STICHTING
MATHEMATISCH CENTRUM
2e BOERHAAVESTRAAT 49
AMSTERDAM

AFDELING TOEGEPASTE WISKUNDE

Report TW 100

On the stability of a difference scheme for the North Sea Problem

by

P.J. v.d. Houwen

March 1966

The Mathematical Centre at Amsterdam, founded the 11th of February, 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications, and is sponsored by the Netherlands Government through the Netherlands Organization for Pure Research (Z.W.O.) and the Central National Council for Applied Scientific Research in the Netherlands (T.N.O.), by the Municipality of Amsterdam and by several industries.

X →

Introduction

After a sequence of articles ([1] - [7]) concerning the analytic computation of the water elevation at the North Sea due to a windfield, it appeared desirable to employ numerical methods to treat more realistic models. The method of nets turns out to be very suitable for the problem, since an arbitrary coast and bottom profile is easily introduced. We are only interested in explicit two-level difference schemes, because of the very long computation time expected with the Electrologica X1 computer of the Mathematical Centre. In [8] a stable difference scheme was constructed with forward time differences for the streamvector as well as for the elevation. This scheme, however, is subject to very stringent stability conditions. To get a more practical scheme, we add a viscosity term, which results in a less stringent stability condition. For stability in the sense of Rjabenki and Filippow we find:

$$\Delta t \leq \frac{2\ \Delta x}{\sqrt{gh}}, \quad \Delta x = \Delta y,$$

while for stability in the sense of O'Brien-Hyman-Kaplan we find:

$$\Delta t < \frac{2\ \Delta x}{\sqrt{gh}} \left(\sqrt{1 + \frac{\lambda^2 \Delta^2 x}{4gh}} - \frac{\lambda \Delta x}{2\sqrt{gh}} \right).$$

For small values of λ these conditions are almost the same.

1. The partial differential equations

We define:

$\vec{w} = \binom{u}{v}$ total stream

ζ elevation of the water-surface. The undisturbed level is given by $\zeta = 0$.

$\vec{s} = \begin{pmatrix} u \\ v \\ \zeta \end{pmatrix}$ state vector

λ a coefficient of friction

Ω the coefficient of Coriolis

g constant of gravity

h depthfunction

$\vec{F} = \begin{pmatrix} U \\ V \end{pmatrix}$ the surface stress due to the windfield

$\vec{k} = \begin{pmatrix} s \\ c \end{pmatrix}$ unitvector tangential to the coast in positive sense

The components of the vectors $\vec{w}$, $\vec{F}$ and $\vec{k}$ are taken in a Carthesian coordinate-system (x,y); the unknown $\vec{w}$ and ζ are functions of the time t and the space coordinates x and y.

$\vec{s}$ satisfies the following equation and boundary conditions:

$$(1.1) \qquad \frac{\partial}{\partial t}\vec{s} = \begin{pmatrix} -\lambda & \Omega & -gh\frac{\partial}{\partial x} \\ -\Omega & -\lambda & -gh\frac{\partial}{\partial y} \\ -\frac{\partial}{\partial x} & -\frac{\partial}{\partial y} & 0 \end{pmatrix} \vec{s} + \begin{pmatrix} U \\ V \\ 0 \end{pmatrix},$$

$$(1.2) \qquad \begin{cases} \vec{w}(x,y,0) = 0, \ \zeta(x,y,0) = 0, \\ \vec{w} \text{ is tangential to the coast part of the boundary,} \\ \zeta = 0 \text{ at the ocean part of the boundary.} \end{cases}$$

We shall construct the equations describing the coast conditions.

From (1.1), we have

$$\frac{\partial}{\partial t}\vec{w} = \begin{pmatrix} -\lambda & \Omega & -gh\frac{\partial}{\partial x} \\ -\Omega & -\lambda & -gh\frac{\partial}{\partial y} \end{pmatrix} \vec{s} + \vec{F}$$

$$= \begin{pmatrix} -\lambda & \Omega \\ -\Omega & -\lambda \end{pmatrix} \vec{w} - gh\nabla\zeta + \vec{F}.$$

Forming the inner product between $\frac{\partial}{\partial t}\vec{w}$ and $\vec{k}$ we obtain

$$(1.3) \quad \frac{\partial}{\partial t}(k \circ \vec{w}) = -\lambda(\vec{w} \circ \vec{k}) + \vec{k} \circ \begin{pmatrix} 0 & \Omega \\ -\Omega & 0 \end{pmatrix} \vec{w} + gh(\vec{k} \circ \nabla\zeta) + \vec{k} \circ \vec{F}.$$

The coast condition is

$$(1.4) \qquad cu - sv = 0,$$

hence, equation (1.3) simplifies to

$$(1.3') \qquad \frac{\partial}{\partial t}(\vec{k} \circ \vec{w}) = -\lambda(k \circ \vec{w}) + gh(\vec{k} \circ \nabla\zeta) + \vec{k} \circ \vec{F}.$$

From (1.4) we have

$$\vec{k} \circ \vec{w} = su + cv = \frac{s^2 + c^2}{s} u = \frac{1}{s} u, \; s \neq 0,$$

$$\vec{k} \circ \vec{w} = su + cv = \frac{s^2 + c^2}{c} v = \frac{1}{c} v, \; c \neq 0.$$

Substitution in (1.3') and combining this with (1.4) we obtain the following equations at the coast:

$$(1.5) \qquad s \neq 0 \quad \begin{cases} u_t = -\lambda u - s\, gh \frac{\partial}{\partial k} \zeta + s\vec{k} \circ \vec{F} \\ v = \frac{c}{s} u \\ \zeta_t = -\nabla \vec{w} \end{cases}$$

$$(1.6) \qquad c \neq 0 \quad \begin{cases} u = \frac{s}{c} v \\ v_t = -\lambda v - c\, gh \frac{\partial}{\partial k} \zeta + c\vec{k} \circ \vec{F} \\ \zeta_t = -\nabla \vec{w} \end{cases}$$

The equations at the ocean offer no difficulties.

After the formulation of the problem, we investigate the operator

$$A = \begin{pmatrix} -\lambda & \Omega & -gh\frac{\partial}{\partial x} \\ -\Omega & -\lambda & -gh\frac{\partial}{\partial y} \\ -\frac{\partial}{\partial x} & -\frac{\partial}{\partial y} & 0 \end{pmatrix}$$

and the behaviour of ζ as a function of the time.
Suppose that the vector $\vec{b}$ is formed from $\vec{s}$ by application of the operator A:

$$\vec{b} = A\vec{s}.$$

We eliminate u and v, obtaining (see [1])

$$\Delta\zeta = f(\vec{b}),$$

where f is a known function of $\vec{b}$. A behaves as an <u>elliptic</u> operator with respect to ζ.
The behaviour of ζ as a function of t follows from the characteristic equation for (1.1) [10 pg 384]

$$(1.7) \qquad \det\left[p + q\begin{pmatrix} 0 & 0 & gh \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} + r\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & gh \\ 0 & -1 & 0 \end{pmatrix}\right] = 0,$$

p, q and r are the direction-cosines of the line-elements (dt,dx,dy) perpendicular to the characteristic-directions.
Equation (1.7) is equivalent to

$$(1.7') \qquad p(p^2 - gh\, q^2 - gh\, r^2) = 0.$$

This equation is satisfied by line-elements parallel to the (x,y)plane and line elements parallel to the rulers of the cone

$$t^2 - gh\, x^2 - gh\, y^2 = 0.$$

The characteristics are given by the directions

(a) parallel to the t-axis

(b) parallel to the rulers of $t^2 - \frac{1}{gh} x^2 - \frac{1}{gh} y^2 = 0$

From (b) it follows that a characteristic line element (dt,dx,dy) satisfies the relation

$$\frac{\sqrt{(dx^2 + d^2y)}}{dt} = \sqrt{gh}. \tag{1.8}$$

$\sqrt{gh}$ can be considered as a disturbance velocity.
From the fact that the characteristic equation has 3 independent solutions

$$(1, 0, 1/\sqrt{gh}), (0, 1, 0), (0, 0, 1)$$

we conclude that equation (1.1) is <u>total hyperbolic</u> [10].

