Combined single domain and subdomain BEM for 3D laminar viscous flow

J. Ravnik *, L. Škerget, Z. Žunič

Faculty of Mechanical Engineering, University of Maribor, Smetanova 17, SI-2000 Maribor, Slovenia

Abstract

A subdomain boundary element method (BEM) is presented for the solution of Laplace, Poisson and diffusion advection equations using a continuous quadratic interpolation of function and discontinuous linear interpolation of flux. By employing compatibility conditions between subdomains an over-determined system of linear equations is obtained, which is solved in a least squares manner. The method, combined with the single domain BEM, is used to solve laminar viscous flows using the velocity vorticity formulation of Navier-Stokes equations. The versatility and accuracy of the method is proven using heat transfer, entry flow, 3D channel flow and 2D and 3D lid driven cavity test cases.

Key words: subdomain boundary element method, velocity-vorticity formulation, laminar viscous fluid flow, Laplace equation, Poisson equation, lid driven cavity
PACS: 47.11.+j, 44.25.+f

1 Introduction

The main advantage of the Boundary Element Method (BEM) is the ability to solve partial differential equation by solving for boundary unknowns only, omitting the discretization of the domain. This advantage is lost when a suitable fundamental solution can not be found and a domain contribution remains in the integral equation. This happens when solving the Poisson equation or the vorticity transfer equation. Several procedures have been proposed to avoid this difficulty, for instance methods based on the expansion of the

* Corresponding author.
Email addresses: jure.ravnik@uni-mb.si (J. Ravnik), leo@uni-mb.si (L. Škerget), zoran.zunic@uni-mb.si (Z. Žunič).

integral kernel (Bebendorf [1]), dual reciprocity method (Jumarhon et al. [2]) or compression of the resulting full matrices (Epplera [3]). We employed the subdomain method (Popov et al. [4]), in which the domain is discretizised into subdomains and BEM is applied on each subdomain. The resulting integral matrices are sparse and as such may be stored efficiently and enable fast algebraic operations.

In the last decade our research group developed several BEM based numerical algorithms for the solution of viscous incompressible and compressible, laminar and turbulent flows by solving the velocity-vorticity formulation of the Navier-Stokes equations. With the aim of increasing computational grid density and decreasing computational times the single domain BEM approach has been coupled by several numerical procedures, such as wavelet compression (Ravnik et al. [5,6]) and the Finite Element Method (Žunič et al. [7]).

In this work, we are presenting a 3D viscous laminar flow solver which is based on a combination of single domain BEM and subdomain BEM. In the subdomain BEM integral equations are written for each subdomain (mesh element) separately. We use continuous quadratic boundary elements for the discretization of function and discontinuous linear boundary element for the discretization of flux. By the use of discontinuous discretization of flux all flux nodes are within boundary elements where the normal and the flux are unambiguously defined. The corners and edges, where the normal is not well defined, are avoided. The singularities of corners and edges were dealt with special singular shape functions by Ong and Lim [8] and by the use of additional nodes by Gao and Davies [9]. By the use of a collocation scheme a single linear equation is written for every function and flux node in every boundary element. By using compatibility conditions between subdomains, we obtain an over-determined system of linear equations, which may be solved in a least squares manner. The governing matrices are sparse and have similar storage requirements as the finite element method. In the paper we present subdomain BEM for the solution of the Laplace, Poisson, diffusion advection and vorticity transport equations. Ramšak et al. [10] employed a similar approach for the 3D Laplace equation, but using a lower order interpolation scheme.

The solution of viscous flow using a velocity-vorticity formulation of Navier-Stokes equations requires an iterative scheme for solution of both velocity and vorticity fields. The main challenge lies in the determination of boundary vorticity values, which are needed for the solution of the vorticity transport equation. Several different approaches have been proposed for the determination of vorticity on the boundary. Daube [11] used an influence matrix technique to enforce both the continuity equation and the definition of the vorticity in the treatment of the 2D incompressible Navier-Stokes equations. Liu [12] recognised that the problem is even more severe when he extended it to three dimensions. Lo et al. [13] used a differential quadrature method to calculate

vorticity from its definition to obtain a solution of a natural convection problem. We will use single domain BEM solution of the kinematics equation for determination of boundary vorticity. This approach was introduced by Škerget et al. [14] in 2D and used coupled by FEM in 3D by Žunič et al. [15].

The rest of the paper is organised as follows: in the second section we introduce the subdomain BEM numerical solution methods for Laplace, Poisson and diffusion advection equations complete with derivations and numerical examples. In the third section we combine the newly developed methods with single domain BEM to solve the Navier-Stokes equations. In the fourth section viscous laminar flow test cases are presented.

2 Subdomain BEM for the Laplace, Poisson and diffusion advection equations

With the final goal of solving the velocity-vorticity formulation of Navier-Stokes equation, we first developed and tested a 3D subdomain BEM for Laplace, Poisson and diffusion advection equations.

2.1 Solution of the Laplace equation by subdomain BEM

The Laplace equation for a scalar field function $u(\vec{r})$, $\vec{r} \in \mathbb{R}^3$ is

$$\nabla^2 u(\vec{r}) = 0. \tag{1}$$

We seek a solution for u in a domain $\Omega \in \mathbb{R}^3$ with a boundary $\Gamma = \partial\Omega \in \mathbb{R}^2$. The function on the boundary

$$u(\vec{r}), \qquad \vec{r} \in \Gamma \tag{2}$$

or the flux on the boundary

$$q(\vec{r}) = \vec{\nabla} u(\vec{r}) \cdot \vec{n}, \qquad \vec{r} \in \Gamma \tag{3}$$

are prescribed as boundary conditions. Vector normal to the boundary pointing out of the domain is denoted by $\vec{n}$. Boundary integral representation of the Laplace equation (Wrobel [16], Ravnik [17]) is

$$c(\vec{\vartheta}) u(\vec{\vartheta}) + \int_\Gamma u \vec{\nabla} u^\star \cdot \vec{n} d\Gamma = \int_\Gamma u^\star q d\Gamma \tag{4}$$

where $\vec{\vartheta}$ is the source or collocation point. $c(\vec{\vartheta})$ is the geometric factor defined as

$$c(\vec{\vartheta}) = \int_0^{\alpha} \frac{d\theta}{4\pi} = \frac{\alpha}{4\pi}, \tag{5}$$

where α is the inner angle with origin in $\vec{\vartheta}$. If $\vec{\vartheta}$ lies inside of the domain then $c(\vec{\vartheta}) = 1$; $c(\vec{\vartheta}) = 1/2$, if $\vec{\vartheta}$ lies on a smooth boundary. The fundamental solution of the Laplace equation in $3D$ is

$$u^{\star} = \frac{1}{4\pi|\vec{\vartheta} - \vec{r}|}. \tag{6}$$

In the subdomain BEM numerical method we divide the computational domain into mesh elements (subdomains). We write the equation (4) for each mesh element. By enforcing compatibility conditions between elements we are able to form an over-determined sparse linear system of equations.

The mesh elements used in this work are hexahedrons. On each face of the hexahedron (Figure 1) we used continuous quadratic interpolation for function and discontinuous linear interpolation for flux:

$$u(\xi,\eta) = \sum_{i=1}^{9} \varphi_i u_i, \qquad q(\xi,\eta) = \sum_{i=1}^{4} \phi_i q_i, \tag{7}$$

where u_i are function values in each function node, q_i are flux values in flux nodes and (ξ,η) are local coordinate system axes. The shape functions for function φ_i are the standard shape functions for a quadratic nine node Lagrangian element, while the shape functions for flux are:

$$\begin{aligned} \phi_1 &= \frac{4}{9}(\xi - \frac{3}{4})(\eta - \frac{3}{4}), \qquad & \phi_2 &= -\frac{4}{9}(\xi + \frac{3}{4})(\eta - \frac{3}{4}), \\ \phi_3 &= \frac{4}{9}(\xi + \frac{3}{4})(\eta + \frac{3}{4}), \qquad & \phi_4 &= -\frac{4}{9}(\xi - \frac{3}{4})(\eta + \frac{3}{4}). \end{aligned} \tag{8}$$

