UNIVERSITY OF TRONDHEIM
THE NORWEGIAN INSTITUTE OF TECHNOLOGY
DIVISION OF HYDRAULIC ENGINEERING

A THREE-DIMENSIONAL NUMERICAL MODEL FOR SIMULATION OF SEDIMENT MOVEMENTS IN WATER INTAKES WITH MULTIBLOCK OPTION

Version 1.2

User's guide

BY
NILS REIDAR B. OLSEN

18. DECEMBER 1993

Foreword

The SSII model was developed in 1990-91 during the work with my dr. ing. degree at the Division of Hydraulic Engineering at the Norwegian Institute of Technology. SSII is an abbreviation for Sediment Simulation In Intakes. The model was originally build around the numerical model Spider, which was made by dr. M. Melaaen during the work on his dissertation in 1989-90. Spider solves a flow problem for a general three-dimensional geometry. SSII was made up of sediment calculation routines for 3D solution of the convection-diffusion equation for the sediments, communications with Spider and a graphical user interface made in OS/2.

The main motivation for making SSII was the difficulty to simulate fine sediments in physical models. The fine sediments, often under 0.2 mm, are important for wear on turbines. It was also an advantage to be able to simulate other problems as for example sediment filling of reservoirs and waterways.

At the time SSII was made I had limited funding for computer equipment. This, together with lack of knowledge of UNIX made it necessary to develop the models on a PC. Then a problem arouse, the 640 kB limit of DOS. The arrays that the model uses is often one order of magnitude larger than the DOS limit. Because of the long computational times, it was also important to have a multi-tasking operating system. Therefore, the operating system OS/2 was used. Compared to UNIX, OS/2 is much more user-friendly, and this has been a major advantage during the development process.

After finishing my dissertation in 1991, I wanted to improve the numerical models. A disadvantage with the SSII and the Spider models for practical situations was that a structured grid was used, and it was only possible to have one block for an outblocked region. A natural improvement was a multi-block model with general outblocking possibilities. This meant considerable changes in Spider. Instead, a new water flow module for multi-block calculation was made. This model was added to SSII, and the resulting model was called SSIIM.

SSIIM, version 1.0, was uploaded on the net 17th of June 1993. Version 1.1 had some bug-corrections and some improvements in the water flow calculation for multiple blocks. Also this manual was improved. Version 1.1 was uploaded on the net 18th of October 1993. In the fall of 1993 version 1.2 was made, with an improved user interface and some additional tools.

Future plans for SSIIM is that there probably will be a bug-fix version of 1.2, perhaps called 1.3. Then maybe the program will be recompiled for the Workplace OS, which runs OS/2 applications on the PowerPC and IBM RISC workstations, maybe with multi-processor capabilities. This could speed up the computational time by one to two orders of magnitude compared with a 486. There is also a remote possibility of making a version with completely non-structured grid. This will be version 2.0, and that is at least a couple of years from now.

Several people have provided me with insight into the various problems I have

encountered in the development of this program. I have benefitted greatly from the knowledge of Prof. Melaaen at the Norwegian Institute of Technology in the subject of computational fluid dynamics. In the topics of hydraulics and sedimentation engineering I have learned from Prof. Lysne at Division of Hydraulic Engineering at the Norwegian Institute of Technology, and from Prof. Julien, Prof, Gessler, Prof. Wohl and Prof. Bienkjewicz at Colorado State University. Mr. Alfredsen at the Norwegian Hydrotechnical Laboratory has helped me with software and hardware problems. The following people have helped me test the program: M. Skoglund, O. Jimenez, A. Løvoll, L. Abrahamsen and S. Stokseth.

Thank you all for your contribution.

Trondheim, 18th of December 1993

Nils Reidar Bøe Olsen

Table of content

Chapter 1. Introduction

1.1 Model purpose and structure

SSIIM is an abbreviation for Sediment Simulation In Intakes with Multiblock option. The program is made for use in River/Environmental/Hydraulic/Sedimentation Engineering. The main motivation for the program is to simulate the sediment movements in general river/channel geometries. This has shown to be difficult to do in physical model studies for fine sediments.

The program solves the Navier-Stokes equations with the k-epsilon model on a three-dimensional almost general non-orthogonal grid. The grid is structured. A control volume method is used for the discretization, together with the power-law scheme or the second order upwind scheme. The SIMPLE method is used for the pressure coupling. The solution is implicit, also over the boundary of the different blocks. This gives the velocity field in the geometry. The velocities are used when solving the convection-diffusion equations for different sediment sizes. This gives trap efficiency and sediment deposition pattern.

The model has a user interface with capabilities of presenting graphical plots of velocity vectors and scaler variables. The plots show a two-dimensional view of the three-dimensional grid. Three plots showing the geometry from above, in a cross-section and in a longitudinal profile are available. In addition it is possible to simulate particle animation for visualization purposes.

The model includes several utilities which makes it easier to give input data. The most commonly used data can be given in dialog boxes. Several of the modules in the program can be run simultaneously as separate threads. This exploits the multi-tasking capabilities in OS/2. There is an interactive graphical grid editor with elliptic and transfinite interpolation. Grid and some of the input data can be changed during the calculation. This can be useful for convergence purposes, and also when optimizing the geometry with respect to the flow field.

For new users it is recommended to read chapter 5.1 and try the tutorial described in chapter 5.2.

1.2 Limitations of program

Some of the most important limitations of the program are listed below. Also note that some combinations of different options may not have been tested, and then there is a risk of a bug in the program.

* The program neglects non-orthogonal diffusive terms.
* The grid lines in the vertical direction have to be completely vertical.
* Internal walls cannot be used within two cells from a multi-block connection.

1.3 Disclaimer and legal matters

I disclaim all warranties with regard to this software and the information in this document, whether expressed or implied, including without limitation, warranties of fitness and merchantability. In no event shall I or my employer, SINTEF NHL, be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortuous action, arising out of or in connection with the use or performance of this software. It is therefore not recommended that the program be used for solving a problem whose incorrect solution could lead to injury to a person or loss of property. If you do use the program in such a manner, it is at your own risk. It is necessary to know that to understand and interpret the program results properly it is required that the user have knowledge and experience in computational fluid dynamics and hydraulic engineering.

Provided the user complies with the above statements, the program can be used freely.
The program can be distributed freely on condition and that an unchanged copy of this manual is distributed with the program.

Nils Reidar B. Olsen

Chapter 2. Theoretical basis

2.1 Water flow calculation

The initial water surface is generated by a standard one-dimensional backwater calculation. This prescribed water surface can afterwards be changed according to the calculated pressure field.

The Navier-Stokes equations for turbulent flow in a general three-dimensional geometry are solved to obtain the water velocity. The k-ε model is used for calculating the turbulent shear stress.

The Navier-Stokes equations for non-transient non-compressible and constant density flow can be modelled as:

(1)

The term on the left side of the equation is the convective term. The first term on the right-hand side is the pressure term. The second term on the left side of the equation is the Reynolds stress term. To evaluate this term, a turbulence model is required.

The k-ε turbulence model

The eddy-viscosity concept with the k-ε model is used to model the Reynolds stress term:

(2)

The first two terms on the right side of the equation forms the diffusive term in the Navier-Stokes equation.

The eddy-viscosity is modeled in the k-ε model as:

(3)

k is turbulent kinetic energy, defined by:

(4)

k is modeled as:

(5)

P_k is given by:

(6)

ε is modelled as:

(7)

In the above equations, the c are different constants.

The equations are discretized with a control-volume approach. An implicit solves is used, also for the multi-block option. The SIMPLE method is used for pressure-correction. The power-law scheme or the second-order upwind scheme is used in the discretization of the convective terms. The numerical methods are further described in [3] and [4].

The RNG model

Compared with the k-ε model, the RNG model for high Reynolds numbers gives the following changes:

$C_\mu = 0.085$, $C_{\varepsilon 1} = 1.42$, , $\sigma_k = 0.719$ and $\sigma_\varepsilon = 0.719$.

The parameter $c_{\varepsilon 2}$ is given by:

(8)

(9)

(10)

(11)

Wall laws

Wall laws for rough boundaries are used. These are given by Schlichting (1979):

2.2 Sediment flow calculation

Sediment transport is traditionally divided in bedload and suspended load. The

suspended load can be calculated with the convection-diffusion equation for the sediment concentration, c:

(10)

The fall velocity of the sediment particles is denoted w. The diffusion coefficient, Γ, is taken from the k-ε model:

(11)

Sc is the Schmidt number, and it is around unity.

The first term in Equation 10 is the convection of sediments, which is the sediments that are transported through the walls of the finite volume because of the velocity of the water at the wall. The right-hand term is the diffusion of sediments. For the case described here, Γ is the diffusion coefficient due to the mixing by turbulence in the water. The term therefore tells how much sediments are transported trough the wall of the finite volume because of turbulence and the difference in concentration between the two sides of the wall. The second term on the left side of Equation 10 is due to the fall velocity of the sediments. This is treated as an extra convective term, and added to the velocities in the vertical direction.

Bed boundary condition

The concentration in the elements closest to the bed are found by various methods described below. The concentration is "forced" on the bed boundary finite volumes in a similar manner as the boundary condition for ε in the k-ε model. The convection-diffusion equation is not solved for the cells closest to the bed. Sediment continuity for these cells are therefore usually not satisfied. The discrepancy in continuity is used to calculate changes in the bed levels. This method also has the advantage of simulating the interaction between the sediment that moves close to the bed and the sediment that move in suspension.

For many situations this approach means that the method for finding the bed boundary concentration becomes very important. There are several methods for finding the bed concentration. These can roughly be divided in three classes:

A. Methods based on a bed sediment concentration formula and a method to find the bed grain size distribution

B. Methods based on extrapolation from the cells above the bed

C. Maximum and minimum values of **A**, **B** and/or other estimates.

For methods in group **A** it is necessary to have a formula for the bed concentration. For a three-dimensional flow situation, van Rijn [5] developed a formula for the equilibrium sediment concentration close to the bed. Van Rijn's formula for bed concentration is given as

(10)

The sediment particle diameter is denoted d_{50}. a is a reference level, set to 1.5 % of the water depth. τ_0 is the bed shear stress, $\tau_{critical}$ is the critical bed shear stress for movement of sediment particles, ρ_w and ρ_s are the density of water and sediment, ν is the viscosity of the water and g is the acceleration of gravity.

