MIKE 21 Flow Model & MIKE 21 Flood Screening Tool

Hydrodynamic Module

Scientific Documentation

DHI headquarters
Agern Allé 5
DK-2970 Hørsholm
Denmark

+45 4516 9200 Telephone
+45 4516 9333 Support
+45 4516 9292 Telefax

mike@dhigroup.com
www.mikepoweredbydhi.com

PLEASE NOTE

CONTENTS

MIKE 21 Flow Model & MIKE 21 Flood Screening Tool
Hydrodynamic Module
Scientific Documentation

1 Introduction

The present Scientific Documentation aims at giving an in-depth description of the equations and numerical formulation used in the hydrodynamic module of the MIKE 21 Flow Model, MIKE 21 HD.

First the main equations and the numerical algorithm applied in the model are described. This is followed by a number of sections giving the physical, mathematical and numerical background for each of the terms in the main equations.

2 Main Equations

The hydrodynamic model in the MIKE 21 Flow Model (MIKE 21 HD) is a general numerical modelling system for the simulation of water levels and flows in estuaries, bays and coastal areas. It simulates unsteady two-dimensional flows in one layer (vertically homogeneous) fluids and has been applied in a large number of studies.

The following equations, the conservation of mass and momentum integrated over the vertical, describe the flow and water level variations:

$$\frac{\partial \zeta}{\partial t} + \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} = \frac{\partial d}{\partial t} \tag{2.1}$$

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left(\frac{p^2}{h}\right) + \frac{\partial}{\partial y}\left(\frac{pq}{h}\right) + gh\frac{\partial \zeta}{\partial x}$$

$$+ \frac{gp\sqrt{p^2+q^2}}{C^2 \cdot h^2} - \frac{1}{\rho_w}\left[\frac{\partial}{\partial x}\left(h\tau_{xx}\right) + \frac{\partial}{\partial y}\left(h\tau_{xy}\right)\right] - \Omega q \tag{2.2}$$

$$- fVV_x + \frac{h}{\rho_w}\frac{\partial}{\partial x}\left(p_a\right) = 0$$

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial y}\left(\frac{q^2}{h}\right) + \frac{\partial}{\partial x}\left(\frac{pq}{h}\right) + gh\frac{\partial \zeta}{\partial y}$$

$$+ \frac{gq\sqrt{p^2+q^2}}{C^2 \cdot h^2} - \frac{1}{\rho_w}\left[\frac{\partial}{\partial y}\left(h\tau_{yy}\right) + \frac{\partial}{\partial x}\left(h\tau_{xy}\right)\right] + \Omega p \tag{2.3}$$

$$- fVV_y + \frac{h}{\rho_w}\frac{\partial}{\partial y}\left(p_a\right) = 0$$

The following symbols are used in the equations:

$h(x,y,t)$	water depth (= $\zeta - d$, m)
$d(x,y,t)$	time varying water depth (m)
$\zeta(x,y,t)$	surface elevation (m)
$p,q\ (x,y,t)$	flux densities in x- and y-directions ($m^3/s/m$) = (uh,vh); (u,v) = depth averaged velocities in x- and y-directions
$C(x,y)$	Chezy resistance ($m^{½}/s$)
g	acceleration due to gravity (m/s^2)
$f(V)$	wind friction factor
$V,V_X,V_Y(x,y,t)$	wind speed and components in x- and y-direction (m/s)
$\Omega(x,y)$	Coriolis parameter, latitude dependent (s^{-1})
$p_a(x,y,t)$	atmospheric pressure ($kg/m/s^2$)
ρw	density of water (kg/m^3)
x,y	space coordinates (m)
t	time (s)
$\tau xx,\ \tau xy,\ \tau yy$	components of effective shear stress

3 Introduction to Numerical Formulation

MIKE 21 HD makes use of a so-called Alternating Direction Implicit (ADI) technique to integrate the equations for mass and momentum conservation in the space-time domain. The equation matrices that result for each direction and each individual grid line are resolved by a Double Sweep (DS) algorithm.

MIKE 21 HD has the following properties:

- Zero numerical mass and momentum falsification and negligible numerical energy falsification, over the range of practical applications, through centring of all difference terms and dominant coefficients, achieved without resort to iteration.
- Second- to third-order accurate convective momentum terms, i.e. "second- and third-order" respectively in terms of the discretisation error in a Taylor series expansion.
- A well-conditioned solution algorithm providing accurate, reliable and fast operation.

The difference terms are expressed on a staggered grid in x, y-space as shown in Figure 3.1.

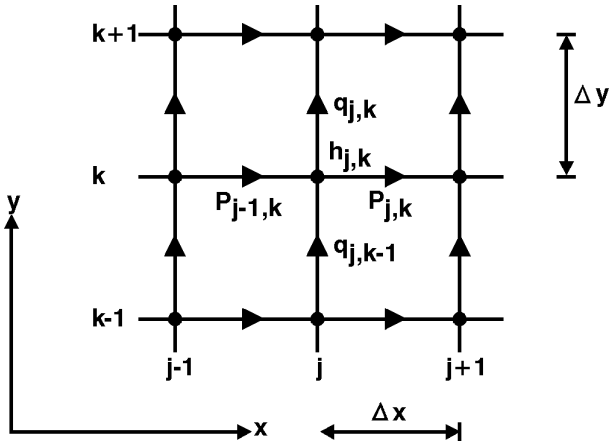

Figure 3.1 Difference Grid in x,y-space

Time centring of the three equations in MIKE 21 HD is achieved as illustrated in Figure 3.2.

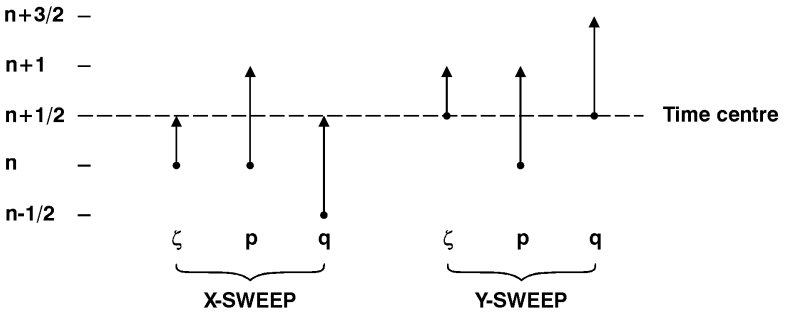

Figure 3.2 Time centring

The equations are solved in one-dimensional sweeps, alternating between x and y directions. In the x-sweep the continuity and x-momentum equations are solved, taking ζ from *n* to *n+½* and *p* from *n* to *n+1*. For the terms involving *q*, the two levels of old, known values are used, i.e. *n-½* and *n+½.*

In the y-sweep the continuity and y-momentum equations are solved, taking ζ from *n+½* to *n+1* and *q* from *n+½* to *n+3/2*, while terms in *p* use the values just calculated in the x-sweep at *n* and *n+1*.

Adding the two sweeps together gives "perfect" time centring at *n+½*, i.e. the time centring is given by a balanced sequence of operations. The word perfect has been put in quotation marks because it is not possible to achieve perfect time centring of the cross derivatives in the momentum equation. The best approximation, without resorting to iteration (which has its own problems), is to use a "side-feeding" technique.

At one time step the x-sweep solutions are performed in the order of decreasing y-direction, hereafter called a "down" sweep, and in the next time step in the order of increasing y-direction, the "up" sweep.

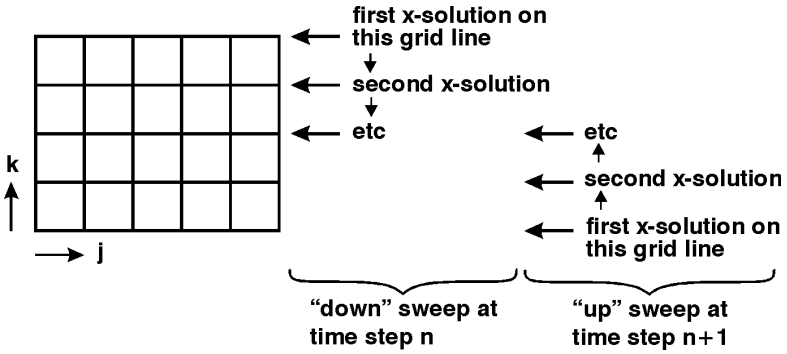

Figure 3.3 Side-feeding

During a "down" sweep, the cross derivative ∂p/∂y can be expressed in terms of $p_{j,k+1}^{n+1}$ on the "up" side and $p_{j,k-1}^{n}$ on the "down" side, and vice versa during an "up" sweep. In this way an approximate time centring of ∂p/∂y at *n+½* can be achieved, albeit with the possibility of developing some oscillations (zigzagging).

The use of side-feeding for the individual cross differentials is described in more detail in the following sections.

Finally, it should also be mentioned here that it is not always possible to achieve a perfect time centring of the coefficients on the differentials.

Centring in space is not generally a problem as will be seen in the next sections.

A mass equation and momentum equation thus expressed in a one-dimensional sweep for a sequence of grid points lead to a three-diagonal matrix

$$MV^{n+1} = W^{n} \tag{3.1}$$

$$A_j \cdot p_{j-1}^{n+1} + B_j \cdot \zeta_j^{n+½} + C_j \cdot p_j^{n+1} = D_j \mid_k$$

(3.2)

$$A_j^* \cdot \zeta_j^{n+½} + B_j^* \cdot p_j^{n+1} + C_j^* \cdot \zeta_{j+1}^{n+½} = D_j^* \mid_k$$

where the coefficients *A, B, C, D* and A^*, B^*, C^*, D^* are all expressed in "known" quantities.

Note that *p* here may be *q* and *j* may as well be *k*.

The system (3.1) is then solved by the well-known Double Sweep algorithm. For reference one may see, for example, Richtmeyer and Morton (1967), Ref. /12/. In developing the algorithm one postulates that relations exist

$$p_j^{n+1} = E_j^* \cdot \zeta_j^{n+½} + F_j^*$$

(3.3)

$$\zeta_j^{n+½} = E_j \cdot p_j^{n+1} + F_{j+1}$$

Substituting these relations back into the Equations (3.2) give recurrence relations for *E, F,* E^* *and* F^*.

$$E_j^* = \frac{-A_j^*}{B_j^* + C_j^* \cdot E_j}$$

$$F_j^* = \frac{D_j^* - C_j^* \cdot F_j}{B_j^* + C_j^* \cdot E_j}$$

(3.4)

$$E_{j-1} = \frac{-A_j}{B_j + C_j \cdot E_j^*}$$

$$F_{j-1} = \frac{D_j - C_j \cdot F_j^*}{B_j + C_j \cdot E_j^*}$$

It is clear that once a pair of E_j, F_j values is known (or E_{j+1}^*, F_{j+1}^*) then all *E, F* and E^*, F^* coefficients can be computed for decreasing j. Introducing the right-hand boundary condition into one of the Equations (3.2) starts the recurrence computation for *E, F* and E^*, F^* - The *E, F*-sweep. Introducing the left-hand boundary condition in (3.3) starts the complimentary sweep in which ζ and *q* are computed.

As discussed earlier, sweeps may be carried out with a decreasing complimentary coordinate or an increasing complimentary coordinate. This is organised in the cycle shown in Figure 3.4.

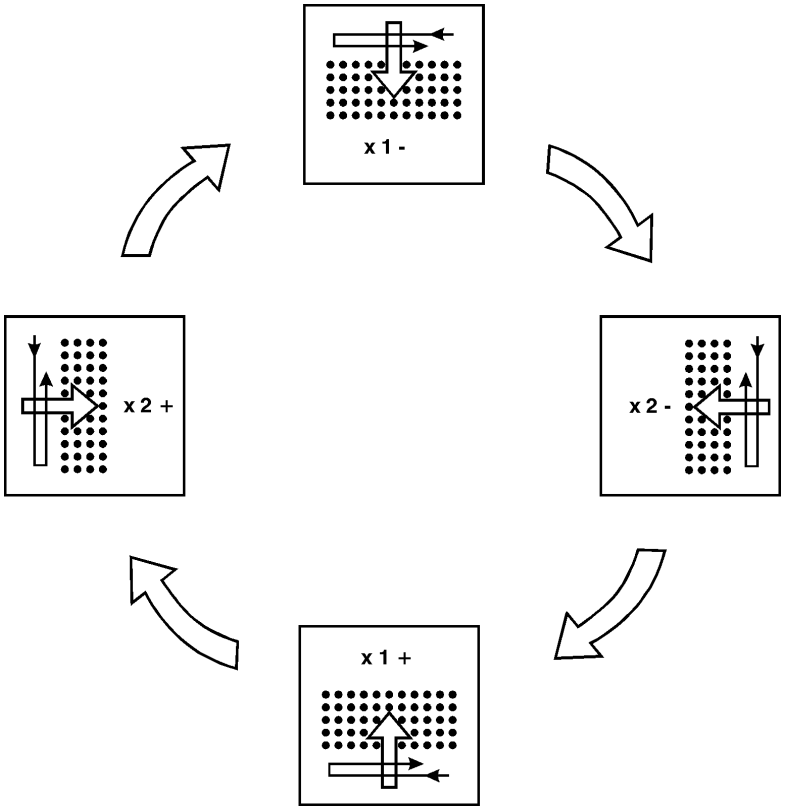

Figure 3.4 Cycle of Computational Sweeps

In Section 6 the numerical properties of the difference scheme in terms of amplification and propagation errors are discussed. Before this, we shall present various difference approximations.

4 Difference Approximations for Points away from Coast

We shall mainly look at the mass and momentum equations in the x-direction. As the mass equation in the y-direction influences the centring of the x-mass equation we shall also consider the difference approximation of this equation. The momentum equation in the y-direction is analogous to the momentum equation in the x-direction and is, accordingly, omitted here.

4.1 Mass Equation in the x-Direction

The mass equation reads

$$\frac{\partial \zeta}{\partial t} + \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} = \frac{\partial d}{\partial t} \tag{4.1}$$

The x- and y-sweeps are organised in a special cycle as shown in the preceding section. In Section 6 it is shown how the computation proceeds in time and how the equations are time centred.

In order to fully understand the balance between the difference approximations employed in the various sweeps it is necessary to read Section 6 in conjunction with the following sections. For the moment it is sufficient to say that the x-mass and x-momentum equations bring ζ from time level *n* to *n+½* while bringing p from *n* to *n+1*. Together with the y-mass equation the terms are centred at *n+½*.

