INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 27, 81–95 (1998)

A PROJECTION METHOD FOR SHALLOW WATER EQUATIONS

M. MORANDI CECCHI,[1]* A. PICA[2] AND E. SECCO[1]

[1]*Dip. di Matematica Pura ed Applicata, Università di Padova, Via Belzoni 7, I-35100 Padova, Italy*;
[2]*Dip. di Matematica ed Informatica, Università Verona, Via delle Scienze 206, I-33100 Verona, Italy*

SUMMARY

The dynamics of shallow water has been studied and an algorithm for this dynamics has been developed. Results have been obtained with data of the Venice lagoon using a model made expressively by a semi-implicit method based on a finite element method in space. Comparison has been made between field data and the results of the simulation. Very good agreement is shown over a long period of simulation. © 1998 John Wiley & Sons, Ltd.

Int. J. Numer. Meth. Fluids, **25**: 81–95 (1998)

KEY WORDS: shallow waters; Navier–Stokes; lagoons; projections; parallel computations

1. INTRODUCTION

In this paper a model is presented to solve the problem of the water motion due to the tides in the Venice lagoon. The physical model considered is the long-wave model. The equations considered are the shallow water equations obtained by the Navier–Stokes equations reduced to a 2D approximation problem. The model is obtained considering different depths of the lagoon and in particular a finer discretization is considered where the water runs with greater speed, namely along the more important canals. A numerical method based on a finite element discretization is presented and numerical results are given. The model is shown to be very stable and capable of preserving the water quantity of the system. The wind effect is also considered and proves to be very influential on the behaviour of the model.

2. NAVIER–STOKES EQUATIONS

In this paper the shallow water equations are derived from the full 3D Navier–Stokes equilibrium and continuity equations for a viscous incompressible Newtonian flow under the following assumptions.

(a) The water elevation with respect to the free surface at rest is at any time much smaller than the depth.
(b) Vertical effects (acceleration and diffusion) are negligible.
(c) Horizontal velocities do not vary with the depth.

*Correspondence to M. Morandi Cecchi, Dip di Matematica Pura ed Applicata, Università di Padova, Via Belzoni 7, I-35100 Padova, Italy.

CCC 0271–2091/98/010081–15 $17.50 *Received May 1996*

During the complete process of derivation of the discretized set of solution equations, viscous and convective terms are neglected; they are eventually considered to some extent and treated in a separate section and the final contributions are included in the original system as additional terms. The technique adopted for the discretization and solution of the governing equations is based on a semi-implicit projection method. Such methods in shallow water analyses were introduced by Chorin[1] and Temam[2] and then successfully adopted at moderate Reynolds numbers by many authors. Because of the boundary conditions implicitly imposed on the pressure, the methods were thought to give from the theoretical point of view an incorrect answer for the pressure in the general case of non-periodic boundary conditions; in contrast, numerical results showed good approximations for the pressure field. Indeed, studies carried out by Gresho,[3] Rannacher,[4] Shen[5] and finally Schwab[6] led to the proof that the pressure obtained by these methods asymptotically converges to the correct answer as the integration time step tends to zero.

Let $\Omega \subset \mathbb{R}^3$ be a bounded domain with a sufficiently smooth boundary $\Gamma = \partial\Omega$. The Navier–Stokes equations governing the dynamical behaviour of an incompressible Newtonian flow in Ω are given by

$$\frac{\partial}{\partial t}\mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p - \nu\nabla^2\mathbf{u} = q, \qquad \nabla \cdot \mathbf{u} = 0, \tag{1}$$

where $\mathbf{u} = u_i(x_1, x_2, x_3, t)$ and $p = p(x_1, x_2, x_3, t)$ are velocities and pressure respectively, ν is the viscosity coefficient, q collects all external forces and t denotes the time.

These equations are to be integrated in the space–time domain $\Omega \times]0, T[$, with T a prefixed time, once an appropriate set of initial and boundary conditions has been defined. While the former conditions in general only need to reproduce a feasible shape of the current solution, the latter have to be set and treated with much care owing to the effect they can have on the evolution process.

A typical approach consists of prescribing the boundary velocities, say

$$u|_\Gamma = b(x_\Gamma, t), \tag{2}$$

with $x_\Gamma \in \Gamma$, allowing, incidentally, for the treatment of globally moving systems. This is definitely realistic from a physical point of view but computationally not very convenient; therefore most usually (2) is decomposed into normal and tangential components, i.e.

$$n \cdot u|_\Gamma = n \cdot b(x_\Gamma, t), \tag{3}$$

$$u - n \cdot u|_\Gamma = b - n \cdot b(x_\Gamma, t), \tag{4}$$

where n is the outward normal to the boundary Γ. The latter represents the so-called *no-slip condition*.

3. SHALLOW WATER EQUATIONS

We consider the case where the wavelength is assumed to be greater than the water depth H (see Figure 1). Under this hypothesis the vertical velocity and acceleration are negligible and the flux becomes almost horizontal. Having performed the vertical integration of the third dimension,

$\eta = \eta(x_1, x_2)$ is the height of the free surface,

$u_1 = u_1(x_1, x_2)$ is the velocity component along the x_1-direction,

$u_2 = u_2(x_1, x_2)$ is the velocity component along the x_2-direction,

averaged over the depth.

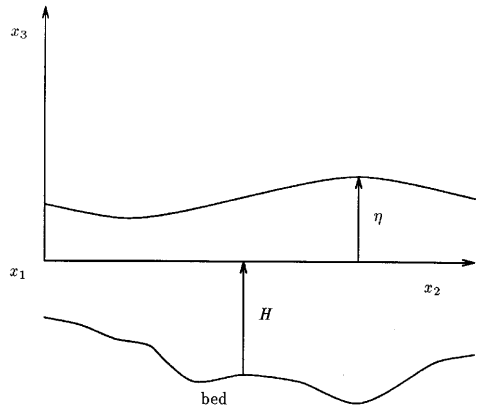

Figure 1. Shallow water scheme

The shallow water equations in conservative form with respect to the Cartesian system of spatial co-ordinates x_1 and x_2 become in compact form[7,8]