2. The difference equations

In the (x,y) plane we take a rectangular net with spatial steps Δx and Δy.
In a netpoint (x,y) we replace the differential operators $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ by difference operators D_x and D_y, defined by

$$D_x f(x,y) = \frac{1}{2(b+2a)\Delta x} \{ a(f(x + \Delta x, y + \Delta y) - f(x - \Delta x, y + \Delta y)) + b(f(x + \Delta x, y) - f(x - \Delta x, y)) + a(f(x + \Delta x, y - \Delta y) - f(x - \Delta x, y - \Delta y)) \}$$

and an analogous expression for $D_y f(x,y)$; a and b are parameters at choice, however, not zero at the same time.
In a more compact form we write:

(2.1)

$$D_x f = \frac{1}{2(b+2a)\Delta x} \begin{vmatrix} -a & 0 & a \\ -b & 0 & b \\ -a & 0 & a \end{vmatrix} f.$$

$$D_y f = \frac{1}{2(b+2a)\Delta y} \begin{vmatrix} a & b & a \\ 0 & 0 & 0 \\ -a & -b & -a \end{vmatrix} f.$$

The cases a = 0 and b = 0 will be called respectively the central and the averaged central difference form.
In the boundary points (we assume that the boundary points are all netpoints) these difference operators are not defined. We postpone the discussion of this problem. The differential and difference operators are related by the following formulae

(2.2)

$$\frac{\partial}{\partial x} = D_x + O(\Delta x^2)$$

$$\frac{\partial}{\partial y} = D_y + O(\Delta y^2)$$

On the t-axis we consider the points $t = k\Delta t$, $k = 0, 1, 2, \ldots$ and we write

$$\vec{s}_k = \vec{s}(x, y, k\Delta t).$$

The difference operator D_t is defined by the formula

(2.3) $$D_t \vec{s}_k = \frac{1}{\Delta t} (\vec{s}_{k+1} - \vec{s}_k).$$

For $\frac{\partial}{\partial t}$ and D_t the following relation holds:

(2.4) $$\frac{\partial}{\partial t} = D_t + O(\Delta t).$$

Let us consider the difference scheme:

(2.5) $$\vec{s}_{k+1} = (I + \Delta t\, D)\vec{s}_k + \Delta t\, \vec{f}_k,$$

where $$D = \begin{pmatrix} -\lambda & \Omega & -gh\,D_x \\ -\Omega & -\lambda & -gh\,D_y \\ -D_x & -D_y & 0 \end{pmatrix}, \quad \vec{f} = \begin{pmatrix} U \\ V \\ 0 \end{pmatrix}.$$

From (2.2) and (2.4) we see that the error of approximation by replacing (1.1) by (2.5) is

$$O(\Delta x^2) + O(\Delta y^2) + O(\Delta t).$$

Difference scheme (2.5) is consistent with (1.1), which means that stability of (2.5) guarantees the convergence of the difference solution to the analytic solution.

In [8] a stability analysis is given for (2.5). The following criteria for stability in the sense of O'Brien-Hyman-Kaplan [11] are established for the averaged central difference form.

(2.6)
$$\Delta t < \frac{\lambda}{gh} \cdot \frac{\Delta^2 x\, \Delta^2 y}{(\Delta^2 x + \Delta^2 y)}$$

$$\Delta t < \left(\frac{2}{3\lambda}, \frac{2\lambda}{\lambda^2 + \Omega^2}\right)_{min}$$

In practice the first condition represents an unacceptable restriction for the time step Δt, because of the following reasons:

(a) Δt depends quadratically on Δx and Δy.

(b) the smallness of the value of λ occurring in the first condition.

(c) the factor $1/h$ prevents an economical choice for Δt (h is a rapid fluctuating function of x and y).

We try to construct an explicit two-level scheme with less stringent conditions for stability.

The inequalities (2.6) show, that an increase of the friction enables us to take a larger time step Δt. That means, a decrease of the internal energy of the watermotion softens the stability conditions for the numerical scheme. This suggests the introduction of an additive viscosity term to simulate an increase of internal energy. In (2.5) therefore, we add the following term

$$\Delta^2 t \; D_0 \; Q \; D_0 \; \vec{s}_k,$$

where

$$D_0 = \begin{pmatrix} 0 & 0 & -gh\,D_x \\ 0 & 0 & -gh\,D_y \\ -D_x & -D_y & 0 \end{pmatrix}, \quad Q = \begin{pmatrix} q_1 & q_2 & q_3 \\ r_1 & r_2 & r_3 \\ s_1 & s_2 & s_3 \end{pmatrix}.$$

Q is a matrix of real numbers.

We obtain the following explicit two-level scheme:

(2.7) $$\vec{s}_{k+1} = (I + \Delta t\; D_0 + \Delta^2 t\; D_0\; Q\; D_0)\vec{s}_k + \Delta t(D-D_0)\vec{s}_k + \Delta t\; \vec{f}_k.$$

It is clear that this scheme is still consistent with (1.1) for every Q.

3. Theory of stability

In this section we give some results from the theory of stability. We consider stability in the sense of Rjabenki and Filippow [12] and stability in the sense of O'Brien-Hyman-Kaplan [11].

Definition 1. Over the interval $0 \leq t \leq T$ scheme (2.7) is called stable in the sense of Rjabenki and Filippow if

(3.1) $$||I + \Delta t\; D + \Delta^2 t\; D_0\; Q\; D_0|| \leq 1 + O(\Delta t)$$

uniformly on $0 < \Delta t \leq \Delta t_0$.

<u>Definition 2.</u> For a certain timestep Δt scheme (2.7) is called stable in the sense of O'Brien- Hyman-Kaplan if

(3.2) $$||I + \Delta t\ D + \Delta^2 t\ D_0\ Q\ D_0|| \leq 1$$

uniformly on $0 \leq T \leq \infty$.

The second definition is used if calculations over very long time intervals $[0,T]$ are performed.
Following Lax and Richtmyer [13] we construct for scheme (2.7) the so-called amplification matrix; this is done by forming the expression:

(3.3) $$A(\vec{w}) \exp i\ \vec{w}\cdot\vec{x} = (I + \Delta t + \Delta^2 t\ D_0\ Q\ D_0) \exp i\ \vec{w}\cdot\vec{x},$$

where $\vec{w}$ is a vector in the (x,y)plane.
The amplification matrix $A(\vec{w})$ is an ordinary matrix-operator depending on $\vec{w}$.
We have the following theorems for difference schemes with constant coefficients [13].

<u>Theorem 1.</u> If the coefficients in difference scheme (2.7) are constant and

(3.1') $$||A(\vec{w})|| \leq 1 + O(\Delta t)$$

uniformly in $\vec{w}$ and Δt, then the scheme is stable in the sense of Rjabenki and Filippow.

<u>Theorem 2.</u> If the coefficients in difference scheme (2.7) are constant and

(3.2') $$||A(\vec{w})|| \leq 1$$

uniformly in $\vec{w}$ and T, then the scheme is stable in the sense of O'Brien-Hyman-Kaplan.

In practice the coefficients of the difference scheme are not constant: the coefficient of friction λ and the depth h are functions of x and y, and in the boundary points the difference operator is of a complete different form. This means that we have in every netpoint an other difference operator.

<u>Assumption</u>. If each of these operators applied in all netpoints is stable, then we assume that the difference operator with variable coefficients itself is stable [12].

In the following sections we study the stability of the difference operators applied at the internal points (internal stability).

4. Internal stability in the sense of Rjabenki and Filippow

Let us consider the difference operator, locally valid at the internal netpoint (x_0, y_0), applied in all netpoints.
We choose a new unit of time with $g(x_0, y_0) \cdot h(x_0, y_0) = 1$ and we construct the amplification matrix $A(\vec{w})$ for the difference operator.
The effect of the operators D_x and D_y on the functions $\exp i\, \vec{w} \cdot \vec{x}$ is given by:

$$D_x \exp i\, \vec{w}\cdot\vec{x} = \frac{1}{2(b+2a)\Delta x} \{a(\exp i(w_1 \Delta x + w_2 \Delta y) - \exp i(-w_1 \Delta x + w_2 \Delta y))$$

$$+ b(\exp i\, w_1\, \Delta x - \exp - i\, w_2\, \Delta y)$$

$$+ a(\exp i(w_1 \Delta x - w_2 \Delta y) - \exp i(-w_1 \Delta x - w_2 \Delta y))\}$$

$$= \frac{2i(b + 2a \cos w_2 \Delta y)\sin w_1 \Delta x}{2(b+2a)\Delta x} \exp i\, \vec{w}\cdot\vec{x}.$$

A similar expression is obtained for $D_y \exp i\, \vec{w}\cdot\vec{x}$.
We define

$$\gamma_1 = \frac{(b + 2a \cos w_2 \Delta y)\sin w_1 \Delta x}{(b+2a)} \cdot \frac{\Delta t}{\Delta x}$$

(4.1)

$$\gamma_2 = \frac{(b + 2a \cos w_1 \Delta x)(\sin w_2 \Delta y)}{(b+2a)} \cdot \frac{\Delta t}{\Delta y}.$$

Substitution in the formulae for $D_x \exp i\, \vec{w}\cdot\vec{x}$ and $D_y \exp i\, \vec{w}\cdot\vec{x}$ gives

$$D_x \exp i \vec{w}\circ\vec{x} = (i \frac{\gamma_1}{\Delta t}) \exp i \vec{w}\circ\vec{x}$$

(4.2)

$$D_y \exp i \vec{w}\circ\vec{x} = (i \frac{\gamma_2}{\Delta t}) \exp i \vec{w}\circ\vec{x}.$$

For the amplification-matrix we find after substitution of (4.1) into (3.3)

$$(4.3)\quad A(\vec{w}) = \begin{pmatrix} 1-s_3\gamma_1^2-\lambda\Delta t & -s_3\gamma_1\gamma_2+\Omega\Delta t & -i\gamma_1-s_1\gamma_1^2-s_2\gamma_1\gamma_2 \\ -s_3\gamma_1\gamma_2-\Omega\Delta t & 1-s_3\gamma_2^2-\lambda\Delta t & -i\gamma_2-s_1\gamma_1\gamma_2-s_2\gamma_2^2 \\ -i\gamma_1-q_3\gamma_1^2+r_3\gamma_1\gamma_2 & -i\gamma_2-q_3\gamma_1\gamma_2-r_3\gamma_2^2 & 1-q_1\gamma_1^2-r_2\gamma_2^2-(r_1+q_2)\gamma_1\gamma_2 \end{pmatrix}$$