The geometry of the hexahedron is defined by the 8 corner nodes, thus each surface is defined by 4 nodes (numbers 1,3,5,7 in Figure 1). One may find the location of flux nodes (a,b,c,d) by using the following transformation

$$\begin{vmatrix} x_a & y_a & z_a \\ x_b & y_b & z_b \\ x_c & y_c & z_c \\ x_d & y_d & z_d \end{vmatrix} = \frac{1}{64} \begin{vmatrix} 49 & 7 & 1 & 7 \\ 7 & 49 & 7 & 1 \\ 1 & 7 & 49 & 7 \\ 7 & 1 & 7 & 49 \end{vmatrix} \cdot \begin{vmatrix} x_1 & y_1 & z_1 \\ x_3 & y_3 & z_3 \\ x_5 & y_5 & z_5 \\ x_7 & y_7 & z_7 \end{vmatrix}. \tag{9}$$

All flux nodes are located within boundary elements, none are located at corners and edges - thus the unit normal and the flux value are unambiguously

defined in each flux node. Having the shape functions in mind, we may rewrite the integral equation (4) to obtain

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \sum_{i=1}^{26} u_i \int_\Gamma \varphi_i \vec{\nabla} u^\star \cdot \vec{n} d\Gamma = \sum_{i=1}^{24} q_i \int_\Gamma \phi_i u^\star d\Gamma, \tag{10}$$

where the first sum goes to 26 and the second to 24 since in each hexahedral mesh element (Figure 2) there are 26 function nodes on the surfaces (8 in corners, 6 on the middle of surfaces and 12 on the middle of edges) and 24 flux nodes (4 on each side). The integrals are traditionally named as

$$H^i_{\vartheta,\Gamma} = \int_\Gamma \varphi_i \vec{\nabla} u^\star \cdot \vec{n} d\Gamma, \quad G^i_{\vartheta,\Gamma} = \int_\Gamma \phi_i u^\star d\Gamma. \tag{11}$$

and thus the discrete equation is

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \sum_{i=1}^{26} u_i H^i_{\vartheta,\Gamma} = \sum_{i=1}^{24} q_i G^i_{\vartheta,\Gamma}. \tag{12}$$

In order to calculate the integrals, a Gaussian quadrature algorithm is used. The integrals are calculated in local coordinate system via weighted summation of up to 48 integration points per coordinate axis. In the case of high aspect ratios of hexahedral elements, the boundary elements are divided into parts whose aspect ratio is approximately equal to one.

Calculation of the free coefficient $c(\vec{\vartheta})$ is preformed indirectly. If we consider a rigid body movement, $u = 1$, $q = 0$, we see that the sum of all $H^i_{\vartheta,\Gamma}$ matrix elements for one source point must be equal to 0, thus we may use this fact to calculate $c(\vec{\vartheta})$. If the source point is located on the surface, we know that $c = 1/2$, also if the source point is inside of the element then $c = 1$. These two relationships are used to check the accuracy of the calculated integrals.

In order to set up a system of equations the source point is set in all function and flux nodes of all mesh elements. Additionally, the source point is set into a node in the centre of the hexahedron, where the function value may be obtained explicitly from known boundary values. Thus all in all we have 51 equations for each element. The corresponding integrals are stored in rectangular matrices, which have 51 times the number of elements rows. The $[H]$ integral matrix has 26 columns, while the $[G]$ matrix has 24 columns. Since neighbouring elements share nodes and since boundary conditions on the outer boundaries of the domain are prescribed, we obtain an over-determined system of equations. The system is sparse. We store the system matrix in compressed row storage format. The system is solved in a least squares manner (Paige and Saunders [18]).

2.1.1 Test cases for the Laplace equation

The validity of programming and the quality of the method was tested on two heat transfer studies. Firstly, we consider heat transfer over a solid cubic domain. Function u is prescribed on two opposite sides, while $q = 0$ is prescribed on the other four. The second investigated example was heat transfer over a solid cube with a cubic empty space inside. On two sides of the cube and on two sides of the empty space the function was prescribed, on all other sides adiabatic boundary condition was employed ($q = 0$).

The accuracy of the calculated function and flux was compared with the analytical values. A RMS error was defined as:

$$RMSerror = \sqrt{\frac{\sum_i (f_i - a_i)^2}{\sum_i a_i^2}}, \tag{13}$$

where f_i is the calculated value in node i and a_i is the analytical value in node i. The accuracy of integration was measured using known values for the free coefficient c (5).

Table 1 summarizes the different meshes used. Meshes a-h were used to calculate heat transfer over a cube, meshes i-j were used for the cube in a cube numerical example. Meshes b and j are displayed on Figure 3.

Both function solutions are plotted in Figure 4. The accuracy of integrals and the comparison of calculated solution versus analytical solution is shown in Table 2. A very high order of accuracy was achieved in all cases. The main conclusion drawn from this analysis is that the accuracy of the solution depends on the accuracy of the calculation of integrals. For elements with large aspect ratio (meshes c and f), we needed to divide the integration surface in order to achieve high order of accuracy. This prolonged integration times severely.

The accuracy of the cube in a cube test case is satisfactory, but lower than the accuracy of the heat transfer in a cube test case. This is due to the fact that the meshes used in the cube in a cube test case employed heavily distorted hexahedra, while we used cubes in the heat transfer in a cube test case.

2.2 Solution of the Poisson equation by subdomain BEM

The non-homogenous Laplace equation is known as the Poisson equation. For a scalar field function $u(\vec{r})$, $\vec{r} \in \mathbb{R}^3$ it may be stated as

$$\nabla^2 u(\vec{r}) = b(\vec{r}), \tag{14}$$

where the non-homogenous part $b(\vec{r})$ is defined in the whole domain Ω. Its integral representation has an additional domain integral compared to the integral representation of the Laplace equation (4):

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \int_\Gamma u\vec{\nabla}u^\star \cdot \vec{n}d\Gamma = \int_\Gamma u^\star q d\Gamma + \int_\Omega u^\star b d\Omega. \tag{15}$$

By using domain shape functions we may interpolate b inside each hexahedra and obtain, similarly to the Laplace equation case, a discrete system of equations:

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \sum_{i=1}^{26} u_i H^i_{\vartheta,\Gamma} = \sum_{i=1}^{24} q_i G^i_{\vartheta,\Gamma} + \sum_{i=1}^{27} b_i B^i_{\vartheta,\Omega}, \tag{16}$$

where the domain integrals are defined as

$$B^i_{\vartheta,\Omega} = \int_\Omega \Phi_i u^\star d\Omega \tag{17}$$

and Φ_i are the domain shape functions. When the source point is set to all function and flux nodes of all mesh elements, we obtain a linear system of equations in exactly the same manner as in the Laplace equation case. The integrals matrix $[B]$ has 51 times the number of elements rows and only 27 columns. Although the domain integral is present in the integral equation, due to the subdomain approach, we are still dealing with matrix sizes, which scale linearly with the number of mesh elements. This is not the case in single domain BEM, where a domain integral requires full matrices, which scale with the square of the number of nodes in the mesh.

2.2.1 Test cases for the Poisson equation

The following Poisson equations were solved on a unit cube:

a) equation $\nabla^2 u = 2$, analytical solution $u = x^2$, boundary conditions $u(0, y, z) = 0$, $u(1, y, z) = 1$, $q(x, 0, z) = q(x, 1, z) = q(x, y, 0) = q(x, y, 1) = 0$;
b) equation $\nabla^2 u = 6x$, analytical solution $u = x^3$, boundary conditions $u(0, y, z) = 0$, $u(1, y, z) = 1$, $q(x, 0, z) = q(x, 1, z) = q(x, y, 0) = q(x, y, 1) = 0$;
c) equation $\nabla^2 u = 12x^2$, analytical solution $u = x^4$, boundary conditions $u(0, y, z) = 0$, $u(1, y, z) = 1$, $q(x, 0, z) = q(x, 1, z) = q(x, y, 0) = q(x, y, 1) = 0$.

The test cases were examined on a, b, d, f, g and h meshes (see Table 1 for descriptions). The RMS errors against the analytical solutions are shown in Table 3, while the CPU times and solver iterations are presented in Table 4. The subdomain BEM solves the first test case (a) up to machine precision on all meshes, since our interpolation is high enough. For the other two, the RMS error quickly diminishes with the increasing mesh density.

As the size of the system of linear equations increases with increasing mesh density the number of solver iterations and the CPU time increase as well. The CPU time was measured on a 3.2Gz Pentium IV processor. The solver stopping criteria was set to 10^{-15}.