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} = \mathbf{R}, \tag{5}$$

where $\mathbf{U} = u(\mathrm{x},\, t) = [h, hu_1, hu_2]^{\mathrm{T}}$,

$$\mathbf{F}_1 = \begin{pmatrix} hu_1 \\ hu_1^2 + \frac{1}{2}g(h^2 - H^2) \\ hu_1u_2 \end{pmatrix}, \qquad \mathbf{F}_2 = \begin{pmatrix} hu_2 \\ hu_1u_2 \\ hu_2^2 + \frac{1}{2}g(h^2 - H^2) \end{pmatrix},$$

$$\mathbf{R} = \mathbf{R}_{\mathrm{s}} + \frac{\partial \mathbf{R}_{\mathrm{d1}}}{\partial x_1} + \frac{\partial \mathbf{R}_{\mathrm{d2}}}{\partial x_2},$$

with

$$\mathbf{R}_{\mathrm{s}} = \begin{pmatrix} 0 \\ fhu_2 + g(h-H)\dfrac{\partial H}{\partial x_1} - \dfrac{g|u|u_1}{C^2H} \\ -fhu_1 + g(h-H)\dfrac{\partial H}{\partial x_2} - \dfrac{g|u|u_2}{C^2H} \end{pmatrix},$$

$$\mathbf{R}_{\mathrm{d1}} = \begin{pmatrix} 0 \\ \dfrac{2\mu_{\mathrm{H}}h}{\rho}\dfrac{\partial u_1}{\partial x_1} \\ \dfrac{\mu_{\mathrm{H}}h}{\rho}\left(\dfrac{\partial u_2}{\partial x_1} + \dfrac{\partial u_1}{\partial x_2}\right) \end{pmatrix}, \qquad \mathbf{R}_{\mathrm{d2}} = \begin{pmatrix} 0 \\ \dfrac{\mu_{\mathrm{H}}h}{\rho}\left(\dfrac{\partial u_2}{\partial x_1} + \dfrac{\partial u_1}{\partial x_2}\right) \\ \dfrac{2\mu_{\mathrm{H}}h}{\rho}\dfrac{\partial u_2}{\partial x_2} \end{pmatrix}.$$

Where $h(x_1, x_2, t) = H(x_1, x_2) + \eta(x_1, x_2, t)$ is the total height of the fluid, $u_1 = u_1(x_1, x_2, t)$ and $u_2 = u_2(x_1, x_2, t)$ are the components of the velocity with respect to the co-ordinate axes, g is the acceleration due to gravity, C and f are the Chezy and Coriolis coefficients respectively and μ_{H} and ρ are the viscosity coefficient of turbulence and the density respectively, having omitted the atmospheric pressure. In (5), $\mathbf{F}_{1,2} = \mathbf{f}_{1,2}(\mathbf{U}, x)$ indicate the flux vectors in the co-ordinate directions x_1

and x_2 respectively, while $\mathbf{R}_s = r_s(\mathbf{U}, x)$ is the load vector and $\mathbf{R}_{di}$, $i = 1, 2$, indicate the diffusion fluxes. If the equations in (5) without consideration of diffusivity are integrated over Ω, a domain in $\mathbb{R}^2$, by use of the divergence theorem one has

$$\frac{\partial}{\partial t}\left(\int_\Omega \mathbf{U}\,\mathrm{d}\Omega\right) = -\int_\Gamma (\mathbf{F}_1 n_1 + \mathbf{F}_2 n_2)\,\mathrm{d}\Gamma + \int_\Omega \mathbf{R}_s\,\mathrm{d}\Omega, \tag{6}$$

where $n = (n_1, n_2)$ is the outward normal to $\Gamma = \partial\Omega$.

Expression (6) shows that the rate of accumulation of $\mathbf{U}$ into Ω is given by the integral of the normal flux along the boundary Γ and of the quantity $\mathbf{U}$ 'generated' inside Ω. Under the long-wave hypothesis, omitting advection and viscous terms, one obtains the shallow water system[9]

$$\begin{aligned} \frac{\partial \eta}{\partial t} + \frac{\partial H u_1}{\partial x_1} + \frac{\partial H u_2}{\partial x_2} &= 0, \\ \frac{\partial u_1}{\partial t} + g\frac{\partial \eta}{\partial x_1} &= f u_2 - \frac{g\sqrt{(u_1^2 + U_2^2)}}{C^2 H} u_1, \\ \frac{\partial u_2}{\partial t} + g\frac{\partial \eta}{\partial x_2} &= -f u_1 - \frac{g\sqrt{(u_1^2 + u_2^2)}}{C^2 H} u_2. \end{aligned} \tag{7}$$

Here, differently from system (5), the non-linearity is present only on the right-hand side of system (7); it is therefore a semi-linear system, while (5) gives the non-linear model of the dynamical behaviour of shallow water (for further details see References 10–13).

According to the type of problem to be solved in a domain $\Omega \subset \mathbb{R}^2$, the system of equations (5) has convenient initial and boundary conditions. To study the tidal motion in the Venice lagoon, the semi-linear model (7) has been considered, omitting the convective terms in the flux vectors. The application to this model of the explicit Taylor–Galerkin method has given good results; therefore this model has been assumed with convenient boundary conditions as the starting point of the simulation of the tidal motion.

4. DOMAIN DECOMPOSITION METHODS

To solve the shallow water equations, it is necessary to use sophisticated numerical methods because of the large number of nodes that are involved. Different approaches have been used by iterative methods such as conjugate gradient and Cholewski decomposition. The domain decomposition method has also attracted great interest according to different ways followed by different authors. We observe that recently there has been an increase in research activities in the area of parallel computing owing to the advent and growth of various parallel processing architectures. The domain decomposition approach can achieve the highest level of parallelism in the numerical solution of partial differential equations. The idea of domain decomposition goes back to Schwarz,[14,15] but only in recent years have researchers developed and extended these ideas to use them with parallel architectures.