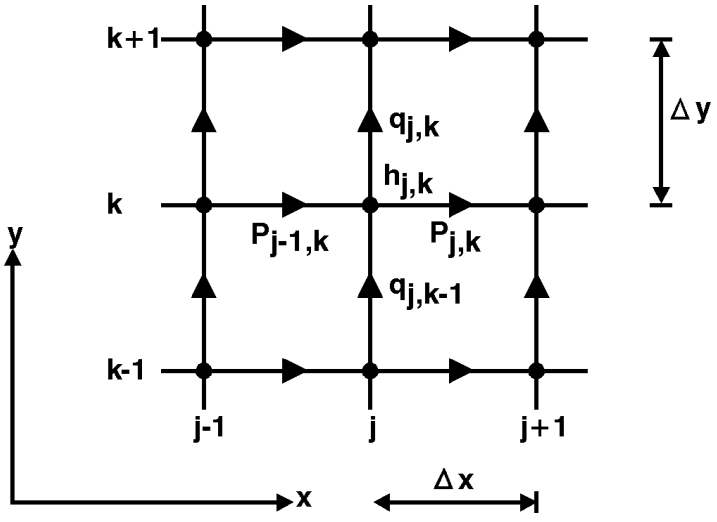

Figure 4.1 Grid Notation: Mass Equation

With the grid notation given in Figure 4.1 above, Equation (4.1) becomes

$$2\cdot\left(\frac{\zeta^{n+½}-\zeta^{n}}{\Delta t}\right)_{j,k}+\frac{1}{2}\cdot\left\{\left(\frac{p_j-p_{j-1}}{\Delta x}\right)^{n+1}+\left(\frac{p_j-p_{j-1}}{\Delta x}\right)^{n}\right\}_{k}$$

$$+\frac{1}{2}\cdot\left\{\left(\frac{q_k-q_{k-1}}{\Delta y}\right)^{n+½}+\left(\frac{q_k-q_{k-1}}{\Delta y}\right)^{n-½}\right\}_{j}=2\cdot\left(\frac{d^{n+½}-d^{n}}{\Delta t}\right)_{j,k} \tag{4.2}$$

4.2 Mass Equation in the y-Direction

The y sweep immediately following the x-sweep, for which the mass equation was just described, brings ζ from time level *n+½* to level *n+1* and helps to centre the x-mass and x-momentum equations. With the grid notation of Figure 4.1, Equation (4.1) becomes

$$2\cdot\left(\frac{\zeta^{n+1}-\zeta^{n+½}}{\Delta t}\right)_{j,k}+\frac{1}{2}\cdot\left\{\left(\frac{p_j-p_{j-1}}{\Delta x}\right)^{n+1}+\left(\frac{p_j-p_{j-1}}{\Delta x}\right)^{n}\right\}_{k}$$

$$+\frac{1}{2}\cdot\left\{\left(\frac{q_k-q_{k-1}}{\Delta y}\right)^{n+3/2}+\left(\frac{q_k-q_{k-1}}{\Delta y}\right)^{n+½}\right\}_{j}=2\cdot\left(\frac{d^{n+1}-d^{n}}{\Delta t}\right)_{j,k} \tag{4.3}$$

Prior to each sweep the bathymetry (when the landslide option is included) is read from the bathymetry data file and interpolated to the respective time step, i.e. *n+½* for a x-sweep and *n+1* for a y-sweep. After completion of each sweep the water depth is updated to the actual value based on surface elevation and bathymetry, yielding $h^{n+½} = \zeta^{n+½} - d^{n+½}$ after the x-sweep and $h^{n+1} = \zeta^{n+1} - d^{n+1}$ after the y-sweep.

We will not discuss general truncation errors at this point. As the approximations are based on a multi-level difference method, centring of terms and the evaluation of truncation errors should be considered in conjunction with a certain set of equations. We will revert to this point in Section 6.

4.3 Momentum Equation in the x-Direction

4.3.1 General

The x-component of the momentum equation reads:

$$\frac{\partial p}{\partial t}+\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)+\frac{\partial}{\partial y}\left(\frac{pq}{h}\right)+gh\frac{\partial \zeta}{\partial x}$$

$$+\frac{gp\sqrt{p^2+q^2}}{C^2\cdot h^2}-\frac{1}{\rho_w}\left[\frac{\partial}{\partial x}\left(h\tau_{xx}\right)+\frac{\partial}{\partial y}\left(h\tau_{xy}\right)\right]-\Omega q \qquad (4.4)$$

$$-fVV_x+\frac{h}{\rho_w}\frac{\partial}{\partial x}\left(p_a\right)=0$$

We shall develop the difference forms by considering the various terms one by one.

The following basic principle is used for the x-momentum finite difference approximations:

All terms in (4.4) will be time-centred at *n+½* and space centred at the location corresponding to *Pj,k* in the space-staggered grid. The grid notation is shown in Figure 4.2.

4.3.2 The time derivation term

The straight forward finite difference approximation to the time derivative term is

$$\frac{\partial p}{\partial t}\approx\left(\frac{p^{n+1}-p^n}{\Delta t}\right)_{j,k} \qquad (4.5)$$

Using a Taylor expansion centred at n+½ leads to

$$\frac{\partial p}{\partial t}\approx\left(\frac{p^{n+1}-p^n}{\Delta t}\right)_{j,k}-\frac{\Delta t^2}{24}\cdot\frac{\partial^3 p}{\partial t^3}+HOT\ (Higher\ \ Order\ \ Terms) \qquad (4.6)$$

In standard hydrodynamic simulations only the first term in (4.6) is included in the scheme. For short wave applications using the BW module (Boussinesq waves) the second term in (4.6) is also included to obtain a higher accuracy of the scheme.

4.3.3 The gravity term

The straight forward approximation to the gravity term reads

$$gh\zeta_x \approx g\left(\frac{h_{j,k}+h_{j+1,k}}{2}\right)^n\left(\frac{\zeta_{j+1,k}-\zeta_{j,k}}{\Delta x}\right)^{n+½} \tag{4.7}$$

where

$$h^n_{j,k} = d_{j,k} + \zeta^n_{j,k}$$

In this way the term has been linearised in the resulting algebraic formulation. Truncation errors embedded in (4.7) can be determined by the use of Taylor expansions centred at *j+½,k* and *n+½*. This leads to

$$gh\zeta_x \approx FDS + g\left[\frac{\Delta t}{2}\zeta_x\zeta_t - \frac{\Delta t^2}{8}\zeta_{tt}\zeta_x - \frac{\Delta x^2}{8}\zeta_{xx}\zeta_x - \frac{\Delta x^2}{24}h\zeta_{xxx}\right] + \quad HOT \quad (Higher \quad Order \quad Terms) \tag{4.8}$$

where FDS is the right hand side of (4.7).

In standard hydrodynamic simulations only the FDS term is included in the scheme. For short wave applications using the BW module, the truncation errors proportional to Δt and Δt^2 are eliminated by shifting the time level of the first bracket in (4.7) from n to *n+½*. This is done in an approximative way by explicit use of the continuity equation. Furthermore, the last term in (4.8) is included in the higher order accuracy scheme used in the BW module.

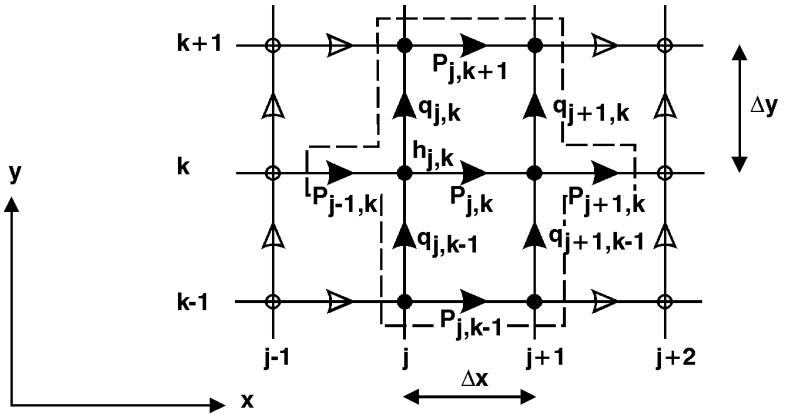

Figure 4.2 Grid Notation: x-Momentum Equation

4.3.4 The convective and cross-momentum correction terms

$$\frac{\partial}{\partial x}\left(\frac{p\ p}{h}\right)+\frac{\partial}{\partial y}\left(\frac{q\ p}{h}\right) \tag{4.9}$$

This requires further discussion. One way of approximating both terms would be to form spatially centred differences of time-centred forms of the bracketed terms. For example,

$$\frac{1}{2\Delta x}\left\{\left(\frac{p\ p}{h}\right)^{n+\frac{1}{2}}_{j+1}-\left(\frac{p\ p}{h}\right)^{n+\frac{1}{2}}_{j-1}\right\}_{k} \tag{4.10}$$

and a similar form for the cross-momentum term. (How the time centring is achieved will be shown in the final difference forms). However, this approximation is not supported by flux at the central point $p_{j,k}$ and this will give rise to zigzagging of flow patterns if variations close to the highest resolvable wave number have to be described. We may, for example, consider the case of a flow concentration around the tip of a pier. The situation, with and without zigzagging, is illustrated in Figure 4.3. The illustration is taken from Abbott and Rasmussen (1977), Ref. /2/, where the problem is discussed, although in a slightly different context.

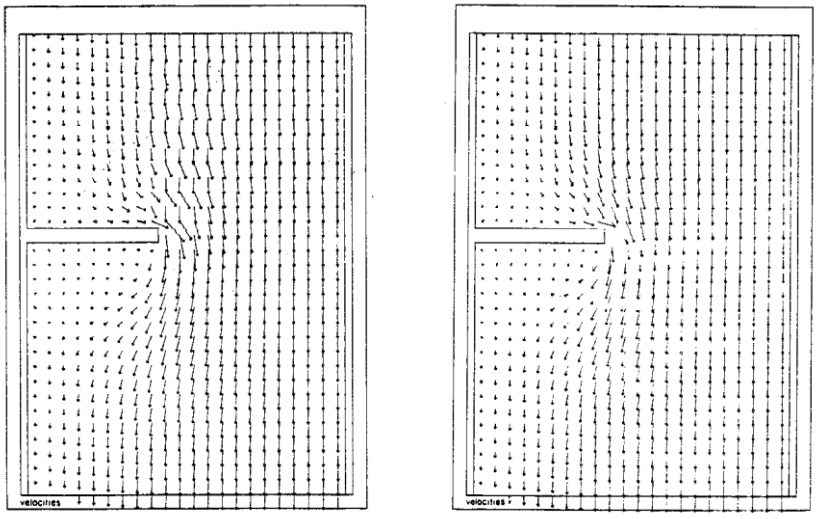

Figure 4.3 Zigzagging in Flow Concentration

A popular exposition of the problem is given by Leonard (1979), Ref. /10/. A central difference form has neutral stability for first order differential terms, being insensitive to the central flux $p_{j,k}$. That is, $p_{j,k}$ may vary without a stabilising positive feedback and erroneously affect the time derivative. This, in fact, is what is occurring during the zigzagging process.

To illustrate this point further, consider $\partial(pq/h)/\partial y$ in connection with a flow concentration giving the variation of p shown in Figure 4.4.

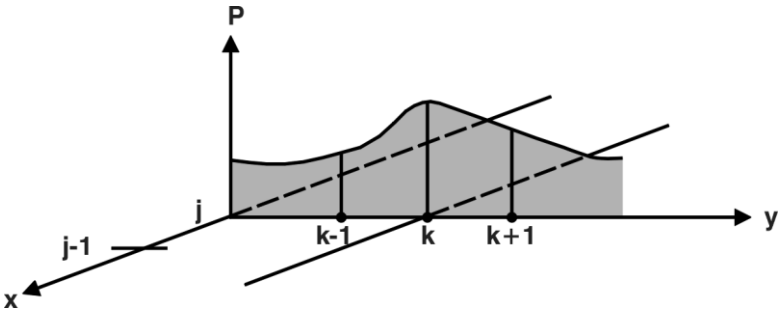

Figure 4.4 Variation of p at a Flow Concentration

Assuming for the present discussion that $v = q/h$ varies much less with y than does $u = p/h$, we have

$$\frac{\partial}{\partial y}\left(\frac{p\ q}{h}\right) \approx \frac{\partial p}{\partial y} \tag{4.11}$$

For $y = k \cdot \Delta y$, then $\partial p/\partial y \approx 0$ and does not contribute to $\partial p/\partial t$. However, $(p_{k+l} - p_{k-l})/2\Delta y$ is not zero. It may be either positive or negative and give an increase or decrease to $\partial p/\partial t$. Thus, the discrete description introduces an exchange of momentum in the y-direction between sweeps in the x-direction which, in the continuous description, may not be present. A similar argument applies for a variation in p in the x-direction, at or close to the highest resolvable wave number.

To improve the situation, clearly we should introduce the curvature of p. This becomes apparent when the transport nature of the terms are considered.

We may rewrite them as

$$u\ \frac{\partial p}{\partial x} + p\ \frac{\partial u}{\partial x} + v\ \frac{\partial p}{\partial y} + p\ \frac{\partial v}{\partial y} \tag{4.12}$$

Following the derivation by Abbott, McCowan and Warren (1981), Ref. /4/ (Chapter 6), we consider the first and third term together with the time derivative, i.e.

$$\frac{\partial p}{\partial t} + u\ \frac{\partial p}{\partial x} + v\ \frac{\partial p}{\partial y} \tag{4.13}$$

(we "forget" for the moment the two other terms). This represents a transport of the x-flux with the resultant of the x-and y-velocities. The integral form of the transport equation, which corresponds to an exact solution of the differential form, is,

$$p\left(x,y,t_2\right) = p\left(x - \int_{t1}^{t2} \bar{u}\, dt, y - \int_{t1}^{t2} \bar{v}\, dt,\ \ t_1\right) \tag{4.14}$$

where the velocities and are averaged quantities over the time interval t_2-t_1.

Now, consider the discrete description over Δt. Equation (4.14) may then be written as

$$p\left(j\Delta x, k\Delta y, (n+1)\Delta t\right) = p\left(j\Delta t - \bar{u}\Delta t, k\Delta t - \bar{v}\Delta t, n\Delta t\right) \tag{4.15}$$

When the right-hand term and the left-hand term are developed with *(n+½)*Δt, jΔx, kΔy as the centre, we obtain

$$\left|\frac{\partial p}{\partial t}+\bar{u}\frac{\partial p}{\partial x}+\bar{v}\frac{\partial p}{\partial y}\right|_{j,k}^{n+½}$$

$$-\frac{1}{2!}\left\{\bar{u}^{2}\,\Delta t\,\frac{\partial^{2} p}{\partial x^{2}}+2\,\bar{u}\,\bar{v}\,\Delta t\,\frac{\partial^{2} p}{\partial x\partial y}+\bar{v}^{2}\,\Delta t\,\frac{\partial^{2} p}{\partial y^{2}}\right. \tag{4.16}$$

$$\left.+\bar{u}\,\Delta t\,\frac{\partial^{2} p}{\partial x\partial t}+\bar{v}\,\Delta t\,\frac{\partial^{2} p}{\partial y\partial t}\right\}_{j,k}^{n+1}+\quad HOT\quad (\,Higher\quad Order\quad Terms)$$

Thus, representing the transport terms on a discrete grid with 2nd order discretisation terms requires the introduction of five correction terms. Two of these terms, $\partial^2 p/\partial x^2$ and $\partial^2 p/\partial y^2$, will bring the central point *j,k* into the difference approximation. We shall retain these two terms and neglect the others (as we have neglected until now the terms $p(\partial u/\partial x)$ and $p(\partial v/\partial y)$. This appears rather arbitrary and, in fact, it is. We cannot argue that, in general, the terms that we intend to neglect are necessarily smaller than the two terms we wish to retain. It should, however, be remembered that in computations with a time scale of the order of tidal motion, the correction terms will all be fairly small. Thus they will not contribute significantly to the accuracy of the principle solution. However, neglecting in particular the terms $\partial^2 p/\partial x^2$ and $\partial^2 p/\partial y^2$ deprives the solution of the support in the central point, allowing small local disturbances to grow to finally poison the entire solution. Experience with MIKE 21 HD has shown that accurate solutions can be obtained with the representation of the convective- and cross-momentum corrections by only two terms.

Now it might be argued that "we just dissipate the higher-order disturbances". Indeed, the second-order space derivatives have the form of stress terms that one would use, for example, in a second-order dissipative interface. However, they are much more selective, being effective only where *u* or *v* are large and then they work towards a more correct solution.