2.3 Solution of the diffusion advection equation by subdomain BEM

The steady state diffusion advection equation is

$$(\vec{v}\cdot\vec{\nabla})u = \frac{1}{Pe}\nabla^2 u, \tag{18}$$

where we used the fluid (advection) velocity $\vec{v}(\vec{r})$, which has been nondimensionalized by v_0. The problem is governed by the Peclet number $Pe = v_0/\alpha$ with diffusivity α. Equation (18) may be written in integral form in the same manner as the Poisson equation, but with a non-linear non-homogenous part:

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \int_\Gamma u\vec{\nabla}u^\star\cdot\vec{n}d\Gamma = \int_\Gamma u^\star q d\Gamma + Pe\int_\Omega u^\star\{(\vec{v}\cdot\vec{\nabla})u\}d\Omega. \tag{19}$$

In this work, we consider only incompressible fluids, $\vec{\nabla}\cdot\vec{v}=0$, thus the domain integral may be written in the following form

$$\int_\Omega u^\star\vec{\nabla}\cdot(\vec{v}u)d\Omega. \tag{20}$$

In order to move the derivative towards the fundamental solution, the following algebraic relation comes in handy $\vec{\nabla}\cdot\{u^*\vec{v}u\} = u^*\vec{\nabla}\cdot(\vec{v}u) + \vec{v}u\cdot\vec{\nabla}u^*$. Using this relationship in (20) as well as the Gauss divergence clause, we write the final integral form of the diffusion advection equation as

$$\begin{aligned} c(\vec{\vartheta})u(\vec{\vartheta}) + \int_\Gamma u\vec{\nabla}u^\star\cdot\vec{n}d\Gamma &= \int_\Gamma u^\star q d\Gamma \\ +Pe\int_\Gamma \vec{n}\cdot\{u^\star\vec{v}u\}d\Gamma - Pe\int_\Omega(\vec{v}u)\cdot\vec{\nabla}u^\star d\Omega. \end{aligned} \tag{21}$$

Each component of the product of velocity and the unknown function is interpolated along the boundary and domain elements using shape functions, i.e. $v_x u = \sum_i \Phi_i (v_x u)_i$. The following integrals must be calculated:

$$\vec{A}^i_{\vartheta,\Gamma} = \int_\Gamma \varphi_i\vec{n}u^\star d\Gamma, \qquad \vec{D}^i_{\vartheta,\Omega} = \int_\Omega \Phi_i\vec{\nabla}u^\star d\Omega. \tag{22}$$

The discrete counterpart of equation (21) is given by

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \sum_{i=1}^{26} u_i H^i_{\vartheta,\Gamma} = \sum_{i=1}^{24} q_i G^i_{\vartheta,\Gamma} + Pe\sum_{i=1}^{26}(\vec{v}u)_i\cdot\vec{A}^i_{\vartheta,\Gamma} - Pe\sum_{i=1}^{27}(\vec{v}u)_i\cdot\vec{D}^i_{\vartheta,\Omega}. \tag{23}$$

When the source point is set into all function and flux nodes of all mesh elements, the system is stored and solved in the same manner, which was described in the Laplace equation case. The advection term and the 3D nature of the problem requires the calculation of three boundary integrals matrices $[\vec{A}]$ and three domain integrals matrices $[\vec{D}]$. The number of rows of all matrices is equal to the number of mesh elements times 51. The number of columns of the boundary matrix is equal to the number of nodes on the boundary of each mesh element, i.e. 26 and the number of columns of the domain matrices is 27, i.e. the number of all nodes in each mesh element.

Since the diffusion advection equation is nonlinear, an iterative scheme is employed to get a converged solution. The integrals depend only on the mesh geometry, shape functions and the fundamental solution. As such they may all be calculated prior to the start of the iteration process and used in each iteration.

2.3.1 Test case for the diffusion advection equation

We tested the subdomain BEM solution of the diffusion advection equation on an entry flow problem. In an interval $x \in [0, 1]$ with a constant velocity field $\vec{v} = (1, 0, 0)$ the analytical solution of the diffusion advection equation (18) for function and flux equals

$$u(x) = 1 - \frac{1 - e^{Pex}}{1 - e^{Pe}}, \qquad du/dx = Pe\frac{e^{Pex}}{1 - e^{Pe}}. \tag{24}$$

We solved the problem on a coarse mesh with $10 \times 1 \times 1$ elements and on a fine mesh with $100 \times 1 \times 1$ elements. The elements in the coarse mesh were concentrated towards the high flux region ($x = 1$). Function was prescribed at $x = 0$ and $x = 1$, while $q = 0$ was imposed on all other walls. The results of simulations for Peclet number values $Pe = 1$, $Pe = 10$ and $Pe = 20$ are shown in Table 5. We are comparing function value in the middle of the domain and flux at $x = 1$. The results obtained on both meshes are in good agreement with the analytical solution. Meshes with concentrated elements give better results than meshes with an equal number of equidistant elements, due to the fact that concentrated elements meshes have more nodes in the high gradient region.

3 Navier-Stokes equations in velocity-vorticity formulation

In this paper we assume an incompressible viscous Newtonian fluid with constant material properties. Vorticity $\vec{\omega}$ is defined as the curl of the velocity

$\vec{\omega} = \vec{\nabla} \times \vec{v}$. Both velocity and vorticity fields are divergence free. The viscous fluid flow is governed by the kinematics equation

$$\nabla^2 \vec{v} + \vec{\nabla} \times \vec{\omega} = 0, \tag{25}$$

which is a vector elliptic partial differential equation of Poisson type and links the velocity and vorticity fields for every point in space and time. It is equivalent to the Biot-Savart law, which connects the electric current and magnetic field density. The same connection that links the electric current and magnetic field density links velocity and vorticity fields in fluid flow (Lundgren and Koumoutsakos [19]). The kinetic aspect of fluid movement is governed by the vorticity transport equation, written in non-dimensional form:

$$\frac{\partial \vec{\omega}}{\partial t} + (\vec{v} \cdot \vec{\nabla})\vec{\omega} = (\vec{\omega} \cdot \vec{\nabla})\vec{v} + \frac{1}{Re}\nabla^2 \vec{\omega}, \tag{26}$$

with the Reynolds number denoted by Re. Equation (26) equates the advective vorticity transport on the left hand side with the vortex twisting and stretching term and the diffusion term on the right hand side. In this paper we are dealing with steady flows only, which makes $\partial\vec{\omega}/\partial t = 0$. The system of equations (25) and (26) is nonlinear. The unknowns are the velocity and vorticity fields. Our solution algorithm is presented below:

- calculate integrals
- begin nonlinear loop
  - calculate boundary vorticity values by solving the kinematics equation by single domain BEM
  - calculate domain velocity values by solving the kinematics equation by subdomain BEM
  - solve vorticity transport equation for domain vorticity values using the boundary values from the solution of the kinematics equation by subdomain BEM
  - check convergence - repeat steps in the nonlinear loop until convergence is achieved
- end nonlinear loop
- output results

The boundary conditions required to obtain the solution are prescribed velocity or velocity flux on the boundary. The boundary conditions for the vorticity transport equation are calculated as a part of the algorithm using single domain BEM. Alternatively, if boundary vorticity or flux is known on a part of the boundary, it may be prescribed, and thus the single domain BEM is not used to calculate boundary values.