The domain decomposition technique can be employed for the solution of problems defined on irregular domains, using the same equations to be solved on all domains.[16,17] This technique allows the use of different numerical schemes and different resolution or different types of elements for the finite element method.[18] The domain decomposition technique can also be used if the problem requires different mathematical models for different subdomains, as often happens in fluid dynamics using a viscous model near the boundary and an inviscid model in the far field.[19,20]

For the domain decomposition we have both overlapping and non-overlapping approaches. For the non-overlapping method, most of the applications up to now have been developed for the interfaces

or, more specifically, to find good preconditioners of the conjugate gradient algorithm[21] for the linear system arising from the discretization of elliptic partial differential equations in two- or three-dimensional domains.[22,23]

Sometimes it is convenient to follow another approach which constructs a decomposed domain preconditioner for the simultaneous iterative procedure on the whole domain. One of the advantages of this approach is that only approximate subdomain solvers are required. This alternative approach was first proposed in References 24 and 25 and further studied in References 26 and 27. Another work along this line on shallow water is Reference 28. Another formulation is given in Reference 29, where a set of first-order non-linear hyperbolic partial differential equations is written for shallow water problems which has very important applications in meteorology and oceanography. These equations can be used to study tides and surface water run-off. They can also be used to study large-scale waves in the atmosphere and ocean if terms representing the effects of the earth's rotation (Coriolis terms) are included. The equations used in this case are the two-dimensional shallow water equations with the Chezy term neglected. These equations are solved using a Galerkin finite element method for the primitive equations of numerical weather prediction.[30] This method gives a good numerical solution in meteorological problems. For the solution the Schur complement is used.[29]

It is well known that in the finite element method the internal degree of freedom can be condensed at the element level before the assembly process, and if the same procedure is applied in a subregion, it can be considered as a new structure at the subregion level. In this procedure, two classes of variables are usually identified, namely the internal variables of nodes within subdomains and the interface variables relevant to nodes belonging to two or more subdomains. When the discretization for finite elements is made, the internal variables can be numbered either before or after the interface ones.

The numerical integration method of the shallow water equations for the Venice lagoon will be discussed in detail in the next section. Here we discuss the domain decomposition of the lagoon because we are particularly interested in the computational speed-up resulting when using the domain decomposition technique. We consider the subdivision of the domain Ω under consideration in the following way: Ω is subdivided into two subdomains Ω_1 and Ω_2, with $\Omega = \Omega_1 \cup \Omega_2 \cup \Gamma$ and $\Omega_1 \cap \Omega_2 = 0$, where Γ is the boundary between the two regions Ω_1 and Ω_2.[9,31] Let us denote the original nodal numbers as the old numbering, while the nodal numbers after renumbering (namely the substructure numbering) will be denoted as the new numbering. This becomes cumbersome if we think of the finite element numbering method in which the relationship among global nodes, local nodes and element numbering turns out to be crucial in the numerical solution of the matrix corresponding to the problem. Better results can be obtained by using various mesh resolutions, mostly dealing with local refinements in the neighbourhood of singularities in the definition of the boundary or near the mouths.

Therefore we use the following rule: the node indices are renumbered in sequence starting with the region Ω_1, following with the region Ω_2 and finally with the boundary Γ. As a consequence of this rule the matrix A of the linear system to be solved,

$$Ax = f,$$

has the form

$$A = \begin{pmatrix} A_{11} & 0 & A_{13} \\ 0 & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix},$$

Table I

	Case 1	Case 2
External nodes	508	350
Internal nodes	251	381
Boundary nodes	131	159
Total time (s)	635	581
Initialization time (s)	170	190
Iteration time (s)	464	390

which is compatible with the topology of the central lagoon. Therefore it would have been too difficult to think of a method with overlapping regions and almost inextensible to such a complicated area.

The nature of the method itself makes it possible to apply any order of decomposition[11,29] necessary to extend this methodology to the entire lagoon. This opens the way to the use of parallel computers; in fact, the times required to make sequential calculations are comparable with those obtained on the same machine with the conjugate gradient solver, which has given in this case from 2 to 3 s for initialization and 351 s for complete solution of the same simulation time with identical time step. Therefore, if such advantages are obtained with sequential calculation, a greater advantage is expected on the global evaluation time owing the use of parallel calculation. The topological properties of the domain also suggest using the domain decomposition to obtain the advantage of evaluating together regions of comparable physical characteristics. Nevertheless, the domain decomposition has to be realized conveniently, namely each subdomain should have almost the same number of nodes and the internal boundaries should have a number of nodes as small as possible. Two cases are considered, for example, in the case of subdivision into only two subdomains (see Table I). As can be seen, Case 1 is unbalanced and Case 2 is balanced.

The time step for the implicit scheme applied to the central lagoon of Venice has been evaluated every 5 min for a space discretization of 890 nodes for the central lagoon. To be able to make a faster model, the domain decomposition idea has been applied to this such region. The choice of the global time step is then based on the triangles of smaller size along the canals of greater depth, although a larger time step would be required in those areas where a coarser mesh was sufficient. Therefore the possibility of using domain decomposition is extremely important, allowing us to subdivide the region studied so as to treat with a Courant number sufficiently homogeneous and to save CPU time.

In the case of the total Venice lagoon we decompose it into 10 subdomains and the nodes of the interfaces are collected into a unique set. The number of nodes of the entire discretization is 1967 and there are 3423 elements. The results of the calculation obtained using the domain decomposition method without parallelization have a speed-up of approximately 25% over the conjugate gradient solver used on a non-decomposed domain (see Plate 1).

5. SEMI-IMPLICIT SCHEME

In order to solve the shallow water equations of the Venice lagoon, a projection method is adopted here. This leads to a split or fractional step model as follows.

Equations (7) can be written in vector form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} = \mathbf{R}_s, \tag{8}$$

where

$$\mathbf{U} = [\eta, u_1, u_2]^{\mathrm{T}}, \qquad \mathbf{F}_1 = [Hu_1, g\eta, 0]^{\mathrm{T}}, \qquad \mathbf{F}_2 = [Hu_2, 0, g\eta]^{\mathrm{T}},$$
$$\mathbf{R}_{\mathrm{s}} = \left[0,\, fu_2 - \frac{g|u|u_1}{C^2H},\; -fu_1 \frac{g|u|u_2}{C^2H}\right]^{\mathrm{T}}.$$

By taking

$$\mathbf{F}_i = \mathbf{F}_i^* + \mathbf{F}_i^{**} \quad (i = 1, 2), \qquad \mathbf{U}^{(n+1)} = \mathbf{U}^{(n)} + \Delta\mathbf{U}^* + \Delta\mathbf{U}^{**},$$

with $\mathbf{F}_i^* = 0$, $\mathbf{F}_i^{**} = \mathbf{F}_i$ and $\Delta\mathbf{U}^*$ and $\Delta\mathbf{U}^{**}$ the corresponding increments in the solution vector from time step n to step $n+1$, system (8) can be split into

$$\frac{\partial \Delta\mathbf{U}^*}{\partial t} = \mathbf{R}_{\mathrm{s}} \qquad \left(\frac{\partial \Delta\mathbf{U}^*}{\partial t} + \frac{\partial \mathbf{F}_1^*}{\partial x_1} + \frac{\partial \mathbf{F}_2^{**}}{\partial x_2} = \mathbf{R}_{\mathrm{s}}\right), \tag{9}$$

$$\frac{\partial \Delta\mathbf{U}^{**}}{\partial t} + \frac{\partial \mathbf{F}^{**}_1}{\partial x_1} + \frac{\partial \mathbf{F}^{**}_2}{\partial x_2} = \mathbf{0}. \tag{10}$$

The semi-implicit scheme consists of solving alternate (9) by an explicit time integration procedure and (10) by an implicit method. It can be observed that owing to the form of the forcing vector $\mathbf{R}_{\mathrm{s}}$, equation (9) never causes any variation in the elevation η.