4.3.5 Convective momentum

We can now write the difference form for

$$\frac{\partial}{\partial x}\left(\frac{pp}{n}\right)-\frac{1}{2}\bar{u}\,\Delta t\frac{\partial^2 p}{\partial x^2} \tag{4.17}$$

On the grid below, we can represent the terms as follows:

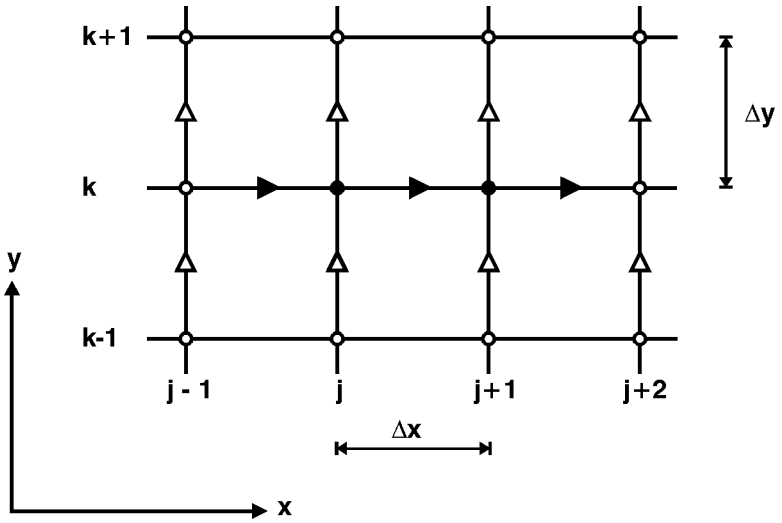

Figure 4.5 Grid Notation: x-Momentum Equation

$$\frac{\partial}{\partial x}\left(\frac{p\,p}{h}\right)\approx\left[\frac{\left(p_{j+1}+p_j\right)^{n+1}}{2}\cdot\frac{\left(p_{j+1}+p_j\right)^{n}}{2}\cdot\frac{1}{h^n_{j+1}}\right.$$

$$\left.-\frac{\left(p_j+p_{j-1}\right)^{n+1}}{2}\cdot\frac{\left(p_j+p_{j-1}\right)^{n}}{2}\cdot\frac{1}{h^n_j}\right]_k\cdot\frac{1}{\Delta x} \tag{4.18}$$

$$\bar{u}^2\;\Delta t\;\frac{\partial^2 p}{\partial x^2}\;\approx\;\Delta t\left(\frac{p^n_{j,k}}{h^*}\right)^2\cdot\left(\frac{p_{j,k}-2p_j+p_{j-1}}{(\Delta x)^2}\right)^{n+1}_k \tag{4.19}$$

with

$$h^*\;=\;\frac{1}{2}\cdot\left(h_{j+1}+h_j\right)^n_k \tag{4.20}$$

One will note that the difference form in (4.18) in fact involves 5 diagonals in the matrix of difference equations, whereas we employ a "3-diagonal" algorithm for its solution. One can extend the "3-diagonal" algorithm to a "5-diagonal" algorithm. Here we have chosen to reduce the form (4.18) to a 3-diagonal form by local substitution.

In (4.19) we note that we approximate , the average velocity over the interval $t_1 = n \cdot \Delta t$ to $t_2 = (n+1) \cdot \Delta t$, by $p_{j,k}/h^*$.

Also, the difference form is written fully on the forward time level. In view of the other errors - neglecting other correction terms - this approximation error is of a higher order.

The difference form in (4.18) is used for flow at low Froude Numbers. For flow at high Froude Numbers, a scheme as described below is used. In this scheme selective introduction of numerical dissipation has been used to improve the robustness of the numerical solution in areas of high velocity gradients, and to provide MIKE 21 with the capability to simulate locally super-critical flows. This numerical dissipation has been introduced through selective “up-winding” of the convective momentum terms, as *Fr* increases. The rationale behind this approach is that the introduction of numerical dissipation at high Froude Numbers can be tuned to be roughly analogous to the physical dissipation caused by increased levels of turbulence in high velocity flows.

Effects of up-winding

The fully space centred description of the convective momentum term considered in (4.17) can be approximated by:

$$\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_j \approx \frac{1}{\Delta x}\left[\left(\frac{p^2}{h}\right)_{j+1/2} - \left(\frac{p^2}{h}\right)_{j-1/2}\right] \tag{4.21}$$

For positive flow in the *x*-direction, the up-winded form of the convective momentum term can be approximation by:

$$\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_{j-1/2} \approx \frac{1}{\Delta x}\left[\left(\frac{p^2}{h}\right)_{j} - \left(\frac{p^2}{h}\right)_{j-1}\right] \tag{4.22}$$

Allowing for the back-centring in space, the up-winded term can be shown to be equivalent to the original space-centred term, plus an additional second order term, as follows:

$$\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_{j-1/2} \approx \frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_{j} - \frac{\Delta x}{2}\frac{\partial^2}{\partial x^2}\left(\frac{p^2}{h}\right)_{j} \tag{4.23}$$

This second order term is highly dissipative for high frequency oscillations, but has little effect on lower frequencies. That is, it will tend to damp out high frequency numerical instabilities, while having little effect on the overall computation.

Selective up-winding

To ensure that the dissipative effects of up-winding are only included when necessary, a Froude Number dependent weighting factor α has been introduced where:

$$\alpha = 0, \quad Fr \le 0.25$$
$$\alpha = \frac{4}{3}(Fr - 0.25), \quad 0.25 < Fr < 1.0 \qquad (4.24)$$
$$\alpha = 1, \quad Fr \ge 1$$

The weighting factor *α* is applied to the convective momentum terms, such that:

$$\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_j \approx (1-\alpha)\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_j + \alpha\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_{j-1/2} \qquad (4.25)$$

This brings the effects of up-winding in gradually as the Froude Number increases from 0.25 to 1.0. For Froude Numbers of *Fr* = 1.0 or more, the convective momentum term is fully up-winded.

Computational form

In the form described in (4.18), the actual representation of the convective momentum equation (for positive flow in the x-direction) can be expressed as follows:

$$\frac{\partial}{\partial x}\left(\frac{p^2}{h}\right)_{j,k}^{n+1/2} \approx$$
$$\left[\frac{\left[(1-\alpha)p_{j+1} + (1+\alpha)p_j\right]^{n+1}}{2}\cdot\frac{\left[(1-\alpha)p_{j+1} + (1+\alpha)p_j\right]^{n}}{2}\cdot\frac{1}{\left[(1-\alpha)h_{j+1} + \alpha.h_j\right]^{n}} - \right. \qquad (4.26)$$
$$\left.\frac{\left[(1-\alpha)p_j + (1+\alpha)p_{j-1}\right]^{n+1}}{2}\cdot\frac{\left[(1-\alpha)p_j + (1+\alpha)p_{j-1}\right]^{n}}{2}\cdot\frac{1}{\left[(1-\alpha)h_j + \alpha.h_{j-1}\right]^{n}}\right]_k \cdot\frac{1}{\Delta x}$$

The weighting factor α for each grid point is calculated every time step, immediately prior to the calculation of the momentum equation coefficients. This ensures that numerical dissipation is only introduced at grid points where high Froude Number flow is occurring, and that the normal high accuracy solution of MIKE 21 is obtained throughout the rest of the model domain.

Selective up-winding is only included on the convective momentum terms and not the cross momentum terms.

With the introduction of selective up-winding of the convective momentum terms, it has been possible to virtually eliminate the unrealistic oscillations and local instabilities that occurred previously when modelling high Froude Number flows. This has improved significantly the robustness of MIKE 21's solution procedure at high Froude Numbers, and has enhanced significantly MIKE 21's capability to include (qualitatively at least):

- Locally super-critical flows
- Weir and levee bank flows (on a grid scale)
- Hydraulic jumps

Selective up-winding also ensures that the high accuracy of solutions in other areas remains unaffected.

4.3.6 Cross-momentum

$$\frac{\partial}{\partial y}\left(\frac{p\ q}{h}\right) - \frac{1}{2}\ \bar{v}^2\ \Delta t\ \frac{\partial^2 p}{\partial y^2} \tag{4.27}$$

The difference approximation will differ between an "up" sweep and a "down" sweep. We shall use "side feeding" as a means to centre the term at level *(n+½)* Δt.

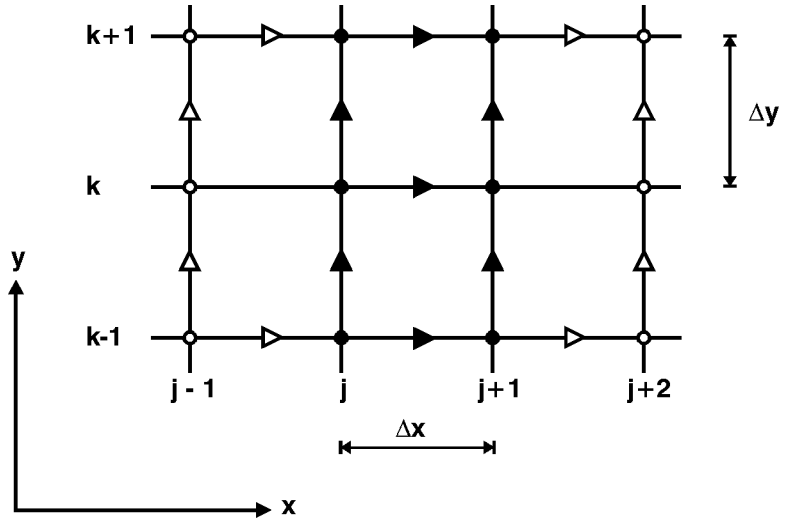

Figure 4.6 Grid Notation: x-Momentum Equation

We write, referring to the grid notation of Figure 4.6,

$$\frac{\partial}{\partial y}\left(\frac{p\,q}{h}\right) \approx \left[\left(\frac{p^a_{k+1} + p^b_k}{2}\right)_j \cdot v^{n+½}_{j+½,k} - \left(\frac{p^a_k + p^b_{k-1}}{2}\right)_j \cdot v^{n+½}_{j+½,k-1}\right] \cdot \frac{1}{\Delta y} \tag{4.28}$$

where: $a = n+1$, $b = n$ for a "down" sweep
$a = n$, $b = n+1$ for an "up" sweep

$$v^{n+½}_{j+½,k} = \frac{2(q_j + q_{j+1})^{n+½}_k}{(h_{j,k} + h_{j,k+1} + h_{j+1,k} + h_{j+1,k+1})^n}$$

$$v^{n+½}_{j+½,k-1} = \frac{2(q_j + q_{j+1})^{n+½}_{k-1}}{(h_{j,k-1} + h_{j,k} + h_{j+1,k-1} + h_{j+1,k})^n} \tag{4.29}$$

$$\bar{v}\ \Delta t\ \frac{\partial^2 p}{\partial y^2}\ \approx\ \Delta t(v^*)^2\ \cdot\ \frac{\left\{p_{k+1}^{\ a}-(p^{n+1}_{\ k}+p_k^n)+p_{k-1}^b\right\}_j}{(\Delta y)^2} \tag{4.30}$$

with *a* and *b* defined as above and

$$v^*\ =\ \frac{1}{2}\ \cdot\ (v_{k+½}+v_{k-½})^{n+½}_{j+½} \tag{4.31}$$

The diagrams in Figure 4.7 and Figure 4.8 may illustrate how the cross terms are built. Note that the main computation we are dealing with in this approximation of the x-momentum equation is in the x-direction. By "down" sweep or "up" sweep we mean in fact computational sweeps in the x-direction, carried out by decreasing or increasing y respectively.

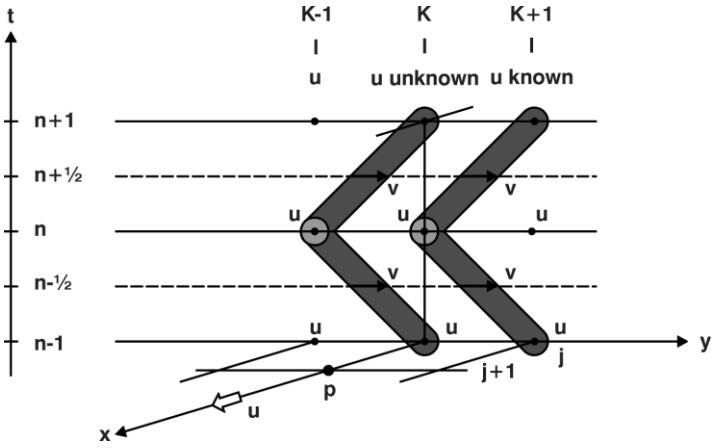

Figure 4.7 "Side-Feeding" for the Cross-Momentum Term. p(n+1,k+1) known, calculated by a "down" sweep. p(n,k-1) known, calculated by an "up" sweep

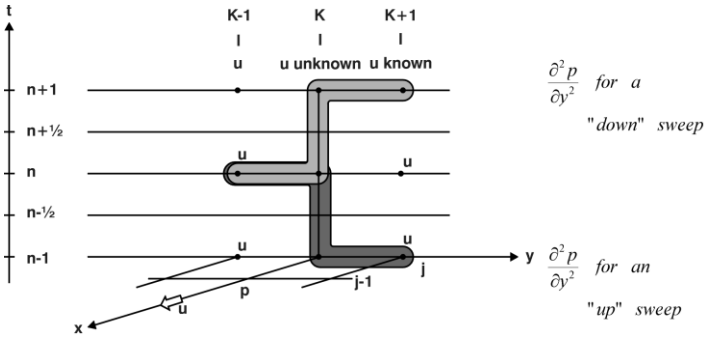

Figure 4.8 "Side-Feeding" for the 2nd-order Cross-Derivative Term

4.3.7 Wind friction term

The wind friction term reads

$$f(v) \cdot V \cdot V_x \tag{4.32}$$

where all variables are known in each grid point. The wind friction factor is calculated in accordance with Smith and Banke (Ref. /16/), see Figure 4.9.

$$f(v) = \begin{cases} f_0 & for\ V < V_0 \\ f_0 + \dfrac{V - V_0}{V_1 - V_0} \cdot (f_1 - f_0) & for\ V_0 \le V \le V_1 \\ f_1 & for\ V > V_1 \end{cases} \tag{4.33}$$

where

$$\begin{aligned} f_0 &= 0.00063 \quad , V_0 = 0 \ m/s \\ f_1 &= 0.0026 \quad , V_1 = 30 \ m/s \end{aligned} \tag{4.34}$$

If the area represented by grid point (*j*,*k*) has been specified to be covered by ice, *f(V)* is set to zero.

