ELSEVIER

Comput. Methods Appl. Mech. Engrg. 166 (1998) 379–390

Computer methods in applied mechanics and engineering

Adaptive finite element simulations of the surface currents in the North Sea

S.Ø. Wille

Faculty of Engineering, Oslo College, Cort Adelersgate 30, N-0254 Oslo, Norway

Received 8 April 1998

Abstract

The Navier–Stokes equations are solved for the pressure and flow of the surface currents in the North Sea. The solution algorithm applied is the nodal adaptive mesh and adaptive time method. The Navier–Stokes equations are split in four equations which are solved sequentially. The first equation, which is solved implicitly, is the diffusion equation. The second equation, which is solved explicitly, is the convection equation. The third equation, which is solved implicitly, is the pressure correction equation and the fourth equation, which is solved explicitly, is the velocity correction equation. The two equations, the diffusion and the pressure correction equation which are solved implicitly, are symmetric, linear and positive definite. The implicit equations are therefore solved by a symmetric conjugate gradient algorithm.

The symmetric conjugate gradient algorithm is performed node by node without storage of the equation matrix. The nodal solution algorithm therefore permits the solution of larger problems as compared to algorithms which apply an assembled and stored equation matrix. The coefficients in the equation matrix in the nodal algorithm are generated whenever needed in the matrix–vector multiplication in the conjugate algorithm.

The initial finite element grid is obtained by adapting the grid to the coastline. A solution is first obtained at a low Reynolds number. The solution is then scaled and used as a start vector for the computation at a higher Reynolds number. At several time steps in the iteration process, the Reynolds number and the Courant number are computed for each element. An element is refined if the element Reynolds number is greater than 1 and an element is recoarsed if the element Reynolds number is much less than 1. The time step is adjusted simultaneously with adapting the grid to the solution. The time step is computed to ensure that the largest element Courant number is less than 0.5.

The simulation results demonstrate that vortices may develop at the coasts outside England and outside Germany.

1. Introduction

The development and the properties of the nodal adaptive finite element method have previously been described in several papers [1–5]. The advantages of the method presented are that no equation matrices need to be stored [6] and fineness and coarseness of the grid are adapted to the solution.

Grid adaption algorithms have also been investigated by other researchers Kallinderis [7] and Greaves et al. [8–10]. Kallinderis uses both rectangular and triangular elements and applies the velocity gradient as refinement-recoarsement indicator. Both Kallinderis and Greaves use quad tree data structure in their grid generation.

Iterative equation solution algorithms [11] are an important part of the nodal operator splitting solution algorithm. Both robustness and efficiency are required in the solution algorithm. Operator splitting algorithms which split the Navier–Stokes equations in three equations have been investigated by Ren and Utnes [12] and Codina [13]. The algorithms investigated proved to be promising.

A posteriori error estimation [14] has proved to be an important parameter in grid refinements. However, the author believes that the element Reynolds number is an equal adequate parameter which in addition has the property of stabilizing the equation system by linearizing the equation system.

PII: S0045-7825(98)00102-9

The test problem during the development of the nodal adaptive finite element method has been the driven cavity flow problem in both two and three dimensions. The aim of the present work is to demonstrate that the method also exhibit advantageous properties for domains with substantial irregular boundary contours. The simulation problem selected is the ocean currents in the North Sea.

2. Governing equations

In order to study sea water currents in detail, the differential equations should be three-dimensional and contain variables for salinity, density and temperature. However, as a first approximation, the present work is limited to the two-dimensional and simplified Navier–Stokes equations given below. These nonlinear Navier–Stokes equations are,

$$\begin{aligned} &\rho\frac{\partial \boldsymbol{v}}{\partial t} - \mu\nabla^2\boldsymbol{v} + \rho\boldsymbol{v}\cdot\nabla\boldsymbol{v} + \nabla p = 0 && \text{in } \Omega \\ &-\nabla\cdot\boldsymbol{v} = 0 && \text{in } \Omega \end{aligned} \tag{1}$$

where $\boldsymbol{v}$ is the velocity vector, p is the pressure, μ is the viscosity coefficient and ρ is the density. The first equation is the equation of motion which contains time, diffusion, convection and pressure terms. The second equation is the equation of continuity.

3. The velocity–pressure operator split algorithm

The original Navier–Stokes equation is reformulated into four equations,

$$\begin{aligned} &\rho\frac{\partial \boldsymbol{v}}{\partial t} - \mu\nabla^2\boldsymbol{v} + \nabla p = 0 && \text{in } \Omega \\ &\rho\frac{\partial \boldsymbol{v}}{\partial t} + \rho\boldsymbol{v}\cdot\nabla\boldsymbol{v} = 0 && \text{in } \Omega \\ &\partial\nabla^2 p - \rho\nabla\cdot\frac{\partial \boldsymbol{v}}{\partial t} = 0 && \text{in } \Omega \\ &\rho\frac{\partial \boldsymbol{v}}{\partial t} + \nabla p = 0 && \text{in } \Omega \end{aligned} \tag{2}$$