6. TIME AND SPACE DISCRETIZATION

6.1. Equation (9)

The discretization in time of equation (9) is obtained by a second-order Taylor expansion, which for a time step Δt, gives

$$(\Delta\mathbf{U}^*)^{(n+1)} = \Delta t\left(\frac{\partial \Delta\mathbf{U}^*}{\partial t}\right)^{(n)} + \frac{\Delta t^2}{2}\left(\frac{\partial^2 \Delta\mathbf{U}^*}{\partial t^2}\right)^{(n).} \tag{11}$$

According to (10) and the obvious relationships

$$\frac{\partial \Delta\mathbf{U}^*}{\partial t} = \mathbf{R}_{\mathrm{s}}, \qquad \frac{\partial^2 \Delta\mathbf{U}^*}{\partial t^2} = \frac{\partial \mathbf{R}_{\mathrm{s}}}{\partial t} = \frac{\partial \mathbf{R}_{\mathrm{s}}}{\partial \Delta\mathbf{U}^*}\frac{\partial \Delta\mathbf{U}^*}{\partial t}, \tag{12}$$

system (11) can be rewritten as

$$(\Delta\mathbf{U}^*)^{(n+1)} = \Delta t(\mathbf{R}_{\mathrm{s}})^{(n)} + \frac{\Delta t^2}{2}(\mathbf{G}\mathbf{R}_{\mathrm{s}})^{(n)}, \quad \text{where } \mathbf{G} = \frac{\partial \mathbf{R}_{\mathrm{s}}}{\partial \Delta\mathbf{U}^*}. \tag{13}$$

Because of the computational complexity in the evaluation of the rightmost term of equation (13), we use a two-step version of the Taylor–Galerkin algorithm.[32] This is given by an approximation of $\mathbf{U}^{(n+1/2)}$ and $(\mathbf{R}_{\mathrm{s}})^{(n+1/2)}$ with the Taylor expansions

$$\mathbf{U}^{(n+1/2)} = \mathbf{U}^{(n)} + \frac{\Delta t}{2}(\mathbf{R}_{\mathrm{s}})^{(n)},$$

$$(\mathbf{R}_{\mathrm{s}})^{(n+1/2)} = (\mathbf{R}_{\mathrm{s}})^{(n)} + \frac{\Delta t}{2}\left(\frac{\partial \mathbf{R}_{\mathrm{s}}}{\partial t}\right)^{(n)} = (\mathbf{R}_{\mathrm{s}})^{(n)} + \frac{\Delta t}{2}(\mathbf{G}\mathbf{R}_{\mathrm{s}})^{(n)},$$

from which we obtain

$$(\mathbf{GR}_s)^{(n)} = \frac{2}{\Delta t}[(\mathbf{R}_s)^{(n+1/2)} - (\mathbf{R}_s)^{(n)}].$$

If the point collocation method based on the mesh nodes is adopted for the discretization in space of system (13), this reduces to a set of uncoupled equations which can be easily solved by a simple evaluation of the right-hand side.

6.2. Equation (10)

For the treatment of equation (10) it is convenient to write it in the extended form

$$\begin{aligned} \frac{\partial}{\partial t}(\Delta\eta^{**}) + \frac{\partial}{\partial x_1}(Hu_1) + \frac{\partial}{\partial x_2}(Hu_2) = 0, \\ \frac{\partial}{\partial t}(\Delta u_1^{**}) + \frac{\partial}{\partial x_1}(g\eta) = 0, \\ \frac{\partial}{\partial t}(\Delta u_2^{**}) + \frac{\partial}{\partial x_2}(g\eta) = 0. \end{aligned} \tag{14}$$

This is discretized in time according to the θ-method proposed in References 10 and 33, giving

$$\begin{aligned} (\Delta\eta^{**})^{(n+1)} + \Delta t\left(\frac{\partial}{\partial x_1}(Hu_1^{(n+\theta_1)}) + \frac{\partial}{\partial x_2}(Hu_2^{(n+\theta_1)})\right) = 0, \\ (\Delta u_1^{**})^{(n+1)} = \Delta t\frac{\partial}{\partial x_1}p^{(n+\theta_2)} = 0, \\ (\Delta u_2^{**})^{(n+1)} + \Delta t\frac{\partial}{\partial x_2}p^{(n+\theta_2)} = 0, \end{aligned} \tag{15}$$

where $p = g\eta$, θ_1 and θ_2 are real parameters in [0,1] and

$$\begin{aligned} u_i^{(n+\theta_1)} = u_i^{(n)} + \theta_1[(\Delta u_i^{*})^{(n+1)} + (\delta u_i^{**})^{(n+1)}], \quad i = 1, 2, \\ p^{(n+\theta_2)} = p^{(n)} + \theta_2(\Delta p^{**})^{(n+1)}. \end{aligned} \tag{16}$$

It should be observed that the term $(\Delta p^{*})^{(n+1)}$ does not appear here since it is always null, as remarked in the previous section.