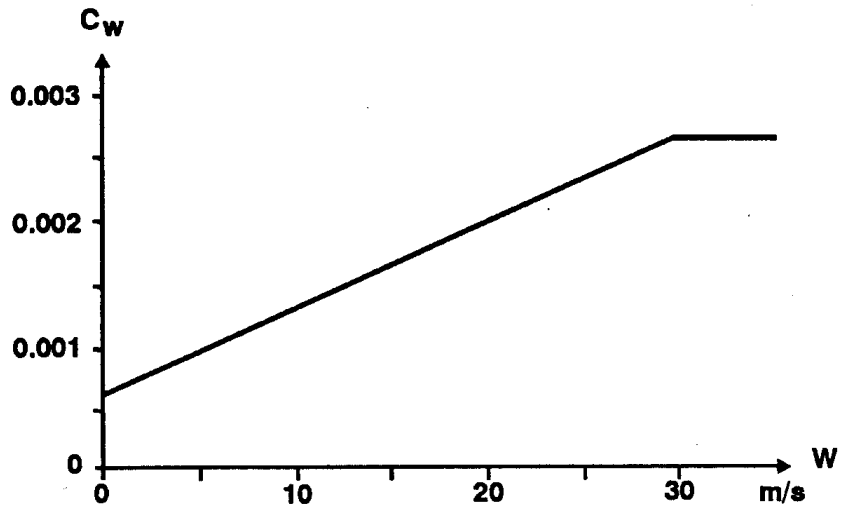

Figure 4.9 Wind Friction Factor

4.3.8 Resistance term

The bed shear stress is represented by the Chezy formulation,

$$\frac{gp\sqrt{p^2+q^2}}{C^2h^2} \tag{4.35}$$

which is approximated as

$$\frac{gp_{j,k}^{n+1}\sqrt{p^{*2}+q^{*2}}}{C^2h^{*2}} \tag{4.36}$$

where

$$\begin{aligned} p^* &= p_{j,k}^{n} \\ q^* &= \frac{1}{8}\,(q_{j,k}^{n-½} + q_{j+1,k}^{n-½} + q_{j,k-1}^{n-½} + q_{j+1,k-1}^{n-½} \\ &\qquad + q_{j,k}^{n+½} + q_{j+1,k}^{n+½} + q_{j,k-1}^{n+½} + q_{j+1,k-1}^{n+½}) \\ h^* &= \begin{cases} h_{j,k}^{n} & for \quad p^* \geq 0 \\ h_{j-1,k}^{n} & for \quad p^* < 0 \end{cases} \end{aligned} \tag{4.37}$$

Up-winding of the water depth used in the friction term was introduced in release 2001, and appeared to overcome some problems associated with previous versions of MIKE 21 flood and dry scheme. With this approach, the friction for flow from a deep grid point to a shallow grid point is calculated on the basis of the water depth in the deep grid point. That is, $h^* = h_{deep}$. Conversely, the friction for flow from a shallow grid point to a deep grid point is calculated on the basis of the water depth in the shallow grid point. That is, $h^* = h_{shallow}$. This makes it relatively easier for water to flow into a shallow grid point, and more difficult for it to flow out. Intuitively, this was considered to be a more physically realistic approach.

The Chezy number, C, is computed from the Manning number, M, as follows:

$$C = M \cdot h^{*1/6} \tag{4.38}$$

The Chezy number, C, is computed from the wave-induced bed resistance as follows:

$$C = \sqrt{g} \cdot \frac{u}{u_{fc}} \tag{4.39}$$

Where g is gravity, u is flow velocity and u_{fc} is the friction velocity calculated by considering the conditions in the wave boundary layer. For a detailed description of the wave induced bed resistance, see Fredsøe (1984), Ref. /6/ and Jones et. al. (2014), Ref. /7/.

4.3.9 Coriolis term

This term

$$\Omega \cdot q \tag{4.40}$$

is approximated explicitly by using q^* as defined in (4.37).

5 Special Difference Approximations for Points near a Coast

The cross-derivatives in the hydrodynamic equations pose a problem when the computational sweep passes near land. Clearly, concepts such as side-feeding become difficult to use. Inaccuracies, asymmetric behaviour between the "up" sweep and the "down" sweep may, especially at corners, create instabilities.

Land boundaries are defined at flux points, with the flux away from the land boundary set to zero. If for the purpose of this discussion, we consider an X-sweep, one can define the three principal situations given in Figure 5.1 to Figure 5.3 below as Case 1, 2 and 3. They are here shown at the "positive" or "north" side of the sweep but have, of course, their counter parts on the negative side. The principal situations can combine to create situations as shown, for example, in Figure 5.1 to Figure 5.3. In fact there are 15 possible combinations. The various situations are identified through a grid code or a combination of grid codes. The difference formulations along a land boundary when it is at an angle to the grid (Figure 5.5) are especially demanding.

In the following we shall show possible approximations for the principal cases of Figure 5.1 to Figure 5.3. The approximations for the other combinations are based on the same principles.

The terms that involve cross-derivatives are - considering an X-sweep - the $\partial q/\partial y$ term in the mass equations, the cross-momentum equation with associated correction term, the eddy viscosity term expressed in combination with this correction term and the cross-gravity term. The $\partial q/\partial y$ term of the mass equation offers no problems as this term is implicitly described in the definition of the land boundary. The other terms will be considered one by one.

5.1 Cross-Momentum Term - without Correction

Consider the general form (4.28) for a "down" sweep

$$\frac{\partial}{\partial y}\left(\frac{pq}{h}\right) \approx \left[\left(\frac{p_{k+1}^{n+1}+p_k^n}{2}\right)_j \cdot v_{j+½,k+½}^{n+½} - \left(\frac{p_{k+1}^{n+1}+p_k^n}{2}\right)_j \cdot v_{j+½,k-½}^{n+½}\right] \cdot \frac{1}{\Delta y} \tag{5.1}$$

with

$$v_{j+½,k+1}^{n+½} = \frac{\frac{1}{2}\left(q_j + q_{j+1}\right)_k^{n+½}}{\frac{1}{4}\left(h_{j,k} + h_{j,k+1} + h_{j+1,k} + h_{j+1,k+1}\right)^n} \tag{5.2}$$

CASE 1: Land to the "North"

In general we shall assume a reflection condition for p. That is p_{k+1} is assumed to be equal to p_k. We assume a flow situation as shown in Figure 5.1.

There is, in fact, no obvious reason for this assumption to be more correct than, for example, the assumption of a distribution as given in Figure 5.2. (We should, however, not be tempted to think of a distribution connected with a "no-slip" boundary condition. In the spatial description that we are dealing with here - Δx, Δy several tenths or hundreds of meters -such a condition is not resolved). However, the distribution of Figure 5.2 would generally give a greater gradient. We have preferred the distribution of Figure 5.1 as it gives a smaller value. The assumptions must be kept in mind in applications where $\partial p/\partial y$ becomes important at the land boundary.

For Case 1 the assumption, however, does not matter. The general form of (5.1) reduces to a reasonable approximation because $v_{j+½,k+½} = 0$.

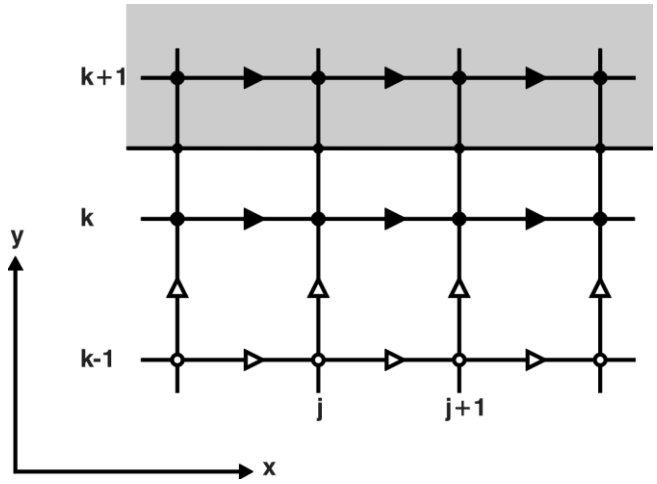

Figure 5.1 Special Situations near Land. Land to the "North" (CASE 1)

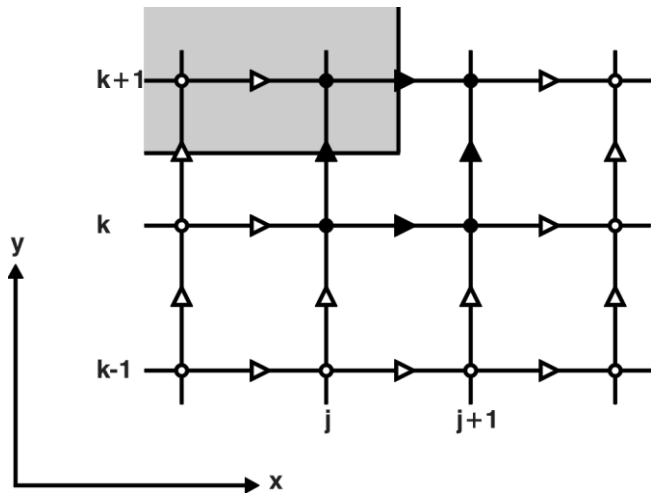

Figure 5.2 Special Situations near Land. Corner - Exit (CASE 2)

CASE 2: Corner - Exit

For *p* the reflection condition is used. The approximation of $v_{j+½,k+½}$ is more difficult. Experience from the regular grid has shown the following assumptions to give good results in general.

$$h_{j,k+1} \approx h_{j,k} \tag{5.3}$$

With this assumption $v_{j+½,k+½}$ can be approximated by the general formula (5.2).

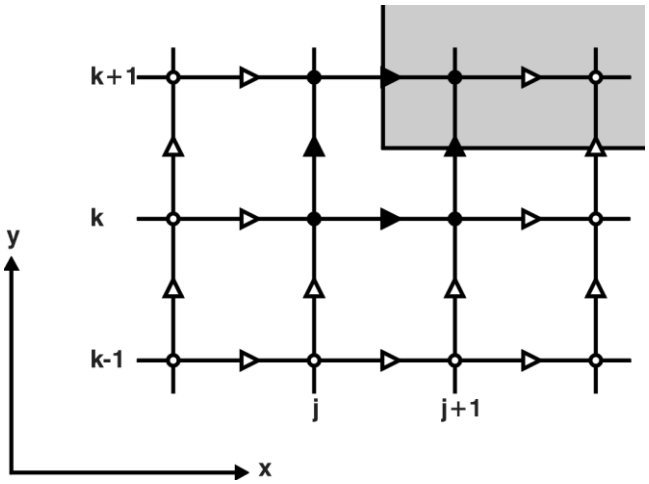

Figure 5.3 Special Situations near Land. Corner - Entry (CASE 3)

CASE 3: Corner - Entry

Similar assumptions to those in Case 2 give reasonable approximations.

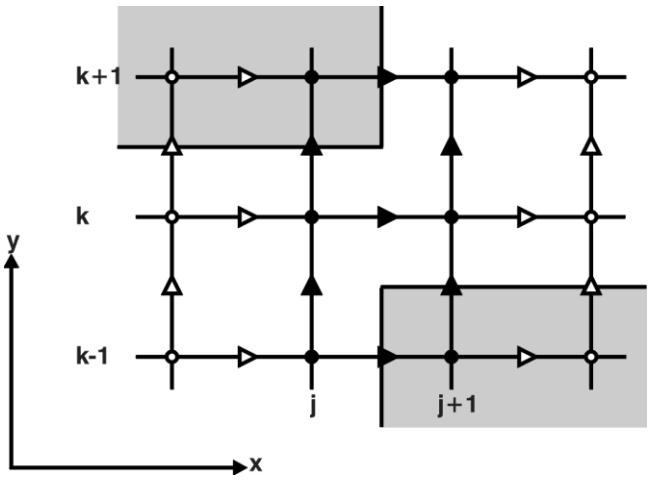

Figure 5.4 Possible Corner Combination (which should be avoided)

5.2 Cross-Momentum Correction and Eddy Viscosity Term

The correction term and eddy viscosity term (using a constant eddy description) require an approximation for $\partial^2 p/\partial y^2$. Consider the general form of the correction term in (4.27) and introduce an additional eddy viscosity by adding a constant coefficient η. We have

$$\left(\eta + \frac{1}{2}\cdot\bar{v}^2\cdot\Delta t\right)\frac{\partial^2 p}{\partial y^2} \approx \left(\eta + \frac{1}{2}\cdot v*^2\cdot\Delta t\right) \cdot\left\{\frac{p_{k+1}^{n+1} - \left(p_k^{n+1} + p_k^n\right) + p_{k-1}^n}{(\Delta y)^2}\right\}_j \tag{5.4}$$

The 2nd - derivative term is approximated using the reflection condition for *p*. For Case 1, *v = 0*, so that the general form provides an automatic approximation. For Case 2, *v** can be approximated as in Case 2 of Section 5.1 above. For Case 3 a similar approximation can be applied.

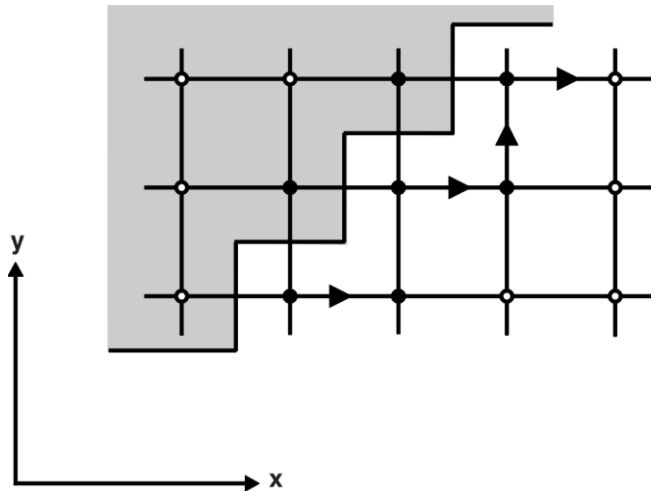

Figure 5.5 Coastline 45º to the grid

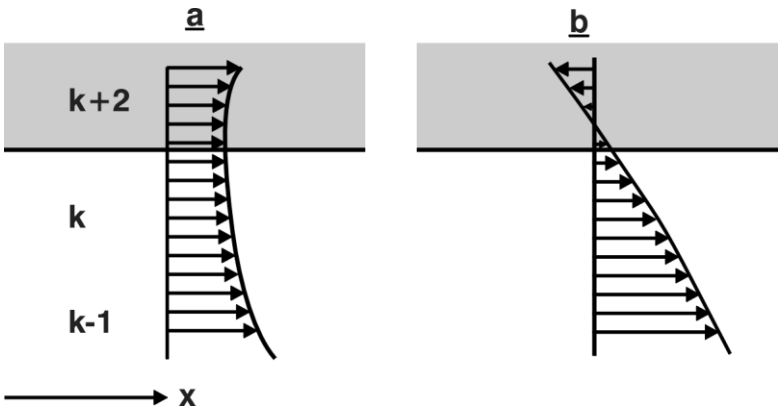

Figure 5.6 Possible Velocity Distributions

6 Structure of the Difference Scheme, Accuracy, Stability

6.1 Time Centring, Accuracy

The difference schemes developed in the previous section must be seen as one component in a computational cycle. Only together with the other component equations in this cycle is time centring obtained. In a simplified, schematic, form we have

$x-mass:$

$$\frac{\zeta^{n+½}-\zeta^{n}}{½\Delta t}+\ldots+\frac{1}{2}\Lambda_x^{n+1}p+\frac{1}{2}\Lambda_x^{n}p \tag{6.1}$$

$$+\frac{1}{2}\Lambda_y^{n+½}q+\frac{1}{2}\Lambda_y^{n-½}q=F_x(p,q)$$

$x-momentum:$

$$\frac{p^{n+1}-p^{n}}{\Delta t}+\ldots+gh\left(\frac{1}{2}\Lambda_y^{n+½}\zeta+\frac{1}{2}\Lambda_y^{n}\ldots\zeta\right)=G_x(p,q) \tag{6.2}$$

$y-mass:$

$$\frac{\zeta^{n+1}-\zeta^{n+½}}{½\Delta t}+\ldots+\frac{1}{2}\Lambda_x^{n+1}p+\frac{1}{2}\Lambda_x^{n}p \tag{6.3}$$

$$+\frac{1}{2}\Lambda_y^{n+3/2}q+\frac{1}{2}\Lambda_y^{n+½}q=F_y(p,q)$$

$y-momentum:$

$$\frac{q^{n+3/2}-q^{n+½}}{\Delta t}+\ldots+gh\left(\frac{1}{2}\Lambda_y^{n+1}\zeta+\frac{1}{2}\Lambda_x^{n}\ldots\zeta\right)=G_y(p,q) \tag{6.4}$$

where the operator Λ_x indicates a difference form, typically as in

$$\Lambda_x^{n+1}p=\frac{(p_j-p_{j-1})^{n+1}}{\Delta x} \tag{6.5}$$

(Our operation notation in the above equations is not meant to be rigorously correct. The idea in the schematic form is only to stress the time stepping structure).

