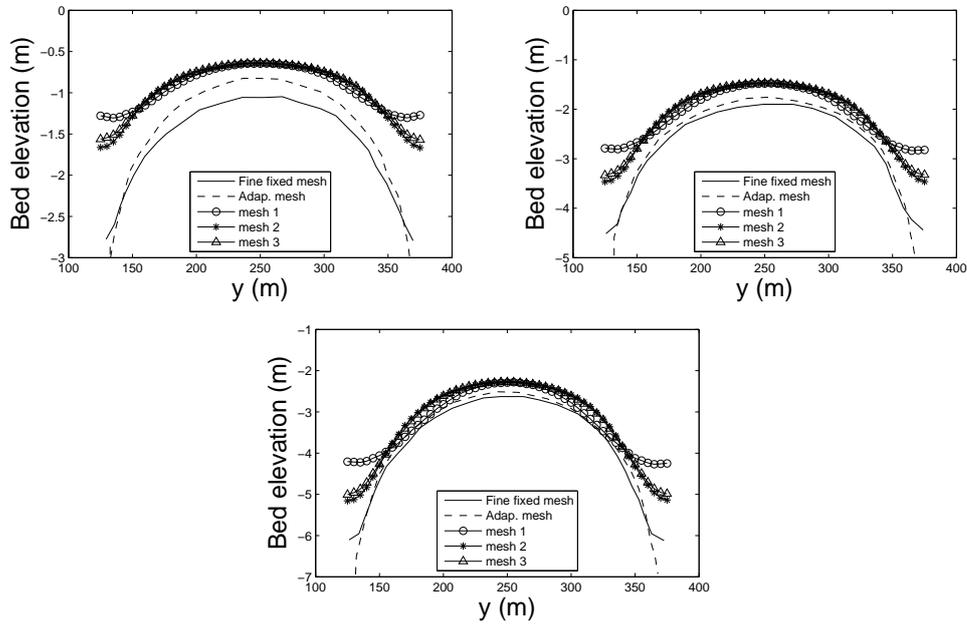


Figure 7.13: Comparison for the bottom profiles in the middle section at times $t = 10\text{h}$ (top left), $t = 25\text{h}$ (top right) and $t = 50\text{h}$ (bottom)

is slightly shifted to the right here, i.e. the lower values for the bedload are reached somewhat before the centre of the channel, and this appears more evident when looking to the bed evolution profiles displayed for the middle of the channel. It is possible to see how in [1] the lower values of the bottom profile are basically reached in the middle of the sections, while in the simulations carried out in this work these peaks are obtained slightly before, and therefore the bottom profiles at the center of the channel show higher values. The use of the fine mesh mitigates this difference, since the results are closer of those of [1]. It has seen how at increasing time steps the profiles obtained with the implicit simulations are more close to those of the study.

This can be explained by considering different grid resolution used in the present study compared to [1] and the fact that in our grids to models do not exactly lay on the channel centreline. Therefore, the variable values on the centreline are obtained by linear interpolation and this may introduce additional errors. However, the main reason of such a disparity of results is

the use of a different numerical method.

Nevertheless, in other references [30, 40] in which a similar test example has been studied by using discontinuous Galerkin methods as numerical schemes the evolution of the bottom profile shows a better agreement with the results of this work, proving the good accuracy that can be reached using implicit simulations.

Chapter 8

Conclusions

The present work has given a contribution to the assessment of the accuracy and efficiency properties at an implicit time-advancing strategy previously developed for morphodynamic Shallow-Water problems. A 1D test-case has been first considered and the investigation has been carried out for different models, speeds of interactions and for different values of the Froude number. In particular, the relationship between the maximum CFL number allowed and the parameter giving the speed of interaction between the bed and the flow, A_g , has been object of this study at increasing Froude numbers. The sediment transport models taken into consideration are the Grass model and the Meyer-Peter Müller one, while to discretize the coupled system composed by the Exner equation, used to describe the bed evolution, and the hydrodynamical part composed by the classical Shallow Water equations, the Modified Roe scheme was used.

Given a Froude number, there are two ways to increase its value: by raising the velocity of the flow or by reducing its height. Then, implicit simulations with both of the models were carried out using these different values of the Froude number. However the MPM model cannot give interesting results when increasing the velocity of the fluid since it is based on a critical velocity threshold. Therefore for the MPM model simulations were carried out only by varying the height. Using a Froude number equal to 0.1, it was found that implicit simulations are extremely convenient from the point of

view of computational costs at weak-intermediate values of the speed of interaction between the flow and the bedload and they stay competitive with the explicit simulations even when this speed reaches high values. When augmenting the Froude number, implicit solutions become less promising since the maximum CFL number allowable decreases, and using more DeC iterations can just partially mitigate this circumstance. Therefore, for the larger considered Froude numbers implicit time advancing remains interesting only for low values of A_g . Making a comparison between the two models, the MPM model shows higher values of CFL allowed by the solution accuracy, confirming what was found in a previous study. However, also for this model, a decrease of the CFL with increasing Froude has been found. The $A_g - Fr$ value couples for whom the implicit simulations are convenient have been identified for each case and model analyzed.

