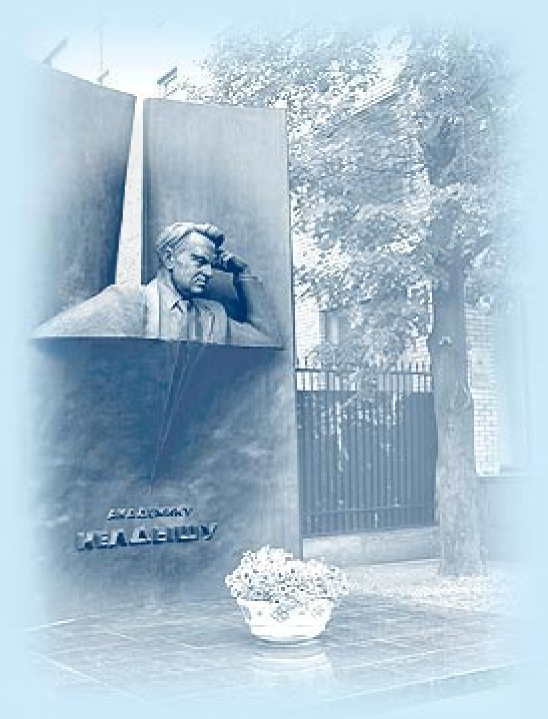

ISSN 2071-2898 (Print)
ISSN 2071-2901 (Online)

P.A. Bakhvalov

Method of local element splittings for diffusion terms discretization in edge-bases schemes

Ордена Ленина
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В.КЕЛДЫША
Российской академии наук

P. A. Bakhvalov

Method of local element splittings for diffusion terms discretization in edge-bases schemes

Moscow — 2020

П. А. Бахвалов

Метод локальных разбиений для дискретизации диффузионных членов в рёберно-ориентированных схемах

Предлагается метод локальных разбиений для аппроксимации диффузионных членов уравнений Навье – Стокса на неструктурированных сетках, состоящих из элементов различных типов. Этот метод является линейным; он схож с классическим методом Галёркина с кусочно-линейными базисными функциями и совпадает с ним на симплициальных сетках. На структурированных сетках доказывается второй порядок точности применительно к уравнению теплопроводности; на сетках общего вида доказывается только первый порядок точности, хотя численные результаты не показывают существенной потери точности по сравнению с методом Галёркина. На декартовых сетках новый метод применительно к аппроксимации лапласиана в 3D вырождается в 7-точечную схему, тогда как метод Галёркина имеет 27-точечный шаблон. Это даёт методу локальных разбиений существенное преимущество при использовании неявных схем, основанных на методе Ньютона, а именно, позволяет без потери сходимости исключить из якобиана элементы, не входящие в 7-точечный шаблон.

Ключевые слова: неструктурированная сетка, рёберно-ориентированная схема, классический метод Галёркина, метод Ритца, диффузионный член

Pavel Alexeevich Bakhvalov

Method of local element splittings for diffusion terms discretization in edge-bases schemes

Method of local element splittings is proposed for the discretization of the diffusion terms of the Navier – Stokes equations on mixed-element unstructured meshes. It is applicable when mesh functions are defined in nodes. This method is a linear method, which has much in common with the classical P1-Galerkin method. In the case of simplicial meshes, these methods coincide, but the new method yields a 7-point approximation of the 3D Laplace operator on a Cartesian mesh. On structured meshes the second order of accuracy is proved for the model heat equation. For general unstructured meshes, only the first order of accuracy is proved, however, the numerical evidence shows that there is no loss in accuracy in comparison with the classical P1-Galerkin method. The new method has an important advantage in the case of implicit time integration based on the Newton method, which implies solving linear algebraic systems with flux Jacobian. It allows to truncate flux Jacobian to a 7-point stencil for a much wider range of Reynolds and Courant numbers without loss of iterations convergence, compared to the P1-Galerkin method.

Key words: unstructured mesh, edge-based scheme, classical Galerkin method, Ritz method, diffusion term

[Added 12.05.22] Lemma 14 and Theorem 2 of this paper are in error. Revised version can be found in: Bakhvalov, Surnachev, Method of averaged element splittings for diffusion terms discretization in vertex-centered framework // JCP, Vol. 450, 110819, https://doi.org/10.1016/j.jcp.2021.110819.

1. Introduction

This paper addresses the simulation of high Reynolds number flows on mixed-element unstructured meshes. Although the numerical methods in CFD are progressing towards the very high-order methods (2-exact or higher for smooth problems), the schemes with linear reconstructions of any kind are still in practical use due to their simplicity and lower computational costs. This especially holds for trans- and supersonic flows, and, in less extent, also for subsonic flows.

The numerical methods for solving compressible Navier – Stokes equations are mainly represented by two main classes, namely vertex- and cell-centered. In cell-centered schemes, the variables are defined in mesh elements. There can be one set of variables per cell (multislope MUSCL-type methods [1, 2], WENO schemes [3–7]) or several sets per cell (discontinuous Galerkin [8–11], flux reconstruction [12], etc.). In contrast, in vertex-centered schemes (also called cell-vertex or node-based schemes) the mesh variables are associated with mesh nodes. In the vertex-centered finite-volume framework, computational domain is divided into dual cells (or control volumes), each of them containing only one mesh node. The shapes of dual cells in the case of 3D unstructured meshes are too complicated to use k-exact polynomial-based vertex-centered schemes. An alternative is the edge-based framework.

In edge-based schemes, the mesh variables are treated as nodal values of fields, and the numerical flux between two control volumes approximates with the first order the point value of the flux function in the center of the corresponding mesh edge (not the integral average over the common boundary of these control volumes). Suprisingly, Roe [13] and Barth [14] found that this approach leads to 1-exact schemes if barycentric control volumes are used. A generalization to mixed-element meshes (however, not suitable for anisotropic meshes) can be found in [15]. Edge-bases schemes are used for solving high Reynolds number problems in many in-house and commercial codes: NOISEtte [16], FUN3D [17], TAU [18], Edge [19], etc. In the last decade, there are two development directions: FC (flux correction scheme) [20–23] and schemes with quasi-one-dimensional reconstruction [24–29].

Availability of the nodal values in edge-based schemes makes it possible to use the P1-Galerkin method for discretization of diffusion terms. Usually, mass-lumped version of the method is in use. This is a traditional approach [14], simple and robust, but limited to the second order of accuracy. Below it will be considered in detail. A high-order approximation of diffusion terms is also possible (see [22] for example). However, the need for the enhanced accuracy for the viscous and heat conductivity terms in aerodynamic or aeroacoustic applications has never been shown in practice. Note also the novel HNS approach [30,31], where gradients are stored in mesh nodes in addition to the conservative variables.

A minor shortcoming of the P1-Galerkin method is its wide stencil for mixed-element meshes. In the case of simplicial meshes, the discretization of diffusion terms in a mesh node contains, besides this node, all the neighboring nodes connected to it by an edge. In the case of mixed-element meshes, it contains all the nodes of the incident mesh elements. For example, on Cartesian hexahedral meshes the stencil consists of 27 nodes, although 7 nodes are sufficient. This problem was considered in [32], where it was suggested to drop some terms in governing equations and use the mesh structure properties, and in [33], where some simplification of the numerical method is proposed. Both approaches lead to improvement in efficiency but with no guarantee of correctness.