Another approach is to use a formula for total load, for example Engelund/Hansen's formula, together with the theoretical vertical sediment and water velocity distribution for uniform flow. An algorithm for finding the bed concentration is obtained by combining these formulas.

When simulating multiple size fractions the resulting effective concentration, $c_{l,eff}$, for sediment size l is given by:

$$c_{l,eff} = c_l * f_l \quad (8)$$

where f is the fraction of size l in the bed material and c is the concentration given by the formula.

It is very important to get the correct value of f. The methods in class **A** can be divided in sub-classes according to which method is used for estimation of f.

A 1. In this method the user gives the size fractions f. This can easily lead to excessive erosion or extreme sediment concentrations if the user does not give correct values.

A 2. The size fractions are calculated by a method based on Shield's formula for critical grain size on the bed. It is assumed that d_{60} is equal to the critical diameter. A grain size curve is constructed based on this point and that the curve is a straight line spanning from $0.37d_{60}$ to $2d_{60}$.

A 3. The size fractions are set to zero initially.

A 4. The size fractions are recalculated based on deposition at the bed. The fraction of size l is proportional to the deposition of size l. After deposition is calculated for all sizes these values are scaled so that the sum does not exceed unity.

Using the **A 4** method, it is possible to start out with the **A 1**, **A 2** or **A 3** method.

The next class of methods are the extrapolation methods, **B**.

B 1. This is a first order extrapolation. It means that the concentration in the bed cell is set equal to the concentration in the cell above.

B 2. This is a second order extrapolation. The concentration in the bed cell is calculated by linear extrapolation from the two cells above the bed cell.

The advantage with the extrapolation methods is that they are independent of the bed sediment concentration formula and the bed sediment grain size distribution. However, because the concentration changes during the iteration, many more iterations are required. It is also possible to get instabilities and non-convergence.

The **A** and **B** methods can be combined in the **C** methods.

C 1 It is important to avoid erosion of sizes that are not present at the bed. This is achieved by using continuity equation to calculate a concentration which can not be exceeded.

C 2 To avoid instabilities it is possible to calculate a maximum concentration at the inlet, and always keep the concentration at any bed cell under this maximum.

C 3 A method which has been applied for sediment deposition in reservoirs is a combination of **C 2** with the maximum of **B 2** and **A 3/4**.

The different options can produce different results for many cases, and may therefore seem somewhat confusing. The algorithms are new and not very well tested. Further research may give recommendations in this matter, and may also produce improved algorithms for solving this problem.

2.3 Porosity calculation

A roughness element in a complex geometry can always be modeled by providing fine enough grid to dissolve the boundary of the roughness element. The disadvantage with this method is that the number of grid cells may be too excessive for practical use. Instead, the roughness elements must be modeled within each grid cell. This can be done in two ways, depending on the magnitude of the roughness elements. If the magnitude is fairly small compared to the size of the grid cell, the roughness can be incorporated in the law of the wall. If the roughness elements are larger, other methods must be used.

The porosity model used in this study is developed by Engelund:

(8)

In the equation the indexes ijk denotes that the variable applies for each three-dimensional cell. Vol is the volume of the cell, ρ is the density of water, β is an empirical parameter, n is the porosity and U is the velocity. The recommended value for β is 3.0, which is used in this study. Equation 13 is discretized by integration over a control volume. The effect is an extra source term in the Navier-Stokes equations.

Note that the porosity will also affect the turbulence in the flow field. To model this it would be necessary to modify the k-ε turbulence model. This is not done in this

study. It is assumed that the source term given by the porosity model is so great that it dominates over the turbulent diffusive terms. Therefore the values of k and ε in the porous domain will only have negligible effects on the flow field.

The porous domain is defined by the user for each cell as a distance above the bed. This is seen in Fig. 1. The grid lines in the vertical direction is completely vertical and parallel. This means that for each horizontal projection of the grid, one must search for the cell where the top of the porosity is located. This cell is treated with wall functions, and the porosity model is applied for the cells below. In the porous area the values for k are set to 0.01, and the values for ε are set according to wall laws.

Two parameters are required for equation 8. This is the porosity and the diameter of the characteristic particle in the porous domain. The porosity is calculated using a routine which is based on a set of measured depths at different locations in the river. For each significant rock in the river points are taken on top of the rock and on its sides. Thus, several points are taken for each bed element. The porosity generation model finds all the points in each bed cell, and calculates how many points are located within certain elevations. The top elevation, e_4, is taken to be equal to the elevation of the highest point in the cell. Three other elevations are defined as follows:

e_1 = bed level
$e_2 = 0.67 * e_1 + 0.33 * e_4$
$e_3 = 0.33 * e_1 + 0.67 * e_4$

The number of points within each elevation is counted for each cell. The following numbers are obtained:

m_{Tot} = total number of bed points within each bed element
m_3 = number of points above level e_3
m_2 = number of points above level e_2

The porosity at the bed level, e_1 is given as a user input. The following formula is used for calculating the porosity at levels e_2 and e_3:

$$n_i = (m_{Tot} - m_i) / (m_{Tot} + 0.5) \quad (9)$$

This factor is corrected by an empirical coefficient, c_i, using the following formula:

$$n_{i,corr} = (1 - c_i(1 - m_i)) \quad (10)$$

This correction formula will give no correction for c_i equal to 1.0 and higher porosity for lower values of c_i.

For each cell the porosity is calculated from interpolation from the set of porosities, n_i, as a function of the elevation, e_i.

Chapter 3. User interface

3.1 The main user interface

The main user interface appears once the program is started and the input files are read or generated.

If a **control** file is not present, the user is prompted for the necessary parameters in a dialog box. A default **control** file is then made and written to the disk.

If a **koordina** file is not present, the user is prompted for the coordinates on a grid point. A rectangular grid is then made between this point and the point (0.0, 0.0). The vertical dimensions of the grid will be given according to the given water depth of the **control** file.

The main user interface consists of a dialog box and a menu bar. The dialog box appears in the lower left corner of the screen, and the menu bar appears on the middle left side of the screen. The dialog box does not have any editfields, only text fields and a push button. The dialog box shows intermediate results. The two top lines in the dialog box are text which is written from the different modules. There are also six numbers on an exponential format. These are the residuals for the six equations in the water flow calculation. The push-button is used for refreshing the text and values in the dialog box.

The menu bar of the main user interface is used for starting different sub-modules and showing graphics. The *File* option gives the possibility of reading and writing different files. Chapter 4 describes these files. The *Input Edit* option gives the possibility of editing the input data through a grid editor or dialog boxes. The *Computations* option is used for starting water flow or sediment calculations. The *Graphics* option is used for displaying the results through different graphics windows.

3.2 Interactive input of parameters

Some of the most commonly used parameters can be set in dialog boxes. These dialog boxes are activated by the *Input Edit* choice in the main menu bar. The choices are:

GridEditor: This option is described in Chapter 3.3.
Sediment data**:** This gives the grain size, fall velocity and inflow of each grain size. Note that the number of grain sizes can not be changed.
Sediment parameters**:** This gives various parameters for calculation of sediment flow.
Waterflow parameters**:** This gives various parameters for calculation of water flow. This is an important dialog box that is used when there are convergence

problems and when parameter tests are done. Note that the parameters can be
changed while the program is calculating the water flow field.

Note that all parameters can not be given in the dialog boxes. The other parameters have to be given in the **control** file.

3.3 The grid editor

The grid editor is invoked by choosing *GridEditor* on the *Input Edit* choice in the main menu. The user can click with the mouse on a grid intersection and drag this to a new location. The mouse button must be pushed down while the movement is made.

The main menu of the grid editor is made up of five main choices plus the *Help* choice.

Move/Scale

The *Move* and *Scale* option are similar to the other presentation graphs. The option *2 Point Enlarge* under *Scale* enlarges the grid. After this choice is made, the user clicks at the lower left corner and upper right corner of the part of the grid that the user wants to see.

Utility

The option *Utility* has three choices on the pull-down menu. The first choice, *Make map*, displays the points in the **geodata** file on the grid plot. The points are shown with a square, and the different color indicate different vertical levels.

The second choice, *Make z interpolations*, generates z values for the bed surface of the grid. This is the same level as given in the **koordina** file. This option can then be used to make the z values in the **koordina** file, if the choice *write koordina* is made from the *File* option in the menu of the main user interface.

The z values are interpolated from a set of geometrical data read from the **geodata** file. If there is no **geodata** file present, an error message is given. The interpolation routine goes through all the grid points i,j, and finds the closest points in the geodata file in all four quadrants where the grid intersection (i,j) is the center of the coordinate system. Then a linear interpolation from these four points are made. If one of the points in the **geodata** file is closer than 5 cm from the grid point, this z value is chosen and no interpolation is done. The outcome of the interpolation is logged to the file **boogie.bed**. If the interpolation routine is unsuccessful in finding the point, the z value is set to zero.

The third choice, *Apply changes*, sets a global change flag. When the waterflow routine sees that this flag is set it updates the calculation geometry according to the present the grid editor geometry.

Define

This option is used for defining different parameters. The parameters are often connected to grid intersection numbers. The points that were last activated by the mouse are default values.

The first option is *Coordinates*. This gives a dialog box where the user can give numerical values for x,y and z for a grid intersection.

The second option is *Set NoMovePoint*. This invokes a mode where the user can define certain points which will not be moved by the interpolation, called **NoMovePoints**. In NoMovePoint mode it is not possible to move the grid points with the mouse. When the user clicks on a grid intersection, a blue star emerges on the intersection as a sign that this is chosen. Up to 200 **NoMovePoints** can be chosen. To verify that this mode is present, the letters "Point mode, 0" is shown on the lower part of the editwindow when the *Set NoMovePoint* is chosen. The integer shows how many points you have chosen. To return to the normal mode, choose *Define* and *Set NoMovePoint* again. It is verified that the normal mode is set because the text "Point mode" disappears. In the normal mode the user can move all points including the **NoMovePoints**.