Replacing now expressions (16) in (15), this becomes

$$\begin{aligned} &\frac{1}{g}(\Delta p^{**})^{(n+1)} + \Delta t\theta_1\sum_{i=1}^{2}\frac{\partial}{\partial x_i}[H(\Delta u_1^{**})^{(n+1)}] = -\Delta t\sum_{i=1}^{2}\frac{\partial}{\partial x_i}(Hu_1^{(n)}) + \Delta t\theta_1\sum_{i=1}^{2}\frac{\partial}{\partial x_i}[H(\Delta u_1^{*})^{(n+1)}], \\ &(\Delta u_1^{**})^{(n+1)} + \Delta t\theta_2\frac{\partial}{\partial x_1}[(\Delta p^{**})^{(n+1)}] = -\Delta t\frac{\partial}{\partial x_1}p^{(n)}, \\ &(\Delta u_2^{**})^{(n+1)} + \Delta t\theta_2\frac{\partial}{\partial x_2}[(\Delta p^{**})^{(n+1)}] = -\Delta t\frac{\partial}{\partial x_2}p^{(n)}, \end{aligned} \tag{17}$$

which allows $(\Delta u_1^{**})^{(n+1)}$ and $(\Delta u_2^{**})^{(n+1)}$ to be obtained from the second and third equations in terms of the increment in pressure and of the pressure at the previous time step. Substitution of these expressions into the first equation of (17) leads to

$$\frac{1}{g}(\Delta p^{**})^{(n+1)} - \Delta t^2\theta_1\theta_2\sum_{i=1}^{2}\frac{\partial}{\partial x_i}\left(H\frac{\partial}{\partial x_i}(\Delta p^{**})^{(n+1)}\right) = -\Delta t\left[\sum_{i=1}^{2}\frac{\partial}{\partial x_i}\{H[u_i^{(n)} + \theta_1(\Delta u_i^*)^{(n+1)}]\} - \Delta t\theta_1\sum_{i=1}^{2}\frac{\partial}{\partial x_i}\left(H\frac{\partial}{\partial x_i}p^{(n)}\right)\right], \tag{18}$$

where the only unknown is the increment in pressure, $(\Delta p^{**})^{(n+1)}$.

The discretization in space is performed here using linear triangular elements. After application of Green's formula to the term including second derivatives in the variational (Galerkin) form of (10), the linear system of equations

$$\left(\frac{1}{g}\mathbf{M} + \Delta t^2\theta_1\theta_2\mathbf{S}\right)(\Delta p^{**})^{(n+1)} = -\Delta t\left(\sum_{i=1}^{2}\mathbf{Q}_i\{H[u_1^{(n)} + \theta_1(\Delta u_i^*)^{(n+1)}]\} + \Delta t\theta_1\mathbf{S}p^{(n)}\right) \tag{19}$$

is obtained, where

$$\mathbf{S} = \sum_{i=1}^{2}\left(\int_\Omega \frac{\partial[\phi]}{\partial x_i}H\frac{\partial[\phi]^{\mathrm{T}}}{\partial x_i}\,\mathrm{d}\Omega\right), \qquad \mathbf{Q}_i = \int_\Omega [\phi]\frac{\partial[\phi]^{\mathrm{T}}}{\partial x_i}\,\mathrm{d}\Omega. \tag{20}$$

Here ϕ is the vector of shape functions at all nodes in the mesh and the same notation has been adopted for the nodal variables as for the corresponding functions in (18).

Once the increment in pressure has been evaluated from (19), it can be replaced in the second and third equations of (17), which in discretized form is given by

$$\mathbf{M}(\Delta u_1^{**})^{(n+1)} = -\Delta t\mathbf{Q}_i[p^{(n)} + \theta_2(\Delta p^{**})^{(n+1)}], \quad i = 1, 2, \tag{21}$$

in order to obtain the velocity sub-increments $(\Delta u_1^{**})^{(n+1)}$ and $(\Delta u_2^{**})^{(n+1)}$.

7. SOLUTION PROCESS

Summarizing what has been seen in the previous section, the solution process involves, for every single time increment, the following steps.

(a) Evaluate $(\Delta u_1^*)^{(n+1)}$ and $(\Delta u_2^{**})^{(n+1)}$ (here $(\Delta p^*)^{(n+1)} = 0$ from (13)).
(b) Solve (19) for $(\Delta\eta^{**})^{(n+1)}$ (obtained as $(1/g)\,(\Delta p^{**})^{(n+1)}$).
(c) Solve the pair of (apparently) uncoupled systems (21) for $(\Delta u_1^{**})^{(n+1)}$ and $(\Delta u_2^{**})^{(n+1)}$.
(d) Evaluate the new solution vector as $\mathbf{U}^{(n+1)} = \mathbf{U}^{(n)} + (\Delta\mathbf{U}^*)^{(n+1)} + (\Delta\mathbf{U}^{**})^{(n+1)}$.

From a computational point of view, systems (13), (19) and (21) have been treated in different ways from each other.

7.1. Solution of (13)

This set of equations has been obtained, as anticipated, by using the point collocation method for the discretization, owing to the complexity of a finite element representation of the right-hand side. Therefore the system has been solved by simply setting the values of the variables to the corresponding values of the right-hand side evaluated at the nodes of the mesh used later for the other equations.

7.2. Solution of (19)

In this case the conjugate gradient method has been adopted in order to contain the storage requirements, since it allows all quantities needed during an iteration to be evaluated at the element level and then assembled in vector form. Both a standard and a preconditioned version of the algorithm have been used with success, showing very fast convergence to the rather strict limit of 10^{-5} for the ratio between the norms of the current residual vector and of the forcing vector at the start of the conjugate gradient iterations.

7.3. Solution of (21)

The matrix of this pair of systems is the usual consistent mass matrix (density equal to one) which comes out in all dynamic analyses, where it has often been successfully replaced by a diagonal lumped version. This analogy has suggested the following iterative method for the solution of (21):

$$[(\Delta u_1^{**})^{(n+1)}]^{r+1} = \mathbf{M}_\mathrm{L}^{-1}\{f_i^{(n)} - (\mathbf{M} - \mathbf{M}_\mathrm{L})[(\Delta u_2^{**})^{(n+1)}]^r\}, \quad r = 0, 1, 2, \ldots, \quad i = 1, 2,$$

where $\mathbf{M}_\mathrm{L}$ is the diagonal matrix obtained from $\mathbf{M}$ by row summation, $f_i^{(n)}$ is the right-hand side of the ith system and suffixes r and $r+1$ stand for two subsequent iterations of the solution process. Usually the velocity sub-increments are assumed to be null at $r = 0$.

This method is based on a decomposition of the mass matrix as

$$\mathbf{M} = \mathbf{M}_\mathrm{L}^{1/2}\{\mathbf{I} + [\mathbf{M}_\mathrm{L}^{-1/2}(\mathbf{M} - \mathbf{M}_\mathrm{L})\mathbf{M}_\mathrm{L}^{-1/2}]\}\mathbf{M}_\mathrm{L}^{1/2}$$

and can be shown to be convergent under the assumption that the norm of the matrix in square brackets is less than one, as in the present case.