Now the discretization of the viscous terms for edge-based schemes is revisited, and a novel finite-element scheme is proposed, namely, the method of local element splittings. This method was implemented in the NOISEtte code [16] and has been successfully used in many simulations (see, for instance, [34–36]), but has not yet been published. It coincides with the mass-lumped P1-Galerkin method on simplicial meshes and exhibits similar behavior on mixed-element meshes. On Cartesian meshes the new method applied to the 3D Laplace operator yields a 7-point approximation. However, on deformed hexahedral meshes the stencil is 27-point, the same as for the P1-Galerkin method. The excessive terms can't be generally dropped. However, the method of local element splittings has an important advantage in the case of implicit time integration based on the Newton method, which implies solving linear algebraic systems with flux Jacobian. It allows to truncate flux Jacobian to a 7-point stencil for a much wider range of Reynolds and Courant numbers without loss of iterations convergence, compared to the P1-Galerkin method.

This paper is organized as follows. In Sections 2 and 3 we describe the P1-Galerkin method and the method of local element splittings. In Section 4 we prove the convergence of the latter with the order $1 - \epsilon$ (for general meshes) and $2 - \epsilon$ (for structured meshes) for the model heat equation with a constant coefficient. In Section 5, the method of local element splittings is applied for the diffusion terms in the Navier – Stokes system. Section 6 contains verification results. Finally, in Section 7 we consider an implicit time discretization and demonstrate the possibility to drop excessive elements of the flux Jacobian.

2. P1-Galerkin method

Consider the initial problem for the heat equation

$$\frac{\partial u}{\partial t}(t, \boldsymbol{r}) = \nabla \cdot (\mu(\boldsymbol{r}) \nabla u(t, \boldsymbol{r})), \quad u(0, \boldsymbol{r}) = v_0(\boldsymbol{r}). \tag{1}$$

For the sake of simplicity, we imply the periodic boundary conditions and the continuity of v_0.

Consider a mixed-element mesh that consists of triangles and quadrilaterals in 2D or of tetrahedrons, quadrilateral pyramids, triangular prisms and hexahedrons in 3D. We do not assume the faces of 3D elements to be planar. Let N be the set of mesh nodes and E be the set of mesh elements. For $j \in N$ and $e \in E$ we write $j \in e$ if the node j is a vertex of the element e. Denote by $E(j)$ the set of mesh elements containing node j, by $\nu(e)$ the number of vertices of element e, and by $n_m(e)$, $m = 1, \dots, \nu(e)$, the vertices of element e. Let $\boldsymbol{r}_j$, $j \in N$, be the radius-vector of node j. Let $\Omega = \cup e$ be the computational domain, $\partial\Omega$ be its boundary.

We begin with the description of the mass-lumped classical (continuous) Galerkin method with piecewise-linear basis functions for the sake of comparison. Let $\phi_j(\boldsymbol{r})$, $j \in N$, be the set of basis functions such that the following properties hold:

(a) ϕ_j is a continuous function on $\bar{\Omega}$;

(b) $\sum\limits_{j \in N} \phi_j(\boldsymbol{r}) \equiv 1$ and $\sum\limits_{j \in N} \boldsymbol{r}_j \phi_j(\boldsymbol{r}) \equiv \boldsymbol{r}$ for $\boldsymbol{r} \in \Omega$;

(c) $\phi_j(\boldsymbol{r}_j) = 1$, $\phi_j(\boldsymbol{r}_k) = 0$ for $j \neq k$;

(d) $\mathrm{supp}\phi_j = \bigcup\limits_{e \in E(j)} e$.

For a simplicial mesh these properties uniquely define the set of basis functions. However, for a mixed-element mesh there are at least three approaches to construct basis functions satisfying these conditions.

A. For each type of mesh elements, introduce a canonical element:

- right triangle with nodes (0,0), (1,0), and (1,1);
- square with nodes (0,0), (1,0), (1,1), and (0,1);
- tetrahedron with nodes (0,0,0), (1,0,0), (0,1,0), and (0,0,1);
- pyramid with nodes (0,0,0), (1,0,0), (1,1,0), (0,1,0), (1/2,1/2,1);
- prism with nodes (0,0,0), (1,0,0), (0,1,0), (0,0,1), (1,0,1), (0,1,1);
- cube with nodes (0,0,0), . . . , (1,1,1).

For each canonical element, define a set of basis functions satisfying (a)–(c). For example, for the triangle they are $\psi_1(\boldsymbol{\xi}) = 1 - \xi_1 - \xi_2$, $\psi_2(\boldsymbol{\xi}) = \xi_1$, $\psi_3(\boldsymbol{\xi}) = \xi_2$ and for the cube they are $\phi_1(\boldsymbol{\xi}) = (1 - \xi_1)(1 - \xi_2)(1 - \xi_3)$, $\phi_2(\boldsymbol{\xi}) = \xi_1(1 - \xi_2)(1 - \xi_3)$, . . ., $\phi_8(\boldsymbol{\xi}) = \xi_1\xi_2\xi_3$. Then the basis functions are defined by

$$\phi_{n_m(e)}(\boldsymbol{r}) = \psi_m(\boldsymbol{\xi}(\boldsymbol{r}, e)), \quad \boldsymbol{r} \in e,$$

where $\boldsymbol{\xi}(\boldsymbol{r}, e)$ is given by identity

$$\boldsymbol{r} \equiv \sum_{m'=1}^{\nu(e)} \boldsymbol{r}_{n_{m'}(e)} \psi_{m'}(\boldsymbol{\xi}(\boldsymbol{r}, e)),$$

and $\phi_j(\boldsymbol{r}) = 0, \boldsymbol{r} \in e, j \notin e$.

B. For each edge, each face and (in 3D) each volume element, define the center as a point, each coordinate of which is the average of the vertices' coordinates. Let the value of each basis function at these points be the average of its values in the corresponding vertices. Then use these points to split all the mesh elements into simplices, and let each basis function be linear inside each simplex.

C. For each type ν of mesh elements, introduce s_ν splittings into simplices such that each face is split $s_\nu/2$ times by one diagonal and $s_\nu/2$ times by another diagonal. The details of the splittings will be described in the next section. For each splitting, the basis functions inside the element are defined uniquely by linearity. It remains to average obtained basis functions over all splittings in consideration. Note that this type of basis functions is not suitable for practice because the measure of $\operatorname{supp} \phi_j \cap \operatorname{supp} \phi_k$ can be nonzero even for nodes j and k that do not belong to a common element.

The basis functions on the uniform Cartesian mesh obtained by these approaches are shown in Fig. 1. Left subfigures correspond to "A" basis function; right ones to the "B" one, which coincides with the "C" one on Cartesian meshes.

Note that if four vertices of a mesh face do not lay on a plane, then all the variants "A", "B", and "C" prescribe the form of the mesh face differently: "A" makes it bilinear; "B" prescribes it as four triangles; "C" makes the boundary double-valued. The last can be interpreted as following: if four vertices of a quadrilateral face do not lay on a plane, then they are vertices of a tetrahedron, and the characteristic function of elements sharing this face is equal to 1/2 inside this tetrahedron.

Speaking about numerical results obtained by the Galerkin method, below we imply the use of the basis functions of type "B" for the sake of simplicity of implementation.