The integral form of the steady vorticity transport equation (26) is

$$c(\vec{\vartheta})\vec{\omega}(\vec{\vartheta}) + \int_\Gamma \vec{\omega}\vec{\nabla}u^\star \cdot \vec{n}d\Gamma = \int_\Gamma u^\star \vec{q}d\Gamma$$
$$+Re\int_\Omega u^\star \left\{(\vec{v}\cdot\vec{\nabla})\vec{\omega} - (\vec{\omega}\cdot\vec{\nabla})\vec{v}\right\} d\Omega, \tag{27}$$

where the domain integral now includes the advection and vortex twisting and stretching terms. $\vec{q}$ is the vorticity flux vector $q_j = \vec{n}\cdot\vec{\nabla}\omega_j$. Let us consider only the j^{th} component of the vector equation (27) and look at the domain integral alone:

$$\int_\Omega \left\{(\vec{v}\cdot\vec{\nabla})\omega_j - (\vec{\omega}\cdot\vec{\nabla})v_j\right\} u^\star d\Omega. \tag{28}$$

Due to the solenoidality of the velocity and vorticity fields, we may use $(\vec{\omega}\cdot\vec{\nabla})v_j = \vec{\nabla}\cdot(\vec{\omega}v_j)$ and $(\vec{v}\cdot\vec{\nabla})\omega_j = \vec{\nabla}\cdot(\vec{v}\omega_j)$ to transform eqaution (28) into

$$\int_\Omega \left\{\vec{\nabla}\cdot(\vec{v}\omega_j - \vec{\omega}v_j)\right\} u^\star d\Omega. \tag{29}$$

In order to move the derivative towards the fundamental solution, the following algebraic relation $\vec{\nabla}\cdot\{u^*(\vec{v}\omega_j - \vec{\omega}v_j)\} = u^*\vec{\nabla}\cdot(\vec{v}\omega_j - \vec{\omega}v_j) + (\vec{v}\omega_j - \vec{\omega}v_j)\cdot\vec{\nabla}u^*$ is used to obtain two integrals:

$$\int_\Omega \vec{\nabla}\cdot\{u^*(\vec{v}\omega_j - \vec{\omega}v_j)\}\, d\Omega - \int_\Omega (\vec{v}\omega_j - \vec{\omega}v_j)\cdot\vec{\nabla}u^\star d\Omega \tag{30}$$

The first integral may be converted to a boundary integral using a Gauss divergence clause. Thus the final integral form of the vorticity transport equation (27) for j^{th} vorticity component may be stated as

$$c(\vec{\vartheta})\omega_j(\vec{\vartheta}) + \int_\Gamma \omega_j\vec{\nabla}u^* \cdot \vec{n}d\Gamma = \int_\Gamma u^* q_j d\Gamma$$
$$+Re\int_\Gamma \vec{n}\cdot\{u^*(\vec{v}\omega_j - \vec{\omega}v_j)\}\, d\Gamma - Re\int_\Omega (\vec{v}\omega_j - \vec{\omega}v_j)\cdot\vec{\nabla}u^* d\Omega. \tag{31}$$

The products of velocity and vorticity field components are interpolated within elements using shape functions. With this, the integrals required to solve the vorticity transport equation are the same as the integrals for the diffusion advection equation (21). We may write the discrete equation as

$$c(\vec{\vartheta})u(\vec{\vartheta}) + \sum_{i=1}^{26} u_i H^i_{\vartheta,\Gamma} = \sum_{i=1}^{24} q_i G^i_{\vartheta,\Gamma}$$
$$+Re\sum_{i=1}^{26}(\vec{v}\omega_j - \vec{\omega}v_j)_i \cdot \vec{A}^i_{\vartheta,\Gamma} - Re\sum_{i=1}^{27}(\vec{v}\omega_j - \vec{\omega}v_j)_i \cdot \vec{D}^i_{\vartheta,\Omega}. \tag{32}$$

The vortex twisting and stretching term does not require any additional integration or storage space. It does, naturally, affect the nonlinearity of the problem.

3.2 Solution of the kinematics equation for domain velocity by subdomain BEM

The integral form of the kinematics equation without derivatives of the velocity and vorticity fields takes the following form (for derivation, see Ravnik et al. [6] eqns. (19)-(24)):

$$c(\vec{\vartheta})\vec{v}(\vec{\vartheta}) + \int_\Gamma \vec{v}\vec{\nabla}u^\star \cdot \vec{n}d\Gamma = \int_\Gamma \vec{v} \times (\vec{n} \times \vec{\nabla})u^\star d\Gamma + \int_\Omega (\vec{\omega} \times \vec{\nabla}u^\star)d\Omega. \quad (33)$$

The boundary integrals on the left hand side are stored in the $[H]$ matrix, the domain integrals on the right hand side are the $[\vec{D}]$ matrices. We define the boundary integral on the right hand side as $[\vec{H}^t]$ integrals in the following manner:

$$\vec{H}^t_{\vartheta,\Gamma,i} = \int_\Gamma \varphi_i(\vec{n} \times \vec{\nabla})u^\star d\Gamma. \quad (34)$$

Since there are no fluxes in the equation, the source point is set to function nodes only. Let vectors of nodal values of field functions be denoted by curly brackets. The discrete kinematics equation written in component wise form is:

$$[H]\{v_x\} = [H^t_z]\{v_y\} - [H^t_y]\{v_z\} + [D_z]\{\omega_y\} - [D_y]\{\omega_z\}, \quad (35)$$

$$[H]\{v_y\} = [H^t_x]\{v_z\} - [H^t_z]\{v_x\} - [D_z]\{\omega_x\} + [D_x]\{\omega_z\}, \quad (36)$$

$$[H]\{v_z\} = [H^t_y]\{v_x\} - [H^t_x]\{v_y\} + [D_y]\{\omega_x\} - [D_x]\{\omega_y\}. \quad (37)$$

Using $[H]$ as the system matrix the three linear systems of equations must be solved repeatedly, until convergence is achieved. This is due to the fact that the right hand sides depend on velocity as well. No under-relaxation was needed in our simulations. We notice, that the $[H]$ and $[\vec{D}]$ integral matrices are needed for the vorticity transport equation as well and are thus used twice.

All in all the subdomain BEM solution of the kinematics and vorticity transport equations requires the calculation and storage of $[H]$, $[G]$, $[\vec{A}]$, $[\vec{H}^t]$ and $[\vec{D}]$ matrices. The total number of integrals that must be calculated and stored is 12540 times the number of mesh elements. In comparison with the single domain BEM this is a very small number. The single domain BEM would require at least three matrices with the number of elements equal to the number of nodes squared. On a cubic mesh with $10 \times 10 \times 10$ elements with 9261 nodes, the ratio between subdomain BEM storage requirements and single domain BEM storage requirements would be approximately 0.04 and on a $20 \times 20 \times 20$ mesh it would be approximately 0.007.

3.3 Solution of the kinematics equation for boundary vorticity by single domain BEM

In order to use the kinematics equation to obtain boundary vorticity values, we must rewrite the system of equations (35) - (37) in a tangential form by multiplying the system with a normal in the source point. This approach has been proposed by Škerget and used in $2D$ by Škerget et al. [14] and in 3D by Žunič et al. [15]. We employed the same procedure as Žunič et al. [15] with the difference of using the second order shape functions, while they used a first order interpolation scheme.

4 Test cases for viscous laminar flow

The proposed numerical scheme was tested on three test cases with known analytical or benchmark solutions.

4.1 2D lid driven cavity

In order to ascertain that the 3D code is capable of yielding 2D results, we have used it to simulate 2D lid driven cavity flow. The simulation was preformed in a block domain $(0, 0, 0) \times (1, 0.05, 1)$. The six walls were named as follows: left wall $x = 0$, right $x = 1$, top $z = 1$, bottom $z = 0$, symmetry walls $y = 0$, $y = 0.05$. The velocity boundary conditions were: top wall $v_x = 1$, bottom, left and right walls $v_x = 0$, symetry walls $\partial v_x/\partial n = 0$. The y component of velocity was set to zero on all walls, $v_y = 0$. The z component was set to zero on the top, bottom, left and right walls $v_z = 0$, while on the symmetry walls its normal derivative was set to zero $\partial v_z/\partial n = 0$. Vorticity ω_x and ω_z were set to zero on all walls. Boundary vorticity ω_y was calculated using single domain BEM from the kinematics equation on all walls, except for the symmetry walls, where $\partial \omega_y/\partial n = 0$. The boundary conditions are sketched on Figure 5.

The simulation was preformed on a mesh with $20 \times 1 \times 20$ elements, having in total 5043 nodes. The elements were concentrated towards the corners in the $x - z$ plane. The ratio of the shortest element width to the longest elements width was 7. The velocity profiles and vectors for $Re = 400$ are shown on Figure 6. We observe good agreement between our results and benchmark results of Ghia et al. [20].