The third option is *Delete NoMovePoint*. This deletes the last point set under the NoMovePoint mode.

The following four options are setting of attraction to certain points or lines in the grid. This is used by the elliptic grid generator. A dialog box emerges when the choice is made, and the user must give two integers which describes the location of the attraction point/line. Then two attraction parameters are given. The **A** value is proportional to the attraction. If negative, the grid lines are moved away instead of attracted. The **B** value gives an attraction proportional to the grid line difference raised to a power of **B**.
Point attraction gives attraction to points, and *Line attraction* gives attraction to lines. Up to 200 attraction points can be defined. The attraction points can be seen on the grid by colored rectangles at the grid intersections.

The last option in the *Define* menu is *Delete last attraction*. This deletes the last defined attraction.

Generate

The first choice in the pull-down menu is *Boundary*. This choice interpolates linearly along the four border lines of the grid. Note that the z values are also interpolated. This will create a rectangle unless a **NoMovePoint** has been defined on the border. Then the interpolation will be between the corners and the **NoMovePoints**.

The second choice is *Elliptic*. This starts the elliptic grid generator. Note that this will not change the z values.

The third choice is *TransfiniteI*. This is transfinite interpolation in streamwise direction. The z values will be interpolated. **IMPORTANT**: In this mode the **NoMovePoints** will also be moved.

The fourth choice is *TransfiniteJ*. This is the same as *TransfiniteI*, except it is in the cross-streamwise direction.

The fifth choice is *TransfiniteM*. This is an average of *TransfiniteI* and *TransfiniteJ*.

The sixth choice is *Average*. This is a procedure where the x,y and z values of a grid intersection are taken to be the average of the four neighboring grid intersections. This procedure is repeated for all the inner grid intersections. This will **not** move the **NoMovePoints**.

Important note:

Note that after having edited the grid, it is advisable to write the content to a **koordina** file. This is done in the **File** option of the main menu. A file named **koordina.new** will be written. The attraction parameters and the fixed point parameters are written to the **control.new** file from the same sub-menu.

3.4 Presentation graphics

There are five graphics modules for presentation of results. These can be invoked any time during the calculation or afterwards. More than one module can run simultaneously. The modules are choices under the *Graphics* option of the main menu:

- *Map*
- *Longprof*
- *Cross*

Map presents the geometry seen from above. It is possible to get velocity vector plots and plots and bar plots of concentration, diffusivity, k, etc. It is also possible to plot the grid and change between different vertical levels.

Longprof presents a longitudinal profile of the geometry. Graphs with different parameters as a function of depth along the longitudinal profile is obtained. It is also possible to view the grid or the velocity vectors. It is possible to change between different longitudinal profiles.

Cross presents a cross-section of the profile. It is only possible to see a velocity vector profile. It is possible to change between different cross-sections.

The modules use the standard GUI for OS/2, and have approximately the same system of menus. The same resource file is used for all the three programs. Not all choices are applicable to all the windows. If a choice is not applicable, it will have no effect to choose this option.

The option *Move* is used to move the plot upwards, downwards or sideways.

The option *Scale* is used to enlarge, shrink or distort the plot. The option *2 Point Enlarge* is used for enlarging a certain section of the grid. After this choice is made, the user clicks at the lower left corner and upper right corner of the part of the grid that the user wants to see. After the second click the enlargement is made.

The third option in the menu bar is *Graph*. The sub-menu displays different parameters that can be shown. The option *Sediment sample no.* is a reference to which of the **N** datasets from the **control** file is placed at which location at the grid. The sub-menu option *Bedchanges* shows the bed changes during and after sediment calculation. The sub-option *Dominant grain size* shows which bed grain size has the highest fraction in each bed element. The *ColorMap* option is be used for displaying a color map of some of the following parameters: water depth, bed shear stress, turbulence, velocity and sediment size. The colors are given by the user in the **H** data sets in the **control** file.

The sub-option *L-val concentration* is used to change between different grain sizes. When choosing 0, the sum of the different grain sizes (total concentration) is shown. The three last options are used for changing which two-dimensional slice is presented. The two first sub-sub options are *i++, i--, j++, j--, k++, k--*, depending on which direction is changed. These options chooses the next(++) or previous(--) slice.

The next option on the menu bar is *Timer*. The timer is a module which updates the graphics at certain time intervals. The numbers of the different sub-options under *Timer* indicates the interval in seconds. The options *Start* and *End* starts and stops the timer.

The last option on the menu bar is *System*. The pull-down menu has the option *Save*. This is used for storing the plot in an OS/2 metafile. It can then be printed or plotted on paper by the standard utilities that comes with OS/2. Note that the name of this file will be **mapfile.met**, **longfile.met** or **crosfil.met**, depending on which of the graphics procedures that produced the file. If there is an existing file with the same name, this will not be overwritten. The command then has no effect. It is therefore recommended to rename the file right after it is made.

The other options from *System* is *Bedchange+* and *Bedchange-*. These options only have effects in the *Map* graphics. The option *Bedchange+* activates the change in the bed levels according to the calculated bed changes. *Bedchange-* changes the bedlevels back.

3.5 Animation

The purpose of the animation module is flow visualization. The module displays the grid as seen from above, and animates the movement of a sediment particle. The movement of the particle will depend on the flow field and the fall velocity of the particle.

The animation window has five menu options. These are *Move* and *Scale* which are similar to the other presentation windows. Then there is the *Particle* option. The pull-down menu is used to set the particle size, which will be used by the program to

calculate a fall velocity. The next menu option is *Run*, which starts the particle movement.

The particle starts from one point in the grid and always from the top cell in the vertical direction. Then it moves along the geometry until it reaches one of the cells that borders the boundary. Then it returns to the starting point. The starting point is changed by clicking with the mouse at a location in the geometry.

The last option in the animation window menu is *Speed*. By choosing *Double* or *Half* the time step is doubled or divided by two. This choice can be repeated for further increase or decrease in time step. Note however that if the time step becomes large, the particle may not move along the steam lines any more, and it may even move out of the geometry during the time step. The accuracy of the particle trajectory will increase with decreasing time step.

****Temporary note**:**

There is a bug somewhere in the program related to the position of the mouse when used in the retained graphics. The retained graphics is used when moving the particle, and also in the *2PointScale* in the *Map* graphics. The error is that the particle will not move to the correct location where the user clicked with the mouse. The location defect is dependent on the screen resolution. I have written to IBM for help for this problem, and they said that they will get back to me. As I'm writing this, I havn't heard anything yet.

To temporarily get around this problem, you can give a **G 15** data set in the **control** file. This following float gives the scaling factor which can be used. On my computer the scaling factor is 1.4 for 800 x 600 screen resolution, and 1.0 (default) for 1048 x 768 screen resolution. For 800 x 600 screen resolution the data set will therefore be:

G 15 1.4

This scaling factor is also implemented in the *Map* graphics.

Also note that this problem does not emerge in the grid editor, because retained graphics is not used. Retained graphics is only necessary when something moves or is saved to a metafile.

Chapter 4. Input/result files

4.1 The file structure

A flowchart describing the various files are given below. Note that most of the files are only used for special purposes and they are normally not required. Some of the files are output files. The program can produces many of the input files. For simpler cases all the necessary input files can be generated by the program.
The two main input files are the **control** file and the **koordina** file. These must be made before the program is started. All the files are ASCII files, and can be created using a standard editor.

4.2 The boogie file

This is a file that shows a print-out of intermediate results from the calculations. It also shows parameters as average water velocity, shear stress and water depth in the initialization. Trap efficiency and sediment grain size distribution is also written here. If errors occur, an explanation is also often written to this file before the program stops. The file contains the data that is normally written to the screen in a DOS program.

The option D on the **F 1** data set will give additional print-out to the file.

Initially in the file it is written how much memory that is occupied by the arrays that are dynamically allocated. To estimate the total memory requirement, add 1 MB to this value.

A table then follows, which shows the cross-sectional area, hydraulic radius, average velocity and water level at the cross-sections that have been used for initializing the water surface. If the option D on the **F 1** data set is used, this information is written for all the cross-sections additionally. Then a table of waterlevels for all cross-sections follows.

If the MB-flow module is used, the residual norms are written. Then follows a sequence of two lines for each iteration of MB-flow. An example with four iterations is shown below:

Iter: 5, Resid: 1.69e-05 4.10e-06 2.73e-05 1.17e-04 1.38e-02 1.13e-02
Cont: 9.23e-08, DefMax: 1.65e-03, U,V,W(96,7,20): 6.40e-01 -5.14e-03 5.76e-02
Iter: 6, Resid: 1.62e-05 3.85e-06 2.62e-05 1.10e-04 1.31e-02 1.08e-02
Cont: 9.23e-08, DefMax: 1.56e-03, U,V,W(96,7,20): 6.40e-01 -5.14e-03 5.76e-02
Iter: 7, Resid: 1.57e-05 3.65e-06 2.50e-05 1.04e-04 1.25e-02 1.03e-02
Cont: 9.23e-08, DefMax: 1.48e-03, U,V,W(96,7,20): 6.40e-01 -5.14e-03 5.76e-02
Iter: 8, Resid: 1.51e-05 3.46e-06 2.38e-05 9.86e-05 1.18e-02 9.77e-03
Cont: 9.23e-08, DefMax: 1.41e-03, U,V,W(96,7,20): 6.40e-01 -5.14e-03 5.76e-02

The first line has the word "Iter" at first. Then an integer follows, which shows the number of the iteration. In the example above this runs from iteration number 5 to 9. Then the residuals for the six equations are shown. The x,y and z velocity equations are first, then the pressure equation and the k and ε equation follow. All these must be under 10^{-3} before the solution has converged.

The second line starts with the word "Cont:". Then a floating point value is shown. This is the sum of all the inflow and outflow in the geometry. This should be a very low value, typically under 10^{-7}. If a larger value is given, check the boundary conditions. Then the word "DefMax" is written. The residual for the cell with largest water continuity defect is then written. The indexes for this cell are then written, with the velocities in the three directions for this cell. In iteration 9 for the example above, the maximum water continuity defect was 1.41e-3 kg/s for cell i=96, j=7, k=20. The velocity in the x-direction for this cell was 0.64 m/s, the velocity in the y direction was -5.14 mm/s and the velocity in the vertical direction was 5.76 cm/s.

