Integration of 3D incompressible free surface Navier-Stokes equations on unstructured tetrahedral grid using distributed computation on TCP/IP networks

N. Evstigneev
Doctor of Science, Department of Nonlinear Dynamic Systems, Institute of System Analysis, Academy of Science, Russia

Abstract

The incompressible system of Navier-Stokes equations for an Initial-Boundary Value Problem is solved on an unstructured tetrahedral grid using a finite volume method. Implementation of a free surface calculation is done by using a combination of Level Set and Volume Of Fluid methods. A numerical scheme utilizes the method of fractional steps based on the predictor-corrector method and the artificial compressibility method. Invariant features of a tetrahedron are used in order to calculate fluxes over a control volume with higher order. A high order approximation in Navier-Stokes and VOF level set advection equations is made by a TVD SuperBEE scheme. The turbulence model is based on LES methodology. In order to decrease the time of solution for a large geometry, a distributed computation routine is incorporated into the method. The distributed calculation is based on a TCP/IP network and can use personal computers under Windows or UNIX. The efficiency of the distributed calculation is shown. The method is verified by comparison of results with other calculations and experiments – cavity flow case, dam break free surface flow case, turbulent flow in a circular pipe case – Poiseuille flow (turbulent energy distribution). The method is successfully used for CFD simulation of water intake on Zagorskaya Hydraulic Power Plant (Russia). The results are close between laboratory experiments and CFD computations.

Keywords: unstructured grid, finite volume method, artificial compressibility method, predictor-corrector method, Navier-Stokes equations, distributed computation.

WIT Transactions on Engineering Sciences, Vol 59, © 2008 WIT Press
www.witpress.com, ISSN 1743-3533 (on-line)
doi:10.2495/AFM080071

1 Introduction

It is a common practice these days to use computational fluid dynamics for complex technical applications. Thus, computational methods must meet certain requirements: high order of space and time discretization, lesser time demands for integration, fair geometry approximation and exact representation of flow specific features. For this purpose, a numerical solution of incompressible viscous fluid equations with free surface, utilizing a weak form of Navier-Stokes equations is designed. The method incorporates a high order of approximation, a decrease in integration time by using distributed computation on cheap available PCs, unstructured mesh topology, allowing approximation of any arbitrary complex geometry.

2 Mathematical model and governing equations

2.1 Governing equations

In order to describe a fluid motion in E^4, the most basic system of governing equations is used, Temam [1]. Three dimensional evolutional equations of mass and momentum conservation for Newtonian incompressible fluid, Navier-Stokes equations, in dimensionless form for the turbulent flow in a weak form are used:

$$\frac{\partial}{\partial t}\int_{\Omega} q \cdot dW + \oint_{\partial\Omega} \vec{f}_i \cdot n \cdot ds - \oint_{\partial\Omega} \vec{f}_v \cdot n \cdot ds - F_s - Fr^{-1} g = 0; \quad (1)$$

$$q = [0; u; v; w]^T;\ \vec{f}_i \cdot n = \left[(\Theta); (u\Theta + n_x \cdot P); (v\Theta + n_y \cdot P); (w\Theta + n_z \cdot P)\right]^T;$$

$$\vec{f}_v \cdot n = [0; (n_x\tau_{xx} + n_y\tau_{xy} + n_z\tau_{xz} + m_{xu} + m_{xz}); (n_x\tau_{yx} + n_y\tau_{yy} + n_z\tau_{yz} + m_{yx} + m_{yz});$$
$$(n_x\tau_{zx} + n_y\tau_{zy} + n_z\tau_{zz} + m_{zx} + m_{zy})]^T$$

here: velocity vector-function $V : \Omega \times [0,T] \to \Re^3$, scalar pressure function $P : \Omega \times [0,T] \to \Re$. We seek solution on an arbitrary bounded domain $\Omega \in E^3$; x_i – directions in E^3 (x,y,z – for Cartesian coordinate system); $\Theta = n_x u + n_y v + n_z w$; Re – Reynolds number; Fr – Froude number; g – gravity unit vector. $\tau_{ij} = \frac{1}{\mathrm{Re}}\left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i}\right)$ - Viscous stress tensor for Newtonian fluid where i=1..3,j=1..3; m_{ij} – turbulent stress tensor, $m_{ij}^L = \overline{V_j V_i} - \overline{V}_j \overline{V}_i - 0.5(V_j \partial_i \overline{V}_j - \overline{V}_j \partial_i V_j)$,see section 2.2. The integral form of (1) assumes that weak solutions exist in R^4 [1,2].