Now the way in which the above component equations are coupled in time is shown in the computational cycle in Figure 6.1.

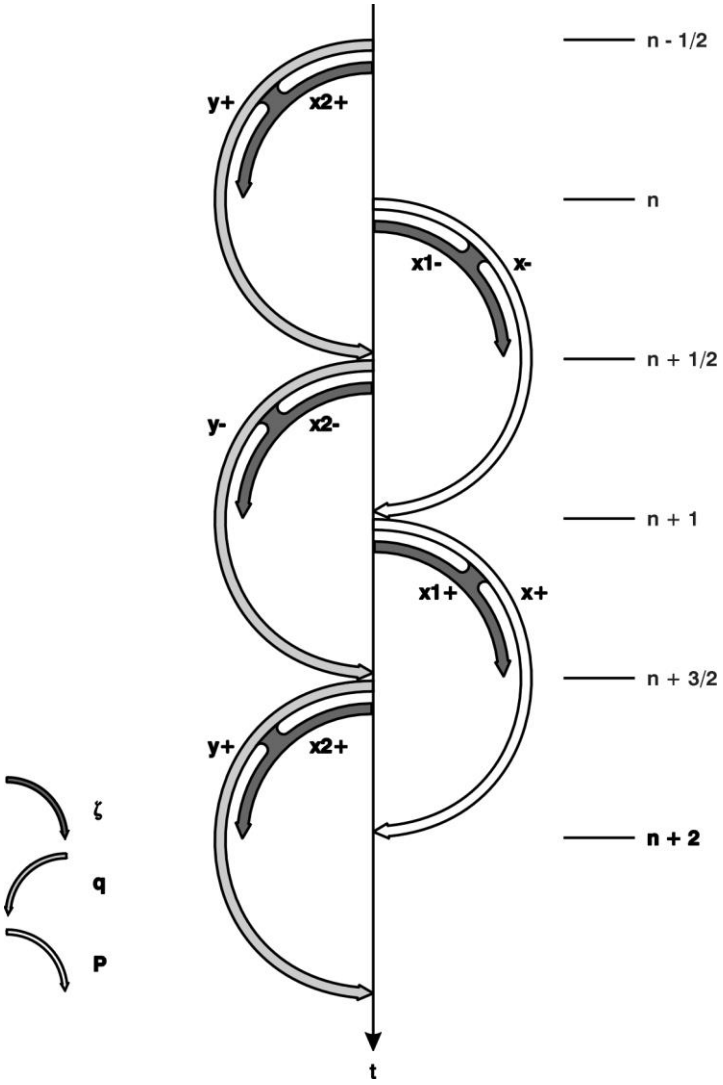

Figure 6.1 Computational Cycle of MIKE 21 HD

Referring to the computational cycle we can now discuss the centring of the various terms in the component equations. Consider the (x-) sweep. Its centre is at *n+½.* This is clear for the $\Lambda_x p$ term in the mass equation and the time derivative in the momentum equation. For the time derivative of ζ, the centring is not obvious. The (x-) sweep alone will not give a centre at *n+½.* The mass equation of the following (y-) sweep has to be involved to provide the centring at *n+½*.

The gravity term in the x-momentum equation $\Lambda_x \zeta$ is correctly centred at *n+½*.

The spatial derivative for *q* in the mass equations may at first hand appear peculiar. If the centre of the (x-) sweep is at n+½, then why not only use $\Lambda(n+½ / y)q$? The explanation lies in the next (y-) sweep. This sweep has its centre at *n+1* and the mass equation therefore has $\Lambda(n+^3/_2 / y)q$ and $\Lambda(n+½ / y)q$. Then, when the mass equation of the (y-)

sweep is considered together with the mass equation of the (x-) sweep, the $\Lambda(n-\frac{1}{2} / y)q$ in the (x-) mass equation is needed to balance the $\Lambda(n+\frac{3}{2} / y)q$ in the (y-) mass equation.

The considerations for the (x-) sweep above can be repeated in a similar manner for the (x+), (y-) and (y+) sweeps.

The open computational cycle of Figure 6.1 is a development of the closed computational cycle employed in an earlier version of MIKE 21 HD. Figure 6.2 shows its structure. This cycle is described in Abbott, Damsgaard and Rodenhuis (1973), Ref. /1/, and in Abbott (1979), Ref./3/. Other implicit difference schemes, for example that of Leendertse (1967), Ref. /9/, are usually based on a closed cycle of similar form. Stability and time centring in such closed cycles is then viewed in terms of a 1 dimensional descent. The x-mass and x-momentum equations, combined in a certain x-sweep, are balanced by the x-mass and momentum equations in a following complimentary x-sweep. Consider, for example, the computational cycle of Figure 6.2. The order of the sweeps is:

x	x-sweep, carried out with decreasing y
y	y-sweep, carried out with decreasing x
y+	y-sweep, carried out with increasing x
x+	x-sweep, carried out with increasing y

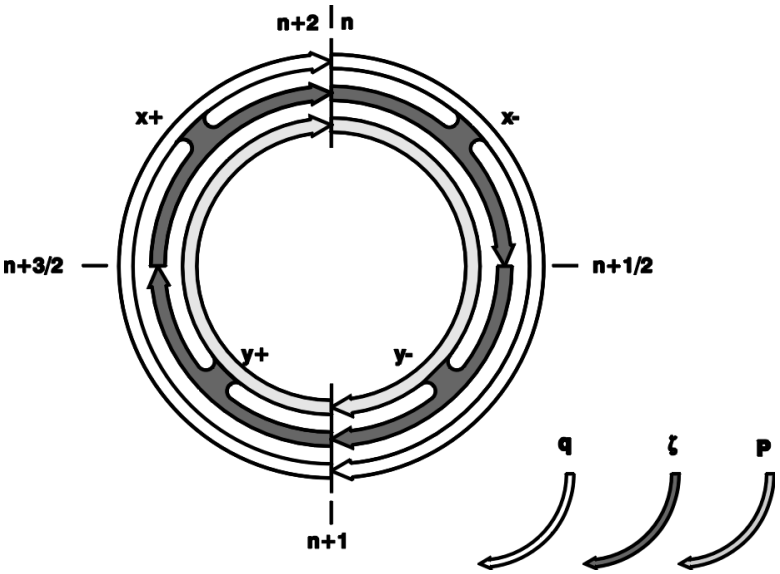

Figure 6.2 Computational Cycle of System 21 Mark 2, an early version of MIKE 21

One observes for terms involving ζ that the (x-) sweep together with the following (x+) sweep provides a centring at *n+1*. The $\partial q/\partial y$ term in the mass equation can only be approximated at the time level in the (x-) sweep, but is centred at n+1 by the $\partial q/\partial y$ term in the mass equation of the later (x+) sweep. However, this centring takes place two "y-sweeps" later and the solution may drift too far "off-centre" to be fully corrected. In that respect the open cycle is an improvement, the correction being provided by the sweep immediately following. The open cycle provides a further simplification in that the (x-) and (x+) sweeps are completely identical apart from the way they are carried out. Instead of 8 component equations - 2 per sweep - in the closed cycle of Figure 6.2, we now have 4 component equations.

The difference scheme, by nature of its central difference forms, is generally of second order. It is second order in terms of the discretisation of the Taylor series expansion, as

well as in the more classical sense, that of the order of the algorithm. This last concept is defined as the highest degree of a polynomial for which the algorithm is exact. The two definitions are often confused, but they do not necessarily always give the same order of accuracy. For the Laplace equation the usual central difference approximation is of second order in terms of the discretisation error but the algorithm is of third order. See Leonard (1979), Ref. /10/.

6.2 Amplification Errors and Phase Errors

6.2.1 General

The behaviour of a difference scheme can be conveniently expressed through amplification portraits and phase portraits. For an earlier version of MIKE 21 HD and the System 21 Mark 6 (and other schemes of this type), such portraits have been derived in Abbott, McCowan and Warren (1981), Ref. /4/. (The System 21 Mark 6 difference scheme is similar to the one used in MIKE 21 HD, but without higher-order correction terms. Furthermore, the $\triangle x$ and $\triangle y$ used in the equations are the distance between a water level point and a flux point, not between two water level points as in MIKE 21 HD). In order to be able to express fully the properties of the scheme with respect to time centring it was found convenient to reduce the scheme to an equivalent 2-level form through a sequence of substitutions. All dependent variables at "half" time levels are written at levels *n+1* and n. The resulting scheme is equivalent for the purpose of amplification and phase error analysis, but is algorithmically intractable. The equations are further reduced to principal form by linearisation. Convective terms, resistance, Coriolis and wind stress terms are all excluded. We will here summarise the main results of the analysis.

6.2.2 Amplification factors and phase portraits of System 21 Mark 6

For the equations in 2-level form a Fourier transform is obtained through the introduction of Fourier series of the following form

$$f_{j,k}^{n} = \sum_{m} f^{*}(m)\ e^{i(\sigma_1 j\Delta + \sigma_2 k\Delta y)} \tag{6.6}$$

with

$$\sigma_1 = \frac{2\pi m}{2L_1} \quad , \quad \sigma_2 = \frac{2\pi m}{2L_2} \tag{6.7}$$

L_1 and L_2 are a characteristic length in the x and y directions respectively and m is the wave number. L_1, L_2 and m are usually so defined that $J\triangle x = L_1$, $K\triangle y = L_2$, and J and K are the number of grid points in the x and y directions respectively. Then, at $m = 1$, L_1 is the half wave length over the total extent in the x-direction, L_2 the half wave length in the y-direction. One may further define the numbers

$$N_1 = \frac{2L_1}{m\Delta x} \quad , \quad N_2 = \frac{2L_2}{m\Delta y} \tag{6.8}$$

to denote the number of grid points per wave length for a certain wave component m.

We introduce the amplification factor ⊥, so that

$$f_{j,k}^{n+1} = \varphi \; f_{j,k}^{n} \tag{6.9}$$

Since the equations are linearised and as ζ, *p* and *q* are all coupled through the hydrodynamic equations it is sufficient to analyse the amplification and phase error for one and the same component m in the Fourier series.

The Fourier transform of System 21 Mark 6 is then

$$\begin{bmatrix} \varphi - 1 & \left(\frac{gh\Delta t}{4\Delta x}\right)(2i\sin\sigma_1\Delta x)(\varphi+1) - \left(\frac{(gh)^2\Delta t^3}{64\Delta y^2\Delta x}\right)(-4\sin^2\sigma_2\Delta y) & 0 \\ \frac{\Delta t}{4\Delta x}(2i\sin\sigma_1\Delta x)(\sigma+1) & (\varphi-1)+\frac{gh\Delta t^2}{16\Delta y^2}\cdot(-4\sin^2\sigma_2\Delta y)(\varphi+1) & \frac{\Delta t}{4\Delta y}(2i\sin\sigma_2\Delta y)\cdot(\varphi+1) \\ 0 & \frac{gh\Delta t}{2\Delta y}(2i\sin\sigma_2\Delta y)\varphi & (\varphi-1) \end{bmatrix} \begin{bmatrix} p^* \\ \zeta^* \\ q^* \end{bmatrix} = 0 \tag{6.10}$$

Setting

$$\frac{gh\Delta t^2}{\Delta x^2}\sin^2\sigma_1\Delta x = C_{r1}^2\sin^2\sigma_1\Delta x_t = \alpha^2 \tag{6.11}$$

$$\frac{gh\Delta t^2}{\Delta y^2}\sin^2\sigma_2\Delta y = C_{r2}^2\sin^2\sigma_2\Delta x_j = \beta^2 \tag{6.12}$$

with *Cr1* and *Cr2* the Courant numbers in the x and y-directions respectively, and

$$A^2 = \frac{\alpha^2}{4} + \frac{\alpha^2\beta^2}{16} + \frac{\beta^2}{4} \tag{6.13}$$

We find, from the condition that the determinant in (6.10) shall be zero,

$$A^2 = \frac{\alpha^2}{4} + \frac{\alpha^2\beta^2}{16} + \frac{\beta^2}{4} \tag{6.14}$$

giving,

$$\varphi^2 = \frac{(A^2-1)}{(A^2+1)} \pm \sqrt{\left(\frac{A^2-1}{A^2+1}\right)^2 - 1} \tag{6.15}$$

It then follows that $|\perp|$ = 1, since A is always real. That is the amplification factor is 1 for all combinations of model parameters. In fact it can be shown that the class of models built upon time-centred implicit difference schemes and all schemes of this class have amplification factors equal to 1, Vreugdenhil (1966), Ref. /17/.

The phase portrait follows from the ratio between the numerical and physical celerity, and is

$$Q = -\frac{\arctan\frac{\mathrm{Im}(\varphi)}{\mathrm{Re}(\varphi)}}{\frac{2\pi Cr}{N}} \tag{6.16}$$

With (6.15) we have

$$\frac{\mathrm{Im}(\varphi)}{\mathrm{Re}(\varphi)} = \pm\frac{i2A}{1-A^2} \tag{6.17}$$

This gives the phase portraits of Figure 6.3, which are, in fact, the phase portraits of all schemes of the class of time-centred implicit models. The relation for A, (6.13) can be written for propagation along grid lines or for propagation at an angle to the grid lines and this is shown in Figure 6.3 for an angle of 45º. The celerity ratios for all other angles are bounded by the graphs for propagation along grid lines and at 45º. One may observe that for tidal problems, where N can be expected to be large, the phase error can be expected to be small, even for large Courant numbers.

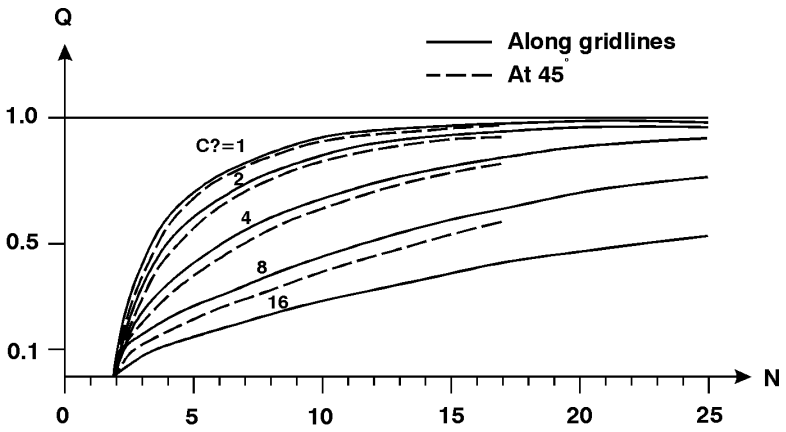

Figure 6.3 Phase Portraits of System 21 Mark 6

7 Boundary Conditions

7.1 General

The main purpose of MIKE 21 HD is to solve the partial differential equations that govern nearly-horizontal flow. Like all other differential equations they need boundary conditions. The importance of boundary conditions cannot be over-stressed.