If 3D changes in the water surface elevation is calculated, the water surface elevations along the centerlines in the two horizontal directions are written to the **boogie** file. This is done for each update. An example of some of these results is shown below:

WL(20,1) = 1.012947e+00
WL(20,2) = 1.012427e+00
WL(20,3) = 1.011113e+00
WL(20,4) = 1.009520e+00
WL(20,5) = 1.007900e+00

If the sedimentation calculation is used, the fluxes through the four side walls and the trapped sediments are written for each size. This is written for each 100th iteration. An example is shown below:

Trap efficiency after 24700 iter: all values in kg/s
l=1: Trapped: 230.015, Fluxes (I1,I2,J1,J2): 229.843, 2.95575e-20, 0, 0 Resid: 0.000750
l=2: Trapped: 485.265, Fluxes (I1,I2,J1,J2): 499.657, 85.56, 0, 0 Resid: 0.142433
l=3: Trapped: -2617.23, Fluxes (I1,I2,J1,J2): 269.815, 3489.59, 0, 0 Resid: 2.233171

Three sizes are given in the example. For size 2 the inflow is 499.657 kg/s, the outflow is 85.56 kg/s and 485.256 kg is trapped. The residual is the continuity defect divided by the inflow. The convergence criteria is given on the **F 4** data set. The solution is conveged when the average residual is under the convergence criteria.

After the sedimentation calculation has finished, the sum of sub-micro time steps is written. This is called the micro time step. A further explanation of the time steps is given in [4], on page 39.

If the bed changes are calculated this is also written to the file. The bed changes are given in meters. An example is given below:

BedMove(10,8) = 1.009495e-03 meters
BedMove(10,9) = 1.030140e-03 meters
BedMove(10,10) = 1.019934e-03 meters

In the example, the deposition in cell i=10, j=10 causes the bed to rise 1.02 mm during the given micro time step.

4.3 The control file

The **control** file gives most of the parameters the model needs except the grid. The main parameters are the grid size, that is how many grid lines there are in the three directions. The number of sediment sizes is also an important parameter. To generate the water surface it is necessary to know a downstream water level, together with the water discharge and the Manning's friction factor. These parameters are given on the **G 1** and the **W 2** data set in the **control** file. If the **control** file does not exist, the user is prompted for these parameters in a dialog box. The user can then later choose *Write control* in the *File* option of the main menu, and get a **control** file written to the disk (as control.new). This can then be edited according to the user's needs. Note that only the most used parameters are written to the **control.new** file.

During the water flow calculations there are several parameters that can be varied. These parameters affect the accuracy and the convergence of the solution. These parameters can be modified while the water flow field is being calculated. A dialog box with the parameters is invoked by choosing *Waterflow parameters* from the *Input Editor* choice in the main menu.

This file contains all other data that are necessary for the program. SSIIM reads each character of the file one by one, and stops if a capital T,F,G,I,S,N,B or W is encountered. Then a data set is read, depending on the letter. A data set is here defined as one or more numbers or letters that the program uses. This can for example be the water discharge, or the Manning's friction coefficient. It is possible to use lower-case letters between the data sets, and it is possible to have more than one data set on each line. Not all data sets are required, but some are. Default values are given when a non-required data set is missing. SSIIM controls the data sets in the **control** file to a certain degree, and if an error is found, a message is written to the **boogie** file and the program is terminated.

For the initialization and graphics, the following data sets are used:

Required: **G 1, G 3, W 1, W 2**
Optional: **T, F 1, F 2, F 7, G 8, P 2-4**

For the sediment calculation, the following data sets are used:

Required: **S, N, B, I**
Optional: **F 4, F 6, F 8-12, F 23**

For the water flow calculation by using MB-Flow, the following data sets are used:

Required: None
Optional: **F 15-17, F 22, F 24-25, G 6-8, G 11, W 3-4, K 1-10**

In the following the data sets for the **control** file is described:

T Title field. The following 30 characters are used in the graphics programs.

F 1 Debugging possibility. If the character that follows is a D, one will get a more extensive print-out to the **boogie** file. If the character is a C, the coefficients in the discretized equations will be printed to the **boogie** file.

F 2 Automatic execution possibility. Some parts of the program will be executed directly after the initialization if a character is placed in this field. The sub-programs will be executed in the order they are given. The possibilities are:

R	Read the **result** file
I	Initialize the sediment calculations
S	Calculate sediment concentration
W	Start the multi-block water flow routine
B	Change the bed according to sediment calculation
M	Write result file
V	Initialize water surface level

F 4 Relaxation factor for second order interpolation of bed concentration, maximum iterations for concentration calculations and convergence criteria for suspended sediment calculation. The second order interpolation method is described in Chapter 2.2. The convergence criteria is given as allowable flux deficit as part of inflowing sediments.

Note that this data set has changed from version 1.1

Defaults: relaxation: 0.5, iterations: 500, convergence criteria: 0.01.

F 6 Coefficients for formula for bed concentration. Default is van Rijn's coefficients: 0.015, 1.5 and 0.3. If one uses this option, the sediment transport formula given in dataset F 10 must be R, which is van Rijn's formula is used as basis.

F 7 Run options. read 10 characters. If the following capital letters are included this will mean:

D: Double the number of grid cells in streamwise direction in comparison to what is given in the **koordina** file. Each cell is divided in two equal parts.

J: Double the number of grid cells in the cross-streamwise direction.

I: Inflowing velocities in the y-direction are set to zero.

A: Diffusion for sediment calculations in non-vertical direction is set to zero.
bed cell will not be lower than the cell above.

B: Correction for sloping bed is used when calculating bed sediment concentration.

G: Cell walls at outblocked area is not changed when there are changes in the cells outside the block.

V: 90 degree turning of the plot seen from above (map).

Z: Vertical distribution of inflowing sediment is uniform.

X: Grid is read in from the "XCYC" file. This is only used in the post-processor, and with presentation of results from Spider where the lines in the k-direction are not vertical.

C: Inflowing and ouflowing water in default walls is set to zero.
This means that the water flow must be specified on the G 7 data sets.

F 8 Maximum bed level change relative to water depth. This is controlled for all the cells. Default: 0.1. This parameter is used to compute the time step for the bed changes.

F 9 Factor that is used to change the turbulent viscosity of the inflowing water.
The factor is proportional to the turbulent viscosity. Default: 1.0.

F 10 Which sediment transport formula is used to calculate the concentration at the bed. The following options are given:

R van Rijn's formula
E Engelund/Hanssen's formula
A Ackers/White's formula
Y Yang's streampower formula
I Einstein's bedload formula
M Mayer-Peter/Mueller's formula
S Shen/Hung's formula

Default: R.

Note that only the option R is fully tested.

F 11 Density of sediments and Shield's coefficient. Default: 2.65 and 0.047.

F 12 Schmidt's coefficient, which is a correction factor for deviation between vertical and isotropic turbulent diffusivity. Default: 1.0

F 15 An integer that determines how the law walls will be used in the cells which borders both the wall and the bed. A value of 0 will make the program use wall laws on both walls. A value of 1 will make the program only use wall laws on the bed wall. For cases where the vertical dimensions are approximately the same as the horizontal dimensions, a value of 0 is recommended. Default: 1.

F 16 Roughness coefficient which is used on the side walls and the bed. If not set, the coefficient is calculated from the Manning's friction coefficient. The file **bedrough** overrides this value for the bed cells.

F 17 Time step in seconds. When this is above 10^{-8} a transient term is included. This is however not tested, and it is not certain that the procedure will give correct results.

F 20 Repeated calculation option. An integer is read, and the calculation sequence on the **F 2** data set will be repeated this many times. Note that the graphical view of the bedlevel changes will only appear on the last iteration when sediment calculations are done. Also note that if a result file is read in the **F 2** data set, it is only read during the first iteration.

F 21 Relaxation coefficient for the Rhie and Chow interpolation. Normally a value between 0.0 and 1.0 is used. When 0.0 is used the Rhie and Chow interpolation will have no effect. When 1.0 is used the Rhie and Chow interpolation will be used normally. Default 1.0.

F 22 Minimum porosity and relaxation factor for porosity calculations. Two floats. Default 0.2 and 2.0.

F 23 Accelerated deposition routine. Two floats are read. The routine fills a reservoir to a certain percentage of the total volume. This volume fraction is given on the first float. The filling is based on only one water flow calculation. Therefore the sediments may fill over the water surface in some

locations. The routine then moves away some sediments so that a certain water depth is kept. This depth is given by the second parameter, in meters. The surplus sediment is distributed to neighboring cells according to the one water flow calculation. The redistribution is iterated so that there are no filling above the minimum water depth.

Note that the bed changes are calculated in the center of the cells, and that changes in the grid therefore are interpolated from four surrounding cells. This means that even if the four surrounding elements are filled to the given criteria, the grid line may not be exactly on this level. This may cause bedlevels to rise above the waterlevel. The used should examine the grid after the bed changes to observe whether this has occurred. Choosing a higher value of the minimum water depth will decrease the chance for such a phenomena to occur.

Note that this routine has not been tested yet.

F 24 Turbulence model. An integer is read, which corresponds to the following models:

0 : standard k-ε model (default)
1 : standard RNG model, see reference [7]
2 : constant eddy-viscosity model
3 : LES RNG model, see reference [7]

Note that only option 0 has been implemented.

F 25 Porosity parameters. Four floats and one integer. The two first floats are identical to the ones on the **F 22** data set. The following two floats give the porosity on the second and third level above the ground. These have default values 0.5 and 0.8. These are used if the roughness height is larger than the levels of the porosity in the **porosity** file. The effective porosity height is set to maximum of bed cell height and roughness height.
The last integer determines the procedure for finding particle diameter in the porosity formula. The following options are given: (default 0)

0 : Maximum of roughness height and porosity height
1 : Maximum of roughness height and 0.33 * porosity height
2 : Equal to height of bed cell
3 : Maximum of height of bed cell and porosity height

F 26 Volume fraction of sediments in deposits. One float is read. Default 0.3. If the water content is 51 % in a fully saturated sample, the volume fraction will be 0.49.