A free surface is described by a modified Volume of Fluid (VOF) method, Sussman et al. [5].; $F_s = \sigma \cdot k \nabla F$ - surface tension; k – free surface curvature; F – color set function, that indicates the level of free surface; σ - wetting coefficient. System of equations (1) is added to by the kinematic and dynamic conditions for

a free surface (continuity of normal stress vector on a free surface), described in the level set advection and gradient equations:

$$\frac{\partial F}{\partial t}+u\frac{\partial F}{\partial x}+v\frac{\partial F}{\partial x}+w\frac{\partial F}{\partial x}=0; F\in[0;1]; k=\nabla\cdot\left(\frac{\nabla F}{|\nabla F|}\right), \tag{2}$$

We assume that a color set function F equals 0 for air and 1 for liquid. It differs form general idea for F>0 for liquid and F<0 for air, as described in the general VOF approach, Sussman et al [5]. So the method can be called a Level Set method with a VOF procedure of calculating stress vectors on a free surface.

2.2 Boundary conditions

System of PDE equations (1) must be completed by initial-boundary values in order to have a well-posed problem in Ω. Boundary conditions in general form are Dirichlet and Neuman conditions for Boundary-value Navier-Stokes equations, Temam [1]. In the physical domain one can define the following boundary conditions: at the inlet, the values of all variables are prescribed. At the outlet, the streamwise gradient for each variable is prescribed to be equal to zero. At the walls, no slip and no penetration are assumed. The pressure gradient normal to the wall is calculated as $\frac{\partial P}{\partial n_i}=\mathrm{Re}^{-1}\nabla^2 u_i+m_{ij}$, and for turbulent flows is assumed to be $\frac{\partial P}{\partial n_i}=-0.5(\overline{V_j\partial_i\overline{V}_j-\overline{V}_j\partial_i V_j})$. Derivatives on the right side are small for the near wall velocities and can be assumed to be equal to 0.

2.3 Large eddy simulation model for turbulent flows

To represent turbulent flows one must recover all flow scales, thus closing a turbulent stress tensor m_{ij} in (1) one way or another. This can be done by four ways: recovering all flow scales directly from Navier-Stokes equations (Direct Numerical Simulation) and, hence, neglecting m_{ij}; imposing a complex turbulent model via additional equations for m_{ij} and averaging (1) by Reynolds averaging procedure, hence computing only mean flow properties; representing large flow scales (where the turbulent flow is anisotropic) directly from Navier-Stokes equations (1) and imposing a simple turbulent model (sub grid model) for small (isotropic) scale turbulence (Large Eddy Simulation); combining the latter two approaches (Detached Eddy Simulation). Here we shall use the LES model for turbulent flows. A detailed description for LES methods and applications can be found in Chunlei et al. [9]. In brief, the Large Eddy Simulation is conducted by averaging (1) in time and space with arbitrary scale L, related with mesh size of Ω discretization. Let's impose a general noncommutative averaging operator in $\Re^4$:

$$\overline{V}=L(V), \tag{3}$$

Thus there exists decomposition in $\Re^4$: $V = L^{-1}(\overline{V})$ with commutative properties for a differential operator:

$$\partial_t V_i = L^{-1}(\partial_t V_i) = \partial_t (L^{-1}(V_i)) \text{ etc for all } x_i. \quad (4)$$

Here we denote $\partial_t = \partial/\partial t$ for brevity.