We now investigate the norm of the matrix $A(\vec{w})$, in which we assume that Δx and Δy are both of the same magnitude as Δt.
The terms $\Omega\Delta t$ and $\lambda\Delta t$ may be omitted, since these terms have an $O(\Delta t)$ effect on the norm of $A(\vec{w})$. To calculate explicitly the eigenvalues of the reduced matrix, we set $s_3 = 0$. For the eigenvalue equation we find:

$$(4.4)\qquad (1-\alpha)(\alpha^2 - S\alpha + P) = 0$$

where

$$S = 2 - q_1\gamma_1^2 - (r_1 + q_2)\gamma_1\gamma_2 - r_2\gamma_2^2$$

$$P = S-1-(i\gamma_1+q_3\gamma_1^2+r_3\gamma_1\gamma_2)(i\gamma_1+s_1\gamma_1^2+s_2\gamma_1\gamma_2)$$
$$-(i\gamma_2+q_3\gamma_1\gamma_2+r_3\gamma_2^2)(i\gamma_2+s_1\gamma_1\gamma_2+s_2\gamma_2^2).$$

If it can be proved that this equation has three different roots less than, or equal to one, we have $||A(\vec{w})|| \leq 1 + O(\Delta t)$.
We use the following criterium

<u>Criterium</u>. The eigenvalues of the equation

$$\lambda^2 - S\lambda + P = 0, \quad S, P \text{ real},$$

are less than, or equal to one in absolute value if and only if

$$P \leq 1, \; 1 - S + P \geq 0 \quad 1 + S + P \geq 0.$$

To ensure that P and S are real we set $q_3 = r_3 = s_1 = s_2 = 0$ and we consider only real values for q_1, q_2, r_1 and r_2. Applying the criterium we obtain:

$$\gamma_1^2 + \gamma_2^2 \geq 0$$

$$\gamma_1^2 + \gamma_2^2 \leq q_1\gamma_1^2 + (r_1+q_2)\gamma_1\gamma_2 + r_2\gamma_2^2 \leq 2+\tfrac{1}{2}(\gamma_1^2+\gamma_2^2).$$

We find immediately a necessary criterium:

$$(4.5) \qquad \gamma_1^2 + \gamma_2^2 \leq 4.$$

This criterium is also sufficient if

$$\gamma_1^2 + \gamma_2^2 = q_1\gamma_1^2 + (r_1+q_2)\gamma_1\gamma_2 + r_2\gamma_2^2.$$

That is $q_1 = r_2 = 1$ and $r_1 = -q_2$; the viscosity matrix Q is of the form

$$(4.6) \qquad Q = \begin{pmatrix} 1 & q_2 & 0 \\ -q_2 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \; q_2 \text{ any real number.}$$

With this choice of Q the eigenvalues are

$$1, \; 1-\tfrac{1}{2}(\gamma_1^2+\gamma_2^2) \pm \sqrt{(\tfrac{1}{4}(\gamma_1^2+\gamma_2^2)-1)(\gamma_1^2+\gamma_2^2)}.$$

They lie all on the unit circle (P = 1) and are different from each other if $\gamma_1^2 + \gamma_2^2 \neq 0$. If $\gamma_1^2 + \gamma_2^2 = 0$ we have

$$A(\vec{w}) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + \Delta t \begin{pmatrix} -\lambda & \Omega & 0 \\ -\Omega & -\lambda & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

so we have proved that (4.6) makes difference scheme (2.7) stable if (4.5) is satisfied.

We consider criterium (4.5):

We set

$$\xi = \sin^2 w_1 \Delta x \ , \quad \eta = \sin^2 w_2 \Delta y.$$

Substitution in (4.1) and (4.5) results in:

(4.7) $$\Delta^2 t \leq 4/(\max_{\substack{0 \leq \xi \leq 1 \\ 0 \leq \eta \leq 1}} \beta^2)$$

where

$$\beta^2 = \frac{\Delta^2 y\ \xi(b+2a\sqrt{1-\eta})^2 + \Delta^2 x\ \eta(b+2a\sqrt{1-\xi})^2}{(b+2a)^2\ \Delta^2 x\ \Delta^2 y}.$$

We distinguish the cases a = 0 and b = 0.

<u>a = 0</u> $$\Delta^2 t \leq \min_{\xi \cdot \eta} 4b^2\Delta^2 x\Delta^2 y/(\xi\Delta^2 y b^2 + \eta\Delta^2 x b^2) \leq 4\ \frac{\Delta^2 x\Delta^2 y}{\Delta^2 x + \Delta^2 y}$$

$$\Delta t \leq 2\ \frac{\Delta x\ \Delta y}{\sqrt{\Delta^2 x + \Delta^2 y}}$$

which means in the original unit of time:

(4.8) $$\Delta t \leq \frac{2}{\sqrt{gh}}\ \frac{\Delta x\ \Delta y}{\sqrt{\Delta^2 x + \Delta^2 y}}.$$

This is the same condition as Fisher [9] gives for the central difference form.

<u>b = 0</u> $\Delta^2 t \leq \underset{\xi \circ \eta}{\text{Min}}\ 4\Delta^2 x \Delta^2 y / (\xi(1-\eta)\Delta^2 y + \eta(1-\xi)\Delta^2 x)$.

The denominator is a harmonic function in ξ and η, so it takes his maximum on the boundary of the region $0 \leq \xi \leq 1$, $0 \leq \eta \leq 1$. The maximum value equals $\Delta^2 x$ or $\Delta^2 y$. We find with the original time scale

(4.9) $$\Delta t \leq \frac{2}{\sqrt{gh}} \min (\Delta x, \Delta y).$$

Comparing (4.8) with (4.9) we see that the averaged central difference-form for the operators $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ is preferred to the central difference-form. From now on we take b = 0.

<u>Remark</u> 1) Criterium (4.9) is a little more stringent than the necessary characteristic criterium:

(4.10) $$\Delta t \leq \frac{2}{\sqrt{gh}} \quad (\Delta^2 x + \Delta^2 y).$$

<u>Remark</u> 2) If Q = 0 we have

$$A(\vec{w}) = \begin{pmatrix} 1-\lambda\Delta t & \Omega\Delta t & -i\gamma_1 \\ -\Omega\Delta t & 1-\lambda\Delta t & -i\gamma_2 \\ -i\gamma_1 & -i\gamma_2 & 1 \end{pmatrix}.$$

When we assume $\Delta t = O(\Delta^2 x) = O(\Delta^2 y)$, which means γ_1 and γ_2 are $O(\sqrt{\Delta t})$, we obtain for the eigenvalues of the matrix

$$N = \begin{pmatrix} 1 & 0 & -i\gamma_1 \\ 0 & 1 & -i\gamma_2 \\ -i\gamma_1 & -i\gamma_2 & 1 \end{pmatrix}$$

$$\alpha(N) = 1,\ 1 \pm i\sqrt{\gamma_1^2 + \gamma_2^2}$$

$$|\alpha(N)| = 1, \quad \sqrt{1 + O(\Delta t)}.$$

Now N is a normal matrix, so

$$||N|| \leq \text{Max } |\lambda(N)| \leq 1 + O(\Delta t).$$

Hence we obtain

$$||A(\vec{w})|| \leq ||N|| + O(\Delta t) \leq 1 + O(\Delta t).$$

This implies the stability of scheme (2.5) in the sense of Rjabenki and Filippow when $\Delta t/\Delta^2 x$ and $\Delta t/\Delta^2 y$ remains constant for vanishing Δx, Δy and Δt. The only condition for the size of Δt is the characteristic-criterium (4.10). We recall that stability in the sense of O'Brien-Hyman-Kaplan for scheme (2.5) requires the very restrictive conditions (2.6).

Remark 3) In actual computation we add to (2.7) the following terms:

$$\Delta^2 t \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \lambda D_x + \Omega D_y & -\Omega D_x + \lambda D_y & 0 \end{pmatrix} \vec{s}_k + \Delta t(\vec{f}_{k+1} - \vec{f}_k) - \Delta^2 t \begin{pmatrix} 0 \\ 0 \\ D_x U + D_y V \end{pmatrix}.$$

The order of approximation and the stability analysis are not altered, the difference scheme, however, can be written in a very simple way

$$(4.11) \quad \begin{cases} \vec{w}_{k+1} = \begin{pmatrix} 1-\lambda\Delta t & \Omega\Delta t & -\Delta t\, D_x \\ -\Omega\Delta t & 1-\lambda\Delta t & -\Delta t\, D_y \end{pmatrix} \vec{s}_k + \Delta t\, \vec{F}_k \\ \\ \zeta_{k+1} = \zeta_k - \Delta t(D_x\, u_{k+1} + D_y\, v_{k+1}) \end{cases}$$

The difference with scheme (2.5) consists in the use of backward time differences when we calculate the elevation.

<u>Remark</u> 4) One may think that it is trivial to compute the elevation with the new values for the stream, however, it appears that first computing the elevation and with these elevation values the stream components, gives rise to an unstable difference scheme for $\frac{\Delta t}{\Delta x}$, $\frac{\Delta t}{\Delta y}$ constant if $\Delta t, \Delta x, \Delta y \to 0$.