It must be pointed out here that the two systems (21) are only apparently uncoupled in the present context. As a matter of fact the presence of the boundary conditions (2), in this case represented by a request of zero velocity in the direction normal to the boundary, can be dealt with in a simple way only by projecting the velocity vectors at the boundary nodes onto the tangent to the boundary; this requires that the two solution processes be carried on simultaneously and the projection be performed at each iteration. The 'inherent' error in the equilibrating forces inevitably introduced in this way (the same also happens for system (13)) has suggested the use of the ratio between the norms of the variation of velocity and of the total velocity sub-increment for the check of convergence; also in this case, very fast convergence can be reported for a tolerance of 10^{-3}.

8. DIFFUSION, CONVECTION AND WIND TRACTIONS

It can be observed that so far the terms related to diffusion and convection have been ignored. This has been done in part for the sake of simplicity in the derivation of the systems of equations and in part because it had been found that their influence on the computed solution was practically insignificant over a period of several hours. In contract, wind tractions seem to be rather important in the area of interest, the Venice lagoon, and could not be ignored in the present context. All these terms have been included in the analysis programme to some extent according to the following discussion.

8.1. Diffusion

The viscous term is not totally ignored. In fact it appears in the friction term with the bed. This term is obtained by integration of the viscous term in the vertical direction under the condition of vanishing velocity on the bed; namely, the conditions imposed are those of no-slip. The same

conditions would be necessary to show turbulence at the boundary. The reason for this treatment is the different order of magnitude of the proposed problem. We should remember that the shallow water equations are defined to study transport phenomena where the vertical dimension is small with respect to the other dimensions. The consequence of this fact is that at every point of the domain the bed is much closer than the boundary, the only exception being a strip of width equal to the depth defined along the boundary. This strip is too narrow to discretize it with the same mesh of the system. For this reason the viscous term has no meaning any more and therefore has been omitted.

8.2. *Convection*

Omitting the complete derivation from their 3D expression, the convective terms result to be additional to the vectors $\mathbf{F}_1$ and $\mathbf{F}_2$ defined in (8) and are given by

$$\mathbf{F}_1^{\mathrm{c}} = [0, hu_1^2, hu_1u_2]^{\mathrm{T}}, \qquad \mathbf{F}_2^{\mathrm{c}} = [0, hu_1u_2, hu_2^2]^{\mathrm{T}}.$$

Acting on these in the same way as for $\mathbf{F}_1$ and $\mathbf{F}_2$, their contribution to (8) can be expressed as

$$\mathbf{R}_{\mathrm{s}}^{\mathrm{c}} = \begin{bmatrix} 0 \\ 2u_1\dfrac{\partial u_1}{\partial x_1} + \dfrac{u_1^2}{H}\dfrac{\partial H}{\partial x_1} + u_1\dfrac{\partial u_2}{\partial x_2} + u_2\dfrac{\partial u_1}{\partial x_2} + \dfrac{u_1u_2}{H}\dfrac{\partial H}{\partial x_2} \\ 2u_2\dfrac{\partial u_2}{\partial x_2} + \dfrac{u_2^2}{H}\dfrac{\partial H}{\partial x_2} + u_1\dfrac{\partial u_2}{\partial x_1} + u_2\dfrac{\partial u_1}{\partial x_1} + \dfrac{u_1u_2}{H}\dfrac{\partial H}{\partial x_1} \end{bmatrix}.$$

Owing to the high non-linearity introduced by this expression and following the linearization process implicitly adopted in treating equation (9), the above vector is moved to the right-hand side of that equation, which now becomes

$$\frac{\partial \Delta \mathbf{U}^*}{\partial t} = \mathbf{R}_{\mathrm{s}} - \mathbf{R}_{\mathrm{s}}^{\mathrm{c}}.$$

Therefore the solution system (13) is given in this case by

$$(\Delta \mathbf{U}^*)^{(n+1)} = \Delta t(\mathbf{R}_{\mathrm{s}} - \mathbf{R}_{\mathrm{s}}^{\mathrm{c}})^{(n)} + \frac{\Delta t^2}{2}[(\mathbf{G} - \mathbf{G}^{\mathrm{c}})(\mathbf{R}_{\mathrm{s}} - \mathbf{R}_{\mathrm{s}}^{\mathrm{c}})]^{(n)},$$

where

$$\mathbf{G}^{\mathrm{c}} = \frac{\partial \mathbf{R}_{\mathrm{s}}^{\mathrm{c}}}{\partial \Delta \mathbf{U}^*}.$$

In terms of splittings the same result would have been obtained by starting from the modified equation (8) given by

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial}{\partial x_1}(\mathbf{F}_1 + \mathbf{F}_1^{\mathrm{c}}) + \frac{\partial}{\partial x_2}(\mathbf{F}_2 + \mathbf{F}_2^{\mathrm{c}}) = \mathbf{R}_{\mathrm{s}},$$

then deriving the new equation (19) as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}_1}{\partial x_1} + \frac{\partial \mathbf{F}_2}{\partial x_2} + \frac{1}{H}\left(\frac{\partial \mathbf{F}_1^{\mathrm{c}}}{\partial x_1} + \frac{\partial \mathbf{F}_2^{\mathrm{c}}}{\partial x_2}\right) = \mathbf{R}_{\mathrm{s}},$$

with $\mathbf{F}_i^* = \mathbf{F}_i^{\mathrm{c}} = 0$, and finally assuming

$$\mathbf{R}_{\mathrm{s}}^{\mathrm{c}} = \frac{1}{H}\left(\frac{\partial \mathbf{F}_1^{\mathrm{c}}}{\partial x_1} + \frac{\partial \mathbf{F}_2^{\mathrm{c}}}{\partial x_2}\right).$$

8.3. Wind tractions

According to Reference 34, the forces due to a wind blowing on the water free surface at a speed v, under the assumption that v is much larger than the water velocity, are proportional to a dimensionless constant ξ determined experimentally and to the square of v and inversely proportional to the water depth h. More precisely, their contribution to the right-hand side of (8) can be expressed as

$$t_w = [0, \xi|v|v_1, \xi|v|v_2]^T.$$

where v_1 and v_2 are the components of v along the two co-ordinate axes.