In general m_{ij} is a nonsymmetrical tensor, which arises from averaging of the advective part in the momentum equation by applying (3) to (1). In general $m_{ij}^L = L(\overline{V}_j L^{-1}(\overline{V}_i)) - \overline{V}_j \overline{V}_i$, so closing the asymmetrical tensor on the selected small scale is done by the modified Leray model, Chunlei et al. [9]:

$$m_{ij}^L = \overline{\overline{V}_j V_i} - \overline{V}_j \overline{V}_i - 0.5(\overline{V_j \partial_i \overline{V}_j - \overline{V}_j \partial_i V_j}) . \quad (5)$$

Adding (5) to (1) closes the system of equations. Hence, the solution of evolutional PDE system (1) with tensor (5) on scales greater than L is conducted as a direct numerical simulation. For the flow scales less than L (for greater wave spectra numbers of turbulent kinetic energy, small scale turbulence Chunlei et al. [9]), the flow is represented by the modified Leray model (5). To have a well-posed problem for LES modeling initial conditions must be properly imposed. For technical problems initial conditions are derived from laminar flow calculations in the same geometry.

3 Numerical method

3.1 Meshing

In order to solve (1) on a given bounded domain, Ω in E^3 must be meshed. In the following method a tetrahedral gird that covers the domain is used. The grid generation is using a Delauny algorithm, Cignoniz et al. [7], that allows the creation of a space-adopted high quality mesh. The algorithm utilizes mesh refinement near given areas. Any arbitrary geometry can be discretized due to an unstructured mesh structure. An example of meshing a 3D power intake of Zagorskaya Hydraulic Power Plant is given in fig. 1.

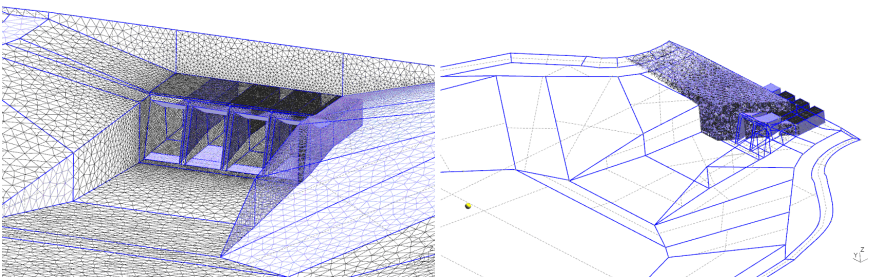

Figure 1: An example of meshing – a power intake of Zagorskaya Hydraulic Power Plant.

3.2 Numerical integration

In accordance with the PDE system (2) two methods are applied to integrate the system in space. A march algorithm similar to the method used in aerodynamics, Evstigneev [12], is used for a continuity equation in (1) is rewritten as $\frac{\partial}{\partial t}\int_{\Omega} P \cdot dW + \oint_{\partial\Omega} \beta \cdot \Theta \cdot ds = 0$, and a predictor-corrector method is used for classic system of Navier-Stokes equations (1). Here β is an artificial compressibility parameter, and equals 80-120, Chorin [6]. The march algorithm is more reliable and much faster due to the lack of Poisson correction equation for pressure function but the predictor-corrector algorithm is more accurate. So the combination of these two methods provides a fast and reliable procedure for flow calculation in complex conditions.

When the space integrals are found, time integration must be applied to the advance solution in time. So, the integration of PDE system (1) consists of two steps: finding space integrals (integral-differential operator in $\Re^3$) and applying the time advancing procedure (solution of ODE in $\Re^1$ with found integrals in $\Re^3$).

Considering a free surface flow, the solution of (1) is only done in the part of the domain where F>0. So the color function equation must be integrated as well.

3.2.1 March scheme

The march method is based on integration of a PDE system like a hyperbolic system of equations, thus flux solvers must be applied to find a solution. The idea of the method originates from Chorin's artificial compressibility solution methodology [6]. In this case a system of singular perturbed PDE is considered thus constructing a strictly hyperbolic system of equations for a void viscosity, i.e. Re equals eternity.