Since the basis functions are defined, we are ready to approximate the model equation (1). To impose the periodic boundary conditions, we extend the basis functions $\phi_j(\boldsymbol{r})$ and the solution to the whole space by periodicity. Multiply (1) by a basis function $\phi_j(\boldsymbol{r})$ and take the integral over Ω:

$$\frac{d}{dt} \int\limits_{\Omega} \phi_j(\boldsymbol{r}) u dV = \int\limits_{\partial\Omega} \phi_j(\boldsymbol{r}) \frac{\partial u}{\partial n} \mu dS - \int\limits_{\Omega} (\nabla \phi_j) \cdot (\nabla u) \mu dV. \tag{2}$$

The boundary term is zero by periodicity. The muss-lumped Galerkin method with the basis $\{\phi_j\}$ leads to the following scheme: find $u^h(t) = \{u_j(t), j \in N\}$, such that

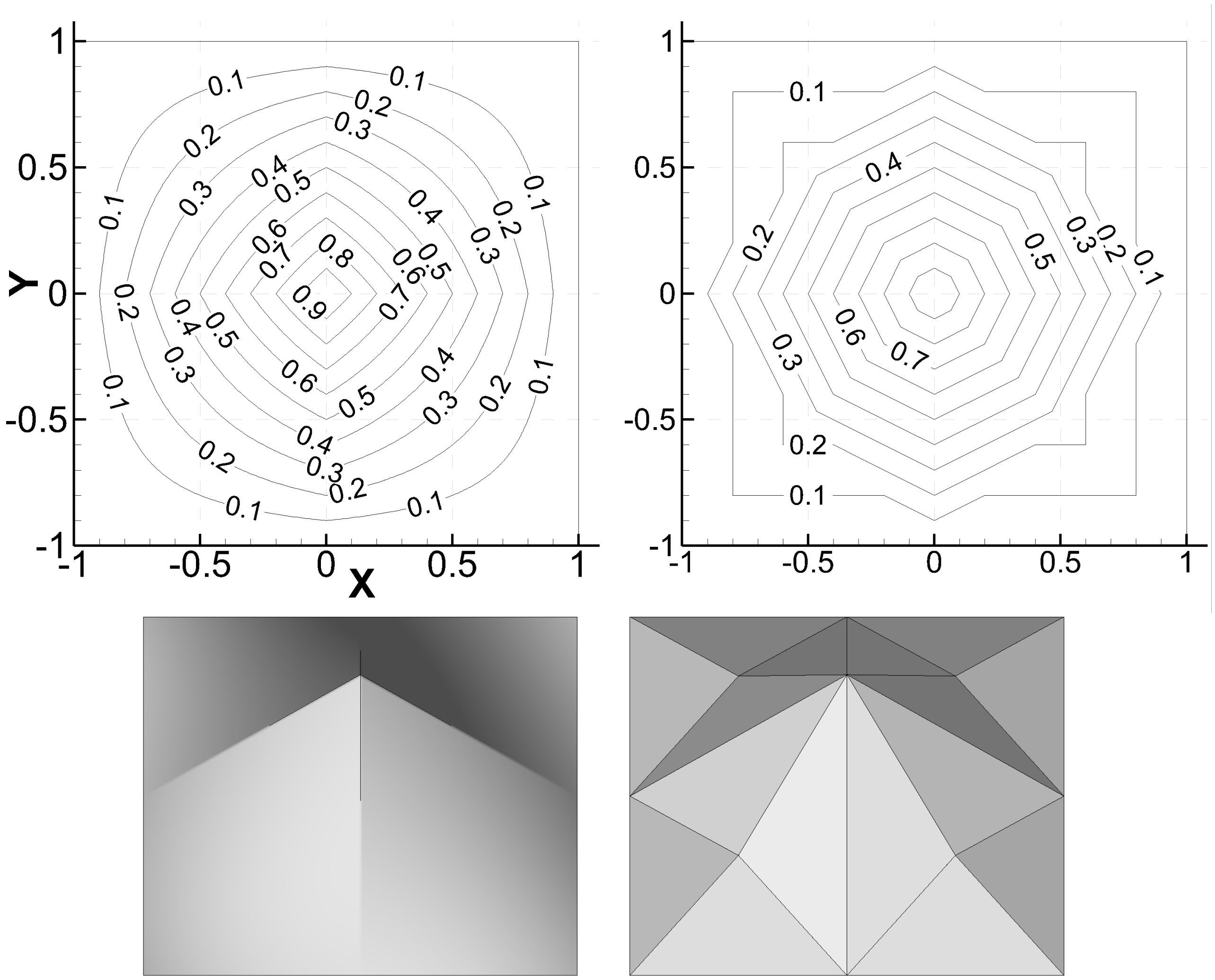

Figure 1. Basis function on the Cartesian mesh with unit steps. Left: bilinear function ("A" approach); right: piecewise-bilinear function ("B" and "C" approaches). Top: isovalues; bottom: 3D plot.

$u_j(0) = v_0(\boldsymbol{r}_j)$ and

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{k \in N} \sum_{\alpha = x,y,z} G_{jk,\alpha\alpha}[\mu] u_k, \quad V_j = \int\limits_{\Omega} \phi_j dV, \tag{3}$$

where

$$G_{jk,\alpha\beta}[\mu] = -\int\limits_{\Omega} (\nabla_\alpha \phi_j)(\nabla_\beta \phi_k) \mu dV. \tag{4}$$

3. Method of local element splittings

Consider a 2D mixed-element mesh first. Each quadrilateral element of the mesh can be split into two triangles in two ways: $e = \tau^{(1)}_{e,1} \cup \tau^{(1)}_{e,2}$ and $e = \tau^{(2)}_{e,1} \cup \tau^{(2)}_{e,2}$. Introduce two simplicial meshes generated by splitting of the original mesh such that each quadrilateral element was split differently in generating these meshes. I. e. define two simplicial meshes $(N, E^{(s)})$, where

$$E^{(s)} = \{e \in E : \nu(e) = 3\} \cup \{\tau^{(s)}_{e,q}, e \in E : \nu(e) = 4, q = 1{,}2\}.$$

The basis functions are defined uniquely on these meshes, as well as the mass-lumped Galerkin approximations (3):

$$V^{(s)}_j \frac{du_j}{dt} = \sum_{k\in N} \sum_{\alpha=x,y} G^{(s)}_{jk,\alpha\alpha}[\mu]\, u_k, \quad s = 1, 2.$$

Now take the average of them:

$$\frac{V^{(1)}_j + V^{(2)}_j}{2} \frac{du_j}{dt} = \sum_{k\in N} \sum_{\alpha=x,y} \frac{G^{(1)}_{jk,\alpha\alpha}[\mu] + G^{(2)}_{jk,\alpha\alpha}[\mu]}{2} u_k. \tag{5}$$

Since the Galerkin approximations are symmetric and positively semidefinite, so is the new approximation.

The resulting scheme differs from the Galerkin method (3) with any of the sets of basis functions considered above. Let us demonstrate the difference on the approximation of the Laplace operator on the 2D Cartesian mesh with unit steps. The coefficients $G_{jk,xx}[1]$ of the Galerkin method (for the approximation of $\partial^2/\partial x^2$), $G_{jk,yy}[1]$ (for the approximation of $\partial^2/\partial y^2$) and their sum (for the approximation of the Laplace operator) are shown in Fig. 2 for the basis functions of type “B” and in Fig. 3 for the basis functions of type “A”. These approximations of the Laplace operator are convex combinations of the “direct cross” 5-point approximation with weight $1 - w$ and the “skewed cross” 5-point approximation with weight w, where $w = 1/4$ for the basis “B” and $w = 1/3$ for the basis “A”.

In contrast, the Galerkin approximation on a Cartesian mesh of right triangles coincides with the 5-point direct cross. Thus, so does the approximation of the method of local element splittings (5) on quadrilateral Cartesian mesh.

Now proceed with the 3D case. In 3D, we should split not only mesh elements, but also their faces. In the general case, it is impossible to split a mixed-element mesh into a conformal tetrahedral mesh without adding extra nodes. Luckily, we do not need this.