The viscosity-matrix Q is now of the form

$$Q = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The eigenvalue equation of the reduced amplification matrix is

$$(\alpha - 1)(\alpha^2 - 2\alpha + 1 + (\gamma_1^2 + \gamma_2^2)(1 - i\gamma_1)) = 0.$$

For $\gamma_1 = 0$ the product of two eigenvalues is $1+\gamma_2^2$, so one of them lies beyond the unit circle.

5. <u>Internal stability in the sense of O'Brien-Hyman-Kaplan</u>

Again we choose gh = 1 in the point at consideration.
We construct the amplification-matrix $B(\vec{w})$ corresponding with scheme (4.11) without omitting $O(\Delta t)$terms. We obtain

$$(5.1) \quad B(\vec{w}) = \begin{pmatrix} 1-\lambda\Delta t & \Omega\Delta t & -i\gamma_1 \\ -\Omega\Delta t & 1-\lambda\Delta t & -i\gamma_2 \\ i(\lambda\Delta t-1)\gamma_1+i\Omega\Delta t\gamma_2 & i(\lambda\Delta t-1)\gamma_2-i\Omega\Delta t\gamma_1 & 1-\gamma_1^2-\gamma_2^2 \end{pmatrix}$$

The eigenvalue equation for $B(\vec{w})$ is

$$(5.2) \qquad \alpha^3 + a_1\alpha^2 + a_2\alpha + a_3 = 0$$

where

$$a_1 = -3 + 2\lambda\Delta t + \gamma_1^2 + \gamma_2^2$$

$$a_2 = 3 - 4\lambda\Delta t + (\lambda^2+\Omega^2)\Delta^2 t - (1-\lambda\Delta t)(\gamma_1^2+\gamma_2^2)$$

$$a_3 = -1 + 2\lambda\Delta t - (\lambda^2+\Omega^2)\Delta^2 t$$

We must find conditions to ensure that $||B(\vec{w})|| \leq 1$. This is certainly the case if all the eigenvalues of $B(\vec{w})$ are within the unit circle, for then we have $||B(\vec{w})|| < 1$. Analogous to [8] we apply the Hurwitz-criterium to force the roots of (3.16) within the unit circle.
We must have:

$$1 + a_1 + a_2 + a_3 > 0$$

$$1 - a_1 + a_2 - a_3 > 0$$

$$3 + a_1 - a_2 - 3a_3 > 0$$

$$1 - a_2 + a_1 a_3 - a_3^2 > 0$$

We obtain the following inequalities:

(a) $(\gamma_1^2 + \gamma_2^2)\lambda > 0$

(b) $(\gamma_1^2 + \gamma_2^2) < 2 \cdot \dfrac{4 - 4\lambda\Delta t + (\lambda^2 + \Omega^2)\Delta^2 t}{2 - \lambda\Delta t}$

(c) $(\gamma_1^2 + \gamma_2^2) > \Delta t^2(\lambda^2 + \Omega^2) \cdot \dfrac{(\lambda^2 + \Omega^2)\Delta t - 2\lambda}{\lambda - (\lambda^2 + \Omega^2)\Delta t}$

(d) $\Delta t < \dfrac{\lambda}{\lambda^2 + \Omega^2}, \quad \Delta t < \dfrac{2}{\lambda}$

(a) We assume $\lambda > 0$; the first inequality can not be satisfied if $\gamma_1 = \gamma_2 = 0$. In this case however, the eigenvalue equation is much simpler:

$$(B(\vec{w}))_{\gamma_1=\gamma_2=0} = \begin{pmatrix} 1-\lambda\Delta t & \Omega\Delta t & 0 \\ -\Omega\Delta t & 1-\lambda\Delta t & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$(\alpha - 1)(\alpha^2 - 2(2\lambda\Delta t - 1)\alpha + (1 - \lambda\Delta t)^2 + \Omega^2\Delta^2 t) = 0.$$

We see that this equation has three different roots and if they lie all within or on the unit circle, we are sure that $||B(\vec{w})|| \leq 1$. We now use the criterium for quadratic equations obtaining:

$$\Delta t \leq \frac{2\lambda}{\lambda^2 + \Omega^2} .$$

This condition is less stringent than inequality (d), so inequality (a) gives no conditions for Δt.

(b) The second inequality is certainly satisfied if we take $\Omega = 0$:

$$\gamma_1^2 + \gamma_2^2 < 2 \cdot \frac{4 - 4\lambda\Delta t + \lambda^2\Delta^2 t}{2 - \lambda\Delta t} = 2 \cdot (2 - \lambda\Delta t).$$

We have seen that (compare with (4.7))

$$\gamma_1^2 + \gamma_2^2 = \Delta^2 t \beta^2$$

where β^2 is independent of Δt.

Hence we obtain the following inequality

$$\beta_{max}^2 \ \Delta^2 t + 2\lambda\Delta t - 4 < 0.$$

This is satisfied if

$$\Delta t < \frac{-2\lambda + \sqrt{4\lambda^2 + 16\beta_{max}^2}}{2\beta_{max}^2} .$$

With the original unit of time we can write

$$\Delta t < \frac{2}{\beta_{max}\sqrt{gh}} \left(\sqrt{1 + \frac{\lambda^2}{4\beta_{max}^2 gh}} - \frac{\lambda}{2\beta_{max}\sqrt{gh}} \right).$$

Because of the smallness of λ we may consider $\lambda/(2\beta_{max}\sqrt{gh})$ as a small quantity. We develop the squareroot in a Taylor series of $\lambda^2/(4\beta_{max}^2 gh)$, obtaining:

$$\Delta t < \frac{2}{\beta_{max}\sqrt{gh}}\left\{1 - \frac{\lambda}{2\beta_{max}\sqrt{gh}} + \frac{\lambda^2}{12\beta_{max}^2 gh} - \frac{\lambda^4}{144\beta_{max}^4 g^2 h^2} \ldots\right\}.$$

(c) The third inequality is always satisfied if (d) is satisfied.

(d) These inequalities need no further reduction.

The conditions for the time step Δt are

$$(5.3) \qquad \Delta t < \frac{2}{\beta_{max}\sqrt{gh}}\left\{1 - \frac{\lambda}{2\beta_{max}\sqrt{gh}} + \ldots\right\},$$

$$\Delta t < \left(\frac{2}{\lambda}, \frac{\lambda}{\lambda^2 + \Omega^2}\right)_{min} .$$

The first condition is very little more restrictive than condition (4.7), which we found for stability in the sense of Rjabenki and Filippow; the second condition is of the same type as the second condition of (2.6) which was found for $Q = 0$.

We recall that in the cases $a = 0$ and $b = 0$ we must substitute respectively

$$\beta_{max} = \frac{\sqrt{\Delta^2 x + \Delta^2 y}}{\Delta x \Delta y}, \quad \beta_{max} = 1/\min(\Delta x, \Delta y).$$

6. The stability of the coast conditions

Before we formulate the difference formulae for the boundary values, we make some remarks concerning the calculation of u, v and ζ in the internal points. As in [8] the stream and the elevation are calculated at different points which form two interlacing nets as shown in figure 1, where the crosses denote points where the components of the stream are calculated and the dots points where the elevation is calculated.

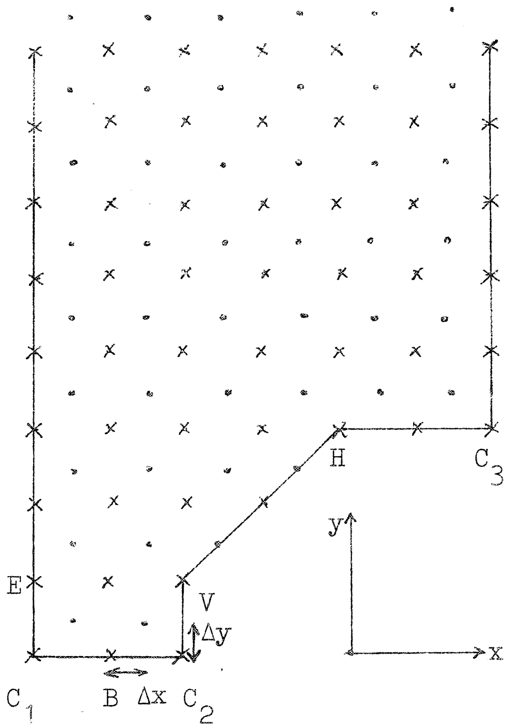


figure 1.

In the internal points the difference formulae (4.11) with b = 0 can be applied. The boundary presenting the coasts is taken through the netpoints of the stream-net. The boundary presenting the ocean is taken through the netpoints of the elevation-net. In corner points, as C_1, C_2 and C_3 in figure 1, we take u = v = 0. In the other boundary points we derive from (1.5) and (1.6) the following difference formulae (in accordance with (4.11))

$$(6.1)\quad s \neq 0 \quad \begin{cases} u_{k+1} = (1-\lambda\Delta t)u_k - sgh\Delta t(sB_x + cB_y)\zeta_k + \Delta t(s^2U_k + scV_k) \\ v_{k+1} = \frac{c}{s}\, u_{k+1} \\ \zeta_{k+1} = \zeta_k - \Delta t B_x u_{k+1} - \Delta t B_y v_{k+1} \end{cases}$$

$$(6.2)\quad c \neq 0 \quad \begin{cases} u_{k+1} = \frac{s}{c}\, v_{k+1} \\ v_{k+1} = (1-\lambda\Delta t)v_k - cgh\Delta t(sB_x + cB_y)\zeta_k + \Delta t c(sU_k + cV_k) \\ \zeta_{k+1} = \zeta_k - \Delta t B_x u_{k+1} - \Delta t B_y v_{k+1} \end{cases}$$

where the difference approximations of $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$, denoted by B_x and B_y, are now weighted averaged central differences; for example in the points B and E we have for B_x respectively B_y in compact notation the following expressions

$$(6.3)\quad B_x = \frac{1}{2\Delta x}\left|\begin{array}{ccc} +1 & & -1 \\ -3 & 0 & +3 \\ & 0 & \end{array}\right| \quad , \quad B_y = \frac{1}{2\Delta y}\left|\begin{array}{ccc} +3 & & -1 \\ 0 & 0 & \\ -3 & & +1 \end{array}\right|$$

We see that the difference operators B_x and B_y are defined in every boundary point (excluding the corner points) of figure 1, if we take for $\vec{k}$ in the points V and H the averaged direction of the boundary segments. We do not consider other situations for boundary points as is shown in figure 1.