In this case a flux splitting scheme for the inviscid part of (2) on a tetrahedron boundary is used. Thus, the flux value on the boundary of a tetrahedral equals:

$$F = 0.5\left(f_r + f_l - |A| \cdot (q_r - q_l)\right), \tag{6}$$

where F is a flux value on the side of a tetrahedron; 'l' and 'r' letters stand for left and right side values of variables; $A = \partial F / \partial q$ -system Jacobean, which has the following decomposition $A = T\Lambda T^{-1}$, the same as the compressible flow equations with four real eigenvalues:

$$\lambda_{1,2} = \Theta, \lambda_{3,4} = \lambda_{1.2} \pm c, \tag{7}$$

where $c = \sqrt{\lambda_{1,2} + \beta}$ - artificial wave speed propagation. In accordance with the found Jacobean eigenvalues, a U-CUSP similar to the Evstigneev [12] scheme is constructed, thus solving an inviscid part of the Navier-Stokes equations. A detail description of the scheme can be found in Evstigneev [12]. In order to have a high accuracy in flux splitting scheme a high order reconstruction of variables is used. A diffusion equation is solved using a Finite Volume scheme,

described in [4,9]. Hence, a solution is found by integrating a singular perturbed system of hyperbolic PDEs in artificial compressibility methodology on a tetrahedral unstructured grid.

3.2.2 Predictor corrector scheme

Consider a classic system of Navier-Stokes equations (1) without singular perturbations. In this case a link between pressure and the solenoid velocity field must be found in an implicit way. There are many ways to do that, for example see [2,4] In the present work we use a predictor-corrector factual step scheme, based on Yanenko et al [2]. The semi-discrete steps are:

$$\text{Step 1. } V' - V^n = -\Delta t \cdot \nabla(V^n V^n) \text{ - explicit scheme.} \tag{8}$$

$$\text{Step 2. } \widetilde{V} - V' = \Delta t \cdot \mathrm{Re}^{-1} \nabla^2 \widetilde{V} + m_{ij} \text{ - implicit scheme.} \tag{9}$$

$$\text{Step 3. } \nabla^2 P = -\nabla \cdot \widetilde{V} / \Delta t \text{ - Poisson equation solution in assumption that } \nabla \cdot V^{n+1} = 0. \tag{10}$$

$$\text{Step 4. } V^{n+1} = \widetilde{V} - \Delta t \nabla P \text{ - correction of V solenoid field.} \tag{11}$$

Let's assume that a vector-function V and pressure function P values (i.e. from initial conditions) are known in $\Omega x T^n$, and vector-function V is solenoidal in Ω. Then on the first step a nonlinear advection equation (inviscid Burgers equation) is solved (8). On the second step a solution of diffusion equation (9) is found. Vector function field V is not solenoid in Ω if only two steps are applied – that's the predictor part. So the velocity field is corrected on the fourth step (11), that's the corrector part. In order to consider solenoidality of velocity vector-function a pressure function is calculated on the third step by the Poisson equation (10) with a right side as the divergence singular term. The convergence of fractional steps method (11) is proved in Yanenko et al. [2]. Numerical integration of fractional steps is done using a finite volume method.

3.2.3 High order variable reconstruction

In order to find a variable value on a boundary side of a tetrahedron with high accuracy, a high order interpolation is applied for both schemes. Unstructured grid solvers usually suffer from the main loss of accuracy due to the difficult topological interpolation of variables. In this work with the staged grid variable storage high accuracy approximation is achieved by applying Taylor series on the tetrahedron boundary side projection, i.e.:

$$q(x, y, z) = q(x_e, y_e, z_e) + \nabla q_e \cdot \Delta L + 0(\Delta L^2); \tag{12}$$

here ΔL – length in E^3 from centers of nearby tetrahedra; q_e – variable value in tetrahedron center; q – variable value in tetrahedron boundary side.