The finite element formulation of the velocity-pressure split equations becomes,

$$\begin{aligned} \boldsymbol{F}_d &= \int_\Omega \left[\rho L_i\frac{\partial \boldsymbol{v}}{\partial t} + \mu\nabla L_i\cdot\nabla\boldsymbol{v} - \nabla L_i p\right]\mathrm{d}\Omega - \int_{\delta\Omega}\mu L_i\frac{\partial \boldsymbol{v}}{\partial n}\,\mathrm{d}\delta\Omega = 0 \\ \boldsymbol{F}_c &= \int_\Omega \left[\rho L_i\frac{\partial \boldsymbol{v}}{\partial t} + \rho L_i\boldsymbol{v}\cdot\nabla\boldsymbol{v}\right]\mathrm{d}\Omega = 0 \\ \boldsymbol{F}_p &= \int_\Omega \left[\rho L_i\nabla\cdot\frac{\partial \boldsymbol{v}}{\Delta t} + \nabla L_i\nabla p\right]\mathrm{d}\Omega - \int_{\delta\Omega} L_i\frac{\partial p}{\partial n}\,\mathrm{d}\delta\Omega = 0 \\ \boldsymbol{F}_m &= \int_\Omega \left[\rho L_i\frac{\partial \boldsymbol{v}}{\partial t} + \nabla p\right]\mathrm{d}\Omega = 0 \end{aligned} \tag{3}$$

The four equations, $\boldsymbol{F}_d$, $\boldsymbol{F}_c$, $\boldsymbol{F}_p$ and $\boldsymbol{F}_m$, which are linear, are solved by the nonlinear Newton algorithm

$$\begin{aligned} &\frac{\partial \boldsymbol{F}_d^n}{\partial \boldsymbol{v}}\Delta\boldsymbol{v}^{n+1} = -\boldsymbol{F}_d^n \qquad && \frac{\partial \boldsymbol{F}_c^n}{\partial \boldsymbol{v}}\Delta\boldsymbol{v}^{n+1} = -\boldsymbol{F}_c^n \\ &\frac{\partial \boldsymbol{F}_p^n}{\partial \boldsymbol{v}}\Delta\boldsymbol{p}^{n+1} = -\boldsymbol{F}_p^n \qquad && \frac{\partial \boldsymbol{F}_m^n}{\partial \boldsymbol{v}}\Delta\boldsymbol{v}^{n+1} = -\boldsymbol{F}_m^n \end{aligned} \tag{4}$$

Divergence free flow is achieved by replacing the pressure by $d^n = p^n - p^{n-1}$ [12]. The Newton formulations of the pressure split equations are

$$
\begin{aligned}
&\int_\Omega \left[\rho L_i \frac{L_j}{\Delta t} + \mu \nabla L_i \nabla L_j\right] \mathrm{d}\Omega\, \Delta \boldsymbol{v} = -\int_\Omega \left[\mu \nabla L_i \cdot \nabla \boldsymbol{v}^n + \nabla L_i p^n\right] \mathrm{d}\Omega \\
&\int_\Omega \rho L_i \frac{L_j}{\Delta t} \mathrm{d}\Omega\, \Delta \boldsymbol{v} = -\int_\Omega \rho L_i \boldsymbol{v}^n \cdot \nabla \boldsymbol{v}^n \,\mathrm{d}\Omega \\
&\int_\Omega \nabla L_i \nabla L_j \,\mathrm{d}\Omega\, \Delta p = -\int_\Omega \left[\rho L_i \frac{\nabla \cdot \boldsymbol{v}^n}{\Delta t} + \nabla L_i \nabla d^n\right] \mathrm{d}\Omega \\
&\int_\Omega \rho L_i \frac{Lj}{\Delta t} \mathrm{d}\Omega\, \Delta \boldsymbol{v} = -\int_\Omega \nabla d^n \,\mathrm{d}\Omega
\end{aligned}
\tag{5}
$$

Let n_d be the spatial dimension, then the exact integrals above can easily be computed both in two and three dimensions by the formula [15]

$$
\int_\Omega L_i^\alpha L_j^\beta L_k^\gamma = \frac{\alpha!\beta!\gamma!}{(\alpha+\beta+\gamma+n_d)!} n_d!\Omega
\tag{6}
$$

The advantage of using the nonlinear Newton formulation for solving linear equations is that the boundary conditions for the correction introduced in the equation system are always zero, while the actual boundary value is inserted in the initial solution vector.

When the velocity field is found, the final pressure is calculated from the Poisson equation with the appropriate boundary conditions included. The Poisson equation is derived from the differentiation of the Navier–Stokes equations and by substitution of the continuity equation.

4. Solution adaption

There are two important parameters in the solution algorithm for the Navier–Stokes equations. These parameters are the Reynolds number and the Courant number

$$
\mathrm{Re} = \frac{\rho\|\boldsymbol{v}\cdot\nabla\boldsymbol{v}\|}{\mu\|\nabla^2\boldsymbol{v}\|} \qquad \mathrm{Co} = \frac{\rho\|\boldsymbol{v}\cdot\nabla\boldsymbol{v}|}{\rho\left\|\frac{\partial \boldsymbol{v}}{\partial t}\right\|}
\tag{7}
$$

The Reynolds number is defined as the ratio of convection to diffusion. The Courant number is defined as the ratio of convection to acceleration and the Dissipation parameter is defined as the ratio of diffusion to acceleration. The Reynolds number is reflecting the degree of non-linearity in the equation system and the Courant number is indicating the degree of hyperbolicity.

4.1. Grid adaption

The element Reynolds number Re_e is computed for each Tri–Tree element from the expression given below. Let L_i^c be the linear basis function evaluated at the geometrical center of the element. Then the different parameters become

$$
\mathrm{Re} = \frac{\Sigma_i L_i^c \left\|\rho \int_\Omega N_i \boldsymbol{v}\cdot\nabla\boldsymbol{v}\,\mathrm{d}\Omega\right\|}{\Sigma_i L_i^c \left\|\mu \int_\Omega \nabla L_i \cdot \nabla \boldsymbol{v}\,\mathrm{d}\Omega\right\|} < \epsilon_{\mathrm{Re}}
\tag{8}
$$

Numerical experiments, Wille [4], have shown that $\epsilon_{\mathrm{Re}} < 10$ in two spatial dimensions and $\epsilon_{\mathrm{Re}} < 30$ in three dimensions in order to obtain a converged solution for the Navier–Stokes equations. In the present work, the element Reynolds number limit is $\epsilon_{\mathrm{Re}} = 1$ for refinement and recoarsement of the grid.