Now we investigate the stability of the boundary formulae in the same manner as we have done for the internal stability. In the stability analysis of the internal points we supposed that the difference operator defined in an arbitrary point was applied in every point of the net extended over the whole (x,y)plane. In the same way we suppose that the boundary operators are applied in every net point of the (x,y)plane.

We write for (6.1) and (6.2) the equivalent formulae

$$(6.4)\quad \begin{aligned} \vec{w}_{k+1} &= \begin{pmatrix} 1-\lambda\Delta t & 0 & -sgh\Delta t(sB_x + cB_y) \\ 0 & 1-\lambda\Delta t & -cgh\Delta t(sB_x + cB_y) \end{pmatrix} \vec{s}_k \\ \zeta_{k+1} &= \zeta_k - \Delta t B_x u_{k+1} - \Delta t B_y v_{k+1} \end{aligned}$$

where we omit the windfield.

For the amplification matrix of (6.4) we find, taking gh = 1

$$\begin{pmatrix} 1-\lambda\Delta t & 0 & -is(s\gamma_1 + c\gamma_2) \\ 0 & 1-\lambda\Delta t & -ic(s\gamma_1 + c\gamma_2) \\ i(\lambda\Delta t-1)\gamma_1 & i(\lambda\Delta t-1)\gamma_2 & 1-(s\gamma_1 + c\gamma_2)^2 \end{pmatrix}$$

where

$$\gamma_1 = \frac{\Delta t}{\Delta x} \sin w_1\Delta x(3 \exp i\, w_2\Delta y - \exp 3i\, w_2\Delta y),$$

$$\gamma_2 = \frac{\Delta t}{\Delta y} \sin w_2\Delta y(3 \exp i\, w_1\Delta x - \exp 3i\, w_1\Delta x).$$

We consider the stability in the sense of Rjabenki and Filippow. Omitting the terms of the order Δt we find for the eigenvalue equation of the reduced matrix

$$(\alpha - 1)(\alpha^2 - (2 - (s\gamma_1 + c\gamma_2)^2)\alpha + 1) = 0.$$

There are three different roots which are within or on the unit circle if for all real $(s\gamma_1 + c\gamma_2)^2$ [1)]

$$2 - (s\gamma_1 + c\gamma_2)^2 + 1 \geq -1,$$

$$\text{and} \quad -2 + (s\gamma_1 + c\gamma_2)^2 + 1 \geq -1.$$

We obtain

$$(s\gamma_1 + c\gamma_2)^2 \leq 4. \tag{6.5}$$

We have to consider the real values of $(s\gamma_1 + c\gamma_2)^2$ only, so it is clearly sufficient if

$$\Delta t \leq \frac{2}{|\beta|_{max}} \tag{6.6}$$

where $\beta = \frac{s}{\Delta x} \sin w_1\Delta x(3e^{iw_2\Delta y} - e^{3iw_2\Delta y}) + \frac{c}{\Delta y} \sin w_2\Delta y(3e^{iw_1\Delta x} - e^{3iw_1\Delta x}).$

---

1) For complex values of $(s\gamma_1 + c\gamma_2)^2$ this follows from the fact that the product of the roots is 1, for real values of $(s\gamma_1 + c\gamma_2)^2$ this follows from the criterium for quadratic equations already mentioned.

The following inequality certainly holds:

$$|\beta|_{max} < 4\left(\frac{|s|}{\Delta x} + \frac{|c|}{\Delta y} \right) .$$

Hence we obtain, with the original unit of time:

$$\Delta t \leq \frac{1}{2\sqrt{gh}} \cdot \frac{\Delta x \Delta y}{|s|\Delta y + |c|\Delta x} . \tag{6.7}$$

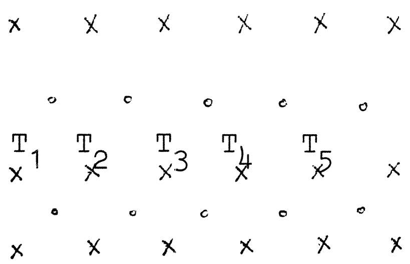
This criterium is far more restrictive than the internal stability-criterium, therefore this criterium prescribes the time step Δt, if we assume that local stability in every netpoint implies the overall stability.

7. Non uniform nets

From the coast stability criterium (6.7) we see that with a variable depth function $h(x,y)$ and a uniform net we cannot choose Δt in an economical way. A rough look at the function $h(x,y)$ for the Northsea (tabel I) reveals that in the Northern part the depth is 200-100 m and then suddenly decreases to 60-20 m in the Southern part. This suggests the transition from a coarse to a refined net. Since we are mainly interested in the Southern part of the Northsea, we approximate the northern coasts by lines parallel to the y-axis, that is $\vec{k} = (0, \pm 1)$ in the northern boundary points. Therefore in these points we have the condition:

$$\Delta t \leq \frac{\Delta y}{2\sqrt{gh}} . \tag{5.1}$$

It is sufficient to increase only Δy in the northern part (figure 2).



In actual computation we took

$\Delta x = 2 \cdot 10^4$ m

$(\Delta y)_{south} = 2 \cdot 10^4$ m, $(\Delta y)_{north} = 4 \cdot 10^4$ m

$g = 9.81$ m/sec^2.

figure 2

	s	c	h in m	Δt in sec	
a)	0	1	200	≃ 450	(according to (5.1))
b)	$\frac{1}{2}\sqrt{2}$	$\frac{1}{2}\sqrt{2}$	20	≃ 500	(according to (6.7))

Case a) is typical for a northern boundary point.
Case b) is typical for a southern boundary point.

In our calculations for the K.N.M.I. we obtained satisfactory results with the given values for Δx and Δy and with $\Delta t = 450$ sec.

Remark 1) In the transition points T_1, T_2, ... we calculate the values for ζ_x and ζ_y from the four neighbouring elevation points with weight-factors according to the geometric configuration.

8. The effect of the Channel

According to Weenink [14] we take into account the Channel-leak-stream by the condition

$$T = f \cdot \zeta_{ch}$$

where T = the quantity of water which leaves the North Sea through the Channel

f = a parameter depending upon the net width

ζ_{ch} = elevation in the Channel (figure 3).

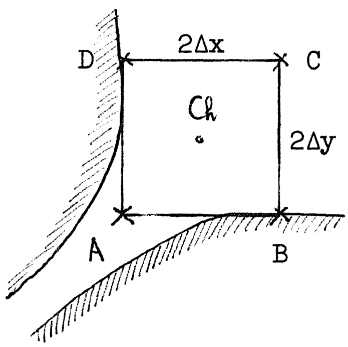


figure 3

According to the equation of continuity

$$-\frac{\Delta t}{2}\{2\Delta x(v_D(t)+v_C(t))+2\Delta y(u_C(t)+u_B(t))\}$$

$$= f\cdot\zeta_{ch}(t)\Delta t+4\Delta x\Delta y(\zeta_{ch}(t)-\zeta_{ch}(t-\Delta t))$$

we obtain

(8.1) $$\zeta_{ch}(t) = \frac{1}{1 + \frac{f\Delta t}{4\Delta x\Delta y}} \cdot \{\zeta_{ch}(t-\Delta t) - \Delta t D_x u(t) - \Delta t D_y v(t)\}.$$

The effect of the Channel on the original difference formula in Ch is a multiplication with

$$1/1 + \frac{f\Delta t}{4\Delta x\Delta y} < 1.$$

If $\Delta x = \Delta y = 2 \cdot 10^4$ m we take $f = 1 \cdot 04\ 10^6\ m^2/sec$.

9. The windfield

In practice it is the wind velocity $\vec{v}_0$ at the seasurface we know; from [2] we have

(9.1) $$\vec{F} = 3 \cdot 10^{-6}\ ||\vec{v}_s|| \cdot \vec{v}_s.$$

We shall consider windfields without sources or sinks, that is

$$\text{div}\ \vec{v}_s = 0.$$

Thus there exists a streamfunction $\psi(x,y)$ with

(9.2) $$\vec{v}_s = \alpha\left(-\frac{\partial}{\partial y}, \frac{\partial}{\partial x}\right)\psi$$

where α is a constant.
We relate the constant α with the maximum wind velocity v_m: from (9.1) we obtain:

(9.3) $$\alpha^2 = v_m^2 / (\psi_x^2 + \psi_y^2)_{max}.$$

In actual computation we used the following streamfunction:

(9.4) $$\psi = \sin\frac{2\pi}{K}(x - C_1 t + X_0) \cdot \sin\frac{2\pi}{L}(y - C_2 t + Y_0)$$

where K, L, C_1, C_2, X_0 and Y_0 are given constants, which represents the wavelength, the disturbance velocity and the phase in the x and y direction respectively.