Flow in a 3D channel with a rectangular cross-section has an analytical solution. The viscous fluid flows in a laminar fashion in positive x direction. The domain extends infinitely in x direction and in $y-z$ plane from $(0,0)\times(L_y, L_z)$. The analytical velocity profile (Chen [21]) is

$$v_x = -\frac{4L_y^2}{\mu\pi^3}\frac{dp_m}{dx}\cdot$$
$$\cdot \sum_{n=1,3,\ldots}^{\infty} \frac{(-1)^{(n-1)/2}}{n^3}\left[1-\frac{\cosh(n\pi(y-L_y/2)/L_y)}{\cosh(n\pi L_z/2L_y)}\right]\cos\left(\frac{n\pi(z-L_z/2)}{L_y}\right),$$
$$\frac{dp_m}{dx} = \frac{12\mu\overline{v_x}}{L_z^2}\left[1-\frac{192}{\pi^5}\frac{L_z}{L_y}\sum_{n=1,3,\ldots}^{\infty}\left(\frac{1}{n^5}\tanh\left(\frac{n\pi L_y}{2L_z}\right)\right)\right]^{-1} \quad (38)$$

We simulated flow in a square channel, $L_y = L_z = 1$ in domains of different lengths L_x. The domain extents were $(0,0,0)\times(L_x,1,1)$, where inflow was at $x = 0$, outflow $x = L_x$, top wall $z = 1$, bottom $z = 0$, front wall $y = 0$ and back wall at $y = 1$.

An uniform velocity profile is prescribed at the inflow $v_x = \overline{v_x} = 1$. The walls of the channel have non-slip velocity boundary conditions. At the outflow normal derivatives of all velocity components are prescribed as zero. Boundary values of vorticity are obtained by solving the kinematics equation by single domain BEM on all walls except, $\omega_x = 0$ at inflow, $\partial\omega_x/\partial n = 0$ at outflow, $\omega_y = 0$ on front and back walls and $\omega_z = 0$ on top and bottom walls. The boundary conditions are sketched on Figure 5.

To do the simulation, we used a mesh with 12^3 elements having in total 25^3 nodes. The elements were concentrated towards the four corners in $y-z$ plane. The ratio between the width of the smallest and largest elements was 27. In Figure 7 we plotted the velocity and vorticity profiles against the height of the channel. We observe good agreement with analytical solution given in equation (38). The Figure 7 also displays velocity vectors on the inflow and outflow planes as well as v_x contours.

4.3 3D lid driven cavity

Flow in a 3D lid driven cavity is one of the standard benchmark test cases used in development of flow solvers. The domain as well as the boundary conditions is unambiguously defined and do not change with the Reynolds number. The flow exhibits a wide variety of phenomena, such as: eddies, complex

three-dimensional patterns and instabilities (Shankar and Deshpande [22]). The research of lid driven cavity flow started with the observations of Koseff and Street [23], who were able to observe the inherent 3D nature of flow phenomena in the cavity.

The simulation was preformed on a unit cube $(0,0,0)\times(1,1,1)$. We named the walls in the following manner: left wall $x=0$, right $x=1$, top $z=1$, bottom $z=0$, front $y=0$ and back $y=1$. Non slip velocity boundary conditions are employed on all wall except the top wall, where a constant velocity in x direction is prescribed $\vec{v}=(1,0,0)$. Dirichlet type boundary conditions are used for the vorticity transport equation. Vorticity on the boundary is obtained by the solution of the kinematics equation for all directions and walls, except for $\omega_x=0$ on left and right walls, $\omega_y=0$ on front and back walls and $\omega_z=0$ on top and bottom walls. The boundary conditions are sketched on Figure 5.

Simulations were run on meshes of 8^3 elements with 4913 nodes, 10^3 elements with 9261 nodes and 12^3 elements with 15625 nodes. The elements were concentrated towards the eight corners. The Reynolds number for this test case is defined with the length of cavity's edge and the top wall velocity. We ran simulations at $Re=100$, $Re=400$ and $Re=1000$.

The moving lid induces a primary vortex inside of the cavity. The size of the vortex increases with Reynolds number. Secondary vortices appear in the corners of the cavity, their position and strength changing with Reynolds number. We compared velocity profiles within the cavity in the $y=0.5$ plane with the results of Yang et al. [24]. Figure 8 shows good agreement for $Re=100$, $Re=400$ on all meshes, while $Re=1000$ profiles are in good agreement with the reference only on the dense mesh. The high Reynolds number induces high gradients, which can only be described correctly on a dense mesh. In order to examine the flow structure within the cavity, we plotted iso-surfaces of velocity $|\vec{v}|=0.13$ on Figure 9 and iso-surfaces of vorticity $|\vec{\omega}|=1$ on Figure 10 for all three Reynolds numbers. The growth of the primary vortex with Reynolds number is evident. Along with the primary vortex the regions of high vorticity are also expanding.

5 Conclusions

We developed a 3D subdomain boundary element method based on the continuous quadratic interpolation of function and discontinuous linear interpolation of flux. Using discontinuous boundary elements for flux enabled us to avoid the undefined flux values in the corners and edges. The resulting over-determined system of linear equations was solved in a least squares manner. The accuracy and versatility of the method was shown by solving the Laplace, Poisson and

diffusion advection equations.

By combining the subdomain BEM with the single domain BEM we are able to solve velocity - vorticity formulation of Navier-Stokes equations and simulate viscous laminar flow in 3D. The method was successfully tested on 3D channel flow and flow in a 3D lid driven cavity up to Reynolds number $Re = 1000$.

In the near future we are planning to expand the method towards time dependent problems using a time dependent fundamental solution. The expansion of the solver to tackle coupled flow and heat transfer problems is under development. It requires a solution of the energy equation, which is of a diffusion advection type and an addition of a coupling term in the vorticity transport equation. Our final goal is the simulation of turbulent flows.

Acknowledgments

The first author gratefully acknowledges the support of the Wessex Institute of Technology during his stay at Ashurst Lodge in the U.K.

References

[1] M. Bebendorf, Approximation of boundary element matrices, Numer. Math 86 (2000) 565–589.

[2] B. Jumarhon, S. Amini, K. Chen, On the boundary element dual reciprocity method, Eng. Anal. Bound. Elem. 20 (1997) 205–211.

[3] K. Epplera, H. Harbrecht, Fast wavelet BEM for 3D electromagnetic shaping, Applied Numerical Mathematics 54 (2005) 537–554.

[4] V. Popov, H. Power, L. Škerget (Eds.), Domain Decomposition Techniques for Boundary Elements: Applications to fluid flow, WIT press, 2007.

[5] J. Ravnik, L. Škerget, M. Hriberšek, The wavelet transform for BEM computational fluid dynamics, Eng. Anal. Bound. Elem. 28 (2004) 1303–1314.

[6] J. Ravnik, L. Škerget, M. Hriberšek, 2D velocity vorticity based LES for the solution of natural convection in a differentially heated enclosure by wavelet transform based BEM and FEM, Eng. Anal. Bound. Elem. 30 (2006) 671–686.

[7] Z. Žunič, L. Škerget, M. Hriberšek, J. Ravnik, Boundary element-finite element method for velocity-vorticity formulation of Navier-Stokes equations, in: WIT transactions on modelling and simulation, Vol. 41, 2005, pp. 793–802.

[8] E. Ong, K. Lim, Three-dimensional singular boundary element method for corner and edge singularities in potential problems, Eng. Anal. Bound. Elem. 29 (2005) 175–189.

[9] X. W. Gao, T. G. Davies, 3D multi-region BEM with corners and edges, Int. J. Solids Struct. 37 (2000) 1549–1560.

[10] M. Ramšak, L. Škerget, 3D multidomain BEM for solving the Laplace equation, Eng. Anal. Bound. Elem. 31 (2007) 528–538.

[11] O. Daube, Resolution of the 2D Navier-Stokes equations in velocity-vorticity form by means of an influence matrix technique, J. Comput. Phys. 103 (1992) 402–414.

[12] C. H. Liu, Numerical solution of three-dimensional Navier Stokes equations by a velocity - vorticity method, Int. J. Numer. Meth. Fl. 35 (2001) 533–557.

[13] D. Lo, D. Young, K. Murugesan, C. Tsai, M. Gou, Velocity-vorticity formulation for 3D natural convection in an inclined cavity by DQ method, Int. J. Heat Mass Transfer 50 (2007) 479–491.

[14] L. Škerget, M. Hriberšek, Z. Žunič, Natural convection flows in complex cavities by BEM, Int. J. Num. Meth. Heat & Fluid Fl. 13 (2003) 720–735.

[15] Z. Žunič, M. Hriberšek, L. Škerget, J. Ravnik, 3-D boundary element-finite element method for velocity-vorticity formulation of the Navier-Stokes equations, Eng. Anal. Bound. Elem. 31 (2007) 259–266.