4.2. *Time adaption*

The element Courant number Co_e and the element Diffusion parameter Dp_e are computed for each Tri–Tree element.

$$\mathrm{Co}_e = \frac{\Sigma_i L_i^c \left\| \rho \int_\Omega L_i \boldsymbol{v} \cdot \nabla \boldsymbol{v} \, \mathrm{d}\Omega \right\|}{\Sigma_i L_i^c \left\| \rho \int_\Omega L_i \frac{\partial \boldsymbol{v}}{\partial t} \, \mathrm{d}\Omega \right\|} < \epsilon_{\mathrm{Co}} \tag{9}$$

For explicit time schemes, it has been shown theoretically that the time marching scheme remains stable if $\epsilon_{\mathrm{Co}} < 1$. These values have been derived for whole geometries where characteristic length and mean velocity are applied in the derivations. The element Courant number has experimentally, been found to be in the range 1.0 to 2.5 when divergence occur. In the present work, the Courant number limit is chosen to be 0.5. The length of the time step is computed from

$$\Delta t < 0.2 \, \Delta t_0 / \mathrm{Max}(\mathrm{Co}_e) \tag{10}$$

5. Generation of the North Sea grid

The generation of the initial grid is shown in Figs. 1–3. The data describing the North Sea is available as a 165 × 150 matrix. The matrix contains zeros indicating land and negative numbers indicating the depth of the sea. The contour of the coastline is obtained from this matrix.

The triangulation of the coastline Fig. 1 is obtained by traversing the coastline in the depth matrix. At each point in the matrix the Tri–Tree is searched and the level of refinement of the triangle containing the point is compared to a preset value. If the refinement level is larger than the preset value, the Tri–Tree triangle is refined. As seen from the figures, not only the triangles at the coastline is refined to a preset value, but also the triangles in a predefined vicinity, Fig. 2. The triangles outside the computational domain are then removed.

The grid in Fig. 2 is then balanced in the sense that every triangle in the Tri–Tree grid has neighbor triangles,

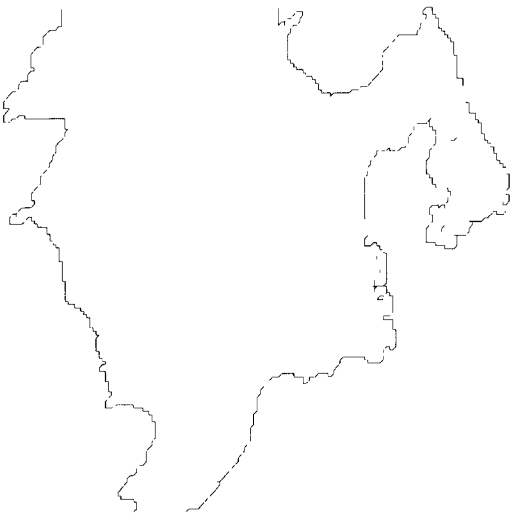


Fig. 1. The figure shows the coastline of the North Sea (left) and the coastline of the North Sea circumscribed by four initial triangles (right).

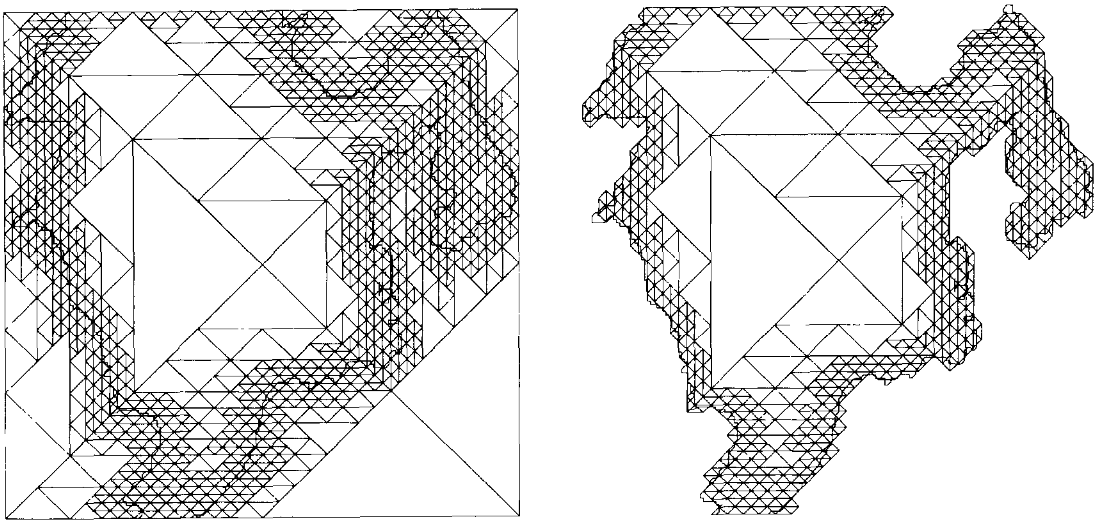


Fig. 2. The figure shows the refinements of the Tri-Tree triangles of the coastline (left) and the triangulation of the North Sea after the removal of the triangles outside the sea (right).