In general the following boundary data are needed:

- Surface levels at the open boundaries and flux densities parallel to the open boundaries
  or
  Flux densities both perpendicular and parallel to the open boundaries
- Bathymetry (depths and land boundaries)
- Bed resistance
- Wind speed, direction and sheer coefficient
- Barometric pressure (gradients)

The success of a particular application of MIKE 21 HD is dependent upon a proper choice of open boundaries more than on anything else. The factors influencing the choice of open boundaries can roughly be divided into two groups, namely

- Grid-derived considerations
- Physical considerations

The physical considerations concern the area to be modelled and the most reasonable orientation of the grid to fit the data available and will not be discussed further here.

The grid itself implies that the open boundaries must be positioned parallel to one of the coordinate axes. (This is not a fundamental property of a finite difference scheme but it is essential when using MIKE 21 HD).

Furthermore, the best results can be expected when the flow is approximately perpendicular to the boundary. This requirement may already be in contradiction with the above mentioned grid requirements, and may also be in contradiction with "nature" in the sense that flow directions at the boundary can be highly variable so that, for instance "360" flow directions occur, in which case the boundary is a most unfortunate choice.

7.2 Primary Open Boundary Conditions

The primary boundary conditions can be defined as the boundary conditions sufficient and necessary to solve the linearised equations. The fully linearised x-momentum equation reads:

$$\frac{\partial p}{\partial t} + gh\frac{\partial \zeta}{\partial x} = 0 \tag{7.1}$$

The corresponding terms in the x-momentum equation of MIKE 21 HD are:

$$\frac{\partial p}{\partial t} + \ldots + gh\frac{\partial \zeta}{\partial x} + \ldots = 0 \tag{7.2}$$

A "**dynamic case**" we define as a case where

$$\frac{\partial p}{\partial t} \approx -gh\frac{\partial \zeta}{\partial x} \tag{7.3}$$

i.e. a case where these two terms dominate over all other terms of the MIKE 21 HD x-momentum equation.

It is then clear that the primary boundary conditions provide "almost all" the boundary information necessary for MIKE 21 HD when it is applied to a dynamic case. The same set of boundary conditions maintain the dominant influence (but are in themselves not sufficient) even in the opposite of the "dynamic case", namely the steady state (where the linearised equations are quite meaningless). This explains why these boundary conditions are called "primary".

MIKE 21 HD accepts two types of primary boundary conditions:

- Surface elevations
- Flux densities

They must be given at all boundary points and at all time steps.

It should be mentioned that - due to the space staggered scheme -the values of the flux densities at the boundary are set half a grid point **inside** the topographical boundary, see Figure 7.1.

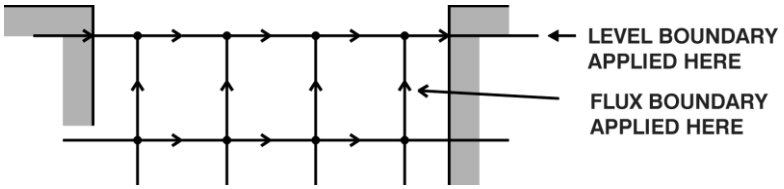

Figure 7.1 Application of boundary Data at a Northern Boundary

7.3 Secondary Open Boundary Conditions

7.3.1 General

The necessity for secondary boundary conditions arises because one cannot close the solutions algorithm at **open** boundaries when using the non-linearised equations. Additional information has to be given and there are several ways to give this. MIKE 21 HD is built on the premise that the information missing is **the discharge or flux density parallel to the open boundary**.

This is chosen because it coincides conveniently with the fact that the simplified MIKE 21 HD - the model that is one-dimensional in space - does not require a secondary boundary condition (i.e. the discharge parallel to the boundary is zero).

As a consequence of the transport character of the convective terms, a "true" secondary condition is needed at inflows, whereas at outflow a "harmless" closing of the algorithm is required. This closing may either be obtained by defining the flow direction at the boundary or by extrapolation of the flux along the boundary from the inside. Furthermore, the fluxes outside the boundaries are needed (for the convective momentum term, the eddy term and the non-linear dissipation term).

7.3.2 Fluxes along the boundary

As described in the previous section, the secondary boundary information has been defined as the Flux Along the Boundary, the FAB.

There are four FAB types implemented in MIKE 21 HD:

FAB type	MIKE 21 action
0	FAB is 0 at all boundary points at all times
1	FAB is obtained by extrapolation mainly in space
2	Flow direction is given whereby FAB can be computed internally in MIKE 21 HD
12	Chooses FAB type = 1 at an outflow and FAB type = 2 at an inflow

The **only** possible FAB type for Flux-boundaries.

FAB TYPE = 0

In this case the FAB will remain 0 at all boundary points during the whole simulation.

Though it appears as a simplification of both FAB type 1 and FAB type 2, it is maintained because it is so simple - both for the user and for MIKE 21 HD.

The physical meaning of FAB type 0 is that one-dimensional behaviour is enforced in the boundary region.

For a two-dimensional model this is principally acceptable only for inflow boundaries, implying that FAB type 0 is a secondary boundary condition typically connected to inflow.

FAB TYPE = 1

This represents extrapolation.

In reality extrapolation gives dummy information and, accordingly, FAB type 1 is meant for outflow boundaries where principally only the primary information is required.

Further, a satisfactory result is often achieved in dynamic simulations (where inflow and outflow replace each other frequently) with FAB type 1.

FAB TYPE = 1 is the default for level-boundaries.

The actual extrapolation is guided by the system parameter

FABD3

where D3 stands for "the Degree of the 3rd derivative".

The FAB is then obtained from the finite difference approximation to the equation, say,

$$\left(\frac{\partial^2 p}{\partial y^2}\right)^{n+1} = FABD3 \left(\frac{\partial^2 p}{\partial y^2}\right)^{n} \tag{7.4}$$

The terms are centred one grid point inside the boundary and new FABs are only computed every second time step (when the sweep direction is towards the boundary). The actual flux along the boundary may therefore - at instants of rapid change - appear rather different from extrapolated values.

When extrapolated values have been obtained according to the formula given above, they are damped and smoothed, i.e. multiplied by

FABDAMP

and smoothed according to the formula:

$$P(j) = FABDISP \cdot P(j-1) + P(j+1) + (1 - 2\,FABDISP) \cdot P(j) \tag{7.5}$$

where j denotes position along the boundary.

The value FABDISP = 0.25 gives maximum smoothing whilst instability occurs if FABDISP > 0.5.

The applied values are

FABD3 = 5; FABDAMP = .99; FABDISP = .05;

corresponding to the proper use of FAB type 1, i.e. for use at outflows. If FABDAMP = 0 then FAB type 1 becomes identical to FAB type 0.

FAB TYPE = 2

By setting a FAB type to 2 the flow **direction** at this boundary is specified, whereafter MIKE 21 HD can compute the FABs. FAB type 2 is typically connected to inflows.

The computation of the FABs is semi-centred in time in the sense that they are actually obtained as the solution to the equation, say

$$FABFW \cdot P + (1 - FABFW) \cdot OLDP = Q \cdot dir \tag{7.6}$$

where FW stands for the Weight on the Front. Thus, the new FABs are explicitly computed and the above given formula becomes time centred at the same time as it reaches its stability limit, namely for FABFW = 0.5. It is, however, recommended not to go below the default of FABFW = 0.6.

When the FABs have been computed, they are smoothed in a similar manner to that described in Equation (7.5), the degree of smoothing being described by FABDISPDIR for FAB type 2 boundaries.

The default is that the flow is at right angle to the boundary, or in other words, the default FAB type 2 is identical to FAB type 0.

FAB TYPE = 12

FAB type 12 meets the theoretical requirements for the two-dimensional, nearly horizontal flow equations.

The number 12 is a code for "1 or 2", and if the FAB type is 12 then MIKE 21 HD simply selects either FAB type 1 or FAB type 2. In order to do this, MIKE 21 HD checks on the total flow through the boundary and, if there is **inflow it uses FAB type 2**, while if there is **outflow it uses FAB type 1**.

After having performed this choice, MIKE 21 HD obtains the FABs exactly as previously described for each of the two FAB types.

8 Flooding and Drying

8.1 General

Flooding and Drying basically is a functionality/set of rules for how the domain of calculation can be expanded or reduced as a function of time. The functionality basically works as a threshold mechanism which uses two user-defined threshold parameters:

- h_{flood} (flooding depth)
- h_{dry} (drying depth)

The following constraint applies: $h_{flood} > h_{dry}$.

These two parameters will regulate when a given cells should be exposed for a flooding or drying check during the simulation.

A wet/flooded cell is a cell, which in the result files will have a water depth greater than zero. A dry cell on the other hand is a cell which appears blank in the result files (delete value). This does however not mean that a dry cell does not contain any water - the water depth in this case is just not supposed to be represented according to the rules.

Apart from the above mentioned threshold parameters, the flooding and drying functionality is about bookkeeping of the water depths when cells goes from wet/flooded to dry and vice versa. This is handled with 2 independent data structures, which interacts at simulation runtime:

- h (water depths while cells are wet/flooded)
- h_flood_dry (water depths while cells are dry)

The fact that dry cells can contain a (non-zero) water depth is important to acknowledge when considering the conservation of mass for water in a simulation, even though the water depths in dry cells are not directly possible to inspect by the user.

Note: All elements in the data structure h_flood_dry have an initial value identical to *EPSF (= 0.0002 m), where EPSF is an internal engine parameter.*. This fact has an important implication: all dry cells in the initial condition will contain a non-zero water-volume even though it is not visible to the user. So when one uses the flooding and drying functionality, then one is implicitly accepting that there is a presence of a small and invisible carpet of water with depth equal to *EPSF* in all dry cells in the floodplain area. This means in practice that if a situation happens during runtime where one or more values in h_flood_dry goes below *EPSF,* then these values will be artificially corrected, and reset to *EPSF*. This is a known/intentional violation of the mass-balance, see also Section 8.5.

8.2 Flooding due to Accumulation of Water from External Sources

If the setup has flooding and drying enabled and has been specified with one, some or all of these features:

- MFLOOD links
- Sources
- Precipitation

Then there is basis for water accumulation in dry cells (if any).

More specifically the following will apply: if (*j,k*) represents a dry cell on the bathymetry, then *h_flood_dry(j,k)* will at all times store the accumulated water depth while the cell is dry.

Since the initial value of *h_flood_dry(j,k)* is *EPSF*, the accumulation can be written:

$$h\_flood\_dry(j,k) = EPSF + \sum \Delta h(j,k) \tag{8.1}$$

Where $\Delta h(j,k)$ is the net water increment per time step at cell (*j,k*) due to the presence of one, some or all of these features: external models (MFLOOD links), Source/Sinks and Precipitation/Evaporation.

This accumulation of *h_flood_dry(j,k)* will continue until it exceeds the specified flooding threshold h_{flood}. When this happens the value of *h_flood_dry(j,k)* will be copied to *h(j,k)* and the cell will go from dry to wet, see Figure 8.1.

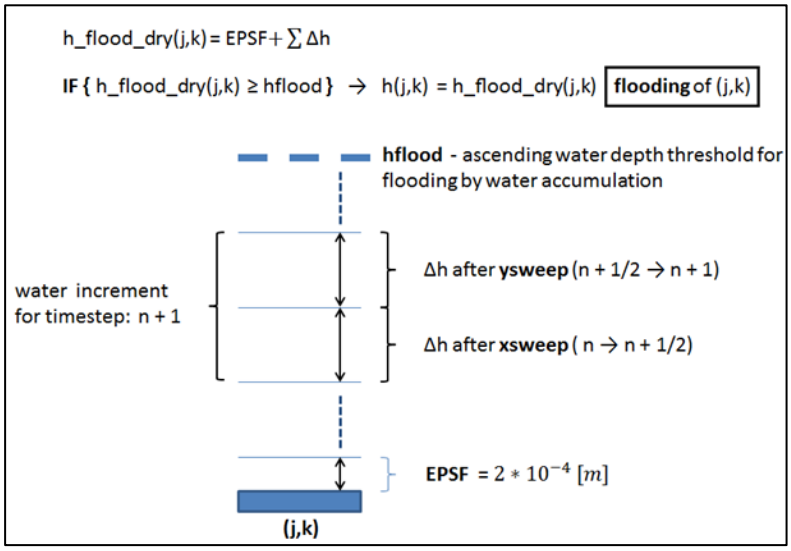

Figure 8.1 Flooding due to water from external sources

8.3 Flooding due to High Water Level in Neighbour-cells (Chain Flooding)

When wet/flooded cells are present at some time steps during the simulation, then the water level in these cells can make the dry neighbour cells flooded as well. The logic can be seen from Figure 8.2.

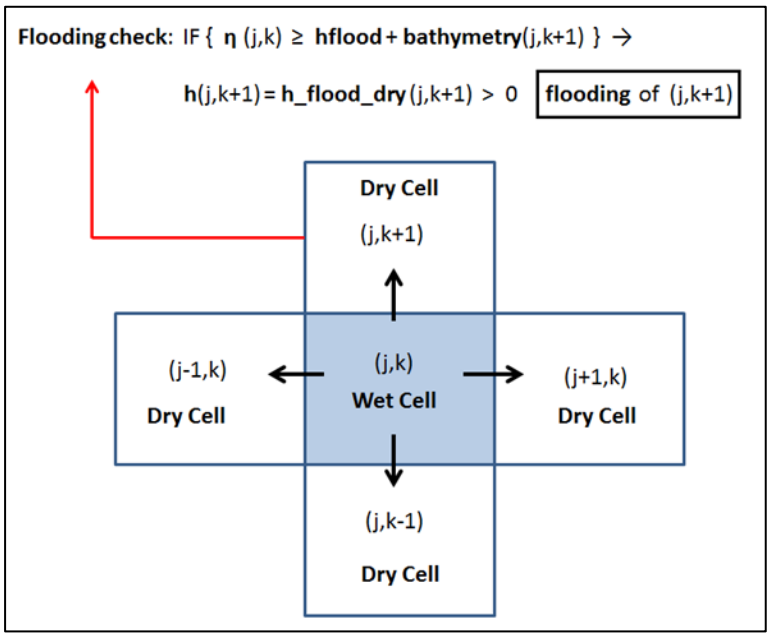

Figure 8.2 Flooding due to high water level in neighbour cells

A flooding check is being performed against all the dry neighbour cells relative to the wet/flooded cell *(j,k)*. The flooding check compares the surface elevation $\eta(j,k)$ at the wet cell up against the sum of the neighbouring bathymetry value at dry cell, bathymetry(j,k+1), and the flooding value h_{flood}. If η(j,k) becomes larger than or equal to h_{flood} + *bathymetry(j,k+1)*, the cell *(j,k+1)* will be flooded so that the water depth stored in *h_flood_dry(j,k+1)* will be copied to *h(j,k+1)*.

8.4 Drying

The drying functionality utilises the drying depth, h_{dry}, and the parameter *EPSH*, reflecting a very small water depth (= 0.0001 m). The drying logic is seen in Figure 8.3.