Consider possible splittings of one element of a mixed-element mesh. We numerate the nodes of an element according to the “neutral” format of the Gambit mesh generator, see Fig. 4. A tetrahedron has only a trivial splitting: 0123. A pyramid has two possible splittings:

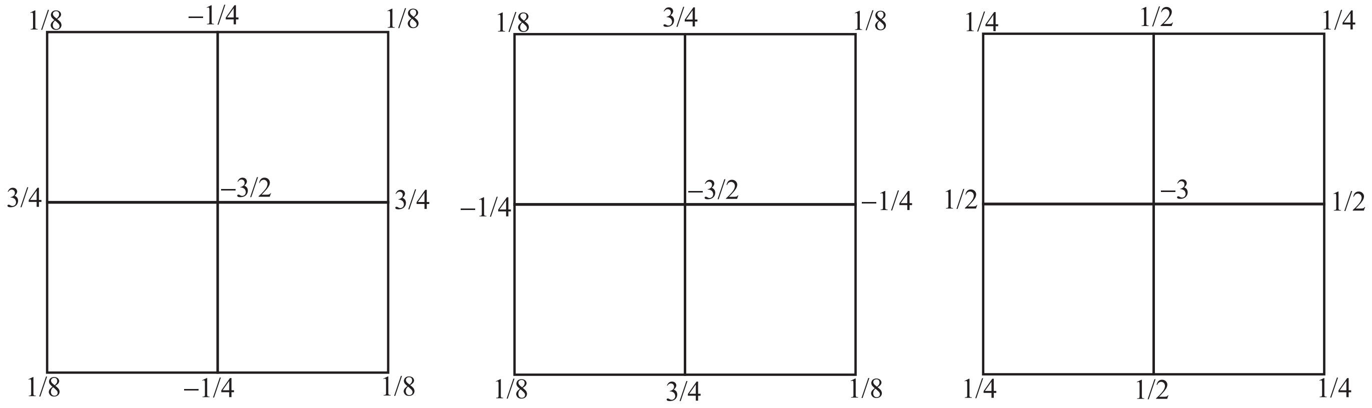

Figure 2. Coefficients for the approximation of second derivatives on the unit-step Cartesian mesh with the use of "B"-type basis. Left: for $\partial^2/\partial x^2$; center: for $\partial^2/\partial y^2$; right: for the Laplace operator

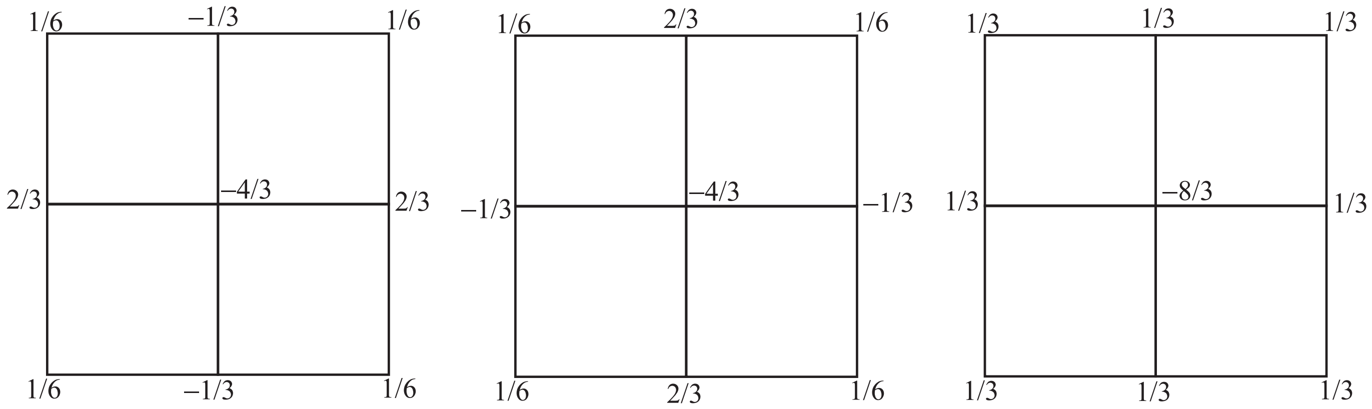

Figure 3. Coefficients for the approximation of second derivatives on the unit-step Cartesian mesh with the use of "A"-type basis. Left: for $\partial^2/\partial x^2$; center: for $\partial^2/\partial y^2$; right: for the Laplace operator

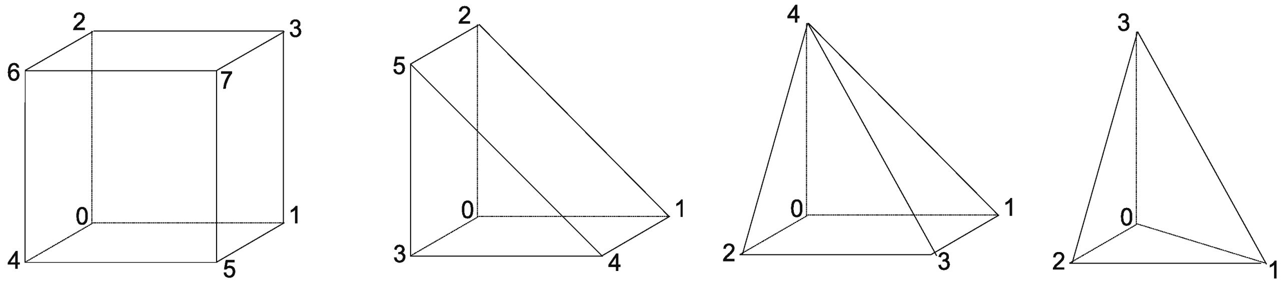

Figure 4. Numeration of mesh nodes

- 0134 + 0234;
- 1024 + 1324.

A prism can be split into 3 tetrahedrons in 6 ways:

- 0534 + 0541 + 0512;
- 1345 + 1352 + 1320;
- 2453 + 2430 + 2401;

- 3201 + 3214 + 3245;
- 4012 + 4025 + 4053;
- 5120 + 5103 + 5134.

A hexahedron can be split into 5 or 6 tetrahedrons in many ways. To construct the scheme it is enough to consider only NGD-type splittings [37], which consist of 6 tetrahedrons with a common edge:

- 1602 + 1623 + 1637 + 1675 + 1654 + 1640;
- 0713 + 0732 + 0726 + 0764 + 0745 + 0751;
- 3420 + 3401 + 3415 + 3457 + 3476 + 3462;
- 2531 + 2510 + 2504 + 2546 + 2567 + 2573.

For the sake of simplicity we duplicate these splittings for each type of mesh elements to have 12 splittings. Then one can check that each quadrilateral face of an element is split by each of the diagonals in 6 splittings.

A simplicial splitting specifies a precise form of the element, so let $e(\omega) \subset \mathbb{R}^3$ be this shape. Let $\phi_j^{(e,\omega)}(\boldsymbol{r})$, $\boldsymbol{r} \in e(\omega)$, be the standard piecewise-linear basis function associated with node j and defined on a simplicial splitting ω, where $\omega = 1, \ldots, 12$, of an element e. Then the method of local element splittings yields the following scheme:

$$\frac{du_j}{dt} = \frac{1}{V_j} \sum_{k \in N} \sum_{\alpha=1,2,3} G_{kj,\alpha\alpha}[\mu] u_k, \tag{6}$$

where

$$G_{jk,\alpha\beta}[\mu] = - \sum_{e \in E(j) \cap E(k)} \frac{1}{12} \sum_{\omega=1,\ldots,12} \int_{e(\omega)} \mu(\boldsymbol{r}) (\nabla_\alpha \phi_j^{(e,\omega)}) (\nabla_\beta \phi_k^{(e,\omega)}) dV, \tag{7}$$

$$V_j = \sum_{e \in E(j)} \frac{1}{12} \sum_{\omega=1,\ldots,12} \int_{e(\omega)} \phi_j^{(e,\omega)} dV = \sum_{e \in E(j)} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega), T \ni j} \frac{1}{4} \int_T dV \tag{8}$$

(compare this with (3), (4)). The last sum is by tetrahedra T belonging to splitting ω of element e such that $j \in T$. Note that $\nabla_\alpha \phi_j^{(e,\omega)}$ and $\nabla_\beta \phi_k^{(e,\omega)}$ are constant over each tetrahedron forming $e(\omega)$. For simplicial meshes, the method of local element splittings actually does not use splittings and thus coincides with the P1-Galerkin method.