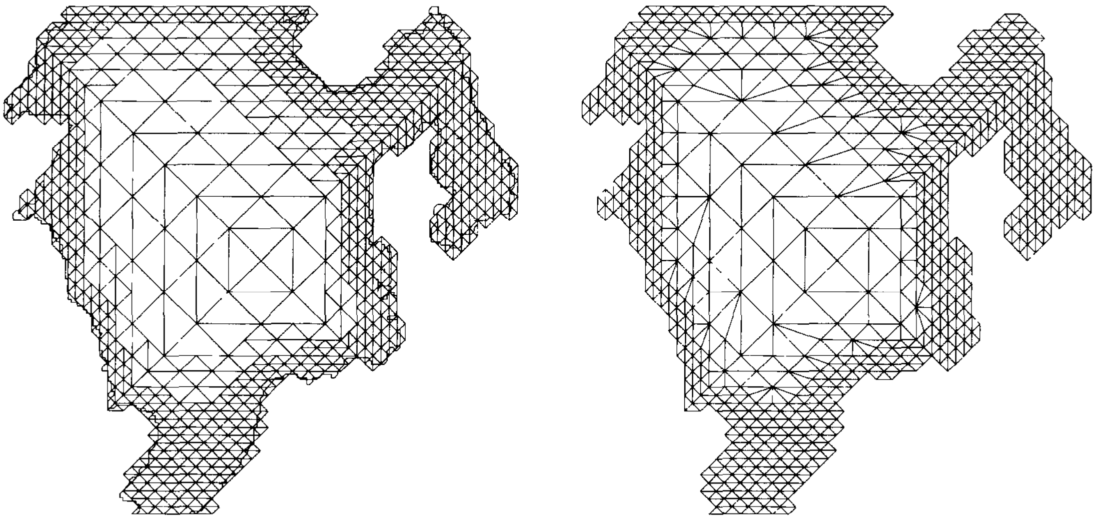


Fig. 3. The figure shows the balanced Tri-Tree triangles (left) and the finite element grid (right).

Fig. 3 (left), which only differ in refinement level by 1. The balanced grid is thereby triangulated to the computational finite element grid, Fig. 3 (right).

The grids used for the computations at each Reynolds number is shown in Fig. 4. The initial grid to the upper left is adapted to the coastline. The velocity and pressure field is first computed for Reynolds number 4. The solution for the higher Reynolds number is computed by scaling the solution for Reynolds number 4 by the factor 2 and applying the scaled solution as a start vector for the iterations at Reynolds number 8. The iterations are thereby guaranteed a good start vector. The grid at each Reynolds number is refined if an element Reynolds number is greater than 1 and the grid is recoarsed if the element Reynolds number is less than 0.5.

The time step in solution algorithm is adapted to the solution every time a grid refinement and grid recoarsement occur.