The last chapter focuses on the application of implicit simulations to a 2D problem modelling the bedload evolution in a contracting channel, a test case that has been object of interest in many studies [1, 30, 40]. Making a comparison to the results of [1] it is possible to notice a good agreement of the results that proves the accuracy of implicit simulations, but some differences were also found. Looking at the bedload evolution in the middle of the channel for different times steps, the results carried out with implicit solutions here show lower peaks than those of [1]: this has due to the different grid resolution and to the use of a different numerical method. In fact, in other references in which a similar test case has been studied the bottom evolution shows a better agreement with the results of this work.

Appendix A

Sediment transport formulae

- **Van Rijn** formula [10, 17, 26, 27]: in this formulation, used especially for rivers, the interaction between fluid forces and sediment grain weight produces bedload and saltation:

$$q_b = c_b u_b \delta_b \quad (\text{A.1})$$

where c_b is sediment concentration for bedload, u_b grain velocity and δ_b the dimension of the layer where sediments are transported. An other formulation proposed by Van Rijn that includes Shields parameter is:

$$\frac{q_b}{\sqrt{(G-1)gd_{50}^3}} = \frac{0.005}{C_d^{1.7}} \left(\frac{d_{50}}{h}\right)^{0.2} (\sqrt{\theta} - \sqrt{\theta_{cr}})^{2.4} \sqrt{\theta} \quad (\text{A.2})$$

while for the suspension load there is the expression:

$$q_s = \int_0^h vcdz = \bar{c}\bar{u}hF \quad (\text{A.3})$$

where c is the solid concentration, \bar{u} is the mean velocity in the field, F is an integral function of the problem and h is the height of the flow.

- **Parker** formula [29, 33] has this expression:

$$q_b = 0.00218 \theta^{3/2} G(\xi) \frac{\sqrt{g\Delta D_s^3}}{(1 - \lambda_p)}, \quad \xi = \frac{\theta}{\theta_r}, \quad \theta_r = 0.0386 \quad (\text{A.4})$$

with

$$G(\xi) = \begin{cases} 5474(1 - 0.853/\xi)^{4.5} & \xi \geq 1.59, \\ \exp[14.2(\xi - 1) - 9.28(\xi - 1)^2] & 1 \leq \xi < 1.59, \\ \xi^{14.2} & \xi < 1 \end{cases} \quad (\text{A.5})$$

- In **Camenen and Larson** formulation [10, 38] the bedload sediment transport has the following expression:

$$|q_b| = C \tau_c^{1.5} \exp\left(-4.5 \frac{\tau_{cr}}{\tau_c}\right) \quad (\text{A.6})$$

where τ_c is the shear stress at the bottom due to the current, τ_{cr} is the critical shear stress computed by the critical Shields theory and C is a constant given by:

$$C = \frac{12}{g\sqrt{\rho}(\rho_s - \rho)} \quad (\text{A.7})$$

where ρ and ρ_s are the water and sediment density.

Bibliography

- [1] Fayssal Benkhaldoun, Salah Daoudi, Imad Elmahi, Mohammed Seaid, 2011, Comparison of unstructured finite-volume morphodynamic models in contracting channel flows *Mathematics and Computers in Simulation*, vol. 81, pp. 2087-2097
- [2] F. Benkhaldoun, S. Sahmim, M. Seaid, Mathematical development and verification of a finite volume model for morphodynamic flow applications, *Numerical Methods for Partial Differential Equations*, pp. 1-21
- [3] F. Benkhaldoun, S. Daoudi, I. Elmahi, M. Seaid, 2012, Numerical modelling of sediment transport in the Nador lagoon (Morocco) *Applied Numerical Mathematics*, pp. 1-11
- [4] Marco Bilanceri, 2011, Numerical Simulations of Barotropic Flows in Complex Geometries, *PhD thesis, Corso di Dottorato in Ingegneria Aerospaziale XXIII Ciclo*
- [5] Bilanceri M., Beux F., Elmahi I., Guillard H., Salvetti M.V., 2010, Implicit simulations of shallow water equations with mobile bed., *J. C. F. Pereira and A. Sequeira (Eds), editors, ECCOMAS CFD 2010*
- [6] M. Bilanceri, F. Beux, I. Elmahi, H. Guillard, M.V. Salvetti, 2012, Linearized implicit time advancing and defect correction applied to sediment transport simulations, *Computers & Fluids*, vol. 64, pp. 13-24
- [7] R. Briganti, N. Dodd, D. Kelly, D. Pokrajac, 2011, An efficient and flexible solver for the simulation of the morphodynamics of fast evolving

flows on coarse sediment beaches *International Journal for Numerical Methods in Fluids*