[16] L. C. Wrobel, The Boundary Element Method, John Willey & Sons, LTD, 2002.

[17] J. Ravnik, Metoda robnih elementov za hitrostno vrtinčno formulacijo simulacije velikih vrtincev, Ph.D. thesis, Univerza v Mariboru, Fakulteta za strojništvo (2006).

[18] C. C. Paige, M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, ACM Transactions on Mathematical Software 8 (1982) 43–71.

[19] T. Lundgren, P. Koumoutsakos, On the generation of vorticity at a free surface, J. Fluid Mech. 382 (1999) 351–366.

[20] U. Ghia, K. Ghia, C. Shin, High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.

[21] C. S. Chen, Numerical method for predicting three-dimensional steady compressible flow in long microchannels, J. Micromech. Microeng. 14 (2004) 1091–1100.

[22] P. Shankar, M. D. Deshpande, Fluid mechanics in the driven cavity, Annu. Re. Fluid. Mech. 32 (2000) 93–136.

[23] J. Koseff, R. Street, The lid-driven cavity flow: a synthesis of qualitative and quantitative observations, J. Fluids Eng. 106 (1984) 385–389.

[24] J. Yang, Y. C. S.C. Yang, C. Hsu, Implicit Weighted ENO Schemes for the Three-Dimensional Incompressible Navier–Stokes Equations, J. Comput. Phys. 146 (1) (1998) 464–487.

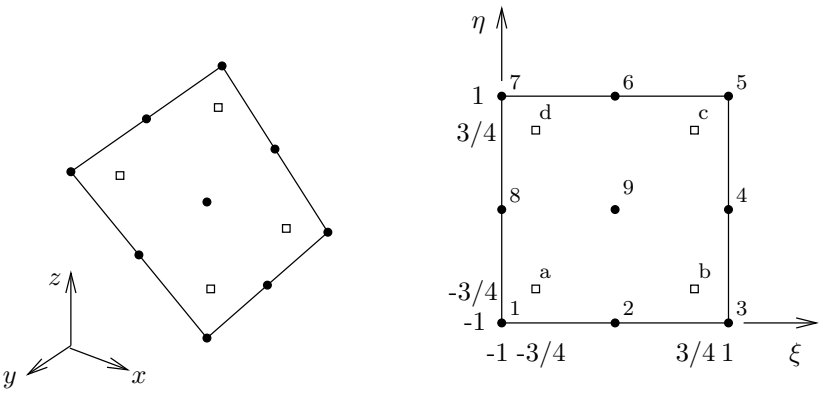

Fig. 1. A boundary element shown with nodes for discontinuous linear interpolation for flux (squares) and nodes for continuous quadratic interpolation for function (circles). Cartesian $\mathbb{R}^3$ space is shown on the left, local coordinate system on the right.

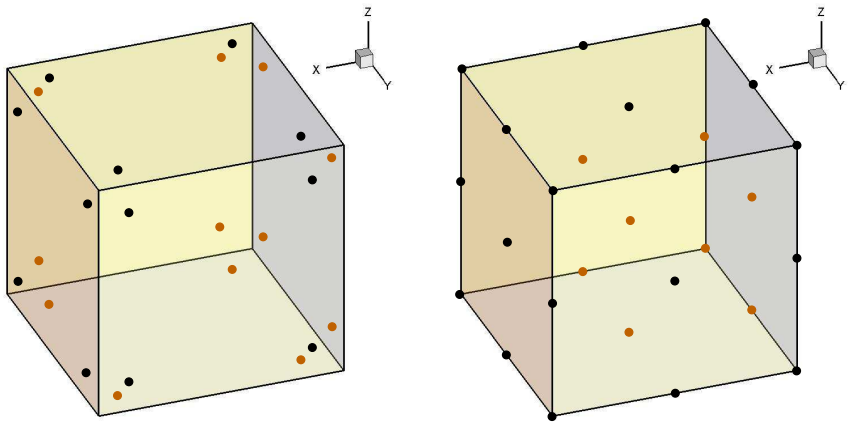

Fig. 2. A hexahedral mesh element with distribution of nodes: (left) nodes for interpolation of flux, (right) nodes for interpolation of function.

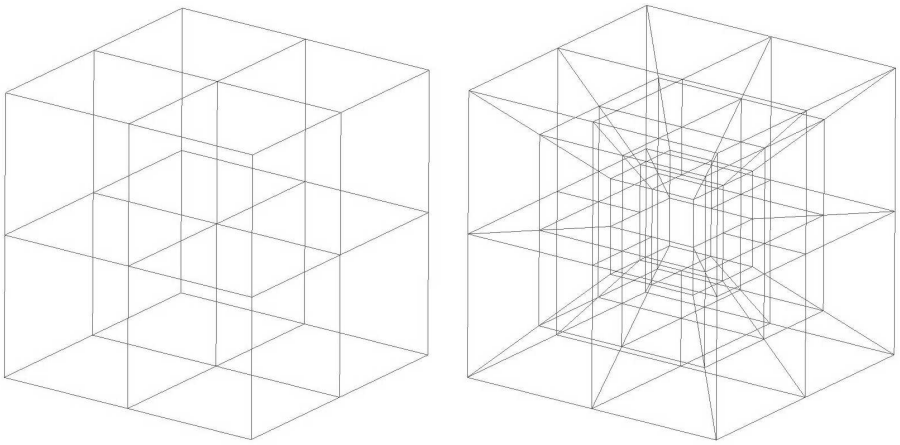

Fig. 3. Meshes used in the heat transfer test cases for the Laplace equation. Left: heat transfer in a solid cube (mesh b; $2 \times 2 \times 2$ elements); right: heat transfer in a solid cube with a cubic empty space inside (mesh j; 48 elements).

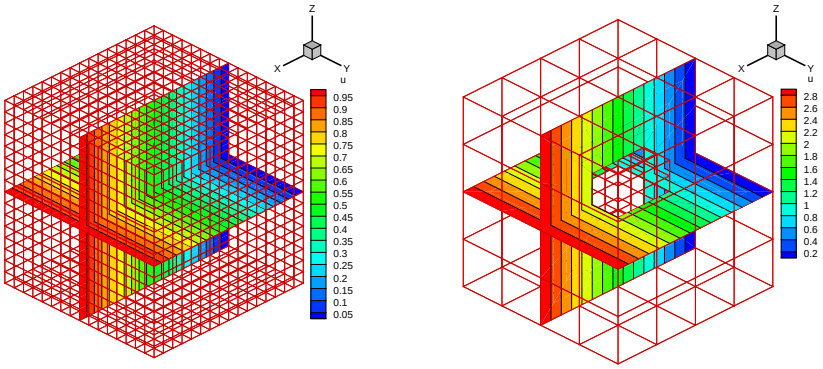

Fig. 4. Isolines of function u of the Laplace equation test cases: (left) heat transfer in a cube on a $8 \times 8 \times 8$ mesh, (right) heat transfer in a cube with an empty cube inside on a 48 element mesh.

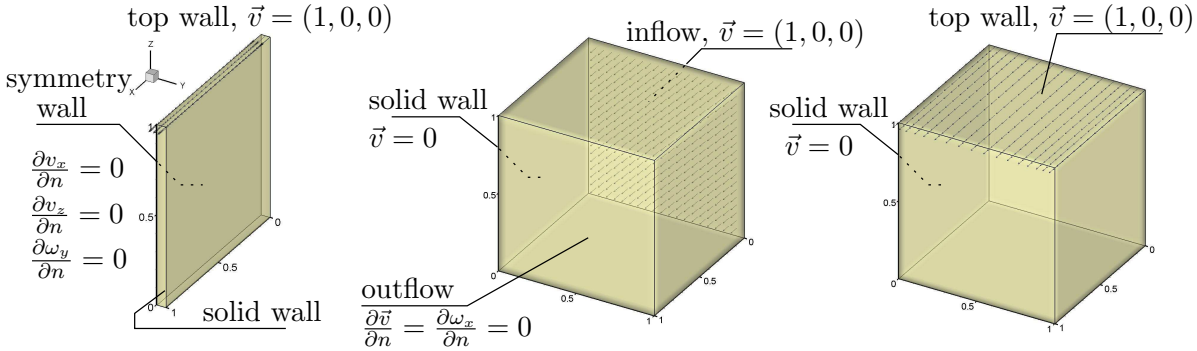

Fig. 5. Boundary conditions for the 2D lid driven cavity test case (left), 3D channel flow (middle) and 3D lid driven cavity (right).