F 30 Bed concentration recalculation methods. Six integers are read. Reference is made to chapter 2.2 where the methods are described. The first integer determines which initial bed sediment grain size distribution is present. The following possibilities are present:

0 : Given by the user, equivalent of the **A 1** method
1 : Shield's graph method, equivalent of the **A 2** method
2 : Zero for all sizes, equivalent of the **A 3** method

The second integer invokes the **A 4** method if it is 1. The bed grain size distribution is then recalculated based on deposition of the different sizes.

The third integer invokes the **B 1** method if it is 1 and the **B 2** method if it is 2. This is extrapolation of first or second order.

The fourth integer invokes the **C 1** routine if it is 1. This is a routine for prevention of erosion.

The fifth integer invokes the **C 2** routine if it is 1. This keeps the concentration under the level of the inflowing concentration.

The sixth integer invokes the **C 3** routine if it is 1. This is a combination of the **C 2**, **B 2** and **A 3** method.

Default: F 30 0 0 0 0 0 0 . Note that six integers must be present, even if only one or two are different from zero.

F 31 Two porosity coefficients are read. The coefficients are used for making the **porosity** file. This is further described in Chapter 2.3. Default: F 31 0.8 0.8

G 1 xnumber, ynumber, znumber and lnumber. There are four integers that show the number of grid lines in the streamwise, cross-streamwise and vertical

direction. lnumber is the number of sediment sizes. This data set must be present in the **control** file. The program will read these values and allocate space for the arrays accordingly. If the space (RAM) on the computer is not sufficient, the program will crash.

G 3 Vertical distribution of grid cells. This is further explained on Fig. 2, where an example is given. This dataset must be present in the file.

G 6 Data set for calculating water surface location with an adaptive grid. Three integers and two floats:

iSurf:
jSurf:
kSurf:

These are three integers that indicate three grid lines. This point is a reference point, and it is not moved. In the present implementation, Ksurf have to be equal to znumber + 1. If not, a warning message is sent to the boogie file, and Ksurf is set to znumber + 1. The computations continue afterwards.

RelaxSurface:

This is a float that relaxes the estimation of the increment to the new recalculated water surface. Recommended values are between 0.5 and 0.95.

ConvSurface:

This float sets the limit for when the water surface should be recalculated. The water surface will be updated when the maximum residual of the equations are below this parameter. Recommended value: 0.01 - 1.0

G 7 This data set specifies water inflow on geometry sides, bed or top. Each surface is given on **one** G 7 dataset. It is possible to have up to 19 G 7 datasets.

On each dataset, seven integers and four floats are read. The names of these variables are:

G 7 type, side, a1, a2, b1, b2, parallel, update, discharge, Xdir, Ydir,

Zdir

Each variable is explained in the following:

- type: 1: outflow, 0: inflow.
- side: 1: plane i=1, -1: plane i=xnumber,
(cross-streamwise plane)
2: plane j=1, -2: plane j=ynumber,
(streamwise plane)
3: plane k=1, -3: plane k=znumber
(horizontal plane)
- a1,a2,b1,b2: four integers that determine the limits of the surface. An example is shown in Fig. 3.
- parallel: direction of the flow:
0: normal to surface
1: parallel to grid lines normal to surface
2: direction is specified (vector directions)

- update: 0 for not update, 1 for update.
(Not implemented by Jan-93)
- discharge: discharge in qm/s. Note that the sign of the discharge must correspond with the direction of the desired flow velocity. Positive discharges indicate discharges in positive directions.
- Xdir: direction vector in x-direction
- Ydir: direction vector in y-direction
- Zdir: direction vector in z-direction

Note that the update option is not implemented by March-93. Still, there must be an integer present for this data in the data set.

Example: G 7 0 1 2 11 2 11 0 0 32.0 1.0 0.0 0.0

This example specifies inflow in the most upstream cross-section. The inflow area is from cell no. 2 to cell no. 11 in both cross-streamwise and vertical direction. The flow direction is normal to the cross-section. The discharge is 32 cubic meters/second.

The parameter "side" can be used to specify flux on sections that have been "amputated" by the multi-block procedure. The parameter side is then evaluated as the number of the block plus 10. Example: A geometry with one multi-block that starts at node i=30. To specify flux on wall i=29, use the G7 data set with the parameter "side" set to 11 (10+1).

Remember to define the walls of the boundary that when this dataset is used. This must be done on the W 4 data set.

G 8 Values for initial velocities. Up to 19 data G 8 data sets can be used. Six integers are read first to specify the volume that is being set. Then three floats are read, which gives the velocities in the three directions.

G 8 i1 i2 j1 j2 k1 k2 U V W

G 11 Source terms for the velocity equations. Six integers and two floats.

i1,i2,j1,j2,k1,k2, source, relax

The first six integers give the cells that are influenced by the source term. The source variable is the form factor times a diameter of a cylinder in the cell. The relaxation variable is recommended set between 1.0 and 2.0

G 12 Sediment source for multi-block border. This is used where there is inflow of sediments in a branch of a block that is cut of. An integer is first read, which tells the number of the block. Then lnumber floats are read, which is inflow of sediments for each size. This is given in kg/s. The option is not fully tested yet.

G 13 Outblocking option that is used when a region of the geometry is blocked out by a solid object. An integer is read first, which determines which sides the wall laws will be applied on. The following options are possible:

0: No wall laws are specified
1: Wall laws are used on the sides of the block
2: Wall laws are used on the sides and the top of the block
3: Wall laws are used on the sides, the top and the bottom of the block

Six integers are then read, i1,i2,j1,j2,k1,k2. These integers define the block.

Up to 19 **G 13** data sets can be used.

H 1 This data set is used for plotting a color graphics map where the plotted parameter is the absolute value of the velocity. A preference curve is given on this data set. The curve has the velocity on the horizontal axis and a preference index on the vertical axis. Three colors are used: red, blue and yellow. The program checks the curve and chooses the index

corresponding to the velocity in the element. If the index is above 0.2, a blue color is used. If the index is below -0.2 a red color is used. If the index is between -0.2 and 0.2 a yellow color is used. These colors are used in the fish habitat calculation. The index is then a preference, and a value above 0.2 is good; below -0.2 and 0.2 is indifferent; and below -0.2 is bad.

An integer is first read, which tells how many points is on the curve. Then two floating points are read for each point, with velocity first. Example:

H 1 6 0.0 0.0 0.049 0.0 0.05 1.0 0.15 1.0 0.16 -1.0 10.0 -1.0

This is the habitat curve for 10 cm salmon. A velocity under 0.05 is indifferent (yellow), a velocity between 0.05 and 0.15 is good (blue) and a velocity above 0.15 is bad (red).

Note that the velocities in the cells closes to the bed are used.

H 2 Color map plotting for water depth instead of velocity. Otherwise it is the same as the **H 1** data set.

H 3 Color map plotting for sediment size instead of velocity. Otherwise it is the same as the **H 1** data set.

H 4 Color map plotting for bed shear stress instead of velocity. Otherwise it is the same as the **H 1** data set.

H 5 Color map plotting for turbulet kinetic energy at the cell closest to the water surface. Othewise it is the same as the **H 1** data set.

P 2 Five floating points that give scaling for the graphical presentation. The first three gives scales in streamwise, cross-streamwise and vertical direction. The fourth and fifth give movements in left-right and vertical direction. Defaults: 1.0 for the scales, and 0.0 for the movements.

P 3 Four integers that give initial location of the graphical plots in streamwise, cross-streamwise and vertical direction, and sediment fraction number.

P 4 A character that indicates initial type of plot. "g" means grid, "v" means velocity vectors, "c" means concentration.

W 1 Manning's number, discharge and downstream waterlevel. This dataset must be present in the file. The parameters given here are used to generate the waterlevel for the calculations. A routine like HEC-2 is used.

W 2 Water surface initialization array of integers. The first integer tells how many numbers there are in the array. The next numbers tell which cross-sections are going to be used in the initialization of the water surface of the grid. The integers must be given in rising order, and start with 1. This dataset must be present in the file.

W 3 Specification of multiple blocks for the multi-block water flow module. First an integer that indicates the number of extra blocks is given. The maximum value is 9. Then two integers for each extra block are given. The first integer (MBSkjot) tells where the block is cut off. The second integer (MBStart) tells where the block is added. If the second integer is negative, the block is added on the left side of the main block. Otherwise it is added on the right side of the block. This is explained with an example on Fig. 4. The corresponding dataset would be: W 3 1 10 -5

W 4 Specification of extra walls for the multi-block water flow module. Seven integers have to be given for each wall. There can be up to 29 walls, and each wall is described on **one** W 4 data set.

The variable names are:

W 4 dir,posneg,node,a1,a2,b1,b2

The first integer, dir, indicates the plane. 1 is the j-k plane (cross-section), 2 is the i-k plane (longitudinal section) and 3 is the i-j plane (seen from above).

The second integer, posneg, indicates if the wall is in the positive or negative direction of the node. The coordinates are given for nodes. 1 or -1 is given. If 0 is given, a previously set wall is deleted.

The third integer is the number of the node plane.

An example is given in Fig. 5 below. The figure shows the i-j plane. The wall is to be given on node i=4. If the second integer, posneg, is 1, then wall laws are applied on the wall upstream of node 4, in the negative i-direction. If posneg = -1, then the wall laws are applied on node 4

if the cell in the downstream i-direction (line i=3) is a wall.

The four following integers are indexes a1,a2,b1,b2, which gives the two-dimensional coordinates for the corner points of the part of the plane that is described. The four integers are further explained on Fig. 3.

Note that this option has not been tested for internal walls yet.

W 5 Different Manning's values than the default value for cross-sections. An integer is first read, which tells how many cross-sections are read. Then an integer and a float is read for each cross-section. The integer tells which cross-section is changed, and the float tells the Manning's value. Several **W 5** data sets can be used.

W 6 NoMovePoint - a point which is used in the **Grid Editor**. Two integers are read, which is the numbers of the i and j grid lines. The intersection of these lines are not moved by the elliptic grid generator. One **W 6** data set is required for each NoMovePoint. Maximum 199 points can be used.