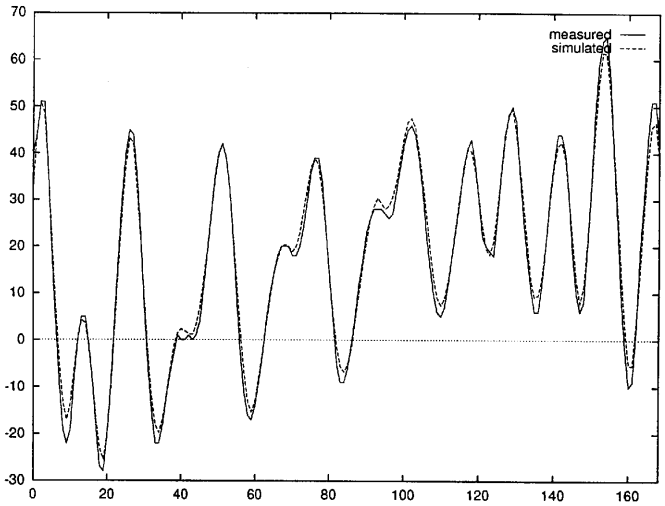

Figure 2. Comparison between field data obtained by measurements (full curve) and results of numerical calculation (broken curve) close to Burano island for first week

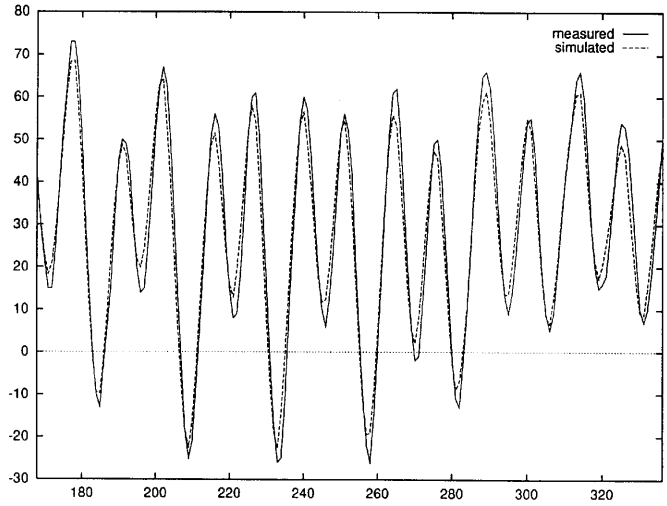

Figure 3: Same as Figure 2 for second week

As is quite evident, such forces are independent of the problem variables, therefore once h has been replaced by H because of the long-wave assumption and they have been divided by H (as done for the second and third equations of (8)), their inclusion in the right-hand side of the explicit system (9) only results in the additional term

$$t_w = \left[0, \xi \frac{|v|v_1}{H}, \xi \frac{|v|v_2}{H}\right]^T.$$

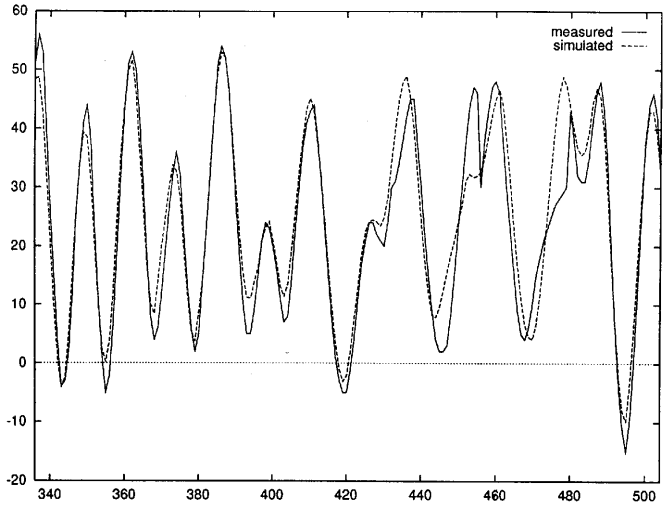

Figure 4: Same as Figure 2 for third week

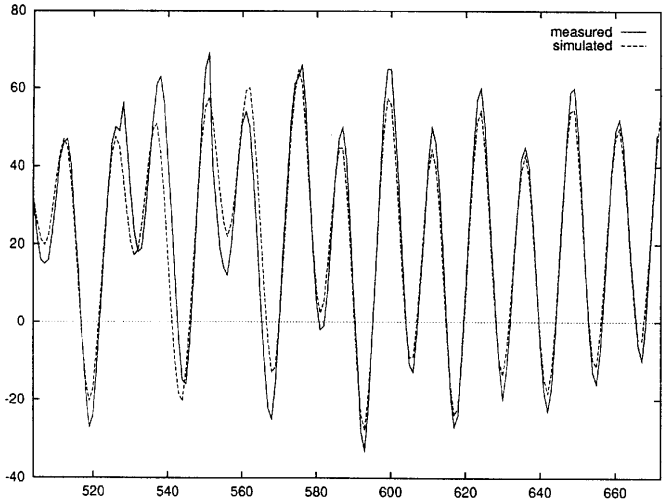

Figure 5. Same as Figure 3 for fourth week

9. RESULTS AND CONCLUSIONS

The programme has been run for long periods of simulation; many months of simulation have been run successfully with no loss of fluid and perfect behaviour. To validate the programme, field data are used taken from the data of the Ufficio Idrografico e Mareografico of Venice. The data have been introduced at the three inlets of the lagoon as boundary conditions on the open boundary. A comparison has been made between the simulation results and the real data from the internal measurement stations of the lagoon. The experimental and simulated data show good agreement at all internal measurement stations where field data have been taken. The station of Burano island has been chosen to illustrate the results (see Figures 2–5). In fact, the results of this station are more significant than others because it is furthest from the inlets. The island has a boundary with a sharp corner that creates vortices of turbulence which are perfectly simulated by the programme. For this reason the station of Burano is often chosen for the control of simulations even though other stations give better agreement.