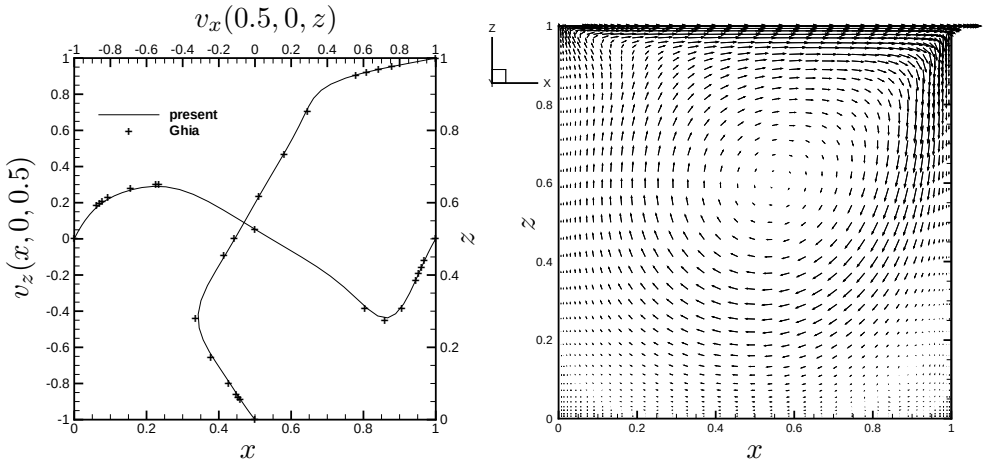

Fig. 6. Results of the 2D lid driven cavity test case at $Re = 400$. Left: velocity profiles through the middle of the domain are compared with the benchmark solution of Ghia et al. [20]; On the right velocity vectors are shown in the $y = 0$ plane.

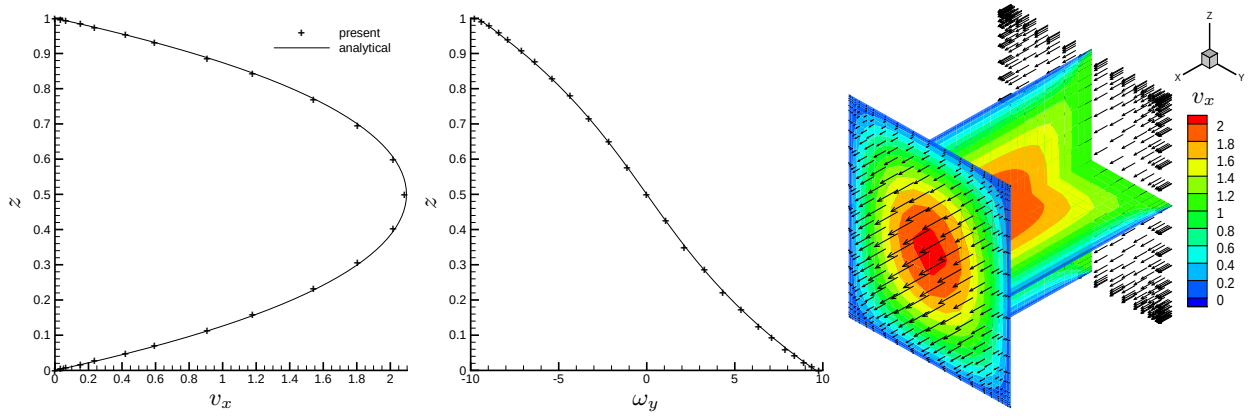

Fig. 7. Laminar flow in a 3D square channel. Velocity (left) and vorticity (middle) profiles ($y = 0.5$) compared against the analytical solution (eqn. (38)). On the right, velocity vectors on the inflow and outflow planes as well as v_x contours are shown.

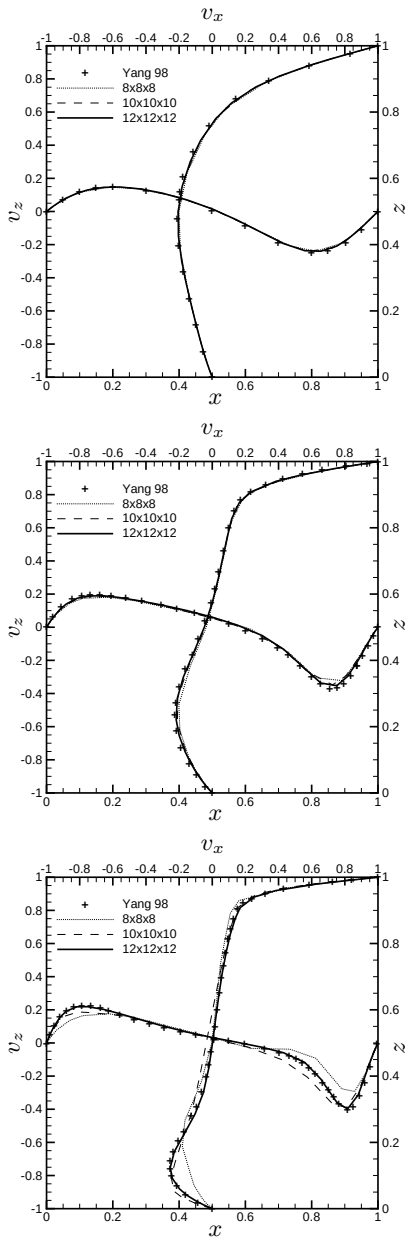

Fig. 8. Flow in a 3D lid driven cavity, Comparison of present results on different meshes with Yang et al. [24] results; $Re = 100$ (top), $Re = 400$ (middle), $Re = 1000$ (bottom).

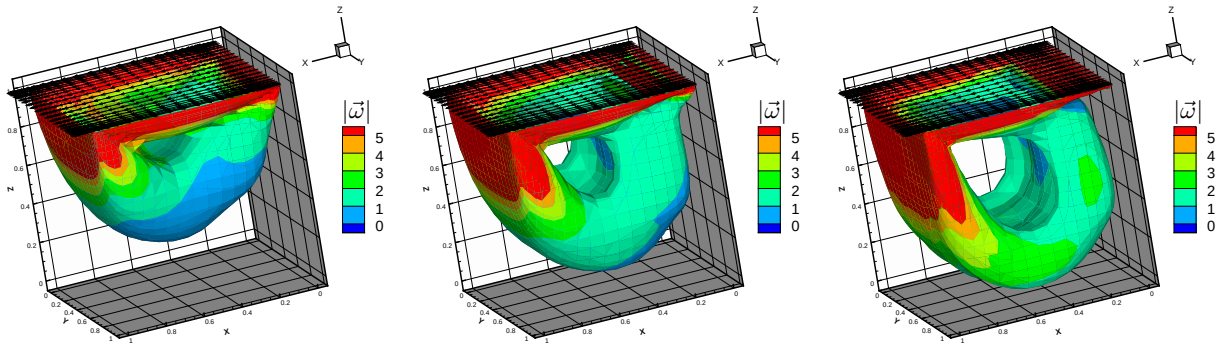

Fig. 9. Flow in a 3D lid driven cavity; iso-surfaces of $|\vec{v}| = 0.13$ with $|\vec{\omega}|$ contours: $Re = 100$ (left), $Re = 400$ (middle), $Re = 1000$ (right).

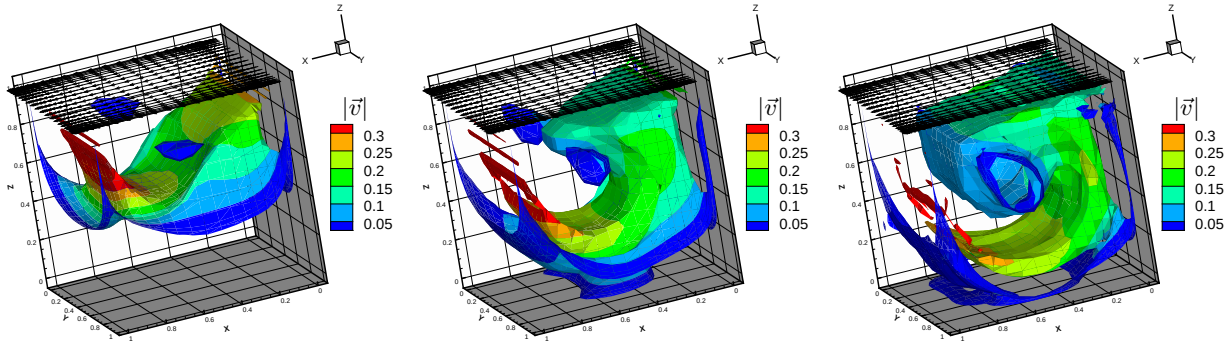

Fig. 10. Flow in a 3D lid driven cavity; iso-surfaces of $|\vec{\omega}| = 1$ with $|\vec{v}|$ contours: $Re = 100$ (left), $Re = 400$ (middle), $Re = 1000$ (right).