We calculate the value of α for the streamfunction (9.4). We write

$$A = \sin(\frac{2\pi}{K}(x - C_1 t + X_0) - \frac{2\pi}{L}(y - C_2 t + Y_0))$$

$$B = \sin(\frac{2\pi}{K}(x - C_1 t + X_0) + \frac{2\pi}{L}(y - C_2 t + Y_0)).$$

For $\nabla\psi$ we obtain in terms of A and B

$$\psi_x = -\frac{\pi}{K}(A - B)$$

$$\psi_y = \frac{\pi}{L}(A + B)$$

$$\psi_x^2 + \psi_y^2 = \pi^2\left(\frac{(A-B)^2}{K^2} + \frac{(A+B)^2}{L^2}\right), \quad |A|,\ |B| \le 1.$$

We consider A and B as independent of each other:

K = L) $||\nabla\psi||^2$ takes its maximum value in the points $(A,B) = (\pm 1, \pm 1)$

K < L) We write

$$||\nabla\psi||^2 = \pi^2\left(\frac{1}{K^2}\left[(A-B)^2 + (A+B)^2\right] - \left(\frac{1}{K^2} - \frac{1}{L^2}\right)(A + B)^2\right).$$

$||\nabla\psi||^2$ is maximal in the points $(A,B) = (1,1), (-1,+1)$.

K > L) In the same way we can show that the maximal value is reached in the points (1,1) and (-1,-1).

In all the cases we have considered, (A,B) indeed equals $(\pm 1, \pm 1)$ for some point (x,y,t) in the (x,y,t)space, so we have

$$\alpha = \begin{cases} \dfrac{L\, v_m}{2\pi} & K \ge L \\[2ex] \dfrac{K\, v_m}{2\pi} & K \le L \end{cases} \qquad (9.5)$$

10. The ALGOL program

The program is mainly used for calculations with the following model (figure 4)

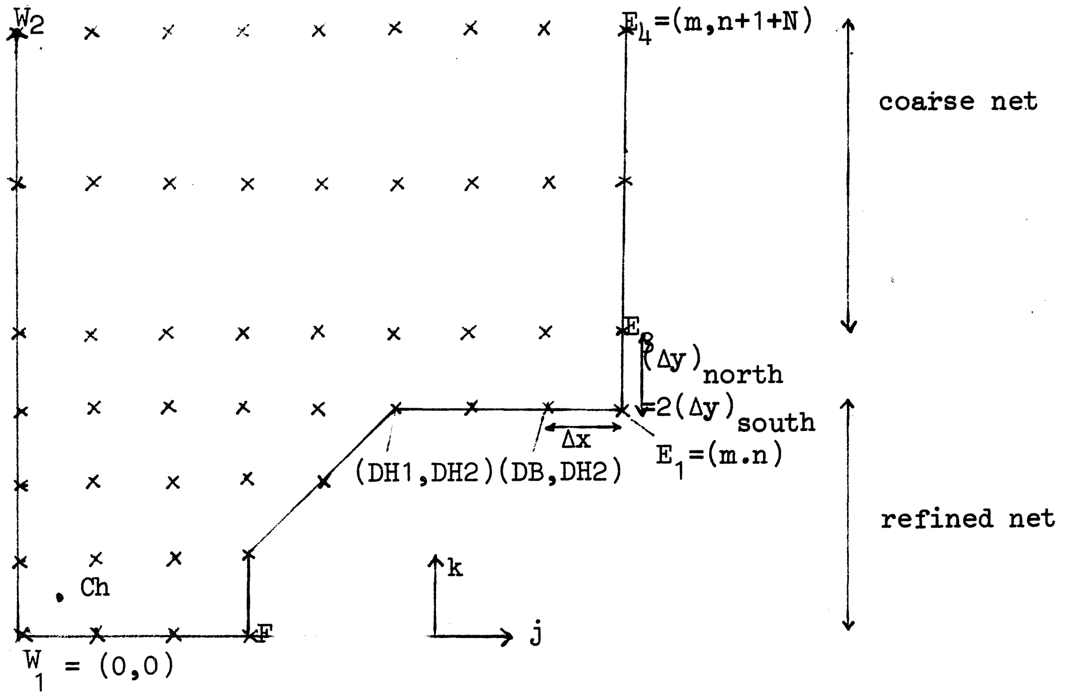


figure 4 Net of the streampoints

For the streampoints and the elevation points we use an integer coordinate system (j,k) with W_1 = (0,0) and Ch = (1,1) respectively (fig. 4). The program also applies to other configurations of the coast segments W_1W_2 and E_1E_4.

Before we deal with the structure of the program we give a list of the parameters which must be specified as input on a paper tape for the program:

N number of elementary meshes $(2\Delta x,\ 2(\Delta y)_{north})$ along E_3E_4

n number of elementary meshes $(2\Delta x,\ 2(\Delta y)_{south})$ along F H_e and E_1E_2

m number of elementary meshes $(2\Delta x,\ 2(\Delta y)_{north})$ along W_2E_4

i	counts the number of elementary timesteps Δt
dt	Δt
j	quotient of the frequencies of the output procedures COASTINF and SEAINFORM
(DH_1, DH_2)	coordinates of Den Helder in the model (figure 3)
DB	see figure 4
P	m * dt/(4 * a), where a is the distance between the left and the right coast in the northern part (for the length of the left coast we take 2a)
p	2 * P
Q	(n+1+2N) * dt/(4 * a)
q	2 * Q
d	Ω * dt
f	1 + f * dt/(dx * $(dy)_{south}$), where f is a given constant considered in section 7
r	friction parameter which we took 0.0024 m/sec [1)]
g	constant of gravity
a1	2 * a * π/(m * K)
a2	4 * a * π/((n+1+2N) * L) if k $\leq$ n+1 else 8 * a * π/((n+1+2N) * L)
b1	2 * dt * π(C1/K - C2/L)
b2	2 * dt * π(C1/K + C2/L)
d1	2 * π * (-X0/K + Y0/L) if k $\leq$ n+1 else 2 * π * (-X0/K + Y0/L) $- \frac{(n+1) * a2}{2}$
d2	2 * π * (-X0/K - Y0/L) if k $\leq$ n+1 else 2 * π * (-X0/K - Y0/L) + (n+2) * a2/2
w	maximum value of the windvelocity
fw	1/2 if K $\leq$ L else L/2K
gw	K/2L if K $\leq$ L else 1/2

---

1) For the friction-coefficient λ we used the expression

$$\lambda = \Omega/h$$

W[k]	abcis of the streampoint lying on W_1W_2 with ordinate k
O[k]	abcis of the streampoint lying on F He or E_1E_4
(SW[k],CW[k])	vector $\vec{k}$ = (s,c) in the streampoint (W[k],k)
(SO[k],CO[k])	vector $\vec{k}$ = (s,c) in the streampoint (O[k],k)
h[j,k]	depth in the streampoint (j,k) (Tabel I)

We now describe the functions of the procedures declared in the program

COEFFICIENT	calculates the coefficients A[jk], B[jk] and C[jk] of the difference scheme from the parameters given above
coefficient	reads from the input paper tape or punches on the output paper tape the coefficients of the difference scheme
ELINFORM	sets equal to zero, reads or punches the elevation z[jk]
INFORMATION	reads or punches the input parameters, sets equal to zero, reads or punches the streamcomponents u[jk] and v[jk], activates the procedures ELINFORM, COEFFICIENT or coefficient
INPUT	activates the procedure INFORMATION; if i = 1 the initial state is set equal to zero and the coefficients of the difference scheme are calculated, if i > 1 the initial state and the coefficients are read
COASTINF	punches the values of the elevation along W_1F, F He, He E_1 and E_1E_2
SEAINF	punches the whole elevation field
ZUIDKUST	calculates the stream along W_1F and He E_1
NOORDZEE	calculates the elevation and the stream in the refined or in the coarse net; z[jk] and (u[jk], v[jk]) are computed at the same time, so it is impossible to find z[jk] correctly in the elevation points along F He

CORRECTIEMODEL WEENINK	corrects the elevation values along F He
wind	calculates the components of the windvelocity
ROOSTEROVERGANG	calculates the stream and the elevation in the net points (j,n+1), j = 0,...,m
OUTPUT	activates the procedure INFORMATION