In the present work a geometrical invariance of a tetrahedron is used, for details see Evstigneev [12]. As the result a gradient would transfer in the following form:

$$\nabla q_e \cdot \Delta L = \partial q / \partial L \cdot \Delta L \cong 0.25 \cdot [1/3(q_B + q_C + q_D) - q_A] \cdot \Delta L / \Delta L ,$$
$$\text{hence: } q_f = q_e + 0.25[(q_B + q_C + q_D)/3 - q_A] , \quad (13)$$

so that the values of left and right variables on a tetrahedron side are found using (13). Here q_A etc are the values of the variable on the vertexes of a tetrahedron; q_f – variable value on a tetrahedron face. Values on tetrahedron vertexes are found using geometrical weighted interpolation.

Using high order in advection equations leads to instability due the existence of nonmonotonous solutions as described by the Godunov's theorem, see Fletcher [4]. To avoid numerical instability a nonlinear scheme must be applied. In the present work a TVD SuperBEE limiter is used as one of the least numerical diffusive. This limiter was tested on modal transport and inviscid Burgers equations [9,12] with very good results. In the author's recent works the limiter was applied to DNS and LES solutions [9–11] and good agreement found with experimental and numerical data.

3.2.4 Time integration

Time integration for the inviscid part of (1) in both schemes (predictor-corrector and march schemes) is done using the 4-th staged explicit Runge-Kutta method [3]. Time advancement of diffusion equation utilizes the implicit 2 staged Runge-Kutta scheme, Hairer et al. [3]. Both methods are described in [9,12] thoroughly. The time step limit in both methods is based on the limit for the advection Burgers equation, hence limited by the CFL [3] condition.

3.2.5 Color set function integration

After the numerical scheme to solve (1) is applied, a free surface must be considered, thus a PDE system (2) must be integrated. The level set equation for the color function (2) is solved using the finite volume method by converting it into appropriate hyperbolic form, using the same approach as for the advection equation in the predictor-corrector scheme, i.e. using finite volume with high order variable reconstruction. The equation (2) changed to:

$$\frac{\partial F}{\partial t} = -\nabla(V \cdot F) + F \nabla \cdot V . \quad (14)$$

If the velocity vector-function field is solenoid, thus $\nabla \cdot V = 0$, equations (14) and (3) are equivalent. But, since the numerical method uses singularity perturbation for artificial compressibility or the fractional steps predictor-corrector method, the divergence term in (14) must be considered to ensure solenoidal velocity vector-function behavior on a free surface.

The free surface tension Fs in (1) is calculated on every time step. Numerical integration of (1) is applied only in the part of the domain where $F>0$, thus decreasing computational time. The pressure and velocity are solved on the liquid side of the interface between air and liquid using the methods described in Sussman et al. [5]. For the edges that intersect the free surface, the pressure that would be calculated via the extrapolation to the control volume face is replaced by the hydrostatic pressure as the differences of distances where $0<F<1$ in the

direction of the gravity vector field. The free surface tension is applied by the Fs in (1) only for the liquid side of the interface between air and liquid, and only in regions where 0<F<1 that come from the equation for k in (3).

3.3 Distributed calculations

The numerical method for Navier-Stokes equations integration, presented in the following work, is developed for complex geometries and turbulent flows. Hence, as any CFD method, it is very resourceful on computer processor power and memory. In order to decrease computer resource dependence, a distributed computation routine is incorporated in the method for TCP-IP networks. Computations can be distributed on any PC network. Tests were made on PCs with MS Windows XP and RedHat Linux 7.0 operational systems. The algorithm is based on division of source domain Ω on separate parts with crosslink boundary tetrahedra. Each PC in the network has a client-sever application that connects to the neighbor computer in the network by the given IP address. Data exchange is done only for neighbor tetrahedra, thus decreasing network usage. Detailed description of the proposed method is given in Evstigneev [12]. Computation acceleration comparison for both schemes as given in fig. 2 was conducted for the lid driven cavity problem with Re=25 000. The grid consisted of 701 434 tetrahedral elements.