REFERENCES

1. A. J. Chorin, 'On the convergence of discrete approximations to the Navier–Stokes equations', *Math. Comput*., **23**, 341–353 (1969).
2. R. Temam, *Navier–Stokes Equations*, North-Holland, Amsterdam, 1977.
3. P. M. Gresho, 'On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via finite element method', *Int. j. numer. methods fluids*, **11**, 587–620 (1990).
4. R. Rannacher, 'On Chorin's projection method for the incompressible Navier–Stokes equations', *Lect. Notes Math*., **1530**, 167–183 (1991).
5. J. Shen, On error estimates of projection methods for Navier–Stokes equations: first order schemes, *SIAM J. Numer. Anal*., **29**, 57–77 (1992).
6. C. Schwab, 'Remarks on pressure approximation in projection method for viscous incompressible flow', in K. Morgan, M. Morandi Cecchi and O. C. Zienkiewicz (eds), *Finite Elements in Fluids*, Wiley, Chichester, 1995, pp. 127–136.
7. O. C. Zienkiewicz, J. Wu and J. Peraire, 'A new semi-implicit or explicit algorithm for shallow water equations', *Math. Model. Sci. Comput*., **1**, 31–49 (1993).
8. J. Peraire, 'A finite element method for convection dominated flows', *Ph.D. Thesis*, University College of Swansea, 1986.
9. M. Morandi Cecchi, E. Secco and A. Pica, 'Domain decomposition of finite element model of a lagoon', in K. Morgan, M. Morandi Cecchi and O. C. Zienkiewicz (eds), *Finite Elements in Fluids*, Pineridge, Swansea, 1993, pp. 990–998.
10. J. Wu and O. C. Zienkiewicz, 'A new semi-implicit or explicit algorithm for shallow water equations', *Proceedings of International Symposium on Environmental Hydraulics, Hong Kong 16–18 Dec*. pp. 971–976, Balchema Press, Rotterdam, 1991.
11. O. C.Zienkiewicz and J. Wu, 'Shallow water equations: recent developments of solution algorithm', *Proceedings of International Symposium on Environmental Hydraulics, Hong Kong 16–18 Dec*. pp. 703–710, Balchema Press, Rotterdam, 1991.
12. O. C. Zienkiewicz and P. Ortiz, 'A split-characteristic based finite element model for the shallow water equations', *Int. j. numer. methods fluids*, **20**, 1061–1080 (1995).
13. O. C. Zienkiewicz and P. Ortiz, 'An improved finite element model for shallow water problems', in *Finite Element Modelling of Enviromental Problems* Wiley, New York, 1995, pp. 61–84.
14. H. A. Schwarz, 'Uber einige Abbildungasaufgaben', *Ges. Math. Abh*., **11**, 65–83 (1869).
15. J. S. Przemienniechi, 'Matrix structural analysis of substructures, *AIAA J*., **1**, 138–147 (1963).
16. D. A. Kopriva, Computation of hyperbolic equations on complicated domains with patched and overset Chebyshev grids', *SIAM J. Sci. Stat. Comput*., **10**, 120–132 (1989).
17. M. M. Rai, 'A conservative treatment of zonal boundaries for Euler equation calculations', *J. Comput. Phys*., **62**, 472–503 (1986).
18. D. A. Kopriva, 'Domain decomposition with both spectral and finite difference methods for the accurate computation of flows with shocks', *Appl. Numer. Math*., **6**, 141–151 (1989).
19. Q. V. Dinh, R. Glowinski, J. Periaux and G. Terrasson, 'On the coupling of viscous and inviscid models for incompressible fluid flows via domain decomposition', in R. Glowinski, G. H. Golub, G. A. Meurant and J. Periaux (eds), *Proc. 1st Symp. on Domain Decomposition Methods for Partial Differential Equations* SIAM, Philadelphia, PA, 1988, pp. 350–369.
20. R. Glowinski, J. Periaux and G. Terrasson, 'On the coupling of viscous and inviscid models for compressible fluid flows via domain decomposition', in T. F. Chan, R. Glowinski, J. Periaux and O. B. Widlund (eds), *Proc. 3rd Int. Symp. on Domain Decomposition Methods for Partial Differential Equations*, SIAM, Philadelphia, PA, 1990, pp. 64–97.

21. P. Concus, G. H. Golub and D. P. O'Leary, A generalized conjugate gradient method for the numerical solution of elliptic partial differential equations, in J. R. Bunch and D. J. Rose (eds), *Sparse Matrix Computations*, Academic, New York, 1976, pp. 309–332.
22. M. Dryja, 'A capacitance matrix method for Dirichlet problem on polygon regions', *Numer. Math*., **39**, 51–64 (1982).
23. M. Dryja, 'A finite-element capacitance method for elliptic problems on regions partitioned into subregions', *Numer. Math*., **44**, 153–168 (1984).
24. J. H. Bramble, J. E. Pasciak and A. H. Schatz, 'An iterative method for elliptic problems on regions partitioned into substructures', *Math. Comput*., **46**, 361–369 (1986).
25. J. H. Bramble, J. E. Pasciak and A. H. Schatz, 'The construction of preconditioners for elliptic problems by substructures', *Math. Comput*., **47**, 103–134 (1986).
26. T. F. Chan and D. Goovaerts, 'A note on the efficiency of domain decomposed incomplete factorizations', *SIAM J. Sci. Stat. Comput*., **11**, 794–803 (1990).
27. C. Borgers, 'The Neumann–Dirichlet domain decomposition method with inexact solvers on the subdomains', *Numer. Math*., **55**, 123–136 (1989).
28. Y. Cai and I. M. Navon, 'Parallel block preconditioning techniques for the numerical simulation of the shallow water flow using finite element methods', *J. Comput. Phys*., **122**, 39–50 (1995).
29. I. M. Navon and Y. Cai, 'Domain decomposition and parallel processing of a finite element model of the shallow water equations', *Comput. Methods. Appl. Mech. Eng*., **106**, 179–212 (1993).
30. J. Steppeler, 'Energy conserving Galerkin finite element scheme for the primitive equations of numerical weather prediction', *J. Comput. Phys*., **69**, 258–264 (1987).
31. M. Morandi Cecchi, L. Padoan and G. Scenna, 'A shallow water model by domain decomposition method', in P. Messina and A. Murli, (eds) *Parallel Computing: Problems, Methods and Applications*, Elsevier, New York, 1992, pp. 361–371.
32. M. Morandi Cecchi, E. Secco and A. Pica, 'Tidal flow analysis with core minimization', in K. Morgan, E. Onãte, J. Periaux, J. Peraire and O. C. Zienkiewicz (eds), *Finite Elements in Fluids*, Pineridge, Swansea, 1993, pp. 1026–1036.
33. W. L. Wood, O. C. Zienkiewicz and N. W. Hine, 'A unified set of single step algorithms. Part 1: General formulation and applications', *Int. j. numer. methods. eng*., **20**, 1529–1552 (1984).
34. G. Di Silvio, 'Modelli matematici per lo studio della propagazione di onde lunghe e del trasporto di materia nei corsi d'acqua e nelle zone costiere', in G. Casadei, G. Di Silvio, G. Gambolati and C. Minnaia (eds), *Equazioni Differenziali dell'Idrologia e dell'Idraulica*, Patron, Bologna, 1979, pp. 11–41.