Like in 2D, on Cartesian meshes, the formula (7) reduces the approximation of the Laplace operator to a 7-point direct cross. In comparison with the P1-Galerkin method, this slightly improves accuracy, but can be considered also as a disadvantage, because for a small deformation of a Cartesian mesh negative coefficients appear, i. e. the approximation of the Laplace operator loses discrete maximum principle (DMP). DMP is not of vital importance for high Reynolds number problems. Moreover, for unstructured simplicial meshes, the P1-Galerkin method does not generally satisfy DMP in the presence of obtuse simplices (see [38], Appendix E), which is almost inevitable in practical applications.

4. Accuracy proof

In this section we establish a near-optimal accuracy result only for structured meshes (hexahedral or prismatical). For general unstructured meshes we prove only the first order of accuracy.

As described in Section 3, for our purpose a mesh element has multiple shapes defined by its tetrahedral splittings and thus up to this point it is not clear whether a mesh is non-overlapping. To fill this gap now we complete the definition of the computational mesh. Let $\Omega = (0,1)^3$. A mesh is a pair of finite sets N and E and an injective mapping $\boldsymbol{r}_j : N \mapsto [0,1)^3$, $j \in N$. Introduce the extended set of nodes $\hat{N} = \{(j, t_x, t_y, t_z)\}$, $j \in N$, $t_x, t_y, t_z \in \{0,1\}$, and $t_i = 0$ unless i-th component of $\boldsymbol{r}_j$ is zero. For $j' = (j, t_x, t_y, t_z)$, by definition $\boldsymbol{r}_{j'} = ((\boldsymbol{r}_j)_x + t_x, (\boldsymbol{r}_j)_y + t_y, (\boldsymbol{r}_j)_z + t_z) \in [0,1]^3$. Each $e \in E$ (called a mesh element) is an ordered $\nu(e)$-tuple of elements of $\hat{N}$. Elements of a tuple e (called vertices of the element) are denoted as $n_m(e) \in \hat{N}$, $m = 1, \ldots, \nu(e)$. For each $e \in E$ and $m, m' \in \{1, \ldots, \nu(e)\}$ there holds $n_m(e) \neq n_{m'}(e)$ if $m \neq m'$. For $j \in N$ and $e \in E$ we shall write $e \in E(j)$ if e contains an element $j' = (j, t_x, t_y, t_z) \in \hat{N}$ for some t_x, t_y, t_z. For each $e \in E$ there holds $\nu(e) \in \{4,5,6,8\}$. We identify elements $j' \in \hat{N}$ with their radius-vectors $\boldsymbol{r}_{j'}$.

Each $e \in E$ has 4 (if $\nu(e) = 4$), or 5 (if $\nu(e) = 5$ or $\nu(e) = 6$), or 6 (if $\nu(e) = 8$) faces, which are 3- or 4-tuples; as shown in Fig. 4. For example, an element $e = < j_0, j_1, j_2, j_3, j_4, j_5, j_6, j_7 >$ has faces $< j_0, j_1, j_3, j_2 >$, $< j_4, j_5, j_7, j_6 >$, $< j_0, j_1, j_5, j_4 >$, $< j_2, j_3, j_7, j_6 >$, $< j_0, j_2, j_6, j_4 >$, $< j_1, j_5, j_7, j_3 >$. Faces of all elements form the set of mesh faces. If all vertices of a mesh face belong to $\partial\Omega$ and lay on the same face of $\partial\Omega$, then this face is called a boundary face. Boundary faces are considered planar, so their form is well-defined. The set of boundary faces must be a decomposition of $\partial\Omega$. We say that two tuples F_1 and F_2 are the same face if either F_1 or (if F_1 is a boundary face) its translation by a vector $(\pm1, 0, 0)$ or $(0, \pm1, 0)$ or $(0, 0, \pm1)$ is identical to F_2 up to a cyclic shift and the inversion of the order. Thus triangular faces are unordered 3-tuples and quadrilateral tuples are partially ordered 4-tuples. The mesh must be *conformal,* i. e. each mesh face is a face of exactly two mesh elements.

For each $e \in E$ there are 12 its tetrahedral splittings (some of them are identical to each other) defined above. For example, for a pyramidal element $e = < j_0, j_1, j_2, j_3, j_4 >$ the first six splittings consist of two tetrahedra:

$T_1 = < j_0, j_1, j_3, j_4 >$ and $T_2 = < j_0, j_2, j_3, j_4 >$ and the remaining six splittings consist of $T_3 = < j_1, j_0, j_2, j_4 >$ and $T_4 = < j_1, j_3, j_2, j_4 >$. A tetrahedron (as a 4-tuple) is uniquely represented by the convex hull of its vertices. Thus a tetrahedral splitting of an element e specifies its shape, denoted by $e(\omega)$, $\omega = 1, \ldots, 12$. We assume that each of these tetrahedra has nonzero volume.

Let F be a mesh face and $e_{F,1}$ and $e_{F,2}$ be elements sharing this face. Each tetrahedral splitting of $e_{F,1}$ and $e_{F,2}$ generates a triangular splitting of F and thus prescribes its form, consisting of one or two planar triangles. We assume that the mesh is *non-overlapping*, i. e. for each F and for each splittings $\omega_1, \omega_2 = 1, \ldots, 12$ of $e_{F,1}$ and $e_{F,2}$ generating the same triangular splitting on F the outer unit normal vectors of $\partial e_{F,1}(\omega_1)$ and $\partial e_{F,2}(\omega_2)$ on the face F are oppositely directed.

A tetrahedron T satisfies Křížek angle conditions [39] with a constant $\bar{\gamma} < \pi$ if the following conditions hold:

- for each vertices $j, k, l \in T$, $j \neq k \neq l \neq j$, the angle between the vectors $\boldsymbol{r}_k - \boldsymbol{r}_j$ and $\boldsymbol{r}_l - \boldsymbol{r}_j$ does not exceed $\bar{\gamma}$;
- the angle between any two faces of T does not exceed $\bar{\gamma}$.

Theorem 1. *Consider the initial value problem for the heat equation* (1) *with $\mu \equiv 1$, in the unit cube $\Omega = (0{,}1)^3$ with the periodic boundary conditions and periodic initial data $v_0 \in C^4(\bar{\Omega})$. Let (N, E) be a conformal non-overlapping mesh and h be the maximal edge length. Let $\bar{\gamma} < \pi$. Suppose that for each element $e \in E$ and for each of its splittings introduced in the previous section the following holds:*

- *tetrahedra forming this splitting have intersection of zero measure;*
- *each tetrahedron satisfies Křížek angle conditions with $\bar{\gamma}$.*

Let Π be the pointwise mapping, i. e. $\Pi u(t) = \{u(t, \boldsymbol{r}_j),\ j \in N\}$. Let $u^h(t) = \{u_j(t), j \in N\}$ be the solution of (6), (7), (8) *with the initial data $u^h(0) = \Pi v_0$. Then for each $\epsilon > 0$ there holds*

$$\|u^h(t) - \Pi u(t)\|_L = \left(\sum_{j \in N} V_j (u_j(t) - u(t, \boldsymbol{r}_j))^2 \right)^{1/2} \leqslant C h^{1-\epsilon}(1 + t), \qquad (9)$$

where C depends on v_0, $\bar{\gamma}$, and ϵ only.