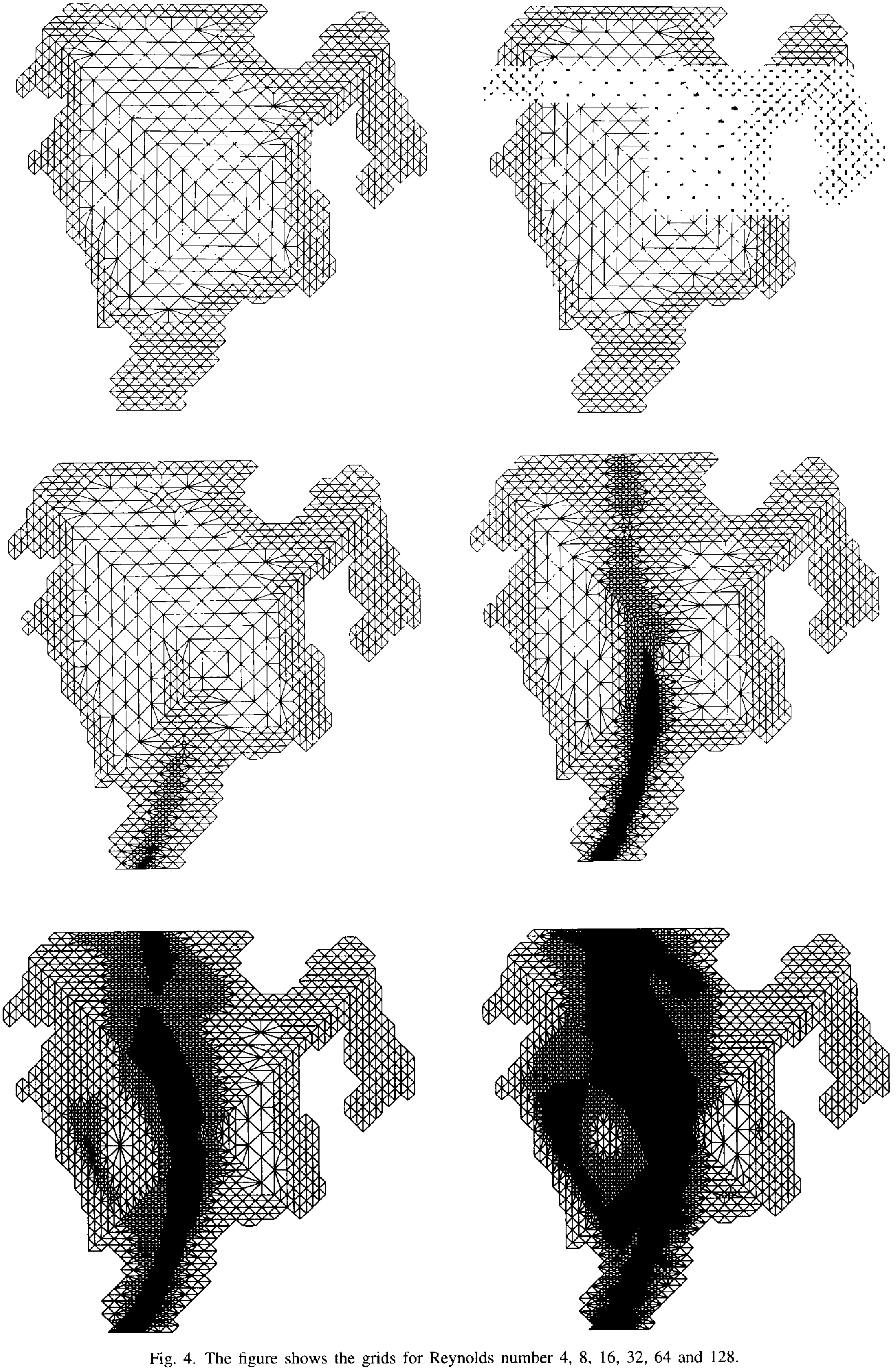
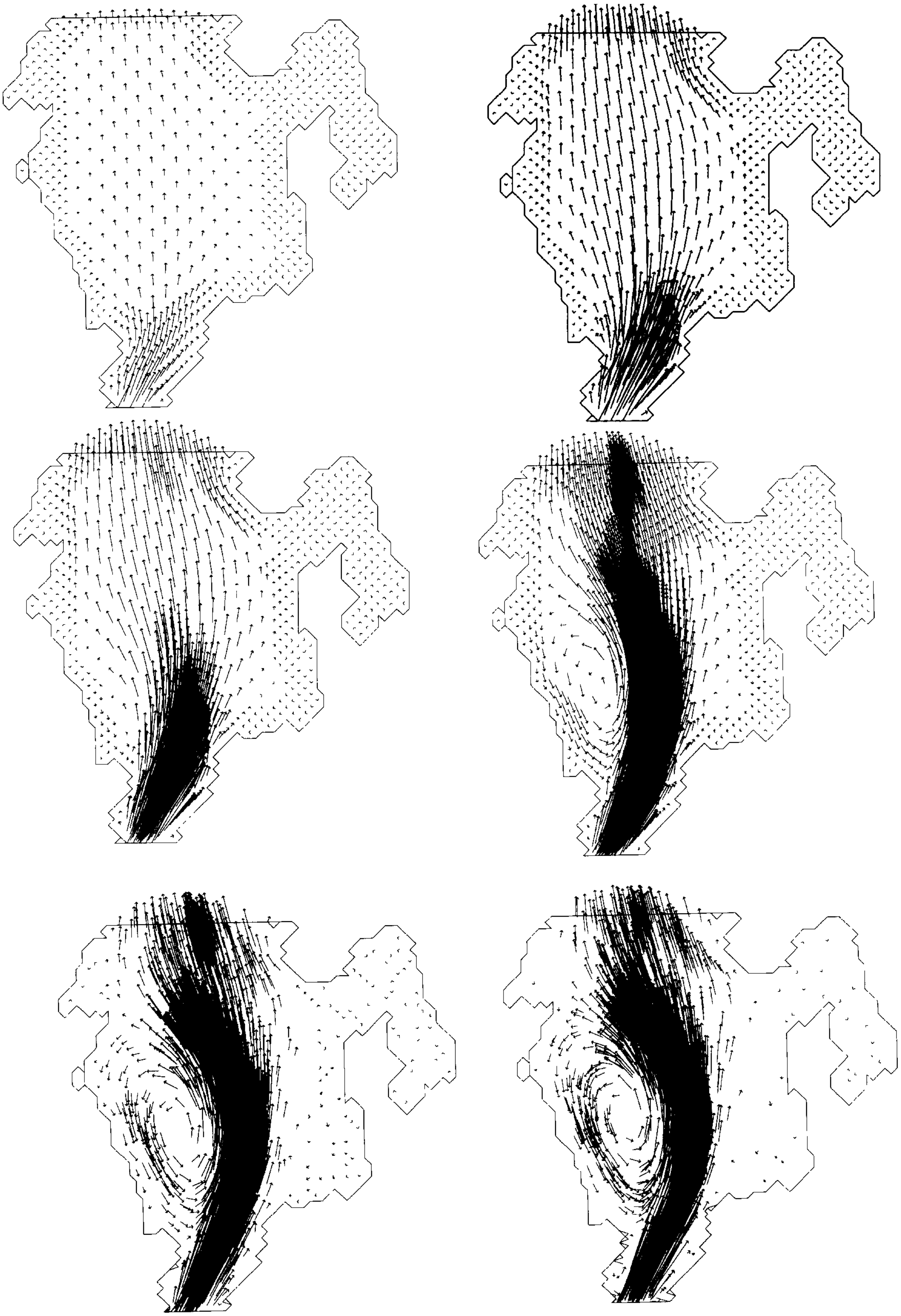


Fig. 4. The figure shows the grids for Reynolds number 4, 8, 16, 32, 64 and 128.



· Fig. 5. The figure shows the velocity vectors for Reynolds number 4, 8, 16, 32, 64 and 128. For the two highest Reynolds numbers a random selection of velocity vector is shown.