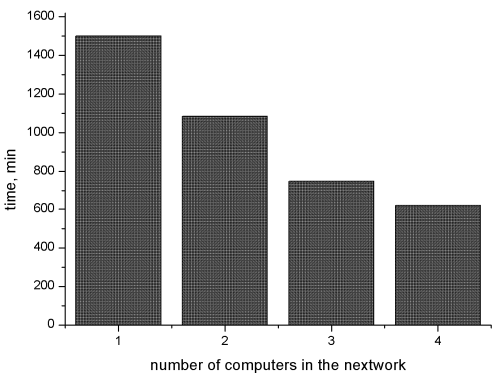

Figure 2: Calculation time for distributed computation.

4 Numerical experiments

4.1 Testing numerical scheme

Thorough tests of the numerical method for incompressible turbulent Navier-Stokes equations were carried out. Only few brief results are presented in the present paper.

4.1.1 Lid driven cavity problem

This test case is the classic task for numerical methods in 2D and 3D geometry for incompressible and compressible low Mach number viscous flows. The test was preformed by many authors for various Reynolds numbers, including fundamental results for nonlinear dynamics, Evstigneev [10]. As a result, this test case has became a main model task for all CFD codes since the end of the 60-th. The test case has a cubic geometry with solid wall boundary conditions. On one of the walls a tangent velocity is set to 1.

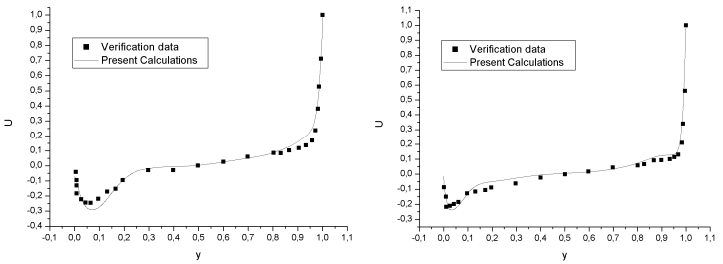

Figure 3: Data comparison form [11]. a – Re=1 000; b- Re=10 000.

Results for Re=1 000 and 10 000 are given in fig. 3 from [10]. For Re=10 000 the flow is turbulent, so averaged velocity profiles are compared. It can be seen that the results of different authors have a close fit. For more results see Evstigneev [10].

4.1.2 Broken dam problem

Broken dam problem is a good model test case for the free surface flow problem that has good experimental and numerical results.

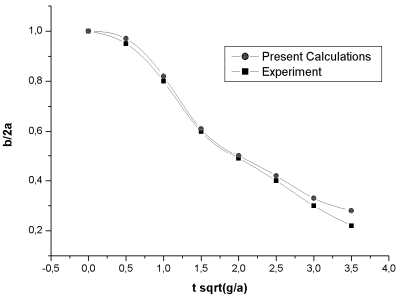

Figure 4: Comparison of numerical and experimental results for the dam break case, Re=500. 2a – initial liquid fraction height, b – current liquid fraction height; t – time in sec.

The test case has a rectangular geometry with ¼ volume filled with liquid. Experimental results were obtained for the same geometry from Sussman et al. [5], results are shown in fig. 4 for Re=500.

4.1.3 Turbulent flow in a circular pipe

This test case is a classic case study for turbulent flow verification for incompressible viscous flows. The test case has detailed data obtained from DNS laminar-turbulent Poiseuille flow transition studies as well as for the developed turbulent regimes, e.g. Chunlei et al. [9].

The case study has a circular geometry with solid wall boundary conditions and periodic boundary conditions on an outflow border. Inflow boundary conditions are laminar as well as initial conditions with no initial perturbations.

Results of turbulent energy distribution from numerical investigation with comparison to the other authors' results are shown in fig. 5. Energy spectrum comparison shows good data correlation with other results. For more information see Evstigneev [11].