We conclude this section with the complete version of the program.

```
begin   comment Noordzee-probleem, Model Weenink.
                Opdracht TW 133/R1158, code-nr vdH 171164/195525;
        integer N, n, m;
        N:= read; n:= read; m:= read;
  begin integer           i, dt, J, DH1, DH2, DB, j, k;
        real              P, p, Q, q, d, f, r,
                          a1, a2, b1, b2, d1, d2, w, fw, gw, Cw, wx, wy;
        integer array     W, O[0:N + n + 1];
        real    array     A, B, C, u, v, z[0:m+1,0:N + n + 2],
                          SW, CW, SO, CO[0:N + n + 1];

        procedure COEFFICIENT(proc); procedure proc;
        begin   real h;
                proc(h); A[j,k]:= 1 - r × dt / h; B[j,k]:= 2 × h × Cw × p;
                h:= if k > n then h × Cw else 2 × h × Cw; C[j,k]:= h × q
        end;

        procedure coefficient(proc); procedure proc;
        begin   proc(A[j,k]); proc(B[j,k]); proc(C[j,k]) end;

        procedure ELINFORM(n1,n2,oost,PROC); value n1, n2, oost;
        integer n1, n2, oost; procedure PROC;
        begin   integer j, k;
                for k:= n1 step -1 until n2 do
                  for j:= 1 step 1 until oost do PROC(z[j,k])
        end;
```

```
procedure INFORMATION(COEFFICIENT, PROC, proc);
          procedure     COEFFICIENT, PROC, proc;
begin   integer west, oost;
        proc(dt); proc(J); proc(DH1); proc(DH2); proc(DB); proc(P);
        proc(p); proc(Q); proc(q); proc(d); proc(f); proc(r); proc(Cw);
        proc(a1); proc(a2); proc(b1); proc(b2); proc(d1); proc(d2);
        proc(w); proc(fw); proc(gw);
        west:= N + n + 1; O[DH2]:= DB;
        for k:= west step -1 until 0 do
        begin   proc(W[k]); proc(SW[k]); proc(CW[k]); proc(O[k]);
                proc(SO[k]); proc(CO[k]); oost:= O[k];
                for j:= W[k] step 1 until oost do COEFFICIENT(proc)
        end; O[DH2]:= DH1;
        for k:= 0 step 1 until west do
          for j:= 0 step 1 until m do PROC(u[j,k]);
        for k:= 0 step 1 until west do
          for j:= 0 step 1 until m do PROC(v[j,k]);
        ELINFORM(N + n + 2,1,m,PROC);
end;

procedure INPUT;
begin   procedure READ(a); real a; a:= read;
        procedure ZERO(a); real a; a:= 0;
        RUNOUT; PUNLCR; PUTEXT(⊰ Resultaten vdH 171164 / 19525.⊱);
        PUNLCR;
        i:= read;
        if i = 1 then INFORMATION(COEFFICIENT, ZERO, READ);
        if i > 1 then INFORMATION(coefficient, READ, READ);
        j:= N + n +2;
        for k:= 0 step 1 until j do z[0,k]:= z[m+1,k]:= 0;
end;
```

```
procedure COASTINF;
begin   integer s;
        ABSFIXP(2,2,i × dt / 3600); PUSPACE(2);
        s:= O[0];
        for j:= 1 step 1 until s do FIXP(1,3,z[j,1]);
        for k:= 1 step 1 until DH2 do FIXP(1,3,z[s + k,k]);
        for j:= DH1 + 1 step 1 until DB do FIXP(1,3,z[j,DH2 + 1]);
        s:= 20 + DH1 - s - DB;
        s:= if s < N + n + 1 then s else N + n + 1;
        for k:= DH2 + 1 step 1 until s do FIXP(1,3,z[O[k],k])
end;

procedure SEAINF;
begin   integer s;
        procedure PUNCH(a); real a; begin FIXP(2,3,a); PUSPACE(s) end;
        RUNOUT; PUNLCR; PUTEXT(⪡Waterstanden Noordzee na⪢);
        ABSFIXP(2,2,i × dt / 3600); PUTEXT(⪡uur⪢); PUNLCR; PUNLCR;
        s:= entier((150 - 8 × m) / m + .1); PUNLCR;
        ELINFORM(N +n +2,1,m,PUNCH); PUNLCR;
        RUNOUT; NLCR; print(i); print(z[DH1,DH2+1])
end;

procedure ZUIDKUST;
begin   integer j, jp1, oost;
        oost:= O[0];
        for j:= 1 step 1 until oost do
        begin   wind(j,0); jp1:= j + 1;
                u[j,0]:= A[j,0] × u[j,0]
                  - B[j,0] × (3 ×(z[jp1,1] - z[j,1]) - (z[jp1,2] - z[j,2]))
                  + dt × wx × Cw
        end;
        for j:= DH1 + 1 step 1 until DB do
        begin   wind(j,DH2); jp1:= j + 1;
```

```
                    u[j,DH2]:= A[j,DH2] × u[j,DH2]
                      - B[j,DH2] × (3 × (z[jp1,DH2+1] - z[j,DH2+1])
                      - (z[jp1,DH2+2] - z[j,DH2+2])) + dt × wx × Cw
        end
end;

procedure NOORDZEE(p,q,n1,n2); value n1, n2;integer n1,n2; real p, q;
begin   integer jm1, j, jp1, km1, k, kp1,
                west, wp1, om1, oost, op1, zuid;
        real    a, sw, so, cw, co,
                U1, U2, U3, V1, V2, V3, Z1, Z2, dZ1, dZ2;
        for k:= n1 step 1 until n2 do
        begin   west:= W[k]; sw:= SW[k]; cw:= CW[k];
                oost:= O[k]; so:= SO[k]; co:= CO[k];
                wind(west,k); wp1:= west + 1; op1:= oost + 1;
                om1:= oost - 1; kp1:= k + 1; km1:= k - 1; zuid:= W[km1];
                V1:= A[west,k] × v[west,k]
                  - cw × cw × (C[west,k] × (3 × (z[wp1,kp1] - z[wp1,k])
                  - (z[west+2,kp1] - z[west+2,k])) - dt × wy × Cw);
                if zuid < west then
                begin   Z1:= z[west,k];
                        V2:= v[west,k]:= V1 - sw × cw ×
                          (B[west,k] × 2 × (z[wp1,k] - Z1) - dt × wx × Cw);
                        U2:= u[west,k]:= sw × V2 / cw;
                        z[west,k]:= Z1 - p × 2 × (u[west,km1] - u[west - 1,km1])
                          - q × 2 × (V2 - v[west,km1])
                end;
                if zuid ≥ west then
                begin   V2:= v[west,k]:= V1 - sw × cw × (B[west,k] × 2 ×
                          (z[wp1,kp1] - z[west,kp1]) - dt × wx × Cw);
                        U2:= u[west,k]:= sw × V2 / cw
                end;
                if zuid > west then
```

```
begin   U2:= U2 + u[wp1,km1] - u[west,km1];
        V2:= v[west,km1]
end;
for j:= wp1 step 1 until om1 do
begin   wind(j,k); jp1:= j + 1; jm1:= j - 1;
        a:= A[j,k]; U1:= u[j,k]; U3:= U2; V1:= v[j,k];
        V3:= V2; Z1:= z[j,k]; dZ1:= z[jp1,kp1] - Z1;
        dZ2:= z[jp1,k] - z[j,kp1];
        U2:= u[j,k]:= a × U1 + d × V1
          - B[j,k] × (dZ1 + dZ2) + dt × wx × Cw;
        V2:= v[j,k]:= a × V1 - d × U1
          - C[j,k] × (dZ1 - dZ2) + dt × wy × Cw;
        z[j,k]:= Z1 - p × (U2 - U3 + u[j,km1] - u[jm1,km1])
          - q × (V2 - v[j,km1] + V3 - v[jm1,km1])
end;
if zuid > west then
z[wp1,k]:= z[wp1,k] - p × (u[wp1,k] - u[west,k])
  - q × (v[wp1,k] - v[wp1,km1]);
wind(oost,k); zuid:= O[km1]; Z2:= z[oost,k];
V1:= A[oost,k] × v[oost,k] - co × co × (C[oost,k] ×
  (3 × (z[oost,kp1] - Z2) -(z[om1,kp1] - Z1))
  - dt × wy × Cw);
V3:= V2; U3:= U2;
if zuid ≥ oost then
begin   V2:= v[oost,k]:= V1 - so × co × (B[oost,k] × 2 ×
          (z[op1,k] - Z2) - dt × wx × Cw);
        U2:= u[oost,k]:= so × V2 / co;
        z[oost,k]:= Z2 - p × (U2 - U3 + u[oost,km1]
          - u[om1,km1]) - q × (V2 - v[oost,km1] + V3
          - v[om1,km1])
end;
```

```
                if zuid > oost then
                z[op1,k]:= z[op1,k] - p × 2 × (u[op1,km1] - u[oost,km1])
                  - q × (3 × (V2 - v[oost,km1]) - (V3 - v[om1,km1]));
                if zuid < oost then
                begin   V2:= v[oost,k]:= V1 - so × co ×(B[oost,k] × (3 ×
                          (z[op1,kp1] - z[oost,kp1])
                          - (z[op1,k+2] - z[oost,k+2])) - dt × wx × Cw);
                        U2:= u[oost,k]:= so × V2 / co;
                        z[oost,k]:= Z2 - p × 2 × (U2 - U3) - q × (3 ×
                          (V3 - v[om1,km1]) - (v[oost-2,k] - v[oost-2,km1]))
                end
        end
end;

procedure CORRECTIE MODEL WEENINK;
begin   integer oost;
        for k:= 1 step 1 until DH2 do
        begin   oost:= O[k];
                z[oost,k]:= z[oost,k] - p × (u[oost,k] - u[oost-1,k]
                  - u[oost,k+1] + u[oost-1,k+1])
        end
end;

procedure wind(j,k); integer j,k;
begin   real Aw, Bw;
        if XEEN(511) > 256 ∧ k ≤ n + 1  then wx:= wy:= Cw:= 0 else
        begin   Aw:= sin(j×a1 - k×a2 - d1);Bw:= sin(j×a1 + k×a2 - d2);
                wx:= -gw×wx×(Aw+Bw); wy:=-fw×wx×(Aw-Bw);
                Cw:= 3₁₀-6 × sqrt(wx×wx + wy×wy)
        end
end;
```

```
procedure ROOSTEROVERGANG(n); value n; integer n;
begin   integer west, wp1, wp2, oost, om1, np1, np2, j, jp1, jm1;
        real    a, U, U1, U2, U3, V, V1, V2, V3, Z1, Z2, Z3, Z4;
        np1:= n + 1; np2:= n + 2; west:= W[np1]; wp1:= west + 1;
        wp2:= west + 2; oost:= O[np1]; om1:= oost - 1;
                wind(west,np1); Z1:= z[wp1,np2]; Z2:= z[wp1,np1]; U2:= 0;
        V2:= v[west,np1]:= A[west,np1] × v[west,np1] - (4/3) × C[west,np1] ×
          (3 × (Z1 - Z2) - (z[wp2,np2] - z[wp2,np1])) + dt × wy × Cw;
        for j:= 1 step 1 until om1 do
        begin   jp1:= j + 1; jm1:= j - 1; wind(j,np1); a:= A[j,np1];
                U1:= u[j,np1]; U3:= U2; V1:= v[j,np1]; V3:= V2;
                Z3:= Z1; Z4:= Z2; Z1:= z[jp1,np2]; Z2:= z[jp1,np1];
                U2:= u[j,np1]:= a × U1 + d × V1 - B[j,np1] ×
                  ((2/3) × (Z1 - Z3) + (4/3) × (Z2 - Z4)) + dt × wx × Cw;
                V2:= v[j,np1]:= a × V1 - d × U1 - C[j,np1] ×
                  ((4/3) × (Z1 - Z2) + (4/3) × (Z3 - Z4)) + dt × wy × Cw;
                z[j,np1]:= Z4 - p × (U2 - U3 + u[j,n] - u[jm1,n])
                  - q × (V2 - v[j,n] + V3 - v[jm1,n])
        end;
        wind(oost,np1); V3:= V2;
        V2:= v[oost,np1]:= A[oost,np1] × v[oost,np1] - (4/3) × C[oost,np1] ×
          (3 × (Z1 - Z2) - (Z3 - Z4)) + dt × wy × Cw;
        z[oost,np1]:= Z2 - p ×(-U2 + u[oost,n] - u[om1,n])
          - q × (V2 - v[oost,n] + v[om1,np1] - v[om1,n])
end;

procedure OUTPUT;
begin   procedure PROC(a); real a; FLOP(12,1,a);
        RUNOUT; PUNLCR; PUTEXT
        (⊰The following papertape can be used as inputtape⊱);
        PUNLCR; FIXP(2,0,N); ABSFIXP(2,0,n); ABSFIXP(2,0,m);
        ABSFIXP(3,0,i); INFORMATION(coefficient,PROC, PROC); TAPEND
end;
```

```
      INPUT; r:=  (n + 1) × a2 / 2;
 SEA: a2:= 0.5 × a2; d1:= d1 + r; d2:= d2 - r; ZUIDKUST;
      NOORDZEE(p,q,1,n);
      if N ≥ 1 then ROOSTEROVERGANG(n);
      a2:= 2.0 × a2; d1:= d1 - r; d2:= d2 + r; NOORDZEE(p,Q,n+2,N+n+1);
      CORRECTIE MODEL WEENINK;
      i:= i + 1; d1:= d1 + b1; d2:= d2 + b2; z[1,1] := z[1,1] / f;
      if XEEN(15)     = GCD(i,XEEN(15))      then COASTINF;
      if XEEN(15) × J = GCD(i,XEEN(15) × J)  then SEAINF;
      if XEEN(63) × J = GCD(i,XEEN(31) × J)  then
      begin   NLCR; PRINTTEXT
              (⊰BVA is followed by an inputtape for continuing the program⊱);
              stop; OUTPUT; RUNOUT; stop
      end;
      if XEEN(127) > 63  then goto SEA;
      if XEEN(255) > 127 then OUTPUT
   end
end
```