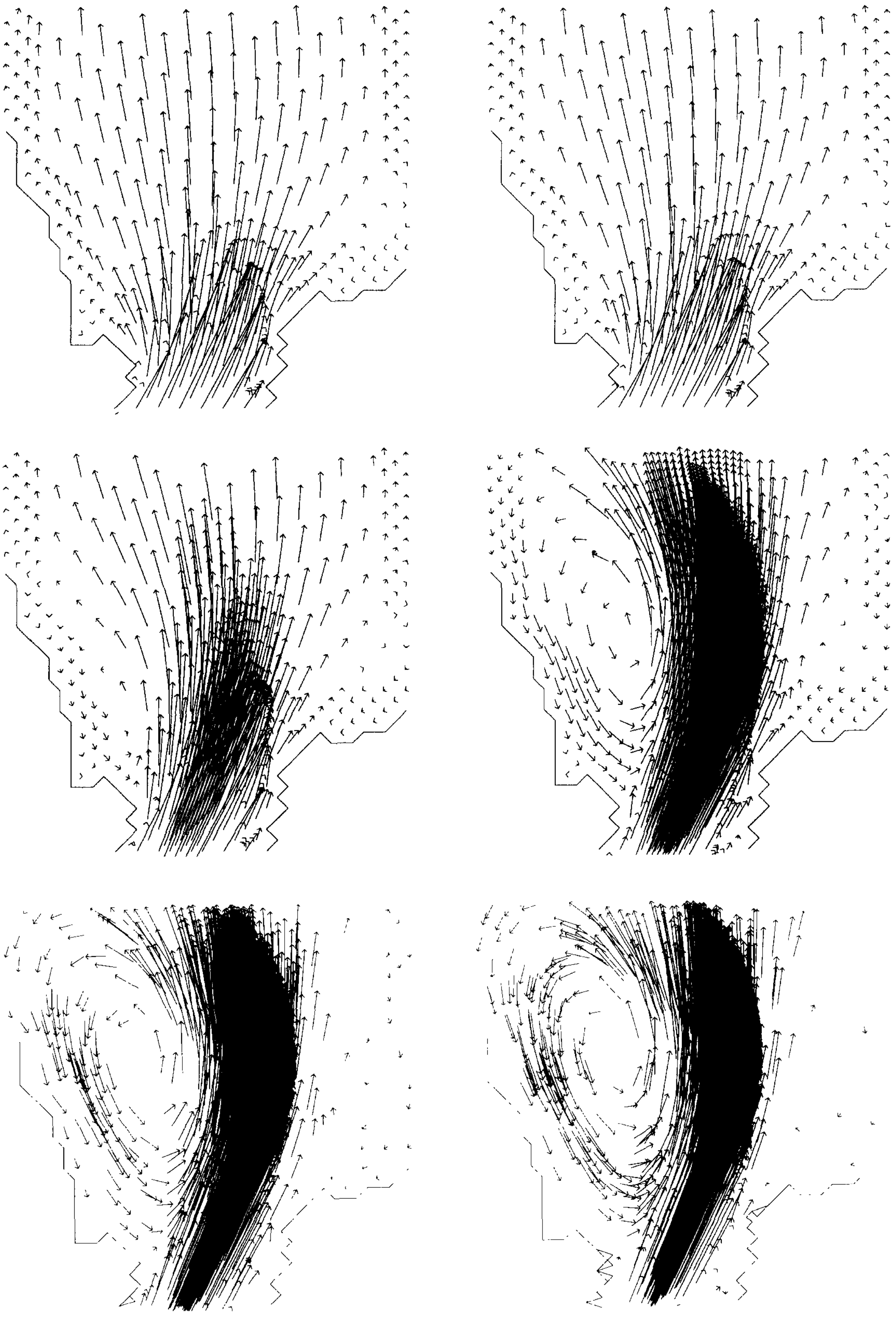


Fig. 6. The figure shows the velocity vectors in the area of the vortices for Reynolds number 4, 8, 16, 32, 64 and 128. For the two highest Reynolds numbers a random selection of velocity vector is shown.