The **W 6** data set is usually generated by the **Grid Editor**.

W 7 Attraction point used in the **Grid Editor**. Each **W 7** data set represents attraction to one grid line or point. Maximum 199 attraction points can be used. An integer is read first which tells the type of attraction. The following options are given:

0: Point attraction in i-direction
1: Point attraction in j-direction
2: Line attraction in i-direction
3: Line attraction in j-direction

Then two integers are read, which tells which grid line intersection the attraction is towards. For the line attraction, only one of the integers are used. Then the two attraction parameters are read, which are floats.

The **W 7** data set is normally generated by the **Grid Editor**.

S Integer that gives the number of the sediment fraction, float that gives the size of the sediments for these fraction, float that gives the fall velocity for the fraction.

There must be lnumber datasets like this in the file, as long as

sediment concentration is to be calculated. All numbers are given in SI units, that is, the grain size is given in meters and the fall velocity is given in meters/second. This data set must be present when sediment transport is calculated.

Note that this dataset defines the grain sizes that are used in the program. This goes for both the grain sizes in the bed and in the suspended sediment calculation.

The grain sizes should be numbered from 1 to lnumber, where size 1 is the coarsest size, and the following sizes have increasingly smaller diameter.

N integer sets

These data sets define different grain size distributions for the sediments that are in the bed when the calculation starts.

Maximum 10 different samples can be used.

Each sample has its own number. This starts with zero, and increases sequentially to the total number of samples - 1.

First, an integer for the number of the sample is given. The second shows which size is described. The third number is a float, which gives the fraction of the size in the sample. It is required to have lnumber N data for each sample. These data sets must be present when sediment transport is calculated.

The number of N datasets that is required is:

(number of grain size distributions) x (number of sediment sizes)

B placed in

This data set gives where the samples (from the N - data sets) are the geometry before the calculation starts.

The first integer indicates the number of the sample. The four following integers give the number of the corner cells on a rectangle of the bed. The sample is placed on the bed from i=second integer to i= third integer and j=fourth integer to j=fifth integer. A B data set overwrite previous B datasets.

This data set must be present when sediment transport is calculated.

The dataset B 0 0 0 0 0 tells that sample no. 0 covers the whole bed.

I Inflowing sediments. First integer shows which sediment fraction is simulated. Second number is a float that is the amount of inflowing sediment of this size in kg/s. lnumber of these data sets must be present when sediment concentrations are calculated.

Note that the I dataset follows the sizes on the S data set, but it has nothing to do with the N and B data sets.

K 1 Number of iterations for flow procedure and number that determines the minimum iterations between updates of water surface. Two integers. Default: 10 9000

K 2 Two integers that indicate of laws of the wall or symmetry conditions are being used. The first number applies to the side walls. The second number applies for the surface. 0 is used for wall laws, and 1 for free surface/symmetry. Wall laws are always used for the bed, if not changed by the W 4 data set. Default: 0 1.

K 3 Relaxation factors. Six floats. For the three velocity equations, the pressure correction equation and the k and ε equation. For further description of the relaxation factors, see a following chapter. Default: 0.8 0.8 0.8 0.2 0.5 0.5

K 4 Number of iteration for each equation. Six integers. Default: 1 1 1 5 1 1

K 5 Block-correction index for each equation. Six integers. If 0, no block-correction. If 1, the block-correction is used. Default: 0 0 0 0 0 0
Note that the block-correction will not work with the SOU scheme.

K 6 Six integers are read which determines whether the SOU or POW scheme is used. If 0 POW is used, if 1, SOU is used. Note that presently the multi-block flow module may not converge completely when the SOU scheme is used with multiple blocks. This may be due to a bug. SOU will work if there is only one block.

Note that the multi-block flow module will always use the POW scheme

for the pressure-correction equation. Default: 0 0 0 0 0 0

K 9 A character deciding whether SIMPLE or SIMPLEC is used. Y= SIMPLE,
N = SIMPLEC. Default: Y

K 10 A character that decides which solver is used. Y = Gauss-Seidel solver, N = TDMA solver. Default: Y

Remember that the order of the data sets may be important. For example the G 1 data set should be early in the file. If the order of the data sets follows the description given here,
this should not be a problem.

4.4 The koordina file

This is the input file where the bed of the geometry is described. An example is show in the figure below. The grid can be made using a map, a spreadsheet or the **GridEditor**.

The necessary input data is the x,y and z coordinates of the points where the grid lines meet. The format of the data is given below.

i j x y z

An example:

1 1 0.34 0.54 0.11
1 2 0.35 0.66 0.12
...

The first two numbers are integers, while the following three are floats. The numbers are read in a free format, which means that the distance between them does not matter. The sequence of the points are not important, as long as all points are included. This is not controlled by the model, so the user must do this by looking at the grid in the graphic modules of the program.

Some words about indexing and numbering of grid lines and cells. The variable names for the number of grid lines in the three directions are:

xnumber : streamwise direction
ynumber : cross-streamwise direction
znumber : vertical direction

The numbering of the grid lines goes from 1 to xnumber in the streamwise direction, and similarly for the other two directions.

However, the grid lines define cells between the grid lines. The variables are calculated in the center of each cell. This means that a numbering for cells also is

required. The word node is often used for the center of a cell.

From a geometrical view of the grid, it is observed that the number of lines always exceeds the number of cells by one in each direction. When the arrays are defined, it is therefore a choice for the programmer to start the numbering of the cells on one or two. The choice that is made in this program is that the numbering starts on two. This means that the cell that is defined by grid lines i=1 and i=2 and j=1 and j=2 has the number (2,2). Cell number (1,1) does not exist. The numbering of the cells is also shown in Fig. 1. The numbering of the grid lines is shown with the <,> sign, while the numbering of the calculation nodes is shown with the (,) sign. The grid is non-staggered.

4.5 The geodata file

This file contains a number of x,y and z coordinates. An example is shown below:

E 2.2 3.3 3.4
E 4.5 3.3 2.2
E 3.3 4.2 1.2

The letter **E** is used for counting the number of points.

The purpose of this file is to use geometrical data that has been obtained from the field, a digitized map or a GIS system. These data would normally not fit into a grid like the **koordina** file.

The file is used in three modules in the program. The first module is the *GridEditor*, which can display the points in the file with the grid. This makes it easier to generate the grid. The *Utility* option of the menu and the *Make map* option in the pull-down menu activates this. The grid points are displayed with different colors according to which level they are.

The second use for this file is also in the *GridEditor*. It is then used for generating the z values for the grid. A linear interpolation procedure is used. The procedure is further described in Chapter 3.3.

The third use is to make the **Porosity** file. The module that does this is activated from the main menu of the user interface, from *File* and *Make porosity file* in the pull-down menu. The procedure to make this file is further described in Chapter 4.7 and Chapter 2.3.

4.6 The bedrough file

This file is used to give a roughness height to individual bed cells. Values in this file overrides the value calculated from Manning's coefficient, and the value given on the **F 16** data set. On each line a character, two integer and a float are given. The first character is a B, and the two following integers are indexes for the bed cell. The float

is the roughness in meters. An example is given below:

B 19 2 0.001
B 19 3 0.001
B 19 4 0.001

4.7 The porosity file

This file is used when the bed of the river is covered by stones, and a porosity term is used in some of the cells. This file describes the location and magnitude of the porosity in the geometry. An example is given below:

P 17 6 3.349774 3.399189 3.450101 3.499517 0.000000 0.700000 0.833333 1.000000
P 17 7 3.358273 3.413603 3.470610 3.525940 0.000000 0.653846 0.807692 1.000000
P 17 8 3.403323 3.426084 3.449536 3.472297 0.000000 0.642857 0.785714 1.000000

First, the character P is read. Then two indexes for the i and j number of the bed cell is read. Then four vertical levels are read, which have the same zero reference as the **koordina** file. The porosities in each of these levels are then read.

The porosity file can be made from a **koordina** file and a **geodata** file. The module that does this is activated from the main menu of the user interface, from *File* and *Make porosity file* in the pull-down menu. The procedure goes through each element and used the points in the **geodata** file to obtain the data. The procedure is described in more detail in Chapter 2.3.

4.8 The innflow file

This file is used to read velocities in three directions for the upstream boundary condition. The program searches for this file, and uses the data on the file if it exists. If the file does not exist, a warning message is written to the **boogie** file, and the program proceeds normally.

On each line the velocities in a cell of the upstream cross-section are given. First, the character E is written. Then the indexes j and k (horizontal and vertical) are given. Then the velocity components in the x,y and z directions are given. An example is given below:

E 2 2 0.299115 0.023009 0
E 2 3 1.79469 0.138055 0
E 2 4 1.9941 0.153394 0
E 2 5 2.19351 0.168733 0

The file does not have to contain values for all the nodes. The normal initialization

procedures are applied first, and then the **innflow** file is read. The nodes that are not present in the **innflow** file will keep the values from before the file was read.

4.9 The loggfil file

This is a file that is used to log bed changes between each time the bed is changed. The bed changes are written with the "append" mode in C, so that for the file to be updated, it must exist before SSII writes to it. What is written to the file will be appended to what is present in the file.

What is written to the file is a time step for each change, together with the bed changes for each cell in meters. The bed sediment grain size distribution for each time step is also written to this file.

When SSII starts, it looks for a file called **loggfil.pre**, and if it exist, it uses this file to change the bed and give initial bed sediment grain size distribution. If one decides to start from a previous time step, one must have copied **loggfil** to **loggfil.pre** before starting SSII.

loggfil is written from the *Bedchange+* option in the *Map* graphics module.

The **loggfil** file is never overwritten by the program. New bed changes are appended to the end of the file. The user must therefore take care so that only the relevant data exist in the **loggfil** and **loggfil.pre** files.

Note that if the program does not find the **loggfil.pre** file, a warning is written to the **boogie** file, and the program proceeds normally.

4.10 The result file

This file contains the results from the water flow calculations. The file is written when the prescribed number of iterations have been calculated or when the solution has converged. The results are velocities in three dimensions, k, ε, pressure, and the fluxes on all the walls of the cells. The data from this file is used as input for the sediment flow calculations. This file can also be read when the user wants to start the water flow calculations from where the **result** file was last written (hot start).