Theorem 2. *Let the conditions of Theorem 1 hold. Suppose that the mesh is either a structured hexahedral mesh, or a structured prismatic mesh (i. e. a deformation of a translationally-invariant prismatic mesh), or consists of tetrahedra only. Then for each $\epsilon > 0$ there holds*

$$\|u^h(t) - \Pi u(t)\|_L \leqslant C h^{2-\epsilon}(1 + t), \qquad (10)$$

where C depends on v_0, $\bar{\gamma}$, and ϵ only.

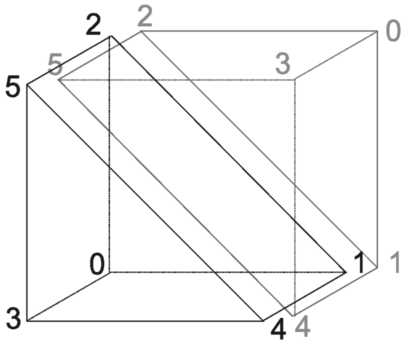

Figure 5. Vertices reordering for two prisms

The rest of this section is the proof of Theorems 1 and 2.

Lemma 3. *Let the conditions of Theorem 2 hold. Then for each* $e \in E$ *there exists a permutation* $\omega_j(e)$, $j = 1, \ldots, 12$, *such that for each quadrilateral face* F *sharing elements* e_1 *and* e_2 *and for each* $j = 1, \ldots, 12$, *the splittings* $\omega_j(e_1)$ *and* $\omega_j(e_2)$ *of these elements generate the same triangular splitting of* F.

If the mesh is tetrahedral, then there is no quadrilateral faces and this the statement is trivial. Otherwise assume without loss that the mesh is Cartesian and translationally-invariant. Note that for each type of elements the set of the splittings preserves when reordering the vertices of the element.

Consider a cubic mesh first. First assume that the order of the vertices preserves under translation. Then for each splitting of a cube the opposite faces have the same triangular splittings by construction. Hence $\omega_j(e) \equiv j$ satisfies the statement of the lemma. Another order of vertices in a cube leads to a permutation of splittings.

For a prismatic mesh, combine adjacent prisms in order to generate a structured hexahedral mesh. Reorder the vertices of prisms in order to make the face 1-2-5-4 sharing two prisms forming one hexahedron, see Fig. 5. Then $\omega_j(e) \equiv j$ will again satisfy the statement of the lemma. □

For each mesh face F, define a one-to-one relation between the splittings of the elements $e_{F,1}$ and $e_{F,2}$ sharing this face. For each quadrangular face, each pair of related element splittings must generate the same triangular splitting of the face. In the proof of Theorem 1, this is the only condition to impose. In the proof of Theorem 2, we will use splittings $\omega_j(e_{F,1})$ and $\omega_j(e_{F,2})$, $j = 1, \ldots, 12$, given by Lemma 3.

We call $f = \{f_{e,\omega}, e \in E, \omega = 1, \ldots, 12\}$ an *admissible function of order* k if the following conditions hold:

1) $f_{e,\omega} \in W_2^k(e(\omega))$;
2) for a face F sharing elements e_1 and e_2, for each related element splittings ω_1, ω_2, and for each $l = 0, \ldots, k-1$ there holds

$$\left.\frac{\partial^l f_{e_1,\omega_1}}{\partial n^l}\right|_F = \left.\frac{\partial^l f_{e_2,\omega_2}}{\partial n^l}\right|_F . \tag{11}$$

These conditions are naturally applied at the periodic boundaries also. Let $\mathcal{H}_k$ be the set of admissible functions of order k. In the proof of Theorem 2 the smoothness condition (11) splits the set of pairs (e,ω) into 12 groups such that each of conditions (11) involves only elements from the same group. Thus

$$\mathcal{H}_k = \left(W^k_{2,per}(\mathbb{R}^3)\right)^{12} . \tag{12}$$

In the setting of Theorem 1 such simple representation is generally impossible.

For each $f, g \in \mathcal{H}_0$, define the scalar product

$$(f,g) = \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \int\limits_{e(\omega)} f_{e,\omega} g_{e,\omega}\, dV, \tag{13}$$

which induces the norm $\|f\| = (f,f)^{1/2}$. For a vector-function $\boldsymbol{f}$ we will use the norm $\|\boldsymbol{f}\|^2 = \sum_j \|f_j\|^2$, where the sum is over the components of $\boldsymbol{f}$.

Let S be the set of $f \in \mathcal{H}_1$ such that there exists $f_j \in \mathbb{R}$, $j \in N$, such that $f_{e,\omega}$ is the linear interpolant based on the values f_j, $j \in T$, on each tetrahedron T belonging to $e(\omega)$. In addition to (13), for each $f, g \in S$, define the scalar product

$$[f,g] = \sum_{j\in N} f_j g_j V_j,$$

where V_j is given by (8). It induces the "lumped" norm $\|f\|_L = [f,f]^{1/2}$.

Lemma 4. *The norms $\|f\|_L = [f,f]^{1/2}$ and $\|f\| = (f,f)^{1/2}$ are equivalent on S.*

Let T be a tetraherdon and $f(\boldsymbol{r}) = f_0 + \boldsymbol{L}\cdot\boldsymbol{r}$. Obviously,

$$\frac{1}{4}\sum_{j\in T} f(\boldsymbol{r}_j)^2 \geqslant \frac{1}{|T|}\int\limits_T (f(\boldsymbol{r}))^2 dV \geqslant \frac{\sigma}{4}\sum_{j\in T} f(\boldsymbol{r}_j)^2,$$

where $\sigma > 0$ does not depend on T and f. Then

$$\|f\|^2 = \sum_{e\in E}\frac{1}{12}\sum_{\omega=1,\ldots,12}\int\limits_{e(\omega)} f^2_{e,\omega} dV \leqslant \sum_{e\in E}\frac{1}{12}\sum_{\omega=1,\ldots,12}\sum_{T\in e(\omega)}\frac{|T|}{4}\sum_{j\in T} f(\boldsymbol{r}_j)^2 =$$

$$= \sum_{j\in N}\left[\sum_{e\in E(j)}\frac{1}{12}\sum_{\omega=1,\ldots,12}\sum_{T\in e(\omega), T\ni j}\frac{|T|}{4}\right] f(\boldsymbol{r}_j)^2 = \|f\|^2_L.$$

In the last equality, we used the fact that the expression in brackets coincides with V_j, see (8). Similarly,

$$\|f\|^2 \geqslant \sigma \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T\in e(\omega)} \frac{|T|}{4} \sum_{j\in T} f(\boldsymbol{r}_j)^2 = \sigma \|f\|_L^2.$$

Thus, the equivalence of the norms is proved. □

Lemma 5. *For each* $\mathbf{r} \in \Omega$ *there exists exactly 12 pairs* (e, ω) *such that* $\mathbf{r} \in \operatorname{Int} e(\omega)$ *unless* $\boldsymbol{r} \in \partial e(\omega)$ *for some* (e, ω).