Table 1
Description of meshes used in test cases for the Laplace and Poisson equations; N_{elem} is the number of hexahedrons, N_{eq} is the number of equations, N_{unk} is the number of unknowns.

	mesh	domain	N_{elem}	N_{eq}	N_{unk}
a	$1 \times 1 \times 1$	$(0,0,0) \times (1,1,1)$	1	51	17
b	$2 \times 2 \times 2$	$(0,0,0) \times (1,1,1)$	8	408	155
c	$2 \times 2 \times 2$	$(0,0,0) \times (1,10^{-2},1)$	8	408	155
d	$4 \times 4 \times 4$	$(0,0,0) \times (1,1,1)$	64	3264	1271
e	$4 \times 4 \times 4$	$(0,0,0) \times (1,10^{-2},1)$	64	3264	1271
f	$8 \times 8 \times 8$	$(0,0,0) \times (1,1,1)$	512	26112	10223
g	$16 \times 16 \times 16$	$(0,0,0) \times (1,1,1)$	4096	208896	81887
h	$32 \times 32 \times 32$	$(0,0,0) \times (1,1,1)$	32768	1671168	655295
i	kvk-6	$(0,0,0) \times (3,3,3)$	6	306	106
j	kvk-48	$(0,0,0) \times (3,3,3)$	48	2448	934

Table 2
Heat transfer test cases of the Laplace equation. Error values for calculation of integrals when the source point is in u nodes and in q nodes and RMS error of function and flux. Nit is the number of iterations of the least squares solver.

mesh	u source	q source	u RMS error	q RMS error	nit
a	$2.0 \cdot 10^{-15}$	$2.0 \cdot 10^{-16}$	$1.3 \cdot 10^{-15}$	$9.4 \cdot 10^{-15}$	6
b	$2.0 \cdot 10^{-15}$	$2.0 \cdot 10^{-16}$	$4.0 \cdot 10^{-15}$	$1.6 \cdot 10^{-14}$	32
c	$1.3 \cdot 10^{-13}$	$1.2 \cdot 10^{-11}$	$1.8 \cdot 10^{-13}$	$8.1 \cdot 10^{-11}$	195
d	$2.0 \cdot 10^{-15}$	$2.1 \cdot 10^{-16}$	$2.6 \cdot 10^{-13}$	$1.5 \cdot 10^{-12}$	85
e	$1.3 \cdot 10^{-13}$	$1.3 \cdot 10^{-11}$	$1.0 \cdot 10^{-12}$	$6.3 \cdot 10^{-10}$	1033
f	$1.9 \cdot 10^{-15}$	$2.1 \cdot 10^{-16}$	$3.4 \cdot 10^{-13}$	$9.7 \cdot 10^{-12}$	225
g	$1.9 \cdot 10^{-15}$	$2.7 \cdot 10^{-16}$	$8.6 \cdot 10^{-13}$	$8.3 \cdot 10^{-11}$	436
h	$1.9 \cdot 10^{-15}$	$1.4 \cdot 10^{-12}$	$2.6 \cdot 10^{-11}$	$2.0 \cdot 10^{-10}$	985
i	$2.6 \cdot 10^{-16}$	$1.1 \cdot 10^{-11}$	$3.1 \cdot 10^{-13}$	$2.9 \cdot 10^{-12}$	112
j	$1.7 \cdot 10^{-15}$	$4.2 \cdot 10^{-14}$	$4.5 \cdot 10^{-4}$	$2.0 \cdot 10^{-3}$	287

Table 3
Function and flux RMS errors of the Poisson equation test cases: (a) $u = x^2$, (b) $u = x^3$ and (c) $u = x^4$.

	mesh	(a) RMS error		(b) RMS error		(c) RMS error	
		u	q	u	q	u	q
a	$1 \times 1 \times 1$	$4.5 \cdot 10^{-14}$	$2.8 \cdot 10^{-14}$	$1.3 \cdot 10^{-13}$	$1.3 \cdot 10^{-3}$	$2.4 \cdot 10^{0}$	$1.9 \cdot 10^{-3}$
b	$2 \times 2 \times 2$	$1.3 \cdot 10^{-14}$	$2.7 \cdot 10^{-14}$	$3.6 \cdot 10^{-14}$	$3.7 \cdot 10^{-4}$	$2.1 \cdot 10^{-2}$	$1.0 \cdot 10^{-3}$
d	$4 \times 4 \times 4$	$4.7 \cdot 10^{-13}$	$1.3 \cdot 10^{-12}$	$1.7 \cdot 10^{-14}$	$1.0 \cdot 10^{-4}$	$1.7 \cdot 10^{-3}$	$4.4 \cdot 10^{-4}$
f	$8 \times 8 \times 8$	$4.7 \cdot 10^{-13}$	$6.7 \cdot 10^{-12}$	$3.4 \cdot 10^{-14}$	$2.8 \cdot 10^{-5}$	$2.9 \cdot 10^{-4}$	$1.9 \cdot 10^{-4}$
g	$16 \times 16 \times 16$	$2.0 \cdot 10^{-12}$	$8.6 \cdot 10^{-11}$	$4.5 \cdot 10^{-14}$	$7.4 \cdot 10^{-6}$	$6.7 \cdot 10^{-5}$	$3.7 \cdot 10^{-5}$
h	$32 \times 32 \times 32$	$2.5 \cdot 10^{-11}$	$1.3 \cdot 10^{-10}$	$2.5 \cdot 10^{-13}$	$1.9 \cdot 10^{-6}$	$1.6 \cdot 10^{-5}$	$9.4 \cdot 10^{-6}$

Table 4
CPU times and solver iterations for the Poisson equation test cases: (a) $u = x^2$, (b) $u = x^3$ and (c) $u = x^4$.

	mesh	solver iterations			solver CPU time [s]		
		(a)	(b)	(c)	(a)	(b)	(c)
a	$1 \times 1 \times 1$	6	9	7	$\leqslant 0.01$	$\leqslant 0.01$	$\leqslant 0.01$
b	$2 \times 2 \times 2$	32	36	33	$\leqslant 0.01$	$\leqslant 0.01$	0.01
d	$4 \times 4 \times 4$	86	109	108	0.07	0.09	0.09
f	$8 \times 8 \times 8$	226	279	278	2.01	2.46	2.45
g	$16 \times 16 \times 16$	437	894	846	33.35	68.06	64.41
h	$32 \times 32 \times 32$	966	2661	2409	612.58	1684.59	1515.99

Table 5
Entry flow test case for diffusion advection equation. Difference between the subdomain BEM solution and the analytical solution is shown for different Peclet numbers Pe. R is the ratio between the longest and the shortest element in the mesh. Elements were concentrated towards $x = 1$.

Pe	mesh	R	$\lvert u(\frac{1}{2}) - u_{analytical} \rvert$	$\lvert q(1) - q_{analytical} \rvert$
1	$10 \times 1 \times 1$	1	$3.7 \cdot 10^{-4}$	$1.9 \cdot 10^{-3}$
1	$10 \times 1 \times 1$	7	$2.2 \cdot 10^{-4}$	$1.1 \cdot 10^{-4}$
10	$10 \times 1 \times 1$	1	$1.2 \cdot 10^{-4}$	$1.9 \cdot 10^{-2}$
10	$10 \times 1 \times 1$	7	$2.7 \cdot 10^{-4}$	$7.9 \cdot 10^{-4}$
10	$100 \times 1 \times 1$	1	$1.1 \cdot 10^{-7}$	$4.3 \cdot 10^{-4}$
20	$100 \times 1 \times 1$	1	$1.2 \cdot 10^{-6}$	$2.8 \cdot 10^{-4}$