An example of a **result** file:

Results from SSIIM - flow, iter = 12910
Residuals: 0.000732 0.000588 0.000002 0.000003 0.001000 0.000000
Roughness : 0.050000
C 54 9 11
i j k u v w k e f1 f2 f3 p
D 1 1 1 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
0.00000000e+00

D 1 1 2 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
D 1 1 3 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00 0.00000000e+00
.....

D 2 4 7 6.31143968e-01 2.02894592e-01 -6.80367016e-05 6.47305125e-03 3.85539263e-04 3.47858729e+03 -5.19507052e+02 1.39776552e+03 1.91355310e+02
D 2 4 8 6.38189711e-01 2.05198514e-01 -2.09522025e-04 6.40727451e-03 3.13804122e-04 3.54203761e+03 -5.25291012e+02 1.09548199e+03 1.91350056e+02
D 2 4 9 6.42687814e-01 2.06670571e-01 -3.70305526e-04 6.37079494e-03 2.71714390e-04 3.58421958e+03 -5.28978103e+02 7.51732364e+02 1.91343817e+02

The first lines gives the residuals, the roughness and the grid size. Then each line gives the nine values for one cell. The letter **D** starts the line, then the three indexes for the cell. Then the three velocities, the k and ε values. Then the fluxes in the three directions and finally the pressure. All the parameters are placed on one line in the file, although this is not shown above because the line was too long.

4.11 The conres file

This file is written after the sediment concentration calculation has finished. Each line in the file contains first three indexes indicating for the node number. Then the total concentration is written and then **lnumber** floats that give the concentration for the sizes. An example is given below:

1 1 21 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00 0.000000e+00
1 2 2 5.075079e-04 0.000000e+00 1.026459e-04 1.030081e-04 1.032351e-04
1 2 3 4.064470e-04 0.000000e+00 1.011788e-04 1.015358e-04 1.017596e-04
1 2 4 4.004061e-04 0.000000e+00 9.967497e-05 1.000267e-04 1.002471e-04

If a file **conres.pre** exist, this will be read by the program before the sediment calculation starts. The **conres.pre** file has the same format as the **conres** file.

4.12 The interpol file

Vertical profiles of velocity or concentration are sometimes needed. Coordinates for the locations where the profiles are wanted are given in this file. When the **write results** routine is activated, it will search for this file. If this file is not found, it will proceed normally and write the **result** file. If the **interpol** file is found, the program will **not** write to the **result** file, but write the interpolated vertical velocities to a file

named **interres**.
An example for an **interpol** file is given below:

M 2.03 0.5
M 4.06 0.39
M 4.06 0.5

The character M is read first, and then the x and y coordinates for the point one wishes to interpolate the vertical profile to. If the concentration has been calculated recently (concentration in cell [lnumber][2][2][0] is above 10^{-8}), then the concentrations are interpolated. Otherwise, the velocities are interpolated. The results are written to a file called **interres**.

Chapter 5. Advice for use of SSIIM

5.1 Advise for new users

If you have used previous versions of ssiim, you will find that version 1.2 can be used in the same manner as version 1.0 and 1.1. The difference is that whereas you previously pushed some buttons in the dialog box, you now have to make the same choices in the main menu.

Generally, it is advisable to start with reading this manual. It is important to understand that the model is made up of several sub-models. This is reflected in the different choices in the main menu which is used to start the sub-models. Some models can be run simultaneously.

The hardware requirements for running the program are mainly focused on having enough RAM. In the beginning of the **boogie** file it is printed how much RAM the program allocates for the arrays. This can be added to the RAM requirement for the program itself, about 600 k, plus what the operating system requires. An estimate for the amount of RAM is thereby obtained. OS/2 will use the harddisk as extra memory if there are not sufficient RAM. The penalty is that the program runs very much slower. This situation can be detected by observing if the system swaps to the harddisk while running only the SSIIM program. The water flow module may take from some hours to some days to converge for a typical case, when there is enough RAM. However, under OS/2 it is not any problem to let the program run in the background while doing other tasks on the computer.

An advice for the first-time user is to run the tutorial described in Chapter 5.2 and one of the example cases. Try to modify some of the parameters and run it again. Often the user wants to simulate a particular case. It is then advisable to try to find a similar example case and modify this step by step.

Some words about crashing the program and input control. There are various controls for input and checking of intermediate results. If any of these controls finds something wrong, an error message is written to the **boogie** file, and the program terminates. Therefore, if the program suddenly stops, and the main user interface disappears, check the **boogie** file for possible error messages.

Two files are required to run the program. These are called **koordina** and **control**. The **koordina** file contain the grid. The **control** file contain the rest of the necessary parameters. Version 1.2 has new options for making the input data for the **control** file, and there is also a grid editor that can be used to make the **koordina** file. The editors are described in separate chapters.

Finally, the results from the program ought to interpreted according to:

- Possibilities of bugs in the program making errors
- Previous cases where the results have been compared with measurements

- Numerical errors, like false diffusion, grid independence, etc.
- Accuracy of boundary conditions

Knowledge and experience in computational fluid dynamics and hydraulic engineering are essential for the assessment of the validity and accuracy of the results. There is a separate chapter in this manual for assistance in interpreting the results.

5.2 Tutorial

This tutorial is meant for first time users of the SSIIM program. The tutorial shows the main features of the user interface, with the presentation graphics, animation and grid editing. The user is not required to edit files. It is required that the user know what a grid is, and that the **control** and **koordina** input files are required for the program to run. During the tutorial the files will be made interactively. The tutorial does not show the more advanced features of the program, which necessitates editing of the input files.

The tutorial is divided in five stages. During the first stage the two input files are made. During the second stage the presentation graphics is shown. The third stage familiarizes the user with the grid editor. The animation graphics is shown in the fourth stage.

First stage

In this stage the two user input files **control** and **koordina** are made.

Start up ssiim from a directory where the files **control** and **koordina** do not exist. This is done by writing *c:\path\ssiim12 <CR>* when you are on this directory. The "path" is here the path were the executable program for SSIIM is placed.

After this a dialog warning box shows up on the screen, telling you that the **control** file is not found, and that you have to fill in the parameters in the next dialog box. Choose *OK* in the warning box, and you see the main parameters dialog box. Default values are present in the edit fields. Click on the edit field for lines in streamwise direction. Change this value from 4 to 10 or a higher value. Also change the number of lines in cross-streamwise direction from 4 to for example 6. After this, push the OK button with the mouse. The **control** file is then automatically made, and written to the disk.

Immediately after this you will get a second warning box, which tells you that the files **koordina** or **geodata** are not found. The file **geodata** can be used initially instead of the **koordina** file, but that is a more advanced topic which you can read more about after this tutorial. Push the OK button on the warning box, and a new dialog box emerges. In this box you give the dimensions of the grid. The default grid that is made initially is rectangular. You can choose a grid which is 11 meters long and 6 meters wide. Change value in the editfields from 10.0 to 11.0 for the length and from 5.0 to 6.0 for the width.
Then push the OK button. After this, the **koordina** file is made and written to the

harddisk. Immediately afterwards the normal user interface for the program is shown and the program is started.

Second stage

In this stage we will solve the flow field for the given grid, and look at the results.

The main user interface is a dialog box and a menu bar. The dialog box will show intermediate results from the calculations and other messages. The menu is used for starting different sub-modules of the SSIIM program.

In the first dialog box to make the **control** file, we gave a Manning's coefficient, water discharge, downstream waterlevel and dimension of the geometry. This is all we need to calculate the flow field with the Navier-Stokes equations. To start the solution of the Navier-Stokes equations you select the option *Computations* from the main menu. Then select *MB-Flow* from the pull-down menu. After this, push the *update* button on the dialog box to see how the residuals develop. The dialog box shows the residual for all the six partial differential equations that is solved for the water flow calculation. The water flow calculation is converged when all the residuals are under 10^{-3}.After 10 iterations the water flow calculation stops. The default number of iterations is 10. To change this, you choose *Edit Input* on the main menu bar, and *Waterflow parameters* from the pulldown menu. This gives you a large dialog box. You change the editfield with number of iterations (on the lower left side) from 10 to 1000. Then push OK. Then restart the water flow calculation and observe the change in the residual until it converges.

After convergence, we want to see the results. Choose the *Graphics* option of the main menu, and the *Map* option in the pull-down menu. This gives you a graphics view of the grid as seen from above. Choose *Graph* from the menu in the **map** window, and *Velocity* from the pull-down menu. Then you see the velocity vectors.

The velocity vectors are not too exciting for this channel and to make a more complex flow pattern, you need to change the grid. This is done in the third stage.

Third stage

In this stage we will concentrate about the grid editor. But first, we want to show several graphs on the same time on the screen. Therefore, use the mouse and push the main menu down in the lower left quadrant of your screen. Scale the **map** window so that it fits in your upper left screen. To fit the grid in the window, choose the *Scale* option in the **map** window and *Shrink* in the pull-down menu together with the *Move* option in the **map** window and *left, down* options in the pull-down menu.

Then start the grid editor by choosing *Input Edit* from the main menu bar and *GridEdit* from the pull-down menu. Scale and move the window and the grid so that it fits in your upper right corner of the screen.

To edit the grid, you first choose some points which will not be affected by the interpolation routines. This is done in a special modus of the editor. This modus is invoked by choosing *Define* from the menu and *Set NoMovePoints* from the pull-

down menu. To verify that this modus is chosen, the letters "Point mode, 0" is shown on the lower part of the editwindow. The integer shows how many points you have chosen. We want to choose four points, all on the upper boundary of the grid. A point is chosen by clicking with the mouse on a grid intersection. If successful, a blue star emerges at the grid intersection. For a further explanation on which points to choose, see Fig. 7 A. After choosing the points we return to normal modus. This is done by choosing *Define* and *Set NoMovePoint* again. We observe that this modus disappears because the text "Point mode, 4" disappears.