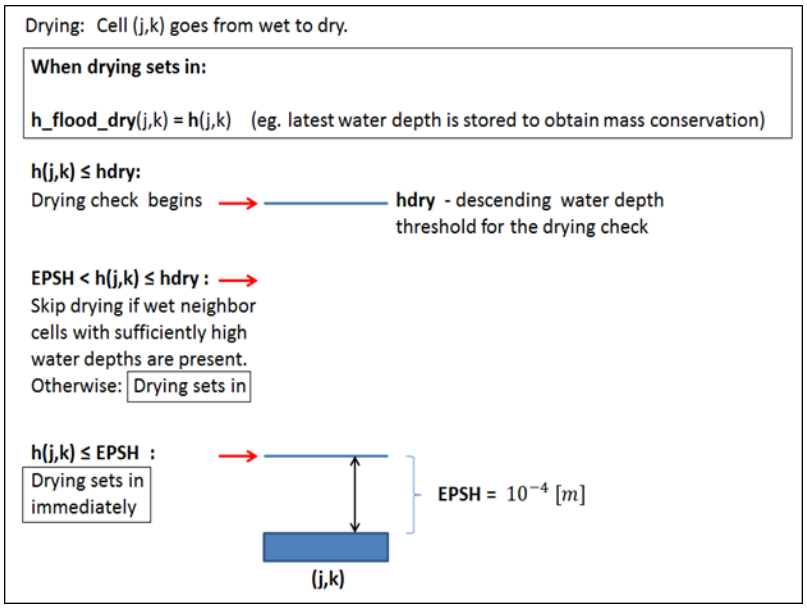

Figure 8.3 Drying functionality

The drying depth, h_{dry}, is used as a water depth threshold (descending) for when to start the drying check - e.g. start investigating if a cell should go from wet to dry. If the water depth at (j,k) is below *EPSH* the drying sets in immediately. However if the water depth at (j,k) is between *EPSH* and h_{dry} and the neighbour cells are sufficiently wet, the drying of cell (j,k) is skipped if just one of the following statements in Equation (8.2) is true:

$$\begin{aligned} &\text{bathymetry}\ (j,k) + h_{flood} \le \eta(j-1,k) \\ &\text{bathymetry}\ (j,k) + h_{flood} \le \eta(j+1,k) \\ &\text{bathymetry}\ (j,k) + h_{flood} \le \eta(j,k-1) \\ &\text{bathymetry}\ (j,k) + h_{flood} \le \eta(j,k+1) \end{aligned} \tag{8.2}$$

When drying sets in, the equations of motion will no longer apply at *(j,k)*, and *h(j,k)* is no longer updated, - instead the value of *h(i,j)* is copied into >*h_flood_dry(j,k)*.

8.5 Internal Engine Parameters

Apart from the parameters h_{flood} and h_{dry}, the MIKE 21 Flow Model also applies 2 internal control parameters to regulate a flooding/drying simulation.

These parameters are:

- *EPSF* (minimum water depth) the default value of this parameter is 0.0002 [m]
- *EPSH* (very small water depth) the default value of this parameter is 0.0001 [m]

The following constraint applies: *hdry > EPSF > EPSH*

These 2 parameters are declared with these default values inside the engine – they cannot be edited from the GUI by the user. It is however possible to modify the values by editing/overwriting their values under option parameters in the simulation file (.m21) as shown in Figure 8.4:

```
[HYDRODYNAMIC_MODULE]
   [OPTION_PARAMETERS]
      epsf = 1e-005
      epsh = 5e-006
   EndSect  // OPTION_PARAMETERS

   [OPEN_BOUNDARY]
   EndSect  // OPEN_BOUNDARY

   [SOURCE_AND_SINK]
      EvaporationType = -1
      Evaporation = 0
      [Evaporation]
         [DATA_FILE]
            FILE_NAME = ''
            ITEM_COUNT = 1
            ITEM_NUMBERS = 1
         EndSect  // DATA_FILE

      EndSect  // Evaporation
```

Figure 8.4 Extract of .m21 input file with manually reduced values of *EPSF* and *EPSH*

In some cases the mass-violation in a simulation can be reduced by overwriting the default values *EPSF* and *EPSH* with some lower values, as seen above. For example when a sink effect applies on a dry cell (*j,k*) and leads to a situation where the water depth goes below *EPSF*.

Figure 8.5 - Figure 8.8 illustrate the significance of *EPSF* and *EPSH* for some potential examples of flooding and drying scenarios. All the illustrations display potential cases of water depth evolution as a time series in a cell (*j,k*). A blue line indicates that the cell is wet and a red line indicates that the cell is dry.

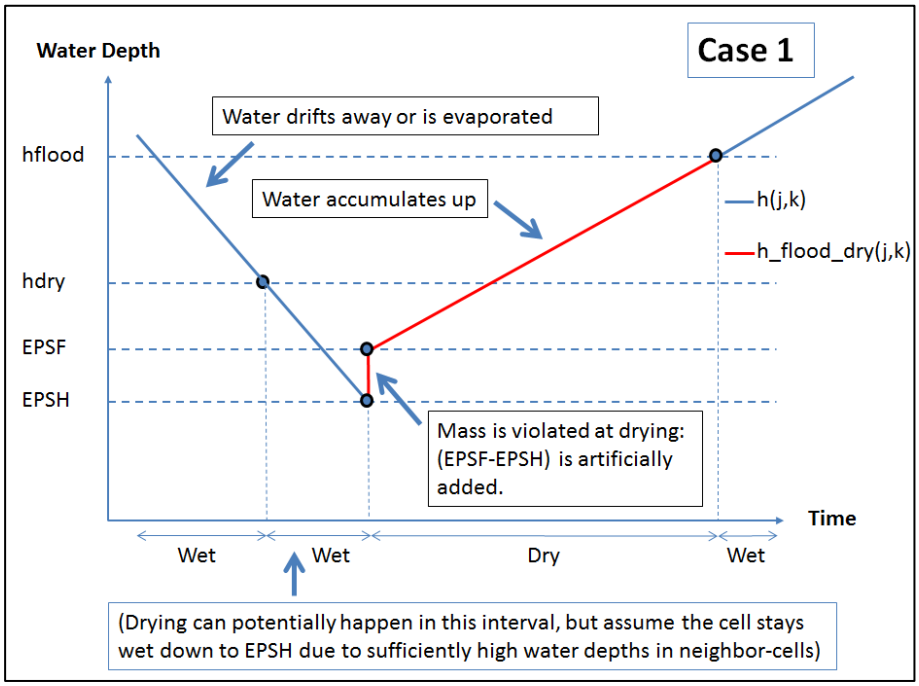

Figure 8.5 Case 1: Water depth drops to *EPSH*

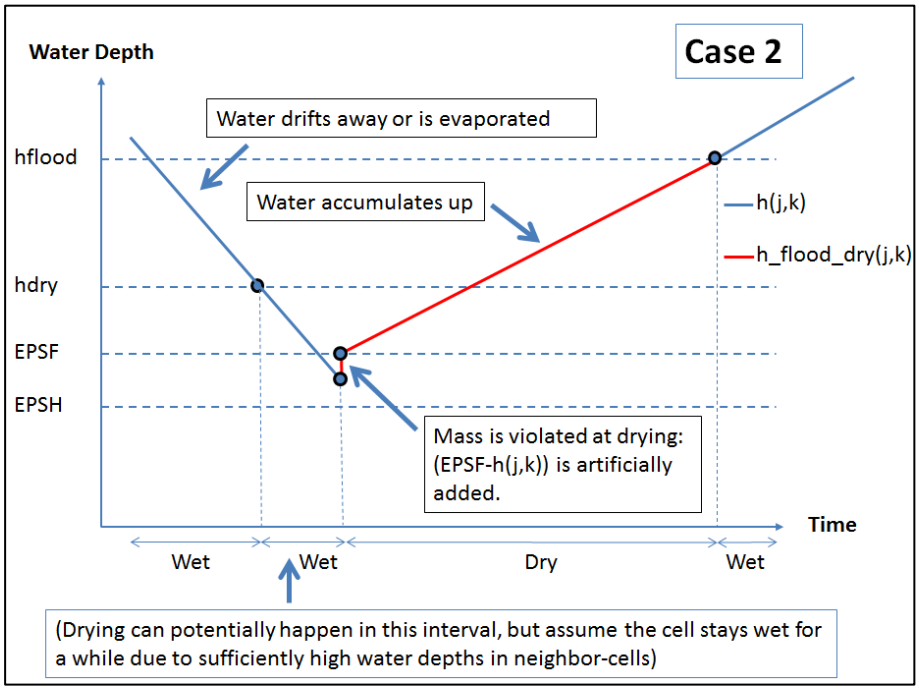

Figure 8.6 Case 2: Water depth drops below *EPSF*, but above *EPSH*

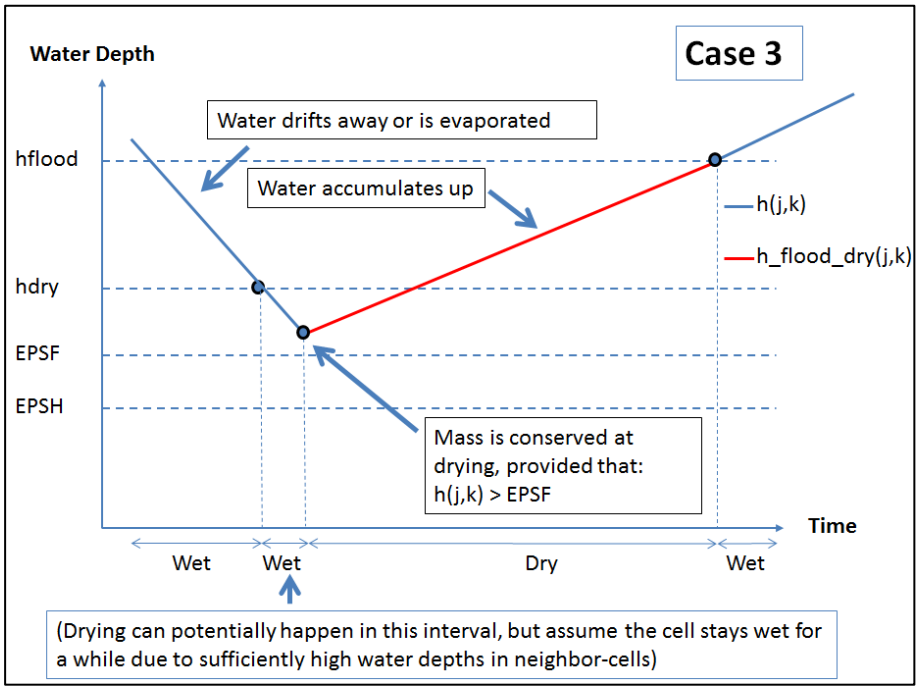

Figure 8.7 Case 3: Water depth drops below h_{dry} but above *EPSF*

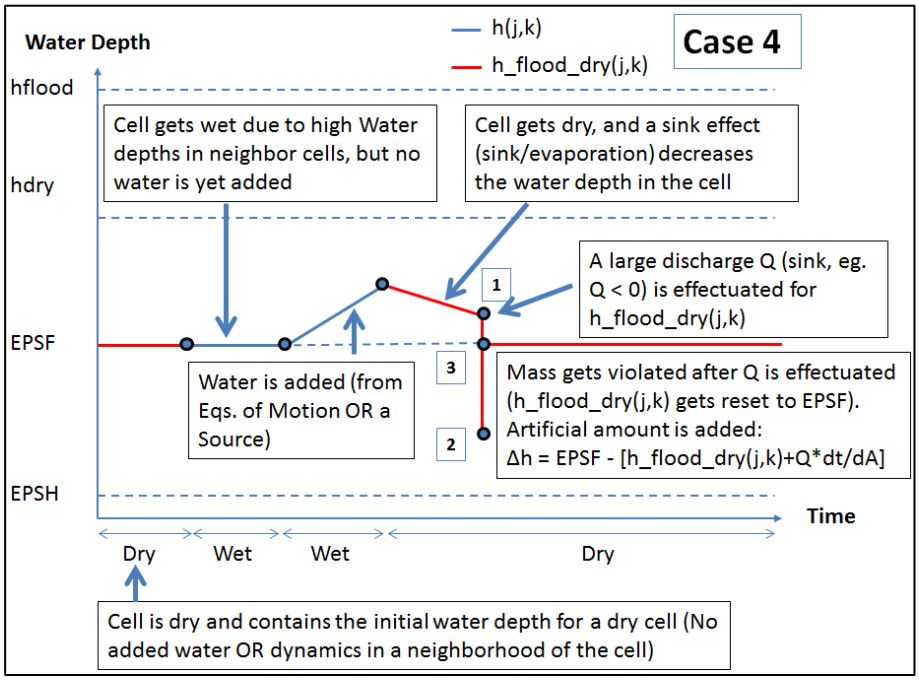

Figure 8.8 Case 4: Complex behaviour

9 Infiltration and Leakage

The effect of infiltration and leakage at the surface zone may be important in cases of flooding scenarios on otherwise dry land. It is possible to account for this in one of two ways: by Net infiltration rates or by constant infiltration with capacity.

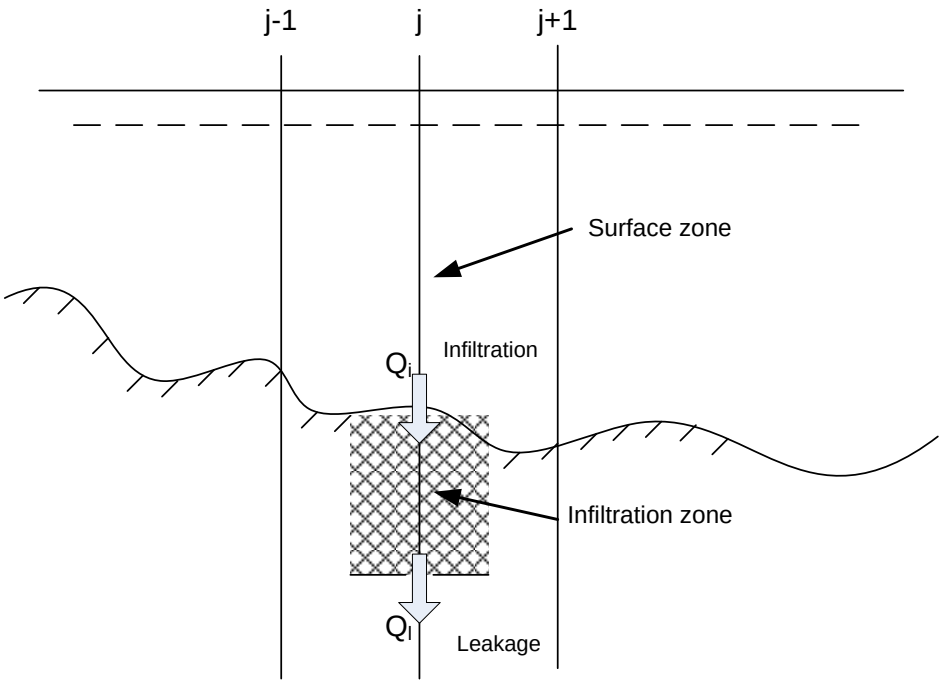

Figure 9.1 Illustration of infiltration process

9.1 Net Infiltration Rates

The net infiltration rate is defined directly. This will act as a simple sink in each grid cell in the overall domain area.

The one-dimensional vertical continuity equation is solved at each hydrodynamic time step after the two-dimensional horizontal flow equations have been solved. The calculation of the new water depth in the free surface zone for each horizontal grid cell is found by

$$H(j,k) = H(j,k) - V_{infiltration}(j,k) \,/(\Delta x \cdot \Delta y) \tag{9.1}$$

Where $V_{infiltration}(j,k)$ is the infiltrated volume in cell (j,k).

If $H(j,k)$ becomes marked as *dry* then cell (j,k) will be taken out of the two-dimensional horizontal flow calculations and no infiltration can occur until the cell is flooded again.

In summary: when using Net infiltration rate an unsaturated zone is never specified and thus has no capacity limits, so the specified infiltration rates will always be fully effectuated as long as there is enough water available in the cell.