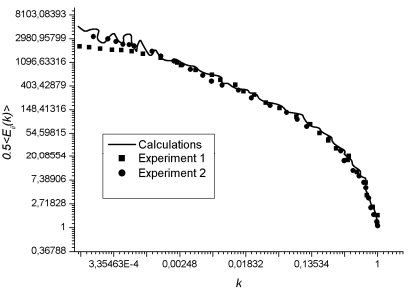

Figure 5: Kinetic turbulent energy spectrum comparison for turbulent Poiseuille flow in a circular pipe for Re=12 000. E – mean turbulent energy, k – wave number.

All tests of the numerical method presented in this paper showed good results and data fit.

4.2 Applications for technical hydrodynamics

The method was designed for technical application and has already been used for hydrodynamic investigation of complex hydraulic power plants. One example of its technical application is presented in the present work.

4.2.1 Investigation of turbulent free surface flow over a power intake of Zagorskaya hydraulic power plant

Input geometry was presented in AutgoCAD and translated to preprocessor format by a Visual Basic adaptive program. An adaptive tetrahedral grid was created automatically by the preprocessor; the fragment of surface is shown in fig. 1. Governing system of equations (1) was used with Carioles acceleration. Integration was conducted on 4 computers on a TCP-IP network using a combination of march and predictor-corrector algorithms.

Boundary conditions on the input boundary are given by the time changing discharge Q(t) through the intake. Other boundary conditions are standard. Free surface is given on a constant level with no perturbations.

The numerical experiment was carried out for 10 hours of working time for the power intake. Averaged vector function V on the surface is presented on fig. 7. A laboratory model scaled 1/50 was made for the given intake geometry for experimental hydrodynamic investigation. Velocities were measured by thermoanemometers. Data was presented to the author by the Hydroelectric Power Plant maintenance organization. Numerical and experimental data comparison for laboratory scaled model on twelve vertical sections is done with 5% mean results deviation. Two verticals for the first section (15sm from the power input in 1:50 scale) are presented in fig. 6.

Average and pulsation velocities and pressure functions, turbulent stress, free surface levels are calculated as the result of numerical investigation. Obtained data was used for power intake construction optimization for the second power plant construction queue.

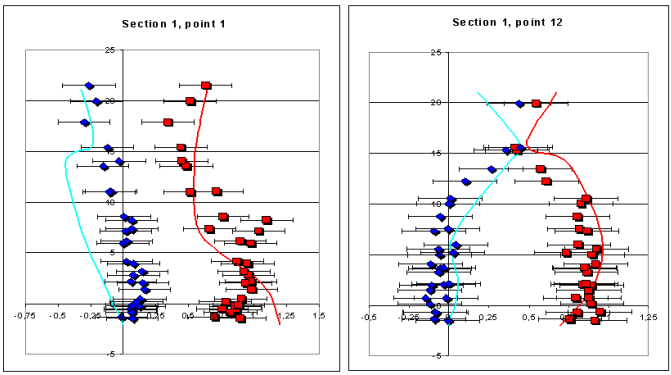

Figure 6: Comparison of computational (lines) and experimental data for averaged streamwise (right) and spanwise (left) averaged velocities.

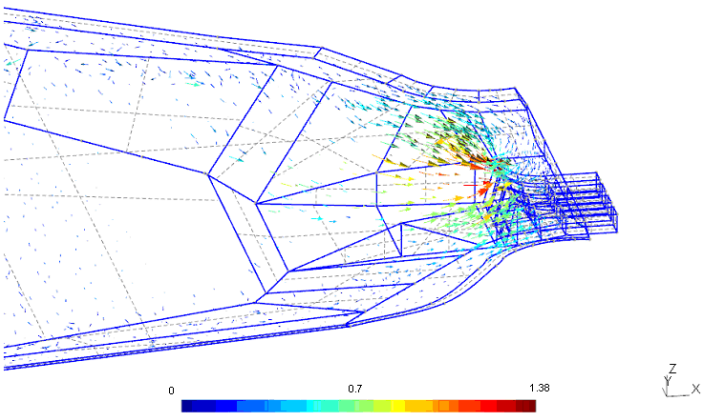

Figure 7: Averaged vector-function V in E^3 on the surface of the power-intake.