Let $\xi_{e(\omega)}$ be the indicator function of $e(\omega)$ and

$$X(\boldsymbol{r}) = \sum_{e\in E} \sum_{\omega=1,\ldots,12} \xi_{e(\omega)}(\boldsymbol{r}).$$

By construction $X(\boldsymbol{r})$ is piecewise-constant, and constant values are separated by triangles belonging to $\partial e(\omega)$, $e \in E$, $\omega = 1, \ldots, 12$. We claim that all these values coincide. Indeed, let τ be one of these triangles. It belongs to a face F shared by elements $e_{F,1}$ and $e_{F,2}$. If all vertices of F lay on a plane (for instance, if F is a triangular face), then $\tau \subset \partial e_{F,1}(\omega)$ and $\tau \subset \partial e_{F,2}(\omega)$ for each $\omega = 1, \ldots, 12$. Otherwise $\tau \subset \partial e_{F,1}(\omega)$ for $\omega = \omega_1, \ldots, \omega_6$ and $\tau \subset \partial e_{F,2}(\omega)$ for $\omega = \omega_1', \ldots, \omega_6'$. Since the mesh is non-overlapping (i.e. the outer normals to $e_{F,1}(\omega)$ and $e_{F,2}(\omega)$ have opposite directions), $X(\boldsymbol{r})$ has no discontinuity at this triangle. Thus $X(\boldsymbol{r})$ is constant a. e. in Ω. Now write

$$\int\limits_{\Omega} X(\boldsymbol{r}) dV = \sum_{e\in E} \sum_{\omega=1,\ldots,12} \int\limits_{e(\omega)} dV = \frac{1}{3} \sum_{e\in E} \sum_{\omega=1,\ldots,12} \int\limits_{\partial e(\omega)} \boldsymbol{r}\cdot\boldsymbol{n}\, d\Sigma,$$

where $\boldsymbol{n}$ is the outer normal to $e(\omega)$. Since the mesh is non-overlapping the internal faces cancel each other, thus

$$\int\limits_{\Omega} X(\boldsymbol{r}) dV = \frac{1}{3} \sum_{F\in\{Boundary faces\}} \sum_{\omega=1,\ldots,12} \int_F \boldsymbol{r}\cdot\boldsymbol{n}\, d\Sigma = \frac{12}{3} \int\limits_{\partial\Omega} \boldsymbol{r}\cdot\boldsymbol{n}\, d\Sigma = 12,$$

where $\boldsymbol{n}$ is the outer unit normal to Ω. Thus $X(\boldsymbol{r}) = 12$ a. e. in Ω. From this, the statement of the lemma is obvious. □

Corollary 6. *For each* $f \in L_p(\Omega)$ *there holds*

$$\int\limits_{\Omega} |f|^p dV = \frac{1}{12} \sum_{e\in E} \sum_{\omega=1,\ldots,12} \sum_{T\subset e(\omega)} \int\limits_{T} |f|^p dV, \tag{14}$$

where the last sum is over tetrahedra belonging to splitting (e, ω).

For $f \in \mathcal{H}_1$ denote $\nabla_j f = \{\nabla_j f_{e,\omega}, e \in E, \omega = 1, \dots, 12\}$. Obviously, for $f \in \mathcal{H}_k$ there holds $\nabla_j f \in \mathcal{H}_{k-1}$. Now we prove the integration-by-parts formula.

Lemma 7. *For each $f, g \in \mathcal{H}_1$ there holds*

$$(f, \nabla g) + (\nabla f, g) = 0. \tag{15}$$

Indeed, for each element by the Gauss theorem

$$\int\limits_{e(\omega)} f \nabla g dV + \int\limits_{e(\omega)} g \nabla f dV = \int\limits_{\partial e(\omega)} fg\, \boldsymbol{n} d\Sigma.$$

Taking the average over splittings and the sum over elements we get

$$(f, \nabla g) + (\nabla f, g) = R(f, g),$$

where

$$R(f, g) = \sum_{e \in E} \frac{1}{12} \sum_{\omega=1,\dots,12} \int\limits_{\partial e(\omega)} fg\, \boldsymbol{n} d\Sigma.$$

Disjoin the integrals over $\partial e(\omega)$ into the sum of integrals over faces and regroup the resulting sum to faces:

$$R(f, g) = \sum_{F \in \{Faces\}} \frac{1}{12} \sum_{\omega=1,\dots,12} \left(\int\limits_F f_{e(F,1),\omega} g_{e(F,1),\omega} \boldsymbol{n} d\Sigma - \int\limits_F f_{e(F,2),\omega} g_{e(F,2),\omega} \boldsymbol{n} d\Sigma \right).$$

Here $e(F,1)$ and $e(F,2)$ are the elements sharing face F, and $\boldsymbol{n}$ is the outer unit normal to $e(F,1)$. The surfaces of face F are prescribed by the splittings ω of the elements $e(F,1)$ and $e(F,2)$, correspondingly. Then reorder the sum by splittings to connect terms associated with related splittings:

$$R(f, g) = \sum_{F \in \{Faces\}} \frac{1}{12} \sum_{j=1,\dots,12} R_{F,j}(f, g),$$

$$R_{F,j}(f, g) = \int\limits_F f_{e(F,1),\omega_j(e(F,1))} g_{e(F,1),\omega_j(e(F,1))}\, \boldsymbol{n} d\Sigma -$$

$$- \int\limits_F f_{e(F,2),\omega_j(e(F,2))} g_{e(F,2),\omega_j(e(F,2))}\, \boldsymbol{n} d\Sigma.$$

The form of face F in the first integral is prescribed by splitting $\omega_j(e(F,1))$ of element $e(F,1)$, and the one in the second integral is prescribed by splitting $\omega_j(e(F,2))$ of element $e(F,2)$. By definition, these forms coincide and so do the traces of integrands. Thus $R_{F,j}(f, g) \equiv 0$. This proves that $R(f, g) \equiv 0$ and thus (15). □

Now we move to the interpolation issues. Let $\|\cdot\|_{k,p,T}$ and $|\cdot|_{k,p,T}$ be the usual norm and seminorm in the Sobolev space $W_p^k(T)$. By $\Pi_T f$ we denote the Lagrange interpolant on tetrahedron T of function f based on its values at vertices.

Theorem 8 (Acosta [40]). *Let T be a tetrahedron satisfying the Křížek angle conditions with $\bar\gamma$. Then for each $2 < p \leqslant \infty$, and each function $f \in W_p^2(T)$ the following estimate for the interpolation error holds:*

$$\|f - \Pi_T f\|_{1,p,T} \leqslant c_{\bar\gamma,p}\, h\, |f|_{2,p,T}, \tag{16}$$

Note that this result does not hold for $p = 2$, see [41,42].

Lemma 9. *The following estimates hold:*

$$\|f - \Pi f\| \leqslant ch^2\, |f|_{2,2}, \quad f \in W^2_{2,per}(\Omega); \tag{17}$$

$$\|\nabla(f - \Pi f)\| \leqslant c_{\bar\gamma,p}\, h\, |f|_{2,p}, \quad f \in W^2_{p,per}(\Omega), \quad p > 2. \tag{18}$$

The inequality

$$\|f - \Pi_T f\|_{0,p,T} \leqslant c_p\, h^2\, |f|_{2,p,T} \tag{19}$$

holds for $p > 3/2$ (see [42]) without any limitations on the tetrahedron shape. To prove (17) we use (19) for $p = 2$, then by (14) and the definition of the norm $\|\cdot\|$ we get

$$\|f - \Pi f\| \leqslant c_2\, h^2 \Bigg(\frac{1}{12} \sum_{e \in E} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega)} |f|^2_{2,2,T} \Bigg)^{1/2} = c_2\, h^2 |f|_{2,2}.$$

For $f \in W_p^2(\Omega)$ by the Hölder inequality we have

$$\|\nabla(f - \Pi f)\|^2 = \sum_{e \in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega)} \int_T |\nabla(f - \Pi_T f)|^2 dV \leqslant$$

$$\leqslant \sum_{e \in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega)} \Bigg(\int_T |\nabla(f - \Pi_T f)|^p dV \Bigg)^{2/p} V_T^{1-2/p}.$$