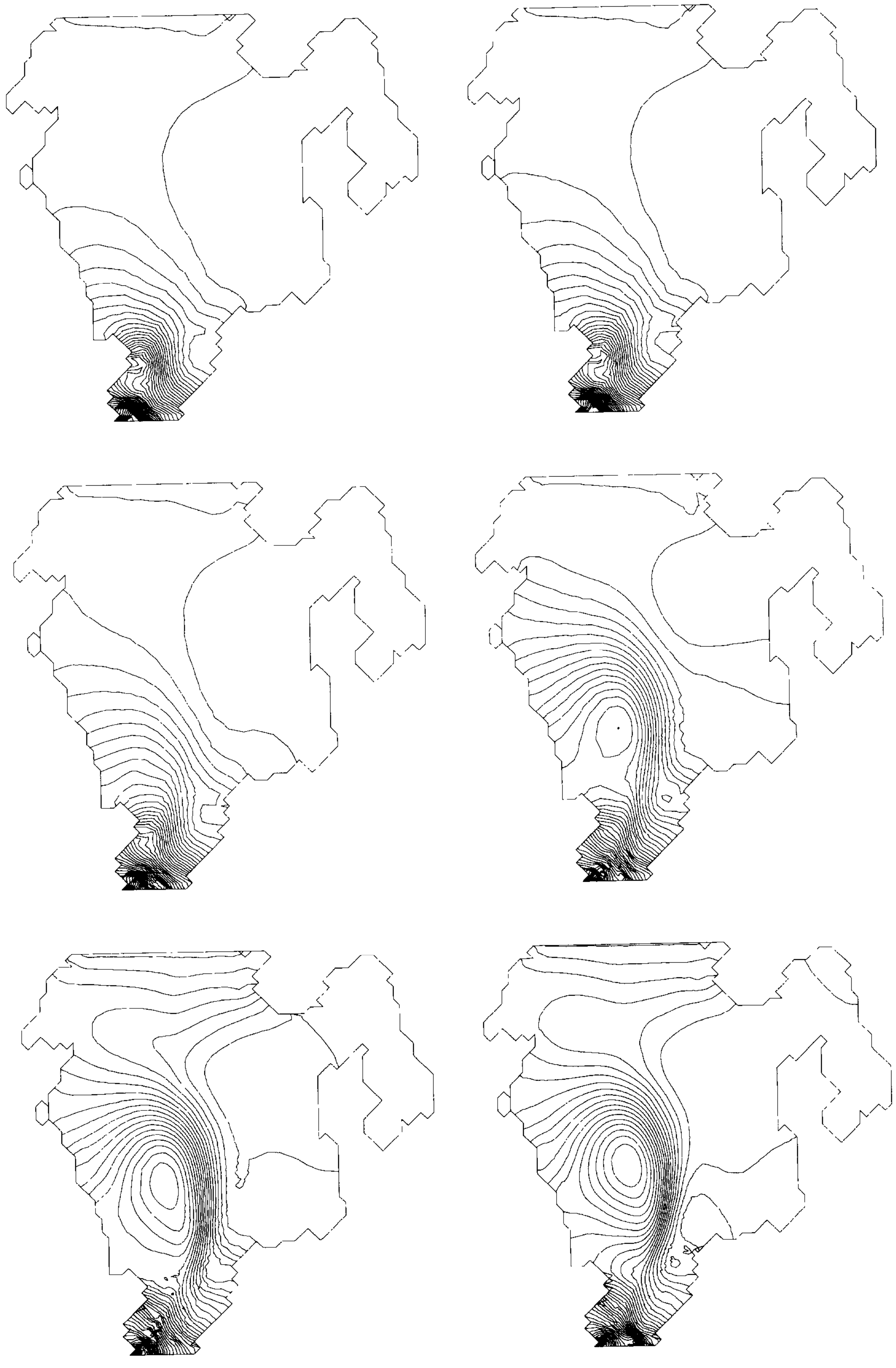


Fig. 7. The figure shows the pressure isobars for Reynolds number 4, 8, 16, 32, 64 and 128.

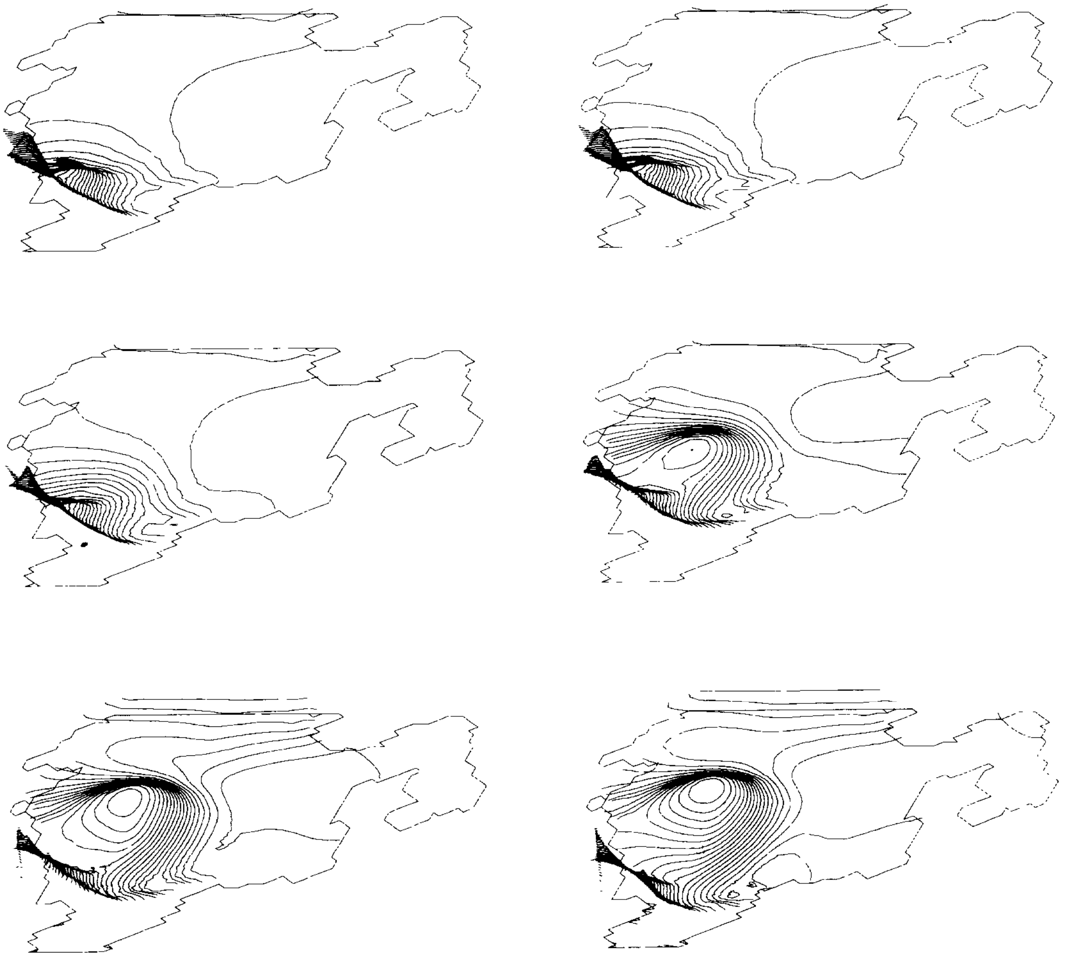


Fig. 8. The figure shows the three-dimensional pressure isobars for Reynolds number 4, 8, 16, 32, 64 and 128.

The computational parameters for the time steps and the grid resolution for a converged solution are shown in Table 1. The diffusion equation and the pressure equation are solved implicitly by a symmetric conjugate gradient algorithm. The number of conjugate gradient iterations for solving the diffusion equation is fixed to 3 and the number of conjugate gradient iterations for solving the pressure equations is fixed to 50. The reason for more pressure iterations than diffusion iterations is that the velocity–pressure operator splitting algorithm requires an accurate pressure solution [12]. During the conjugate gradient iterations, the coefficients in the matrix–vector products are computed whenever needed. This implies that no equations matrix needs to be stored neither in central memory or at disks. The algorithm therefore permits a considerable increase in size of problems to be solved.