TABEL 1: h(x,y) in meters in the streampoints of the Weenink-model with grid-transition and (N,n,m) = (5,9,10)

200	200	200	200	200	200	200	200	200	200	200
150	175	175	175	175	175	175	200	200	200	200
100	150	150	150	150	150	150	175	200	200	200
60	100	100	100	100	100	125	150	175	200	200
40	60	100	100	100	100	100	125	150	175	200
40	60	100	100	100	100	100	60	60	60	40
40	60	60	60	60	60	60	60	60	40	40
40	60	60	40	40	40	40	40	40	40	20
40	60	40	40	40	40	40	40	40	20	20
20	40	40	40	40	40	40	40	40	20	20
20	20	40	40	40	40	40	40	40	20	20
20	20	40	40	40	40	40	40	40	20	20
20	20	40	20	20	20	20	20	20	20	20
20	20	40	20	20						
20	40	20	20							
40	20	20								

In the second tabel we have tabulated the elevation $\zeta(j,1)$, computed in a rectangular model with uniform depth of 65 m, a step-windfield $(U,V) = (0, -41^2.3.10^6)m^2/sec^2$ and a Coriolis-parameter $\Omega = 1{,}22.10^{-4}$/sec. We give in each point two values. The first value is obtained in a net (N,n,m) = (-1, 16, 8), the second value is obtained in a net (N,n,m) = = (4, 7, 8).

TABEL II t in hr	$\zeta(j,1)$ in meters in a rectangular model with and without a uniform net.							
3	2,09 2,09	1,96 1,96	1,81 1,81	1,66 1,66	1,53 1,53	1,42 1,42	1,34 1,34	1,29 1,29
6	4,11 4,11	3,89 3,90	3,62 3,63	3,31 3,32	2,98 2,99	2,69 2,70	2,43 2,44	2,25 2,25
9	5,30 5,47	5,16 5,32	5,04 5,20	4,96 5,11	4,88 5,03	4,79 4,94	4,69 4,85	4,65 4,80
12	6,36 6,71	6,22 6,54	6,09 6,37	5,95 6,20	5,84 6,06	5,75 5,95	5,69 5,87	5,67 5,82
15	6,14 6,52	6,08 6,43	6,06 6,39	6,05 6,40	6,05 6,44	6,03 6,45	5,99 6,44	5,97 6,41
18	6,53 6,82	6,44 6,71	6,36 6,64	6,32 6,63	6,31 6,65	6,32 6,65	6,34 6,62	6,36 6,59

References

[1] Van Dantzig, D. and H.A. Lauwerier, General considerations concerning the hydrodynamical problem of the motion of the North Sea. The North Sea Problem I. Proc. Kon. Ned. Ak. v. Wetensch. A 63, 170-180 (1960).

[2] Lauwerier, H.A., Influence of a stationary windfield upon a bay with a uniform depth. The North Sea Problem II. Proc. Kon. Ned. Ak. v. Wetensch. A 63, 266-278 (1960).

[3] Lauwerier, H.A., Influence of a stationary windfield upon a bay with an exponentially increasing depth. The North Sea Problem III. Proc. Kon. Ned. Ak. v. Wetensch. A 63, 279-290 (1960).

[4] Van Dantzig, D. and H.A. Lauwerier, Free oscillations of a rotating rectangular sea. The North Sea Problem IV. Proc. Kon. Ned. Ak. v. Wetensch. A 63, 339-354 (1960).

[5] Lauwerier, H.A., Free motions of a rotating rectangular bay. The North Sea Problem V. Proc. Kon. Ned. Ak. v. Wetensch. A 63, 423-438 (1960).

[6] Lauwerier, H.A., Non-stationary windeffects in a rectangular bay (Theoretical part). The North Sea Problem VI. Proc. Kon. Ned. Ak. v. Wetensch. A 64, 104-122 (1961).

[7] Lauwerier, H.A., Non-stationary windeffects in a rectangular bay (Numerical part). The North Sea Problem VII. Proc. Kon. Ned. Ak. v. Wetensch. A 64, 418-431 (1961).

[8] Lauwerier, H.A. and B.R. Damsté, A numerical treatment. The North Sea Problem VIII. Proc. Kon. Ned. Ak. v. Wetensch. A 66, 167-184 (1963).

[9] Fischer, G., Ein numerisches Verfahren zur Errechnung von Windsteu und Gegeiten in Randmeeren. Tellus, II, I (1959).

[10] Forsythe, G.E. and W.R. Wasow, Finite-difference methods for partial differential equations.

[11] O'Brien, G.G., M.A. Hyman and S. Kaplan, A study of the numerical solution of partial differential equations. J. of Math. and Phys. Vol. 29, 223-251 (1950).

[12] Rjabenki, V.S. and A.F. Filippow, Über die Stabilität von differenzengleichungen.

[13] Lax, P.D. and R.D. Richtmyer, Survey of the stability of linear finite difference equations. Comm. Pure Appl. Math. Vol. 9, 267-293 (1956).

[14] Weenink, M.P.H., A theory and method of calculation of wind effects on sea levels in a partly enclosed sea, with special application to the southern coast of the North Sea. Kon. Ned. Meteorologisch Instituut, Mededelingen en Verhandelingen vol.73.