- [8] V. Caleffi, A. Valiani, A. Bernini, 2007, High-order balanced CWENO scheme for movable bed shallow water equations *Advances in Water Resources*, vol. 30, pp. 730-741
- [9] A. Canestrelli, A. Siviglia, M. Dumbser, E. F. Toro, 2008, Well-balanced high-order centred schemes for non-conservative hyperbolic systems. Applications to shallow water equations with fixed and mobile bed *Advances in Water Resources*, vol. 32, pp. 834-844
- [10] M.T.Capilla A. Balaguer-Beser, 2012, A well-balanced high-resolution shape-preserving central scheme to solve one-dimensional sediment transport equations *Advances in Engineering Software*, vol. 50, pp 19-28
- [11] M. Castro Diaz, E. Fernández-Nieto, A. Ferreiro, and C. Parés, 2009, High order extensions of Roe schemes for two-dimensional nonconservative hyperbolic systems, *J Sci Comput.*, vol. 39, pp. 73-85
- [12] M.J. Castro Diaz , E.D. Fernández-Nieto, A.M. Ferreiro, C. Parés, 2009, Sediment transport models in Shallow Water equations and numerical approach by high order finite volume methods *Computers & Fluids*, vol. 37, pp. 299-316
- [13] M.J. Castro Diaz , E.D: Fernández-Nieto, A.M. Ferreiro, C. Parés, 2009, Two-dimensional sediment transport models in shallow water equations. A second order finite volume approach on unstructured meshes *Comput. Methods. Appl. Mech. Engrg.*, vol. 138, pp. 2520-2538
- [14] P. Cinat, 2012, Sediment transport models for Shallow Water equations *Università degli studi di Pisa*
- [15] N. Crnjaric-Zic, S. Vukovic, L. Sopta, 2004, Extension of ENO and WENO schemes to one-dimensional sediment transport equations *Computers & Fluids*, vol. 33, pp. 31-56

- [16] A.I. Delis, I. Papoglou, 2008, Relaxation approximation to bed-load sediment transport *Journal of Computations and Applied Mathematics*, vol. 213, pp. 521-546
- [17] R. Faeh, 2007. Numerical Modeling of Breach Erosion of River Embankments Fluent *J. Hydraul.*, vol.133, pp. 1000-1009.
- [18] Christian Ferrarin, Georg Umgiesser, Andrea Cucco, Tai-Wen Hsu, Aron Roland, Carl L. Amos, 2008, Development and validation of a finite element morphological model for shallow water basins *Coastal Engineering*, vol. 55, pp. 716-731
- [19] Giulia Garegnani, Giorgio Rosatti, Luca Bonaventura, 1999, Mathematical and numerical modelling of free surface flows over mobile-bed with high and low sediment concentration *Mathematical Modelling and Numerical Analysis*
- [20] Godlewski E, Raviart PA., 1996, Numerical approximation of hyperbolic systems of conservation laws, *Applied Mathematical Sciences*, n. 118
- [21] Gottlieb S. Shu CW, 1998, Total variation diminishing Runge-Kutta schemes, *Math Comput.*, vol. 67, pp. 73-85
- [22] Grass AJ., 1981, Sediment transport by wave and currents, *SERC London Cent Mar Technol Report No. FL29*
- [23] Guillard H., 1995, Mixed finite element-volume method in computational fluid dynamics, *von Karman Institute Lecture Series*, number VKI LS 1995-02
- [24] H. Guillard, M.Bilanceri, P. Cinat, F. Beux, I. Elmahi, M.V. Salvetti 2012, Implicit time advancing applied to shallow water problems coupled with different models of sediment transport, *European Congress on Computational Methods*
- [25] L. Hascoët and V. Pascual, 2004, TAPENADE 2.1 User's Guide, *Technical Report n 300, INRIA*