Table 1
The table shows the length of the time steps, the number of element nodes, the number of grid elements, the number of Tri–Tree nodes and the number of Tri–Tree elements for Reynolds number 4, 8, 16, 64 and 128

Computation parameters

Reynolds no	4	8	16	32	64	128
Time step	3.8×10^{13}	1.9×10^{13}	9.5×10^{12}	1.4×10^{12}	3.3×10^{11}	4.8×10^{10}
Element nodes	831	831	1010	2363	7081	13959
Grid elements	1459	1459	1815	4516	13932	38456
Tree nodes	1162	1162	1342	2695	7413	19695
Tree elements	2801	2801	3185	6449	18217	49397

The number of outer split iterations is fixed to 400 for all Reynolds numbers. At every 20 split iterations, the grid is refined and a new time step is computed according to the solution at that time. Since time consumption for grid refinement and grid recoarsement is only a few per cent of the computational time, the overall time used is not very sensitive to frequent grid adaption [4].

6. Numerical simulations

The boundary conditions for the ocean current simulations are that a parabolic velocity profile is specified in the English channel, the velocities are specified to be zero along the coasts and the normal derivatives of the velocities are fixed to zero at the outlet. The pressure is also fixed to zero at the outlet boundary.

The results of the ocean current simulations are shown in Figs. 5–8. Fig. 5 shows the velocity fields for increasing Reynolds numbers. At Reynolds number 16 two vortices begin to develop. One vortex is located at the coast of England and the other one at the coast of Germany. The water transport is taking place in a central core through the North Sea (Fig. 5). In Fig. 6 the velocity field of the vortices is enlarged. Fig. 7 shows the pressure isobars. A local high pressure sone is appearing at the site of the vortex outside England for the highest Reynolds numbers. The high pressure sones are more clearly seen in Fig. 8 which is a three-dimensional drawing of the pressure isobars.

7. Discussion

The present work demonstrate that the nodal adaptive finite element method developed is well suited for flow problems with complex boundary geometries. The nodale adaptive method does not require storage of large equation matrices. The equation systems to be solved implicitly are symmetric and positive definite. The nodal adaptive method can resolve the resolution in both predetermined interesting areas and in area with high convection.

Acknowledgments

The author is grateful to Trygve Svoldal for valuable suggestions and corrections of the manuscript. The project has been supported by The Norwegian Research Council, grant no. NN2461K., for partial financing of the computer runtime expenses.

References

[1] S.Ø. Wille, A structured Tri–Tree search method for generation of optimal unstructured finite element grids in two and three dimensions, Int. J. Numer. Methods Fluids 14 (1992) 861–881.

[2] S.Ø. Wille, Adaptive linearization and grid iterations with the Tri–Tree multigrid refinement—Recoarsement algorithm for the Navier–Stokes equations, Int. J. Numer. Methods Fluids (1996) in press.

[3] S.Ø. Wille, A non-linear adaptive Tri–Tree multigrid solver for mixed finite element formulation of the Navier–Stokes equations, Int. J. Numer. Methods Fluids 22 (1996) 1041–1059.

[4] S.Ø. Wille, A local predictive convection–diffusion refinement indicator for the Tri–Tree adapted finite element multigrid algorithm of the Navier–Stokes equations. Comput. Methods Appl. Mech. Engrg. 134 (1996) 181–196.

[5] S.Ø. Wille, The prolonged adaptive multigrid method for finite element Navier–Stokes equations. Comput. Methods Appl. Mech. Engrg. 138 (1996) 227–271.

[6] S.Ø. Wille, The three dimensional prolonged adaptive unstructured finite element multigrid method for The Navier–Stokes equations, Int. J. Numer. Methods Fluids 25 (1997) 371–392.

[7] Y. Kallinderis, Adaptive hybrid prismatic/tetrahedral grids, Int. J. Numer. Methods Fluids 20 (1992) 1023–1037.

[8] D.M. Greaves and A.G.L. Borthwick, Computation of flow past a square cylinder using adaptive hierarchical mesh in: Proc. 14th Int. Conf. on Offshore Mechanic and Arctic Engineering, Copenhagen, Denmark 1995.

[9] D.M. Greaves, A.G.L. Borthwick, R. Eatock Taylor and G.X. Wu, Analysis of wave-body interactions using adaptive finite element meshes in: Proc. 10th Int. Workshop on Water Waves and Floating Bodies, Oxford, UK, 1995.

[10] D.M. Greaves, A.G.L. Borthwick, R. Eatock Taylor and G.X. Wu, A moving boundary finite element method for fully nonlinear wave simulations, J. Ship Res. 41 (1997) 181–194.

[11] A.M. Bruaset and H.P. Langtangen, Object-oriented design of preconditioned iterative methods in Diffpack, ACM Trans. Math. Software 23 (1997) 50–80.

[12] G. Ren and T. Utnes, A finite element solution of the time-dependent Navier–Stokes equations using a modified velocity correction method, Int. J. Numer. Methods Fluids 17 (1993) 349–364.

[13] R. Codina, M. Vazquez and O.C. Zienkiewicz, A general algorithm for compressible and incompressible flow. Part III: The semi-implicit form, Int. J. Numer. Methods Fluids 27 (1998) 13–32.

[14] J.T. Oden and S. Prudhomme, New approaches to error estimation and adaptivity for the Navier–Stokes equations, Proc. of the Tenth Int. Conf. on Finite Elements in Fluids. Tucson, Arizona (1998) 144–149.

[15] S.Ø. Wille, D. Skipitaris, A dynamic adaption algorithm of grid, time and satellite processors for nodal finite element formulations of Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 161 (1998) 215–228.