9.2 Constant Infiltration with Capacity

Constant infiltration with capacity describes the infiltration from the free surface zone to the unsaturated zone and from the unsaturated zone to the saturated zone by a simplified model. The model assumes the following:

- The unsaturated zone is modelled as an infiltration zone with constant porosity over the full depth of the zone.
- The flow between the free surface zone and the infiltration zone is based on a constant flow rate, i.e. $V_{infiltration} = Q_i \cdot \Delta t$ where Q_i is the prescribed flow rate.
- The flow between the saturated and unsaturated zone is modelled as a leakage Q_l having a constant flow rate, i.e. $V_{leakage} = Q_l \cdot \Delta$.

The simplified model described above is solved through a one-dimensional continuity equation. Feedback from the infiltration and leakage to the two-dimensional horizontal hydrodynamic calculations is based solely on changes to the depth of the free surface zone – the water depth.

Note that the infiltration flow cannot exceed the amount of water available in the free surface water zone nor the difference between the water capacity of the infiltration zone and the actual amount of water stored there. It is possible that the infiltration flow completely drains the free surface zone from water and thus creates a dried-out point in the two-dimensional horizontal flow calculations.

The one-dimensional vertical continuity equation is solved at each hydrodynamic time step after the two-dimensional horizontal flow equations have been solved. The solution proceeds in the following way:

1. Calculation of the volume from leakage flow in each horizontal grid cell – $V_{leakage}(j,k)$

$$V_{leakage}(j,k) = Q_l(j,k) \cdot \Delta t \cdot \Delta x \cdot \Delta y \tag{9.2}$$

$$V_{leakage}(j,k) = \min(V_{leakage}(j,k), V_i(j,k)) \tag{9.3}$$

$$V_i(j,k) := V_i(j,k) - V_{leakage}(j,k) \tag{9.4}$$

   Where $V_i(j,k)$ is the total amount of water in the infiltration zone and $Q_l(j,k)$ is the leakage flow rate.

2. Calculation of the volume from infiltration flow in each horizontal grid cell – $V_{infiltration}(j,k)$

$$V_{infiltration}(j,k) = Q_i(j,k) \cdot \Delta t \cdot \Delta x \cdot \Delta y \tag{9.5}$$

$$V_{infiltration}(j,k) = \min\left(V_{infiltration}(j,k), SC_i(j,k) - V_i(j,k), H(j,k) \cdot \Delta x \cdot \Delta y\right) \tag{9.6}$$

$$V_i(j,k) := V_i(j,k) + V_{infiltration}(j,k) \tag{9.7}$$

   Where $Q_i(j,k)$ is the infiltration rate, $SC_i(j,k)$ is the water storage capacity and $H(j,k)$ the depth of the free surface.

3. Calculation of the new water depth in the free surface zone for each horizontal grid cell

$$H(j,k) = H(j,k) - V_{infiltration}(j,k)/(\Delta x \cdot \Delta y) \quad (9.8)$$

If $H(j,k)$ becomes marked as *dry* then cell *(j,k)* will be taken out of the two-dimensional horizontal flow calculations. The cell can still *leak* but no infiltration can occur until the cell is flooded again.

The water storage capacity of the infiltration zone is calculated as

$$SC_i(j,k) = Z_i(j,k) \cdot \Delta x \cdot \Delta y \cdot \gamma(j,k) \quad (9.9)$$

Where $Z_i(j,k)$ is the depth of the infiltration zone and $\gamma(j,k)$ is the porosity of the same zone.

In summary, when using Constant infiltration with capacity there can be situations where the picture is altered and the rates are either only partially effectuated or not at all:

- If $= H(j,k) < H_{dry}$ on the surface (dry surface) => infiltration rate is not effectuated
- If: the water volume in the infiltration zone reaches the full capacity => infiltration rate is not effectuated
- If: the water volume is zero in the infiltration zone (the case in many initial conditions) => leakage rate is not effectuated
- Leakage volume must never eclipse the available water volume in the infiltration zone, if so we utilise the available water volume in infiltration zone as leakage volume
- Infiltration volume must never eclipse the available water volume on the surface, if so we utilise the available water on the surface as infiltration volume

10 Multi-Cell Overland Solver

The MIKE 21 multi-cell overland solver is designed for simulating two-dimensional flow in rural and urban areas. The overall idea behind the solver is to solve the modified equations on a coarse grid taking the variation of the bathymetry within each grid cell into account. Results are presented on the grid that takes the fine scale bathymetry into account.

10.1 The Modified Governing Equations

The control volume for the governing equations is taken as being one coarse grid cell. Within this grid cell the topography may vary as illustrated in Figure 10.1.

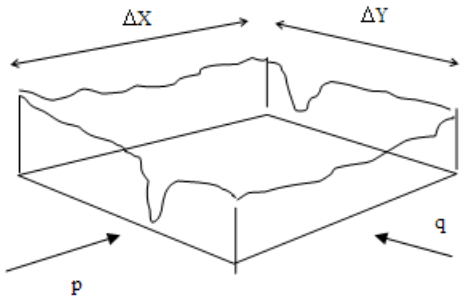

Figure 10.1 The topography within a coarse grid cell illustrating the control box used for deriving the fluxes

The mass balance reads

$$\frac{\partial h}{\partial t} + \frac{\partial p}{\partial y} + \frac{\partial q}{\partial x} = s \tag{10.1}$$

where *s* is the added sources/sinks per area.

By integration over a coarse grid cell area *A*, the equation reads

$$\int_A \frac{\partial h}{\partial t}(dx)dy + \int_A \frac{\partial p}{\partial y}(dx)dy + \int_A \frac{\partial q}{\partial x}(dx)dy = \sum_{s(A)} Q_S \tag{10.2}$$

where Q_S are sources and sinks within area *A* and the summation is to be taken over all sources and sinks with the area.

By selecting the flooded area *A* within a calculation cell and also assuming that the water level is constant within this cell, we obtain by the use of Green's theorem

$$A_{flood}\frac{\partial h}{\partial t} + \int_{\partial A} pdy + \int_{\partial A} qdx = \sum_{s(A)} Q_S \tag{10.3}$$

The summation is taken over the whole of the calculation cell.

The momentum equation to be solved is modified from the standard shallow water equation solved in MIKE 21. The approach taken is a “channel” like description for the J and K direction separately. Further, the coriolis force, wind forcing, and wave radiation stress are not included.

$$\int_S \frac{\partial p}{\partial t} dy + \frac{\partial}{\partial x}\left(\int_S \frac{p^2}{h} dy\right) + g \int_s h \frac{\partial \zeta}{\partial x} dy + \int_S \frac{\partial}{\partial y}\left(\frac{qp}{h}\right) dy + \int_S \frac{gp\sqrt{p^2+q^2}}{C^2 h^2} dy + \int_S eddy\ viscosity\ terms = 0 \tag{10.4}$$

The integration is taken over the length of a coarse grid cell in the J direction (ΔY). The depth in the convective term and the cross momentum is approximated by

$$h \approx \bar{h} = \frac{A_X}{\Delta Y} \tag{10.5}$$

where A is the “cross sectional area” in the J direction given by

$$A_X = \int_S h dy \tag{10.6}$$

The friction term is modified to reflect that the friction is effective along the wetted perimeter thus

$$h^2 \approx R_X \bar{h} = R_X \frac{A_X}{\Delta Y} \tag{10.7}$$

Finally, taking the flux as being constant within a coarse grid cell and dividing by ΔY one obtains

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial x}\left(\frac{p^2 \Delta Y}{A_X}\right) + g \frac{A_X \partial \zeta}{\Delta Y \partial x} + \frac{\partial}{\partial y}\left(\frac{gp \Delta Y}{A_X}\right) + \frac{gp\sqrt{p^2+q^2}}{C^2 R_X A_X} \Delta Y + (eddy\ visc\ terms) = 0 \tag{10.8}$$

The equation for the K direction reads

$$\frac{\partial p}{\partial t} + \frac{\partial}{\partial y}\left(\frac{q^2 \Delta X}{A_Y}\right) + g \frac{A_Y \partial \zeta}{\Delta X \partial y} + \frac{\partial}{\partial x}\left(\frac{gp \Delta X}{A_Y}\right) + \frac{gp\sqrt{p^2+q^2}}{C^2 R_Y A_Y} \Delta X + (eddy\ visc\ terms) = 0 \tag{10.9}$$

The equations are discretised with the cross sectional areas and hydraulic radius at the latest evaluated water level. Both quantities are taken as constant throughout the cell. The latter is achieved by taking the mean throughout the cell.

10.2 Determination of Fluxes on the Fine Scale

The fluxes on the fine grid scale are determined through linear interpolation in the primary direction and a distribution according to the water depth to the power of 3/2 in the transversal direction.

The interpolated fluxes may be written as

$$p_{j,k,fine} = \left(\frac{j + 1 + (J-1)N_{factor,J}}{N_{factor,J}} p_{J,K} + \frac{JN_{factor,J} - j - 1}{N_{factor,J}} p_{J-1,K} \right) \cdot N_{factor,K} \cdot h_{j,k}^{3/2} \cdot \left(\sum_{k=(K-1)N_{factor,K}}^{KN_{factor,K}-1} h_{j,k}^{3/2} \right)^{-1} \quad (10.10)$$

$$q_{j,k,fine} = \left(\frac{k + 1 + (K-1)N_{factor,K}}{N_{factor,J}} q_{J,K} + \frac{KN_{factor,K} - k - 1}{N_{factor,K}} q_{J,K-1} \right) \cdot N_{factor,J} \cdot h_{j,k}^{3/2} \cdot \left(\sum_{j=(J-1)N_{factor,J}}^{JN_{factor,J}-1} h_{j,k}^{3/2} \right)^{-1} \quad (10.11)$$

which are valid for

$(J-1)N_{factor,J} < j \leq JN_{factor,J}$

$(K-1)N_{factor,K} < k \leq KN_{factor,K}$

Note that the fluxes estimated through this process are not the result of a mass balance on the fine scale. Thus, the fluxes are indicative of the flow pattern, but are a post processed result and should be evaluated as such.

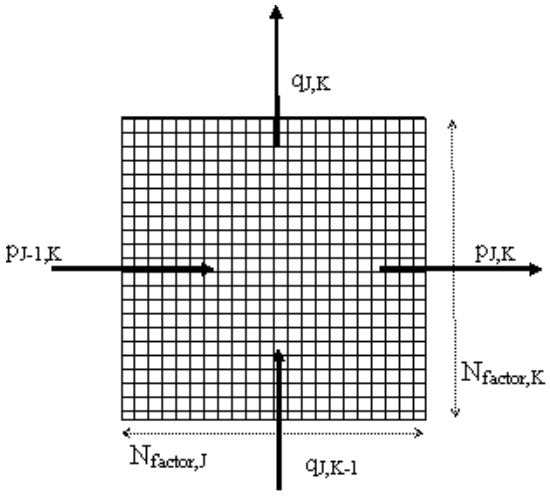

Figure 10.2 Interpretation of fluxes

11 Flood Screening Tool

The numerical engine for MIKE 21 FST and MIKE 21 HD is the same. The functionality in MIKE 21 FST calculations is however different from the 'normal' MIKE 21 HD as follows:

- The convective terms are omitted from the model
- The viscous terms are omitted in the governing equations
- Influence from wind, atmospheric pressure variation, Coriolis and wave radiation stresses cannot be included

This will result in faster runtime for inland flooding computations.

By neglecting the viscous terms, the convective terms, wind, atmospheric pressure variations and the Coriolis force, Eqs.(2.1) to (2.3) describing the flow and water level variation then read:

$$\frac{\partial \zeta}{\partial t} + \frac{\partial p}{\partial x} + \frac{\partial q}{\partial y} = \frac{\partial d}{\partial t} \tag{11.1}$$

$$\frac{\partial p}{\partial t} + gh\frac{\partial \zeta}{\partial x} + \frac{gp\sqrt{p^2+q^2}}{C^2 \cdot h^2} = 0 \tag{11.2}$$

$$\frac{\partial q}{\partial t} + gh\frac{\partial \zeta}{\partial y} + \frac{gq\sqrt{p^2+q^2}}{C^2 \cdot h^2} = 0 \tag{11.3}$$

The following symbols are used in the equations:

h(x,y,t)	water depth (= *ζ–d, m*)
d(x,y,t)	time varying water depth (m)
ζ(x,y,t)	surface elevation (m)
p,q (x,y,t)	flux densities in x- and y-directions ($m^3/s/m$) = *(uh,vh); (u,v)* = depth averaged velocities in x- and y-directions
C(x,y)	Chezy resistance ($m^{½}/s$)
g	acceleration due to gravity (m/s^2)
x,y	space coordinates (m)
t	time (s)

12 References

/1/ Abbott, M.B., Damsgaard, A. and Rodenhuis, G.S., System 21, Jupiter, A Design System for Two-Dimensional Nearly-Horizontal Flows. J. Hydr. Res., 1, 1973.

/2/ Abbott, M.B. and Rasmussen, C.H., On the Numerical Modelling of Rapid Contractions and Expansions in Models that are Two-Dimensional in Plan, Proc. 17th Congress, IAHR, Baden-Baden, 2, 1977.

/3/ Abbott, M.B., Computational Hydraulics - Elements of the Theory of Free Surface Flows, Pitman, London, 1979.

/4/ Abbott, M.B., McCowan and Warren, I.R., Numerical Modelling of Free-Surface Flows that are two-Dimensional in Plan, Proceedings of a symposium on Predictive Ability of Transport Models for Inland and Coastal Waters, Academic Press, 1981.

/5/ Aupoix, B., Eddy Viscosity Subgrid Scale Models for Homogeneous Turbulence, in Macroscopic Modelling of Turbulent Flow, Lecture Notes in Physics, Proc. Sophie-Antipolis, France, 1984.

/6/ Fredsøe, J. (1984), Turbulent boundary layers in Combined Wave Current Motion. J. Hydraulic Engineering, ASCE, Vol 110, No. HY8, pp. 1103-1120.

/7/ Jones, O., Zyserman, J.A. and Wu, Yushi (2014), Influence of Apparent Roughness on Pipeline Design Conditions under Combined Waves and Current, *Proceedings of the ASME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering*.

/8/ Launder, B.E. and Spalding, D.B., Mathematical Models of Turbulence. Academic Press, 1972.

/9/ Leendertse, J.J., Aspects of a Computational Model for Long Water Wave Propagation, Rand. Corp., RH-5299-RR, Santa Monica, California, 1967.

/10/ Leonard, B.P., A Survey of Finite Difference of Opinion on Numerical Muddling of the Incomprehensible Defective Confusion Equation, Proc. Am. Soc. Mech. Eng., Winter Annual Meeting, Publ. No. AMD-34, 1979.

/11/ Main, P. and Kun, J., Numerical Investigations of Turbulent Channel Flow, J. Fluid Mech., Vol. 118, pp. 341-377, 1982.

/12/ Richtmeyer, R.D. and Morton, K.W., Difference Methods for Initial Value Problems, 2nd Ed., Interscience, New York, 1967.

/13/ Rodi, W., Turbulence Models and their Applications in Hydraulics. A State of the Art Review. SFB 80/T/127, 1980.

/14/ Schuman, V., Subgrid Scale Model for Finite Difference Simulations of Turbulent Flows in Plane Channels and Annuli, J. Compt. Phys., Vol. 18, pp. 376-404, 1975.

/15/ Smagorinsky, J., General Circulation Experiments with the Primitive Equations, Monthly Weather Review, Vol. 91, pp. 91-164, 1963.

/16/ Smith, S.D. and Banke, E.G., Variation of the sea drag coefficient with wind speed, Quart. J.R. Met.Soc., 101, pp. 665-673.

/17/ Vreugdenhil, C.B., De Invloed van de Wrijvingsterm op de Stabiliteit van Differentie Methoden voor Hydraulische Problemen, De ingenieur, JRG. 78, Nr. 20, 1966.