- [26] Justin Hudson, Peter K. Sweby, 2002, Formulations for Numerically Approximating Hyperbolic Systems Governing Sediment Transport *Journal of Scientific Computing*, vol. 19, Nos 1-3
- [27] J. Hudson, J. Damgaard, N. Dodd, T. Chesher, A. Cooper, 2005, Numerical approaches for 1D morphodynamic modelling, *Coastal Engineering*, vol 52, pp. 691-707
- [28] David M. Kelly, Nicholas Dodd, 2009, Floating grid characteristics method for unsteady flow over a mobile bed *Computers & Fluids*, vol. 38, pp. 899-909
- [29] D.M. Kelly, N. Dodd, 2010, Beach-face evolution in the swash zone *J. Fluid. Method.*, vol. 661, pp. 316-340
- [30] Ethan J. Kubatko, Joannes J. Westerink, Clint Dawson, 2005, An unstructured grid morphodynamic model with a discontinuous Galerkin method for bed evolution, *Ocean Modelling*, vol. 15, pp. 71-89
- [31] X. Liu, B.J. Landry, M.H. García, 2008, Two-dimensional scour simulations based on coupled model of shallow water equations and sediment transport on unstructured meshes *Coastal Engineering*, vol. 55, pp. 800-810
- [32] Martin R, Guillard H., 1996, A second order defect correction scheme for unsteady problems, *Computer Fluids*, vol. 25, pp. 9-27
- [33] Meyer-Peter & Müller's, 1948, Formulas for bed-load transport. Report on 2nd meeting on international association on hydraulic structures research, *Stockholm*, pp 39-64
- [34] J. Murillo, P. García-Navarro, 2010, An Exner-based coupled model for two-dimensional transient flow over erodible bed *Journal of Computational Physics* 229, 8704-8732
- [35] J. Noh, S. Lee, Ji-Sung Kim, A. Molinas , 2011, Numerical modeling of flow and scouring around a cofferdam *Journal of Hydro-environment Research*, pp. 1-11

- [36] Marco Petti, Silvia Bossa, 2006, Il rilievo geomorfologico dei corsi d'acqua: la loro evoluzione e stima del trasporto solido *Dipartimento di Georisorse e Territorio, Università degli Studi di Udine*
- [37] Matteo Postacchini, Maurizio Brocchini, Alessandro Mancinelli, Marc Landon, 2011, A multi-purpose, intra-wave, shallow water hydro-morphodynamic solver, *Advances in Water Resources*, vol. 38, pp. 13-26
- [38] Guy Simpson, Sebastien Castelltort, 2006, Coupled model of surface water flow, sediment transport and morphological evolution *Computers & Geosciences*, vol. 32, pp. 1600-1614
- [39] A. Siviglia, G. Stecca, D. Vanzo, G. Zolezzi, E. F. Toro, M. Tubino, 2013, Numerical modelling of two-dimensional morphodynamics with applications to river bars and bifurcations *Advances in Water Resources*, vol. 52, pp- 243-260
- [40] P.A. Tassi, S. Rhebergen, C.A. Vionnet, O. Bokhove, 2008, A discontinuous Galerkin finite element model for river bed evolution under shallow flows, *Comput. Methods Appl. Mech. Engrg.*, vol. 197, pp. 2930-2947
- [41] Z. Zhu, 2011, Simulation of suspended sediment and contaminant transport in shallow water using two-dimensional depth-averaged model with unstructured meshes *University of Illinois at Urbana-Champaign, 2011*

Ringraziamenti

Non posso che cominciare questa rassegna di ringraziamenti iniziando con la professoressa Salvetti, per il sostegno continuo datomi durante questi mesi e per la pazienza nel cercare di comprendere la mia parlantina a volte ben poco comprensibile; un grazie sentito va anche a Marco Bilanceri che, seppur la lontananza, é stato per me un validissimo aiuto, e anche e soprattutto a Paolo Cinat, per avermi pazientemente sopportato per ore via facebook e solo per gentilezza, non essendo affatto tenuto a prestarmi la sua attenzione e il suo tempo.

Questi anni di universitá sono stati fondamentali per la mia crescita, e se sono divenuti realtá é solo grazie ai miei genitori, ai miei fratelli e a mia sorella con sacrifici continui, e per questo saró loro sempre grato. Un pensiero anche ai miei nonni, zii e parenti, per aver contribuito anche loro a questo traguardo con i loro incoraggiamenti.

Infine, ma non come ultimi, vorrei ringraziare tutte le persone, amici, conoscenti e altri, che in un modo o nell'altro hanno fatto parte della mia vita, anche solo per poco: ognuno di essi ha contribuito, chi piú o chi meno, a farmi crescere e a cercare di diventare una persona migliore...fare nomi non é semplice, e tantomeno in ordine di importanza, ma come non sentirmi grato ad Alessia, Melania, Elisabetta, Pia, Francesca, Daniela, Marco, PierFrancesco, Danilo, Maria Concetta, Annalisa, Ezia, Claudia, Annarita, Lucia, Astrid e tanti, tanti altri...

Credo di aver finito...o forse no, resta una persona che devo ringraziare...me stesso. Avrei potuto e dovuto fare di meglio, sicuro, ma resta un dato di fatto: siamo noi gli artefici di ciò che ci succede, ciò che perdiamo e ciò che otteniamo...Come consiglio per il futuro non posso quindi che darvi questo: noi possiamo essere il nostro peggior nemico, ma se lo vogliamo siamo anche la persona piú preziosa che possa aiutarci.