5 Conclusions

The purpose of this paper is to present a new numerical method for viscous incompressible flow calculation in R^4 with utilization of distributed computations and large eddy simulation methodology of free surface flows, governed by Navier-Stokes equations. The integration is conducted on a unstructured tetrahedral grid. This allows the use of the method for complex geometry of technical challenging tasks. A high order finite volume method for flux calculation is used. To maintain monotonous solutions a TVD SuperBEE flux limiter is applied. A large eddy simulation method is considered for almost any averaging scale. The distributed computation on cheap PC networks allows the use of any arbitrary network as a computational environment.

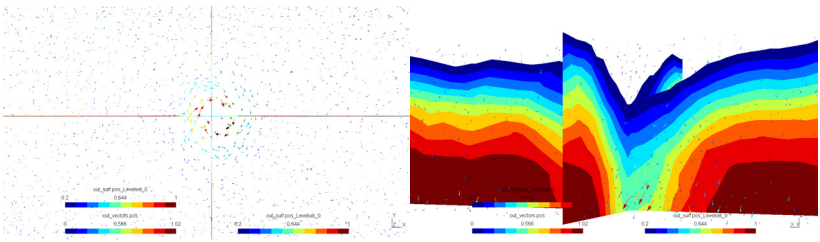

Figure 8: Zoomed velocity vectors (V) and free surface depression (F) near one of the vertexes.

Numerical results and data comparison for case studies demonstrated good performance. The method is used to calculate turbulent free surface flows on complex geometries. Two technical tasks are considered to demonstrate the ability of the method in handling complex turbulent regimes and geometries.

The numerical method, presented in the paper, is part of a complex CFD module that was developed by the author. Other parts of the complex, used for compressible viscous and inviscid flow computations, are presented in other papers [9,12].

References

[1] Temam R. Navier-Stokes equations – theory and numerical analysis.- North Holland Publishing Company, Oxford, 1981.

[2] Yanenko N.N., Kuznetsov B.G., Smagulov Sh. On the approximation of the Navier-Stokes equations for an incompressible fluid by evolutionary-type equations // Numerical Methods in Fluid Dynamics. - Moscow, 1984. - P.290–313.

[3] Hairer E.,Norsett S.P., Wanner G. Solving ordinary differentional equations. - V1., Springer-Verlag Berlin Heidelerg, 1987.

[4] Fletcher C.A.J., Computational Techniques for Fluid Dynamics - Springer-Verlag, 1056 p., 1988.

[5] Sussman M., Smereka P. and Osher S. A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow //, J. Comput. Phys. 114, pp.146–159, 1994.
[6] Chorin A., A Numberical Method for Solving Incompressible Viscous Flow Problems, // J. Comput. Phys. 2, pp.12–26, 1967.
[7] P. Cignoniz, C. Montaniz, R. Scopigno. DeWall: A Fast Divide & Conquer Delaunay Triangulation Algorithm in Ed. // The Computer J., 19(2):pp178–181, 2006.
[8] Roe P.L. Characteristic based schemes for the Euler equations. // Annual Review of Fluid Mechanics. Vol. 18, pp. 337–365., 1986.
[9] Chunlei Liang, Evstigneev N., A study of kinetic energy conserving scheme using finite volume collocated grid for LES of a channel flow. // Proceedings of the international conference on numerical methods in fluid dynamics. King's College London, Strand, WC2R 2LS, 2006.
[10] Evstigneev N.M. Lid driven cavity flow on various Reynolds numbers using 3D unstructured method for Navier-Stokes equations. // Moscow, Journal of Fundamental and technical science 1, 2006. (in russian)
[11] Evstigneev N.M. LES of Poiseuille flow in a circular pipe for transition and turbulent regimes. // Moscow, Journal of Fundamental and technical science №3, 2006г.
[12] Evstigneev N.M. Solution of 3D nonviscous compressible gas equations on unstructured meshes using the distributed computing approach. // J. of Comp. Math. and Math. Physics. V.8, pp. 252–264, 2007.