Now we want to move the points. You can move any of the grid intersections by clicking on the intersection with the mouse and dragging it to another place and then release the mouse button. Try this with one of the intersections between the marked points. In the following the four marked points are denoted 1,2,3 and 4, starting from left. Move point 2 and 3 down midway into the channel. The grid should look like Fig. 7 B afterwards. Then choose *Generate* from the menu and *Boundary* from the pull-down menu. This makes straight lines at the boundary. The grid will look like Fig. 7 C. Now you see why we had to make the "NoMovePoints". Then choose *Generate* from the menu and *Elliptic* from the pull-down menu. Do this a couple of times until the grid looks ok. Now you have generated an elliptic grid, which looks something like Fig. 7 D.

Next, we want to calculate the flow field for this grid. To increase the visual effect, first go to the **map** window in the left upper corner, and start the timer with 5 seconds interval. This makes the map window and the results dialog box refresh automatically every 5 seconds. The timer is started by first choosing *Timer* in the **map** window, and choosing *5 S* on the pull-down menu. Then choose *Timer* once more, but this time choose *Start* from the pull-down menu. You can observe that the window is refreshed every 5 seconds.

To apply the changes from the grid editor, back to the **grid editor** window in the upper right quadrant, and choose *Utilities* from the menu and *apply changes* from the pull-down menu. Then restart the water flow calculation by choosing *Computations* from the main menu and *MB-Flow* from the pull-down menu. Watch the results in the **map** window and the residuals as the solution converges.

Fourth stage

In this stage we look at the animation. This is started by choosing *Graphics* from the main menu and *Animation* from the pull-down menu. The grid seen from above is shown. Move and scale the window and the grid so that it fits in the lower right quadrant of your screen. Click with the mouse inside the grid at the upstream part. Watch the particle that moves to the place you chose. Start the animation by choosing *Particle* in the menu and *run* on the pull-down menu.

If the particle does not move to the exact location where you clicked with the mouse, see **temporary note** in Chapter 3.5

After a while, click with the mouse on another part of the grid, and watch what happens.

Fifth stage

Keeping the screen as it is now, change the grid again, and start another water flow calculation. Observe the changes in the **map** and the **animation** windows.

5.3 Examples

There are four example cases accompanying this program. Each case consist of a **control.xxx** and a **koordina.xxx** file. The files have extensions and need to be renamed/copied to **control** and **koordina** without extensions. Each case is explained in the following:

X shaped channel crossing

The files have extensions **.x**. In this example two channels cross. The example is used to test the multi-block flow routine. Two extra blocks are added to create the geometry. The geometry is symmetric, so the result should also be symmetric.

Y shaped channel

The files have extension **.y**. A channel branches in two at an angle of 45 degrees. The two branches have identical boundary geometry. The case was initially thought of being used for symmetry testing. However, the grid is not symmetrical, and therefore the flow field will not be completely symmetrical. This case is instead used for demonstrating sediment transport.

Curved channel

The files have extensions **.svi**. The channel is used for testing the programs ability to calculate the secondary flow pattern in a curved channel. It is also used for testing the routine that recalculates the water surface location based on the 3D flow field. The cross-sectional slope corresponds very well to theoretical solutions.

Fish farm tank

The files have extensions **.kar**. This case is used for demonstrating inflow and outflow at different locations in the geometry. The inflow is on one side, at 45 degrees to the wall. The outflow is at the bottom of the tank. The case is also used for demonstration of fish habitat. It is also suited for particle animation because the particle will move in a circular pattern.

5.4 Experience with convergence for water flow calculation

In computational fluid mechanics cases it is often a problem to get the solution to converge. Two factors are important:

- A good grid.
- Good relaxation coefficients.

It is presumed that the boundary conditions are correct, which is always a good thing to check first.

Experience shows that the degree of non-orthogonality of the grid will affect the convergence. A higher degree of non-orthogonality will give slower convergence. A slower convergence will also be experienced where strong gradients are present. This applies for example at the inflow of a jet from a wall.

For the convergence of the k and ε equations for river problems, the size of the cell closest to the bed is important. This can be changed by changing the first number in the G 3 data set in the **control** file. This parameter also depends on the roughness of the bed. The size of the bed cell should increase with increasing roughness. The height of the bed cell must however be greater than the roughness of the bed. The following formula can be used to determine the roughness of the bed, given the friction coefficient M of Manning's formula:

The relaxation coefficients are set to the default values of 0.8 for the velocity equations and 0.2 for the pressure-correction equation. For k and ε a value of 0.5 is set. The values of the relaxation coefficients for the velocity equations and the pressure-correction equations are presumed to give an optimum convergence for the average flow case. Values of 0.5 for all equations will often give a slower convergence, but with less probability of blowing up the solution. Theoretically, the sum of the relaxation coefficient for the velocity equations and the relaxation coefficient for the pressure-correction equation should be unity for optimum convergence. However, for some difficult cases, this rule has to be abandoned.

If the solution blows up after the first few iterations, it is possible to set the relaxation coefficients very low, and then increase the coefficients for the following iterations.

For some cases where the equation for k is slowest in convergence, a more rapid convergence has been archived when changing the relaxation coefficient for k from 0.5 to 1.0 after some iterations.

On general it can be said that lower relaxation coefficients will give less instabilities during the convergence, but a slower convergence. Higher relaxation coefficients will give more rapid convergence if there are no instabilities. Instabilities can be observed during the iterations when the residual or the velocities increase and decrease periodically.

Use of block-correction will lead to a more rapid convergence. There is however a possibility of this procedure leading to negative values of k. The program will then crash because in the wall laws the square root of k is calculated. The multi-block flow module avoids this problem never using a k value below 10^{-9}. This value is

equivalent bed shear stress of $3.0x10^{-7}$ N/m^2, and an interpretation of this minimum value is that the bed shear stress also has a minimum value of $3.0x10^{-7}$ N/m^2.

For some cases there have been problems getting the solution to converge if there are parts of the geometry that has relatively low total velocity. Experience has shown that the initial conditions then may be important. If such a situation is present it is important to start the iterations with very low initial velocities.

5.5 Advise for interpretation of results

As mentioned earlier, it is advisable to have experience in computational fluid dynamics when proper interpretation of the results is required. Some guidance is given below.

An important numerical effect that can deteriorate the results is false diffusion. This effect is most profound for first-order schemes, including the POW scheme. The effect depends on how well the flow velocity vectors are aligned with the grid lines. For small alignment angles, the effect is small. Maximum false diffusion will happen when the grid lines are aligned 45 degrees with the flow. The amount of false diffusion also depends on the size of the grid cells.

There are three methods to decrease the amount of false diffusion:

1. Decrease the size of the grid cells == increase the number of cells
2. Align the grid with the flow field
3. Use the second-order upwind scheme

Point 2 may be difficult in a practical situation. However, the calculations should be carried out using approach 1 and/or 3 to assess the effect of false diffusion.

Another important characteristic is the boundary condition. This especially applies to the inflowing boundary. If the velocity field at the inflowing boundary is not known exactly, one should try different velocity distributions to try to assess the effect of this parameter. For a river running into a reservoir, it is possible to model a part of the river upstream of the reservoir, and thereby obtaining a better estimate for the velocity distribution where the river enters the reservoir. The upstream boundary condition is also important for sediment calculations, where both the total amount of sediment inflow and the sediment grain size distribution can be varied. For the bed boundary, it is possible to vary the roughness to investigate the effect of this parameter. It can also sometimes be advantageous to make variations for the formula for sediment concentration close to the bed. This especially applies for sediment particles outside the range for which the formula is applied for.

When interpreting the results from the model it is also important to keep the accuracy of model in mind. The k-ε turbulence model has limitations in how accurate the turbulence field is predicted. This will also affect the velocity field. For example, when calculating the recirculation zone for a step case, the length of the recirculation zone can often not be predicted with any better accuracy than 10 - 30 %.

In some situations the water flow will be time-dependent. An example can be oscillations behind a cylinder or in an expansion. The equations that are solved by the program do not have time-dependent terms. Therefore the converged solution will be steady. It is possible to obtain a steady solution from the model although the physical problem is unsteady. This must be considered when interpreting the results. The effects of an unsteady solution compared to the given solution should be assessed if this is probable. When an unsteady case is solved with a steady method there can be convergence problems. If the relaxation factors have to be fairly low to get the solution to converge, this can be an indication that the flow field may be unsteady.

Another topic of interpretation is the resolution of the calculated flow field compared with the size of the grid cells. Several cells are required to dissolve a recirculating zone. Flow field characteristics smaller than about 4-7 cells does often not show up in the solution.

Literature

[1] Engelund, F., and Hansen, E., "A monograph on sediment transport in alluvial streams", Teknisk Forlag, Copenhagen, Denmark, 1967.

[2] Mayer-Peter, E. and Mueller, R., "Formulas for bed load transport", Report on Second Meeting of International Association for Hydraulic Research, Stockholm, Sweden, 1948.

[3] Melaaen, M. C., "Analysis of curvilinear non-orthogonal coordinates for numerical calculation of fluid flow in complex geometries", Dr. Ing. Thesis, The Norwegian Institute of Technology, Trondheim, 1990.

[4] Olsen, N. R., "A numerical model for simulation of sediment movements in water intakes", Dr. Ing. Dissertation, The Norwegian Institute of Technology, Trondheim, 1991.

[5] Van Rijn, L. C., "Mathematical modeling of morphological processes in the case of suspended sediment transport", Ph.D Thesis, Delft University of Technology, 1987.

[6] Vanoni, V., et al, "Sedimentation Engineering", ASCE Manuals and reports on engineering practice - No54, 1975.

[7] Ashworth, R., "Renormalisation Group Turbulence Model", Eight International Conference on Numerical Methods in Laminar and Turbulent Flow, Swansea, July, 1993.

[8] Olsen, N. R. B., and Melaaen, M. C., "Numerical Modeling of Erosion around a Cylinder and Sediment Deposition in a Hydropower Reservoir", Eight International Conference on Numerical Methods in Laminar and Turbulent Flow, Swansea, July 1993.

[9] Olsen, N. R. B., and Melaaen, M. C., "Three-dimensional numerical modeling of scour around cylinders", ASCE Journal of Hydraulic Engineering, Vol. 119, No. 9, September 1993.