Applying (16), the Hölder inequality for sums, and (14) we obtain

$$\|\nabla(f - \Pi f)\|^2 \leqslant \sum_{e \in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega)} c^2_{\bar\gamma,p} h^2 |f|^2_{2,p,T} V_T^{1-2/p} \leqslant$$

$$\leqslant h^2 c^2_{\bar\gamma,p} \Bigg(\frac{1}{12} \sum_{e \in E} \sum_{\omega=1,\ldots,12} \sum_{T \subset e(\omega)} |f|^p_{2,p,T} \Bigg)^{2/p} = h^2 c^2_{\bar\gamma,p} |f|^2_{2,p}.$$

Thus we have (18). □

For each $f, g \in \mathcal{H}_1$, define the "energy" product

$$a(f,g) = (\nabla f, \nabla g) = \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\dots,12} \int\limits_{e(\omega)} (\nabla f_{e,\omega}) \cdot (\nabla g_{e,\omega})\, dV. \tag{20}$$

Obviously, $a(f,f) \geqslant 0$. Let $P : \mathcal{H}_1 \to S$ be the Ritz projection, i. e. the linear operator taking each $f \in \mathcal{H}_1$ to $Pf \in S$ giving the minimum of the functional

$$a(Pf - f, Pf - f) + (Pf - f, 1)^2.$$

By the variational principle, for each $v \in S$ there holds

$$a(Pf - f, v) + (Pf - f,1)\,(v,1) = 0.$$

Since $a(v,1) \equiv 0$, taking $v = 1$ we get

$$(Pf - f,1) = 0, \tag{21}$$

and thus Pf gives the minimum of $\|\nabla(Pf - f)\|$. Taking v such that $(v,1) = 0$, we get $a(Pf - f, v) = 0$. The latter also holds for $v = 1$. Therefore, for each $v \in S$ and $f \in \mathcal{H}_1$ there holds

$$a(Pf - f, v) = 0. \tag{22}$$

Lemma 10. *For each $w \in S$ there holds*

$$\|w\|^2 \leqslant c(\|\nabla w\|^2 + (w,1)^2),$$

where c does not depend on the mesh and w.

Let ϕ_j be the basis functions of the P1-Galerkin method of type "C" (see Section 2), i. e.

$$\phi_j(\boldsymbol{r}) = \sum_{e\in E(j)} \frac{1}{12} \sum_{\omega=1,\dots,12} \phi_j^{(e,\omega)}(\boldsymbol{r}).$$

This yields $V_j = \int \phi_j dV$, where V_j is given by (8). Let G be the interpolation operator taking each $u \in S$ to

$$(Gu)(\boldsymbol{r}) = \sum_{j\in N} u_j \phi_j(\boldsymbol{r}).$$

By Lemma 5 for each $\boldsymbol{r} \in \Omega$ there exists not more than 96 triplets (j, e, ω) such that $\boldsymbol{r} \in \operatorname{Int}\operatorname{supp} \phi_j^{(e,\omega)}$. Thus there exists a finite decomposition $\{W_k\}$ of Ω, i. e.

$W_k \cap W_l = 0$ for $k \neq l$ and $\cup W_l = \Omega$, such that for each k the number of triplets (j, e, ω) such that $\phi_j^{(e,\omega)}(\boldsymbol{r}) \not\equiv 0$ on W_k is not greater than 96. Hence,

$$\begin{gathered}\|\nabla(Gu)\|^2 = \frac{1}{144} \int\limits_\Omega \left(\sum_{e \in E} \sum_{\omega=1,\ldots,12} \sum_{j \in e} u_j \nabla \phi_j^{(e,\omega)} \right)^2 dV \leqslant \\ \leqslant \frac{96}{144} \sum_{e \in E} \sum_{\omega=1,\ldots,12} \int\limits_\Omega \left(\sum_{j \in e} u_j \nabla \phi_j^{(e,\omega)} \right)^2 dV = 8\|\nabla u\|^2.\end{gathered} \tag{23}$$

The function Gw is single-valued, so using the Fourier representation we get

$$\|Gw\|^2 \leqslant \frac{1}{4\pi^2} \|\nabla(Gw)\|^2 + (Gw, 1)^2. \tag{24}$$

The first term on the right-hand side is estimated using (23), and for the second term we have

$$(Gw, 1) = \int\limits_\Omega \sum_{j \in N} w_j \phi_j(\boldsymbol{r}) dV = \sum_{j \in N} V_j w_j = (w, 1).$$

Similarly to Lemma 4 it can be proved that $\|w\| \leqslant \tilde{c}\|Gw\|$, where $\tilde{c}$ does not depend on the mesh and w. This completes the proof of the lemma. □

Lemma 11. *For each $p > 2$ and $f \in W^2_{p,per}(\Omega)$ there holds*

$$\|Pf - f\| \leqslant \hat{C}_{\bar{\gamma},p} h |f|_{2,p}. \tag{25}$$

Put $w = Pf - \Pi f \in S$. By Lemma 10 we have

$$\|Pf - \Pi f\|^2 \leqslant c\|\nabla(Pf - \Pi f)\|^2 + c(Pf - \Pi f, 1)^2. \tag{26}$$

For the first term on the right-hand side of (26), write

$$\|\nabla(Pf - \Pi f)\| \leqslant \|\nabla(Pf - f)\| + \|\nabla(\Pi f - f)\| \leqslant 2\|\nabla(\Pi f - f)\|.$$

The last inequality is by the definition of Pf. For the second term on the right-hand side of (26), using (21) and the Cauchy–Schwarz inequality we get

$$(Pf - \Pi f, 1)^2 = (f - \Pi f, 1)^2 \leqslant \|f - \Pi f\|^2.$$

Thus

$$\|Pf - \Pi f\| \lesssim \|\nabla(\Pi f - f)\| + \|f - \Pi f\|.$$

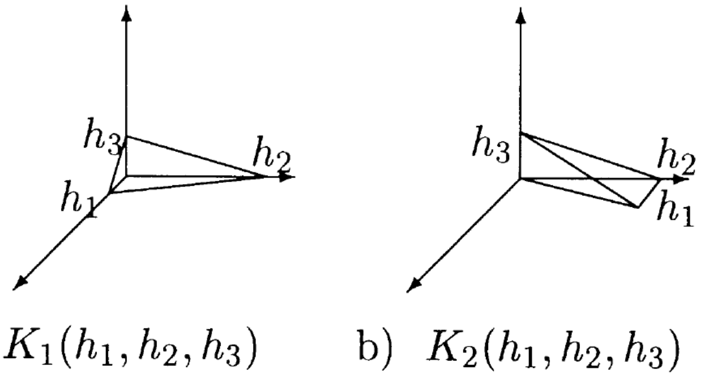

Figure 6. Reference tetrahedra (the figure is copied from [40])

Now write

$$\|Pf - f\| \leqslant \|Pf - \Pi f\| + \|\Pi f - f\| \lesssim \|\nabla(\Pi f - f)\| + \|\Pi f - f\|.$$

The inequality to prove is by (17) and (18), taking into account that $h < 1$. □

Lemma 12. *Let T be a tetrahedron satisfying Křížek angle conditions with the constant $\bar\gamma < \pi$ and $f \in W_2^1(T)$. Then for each $0 < \epsilon < 1$ there holds*

$$|f|_{0,p(\epsilon),T} \leqslant c(\bar\gamma) V_T^{-\epsilon/3} \left(|f|_{0,2,T} + (|f|_{0,2,T})^{1-\epsilon} (h|f|_{1,2,T})^{\epsilon} \right), \tag{27}$$

where $p(\epsilon) = ((1-\epsilon)/2 + \epsilon/6)^{-1} > 2$, and $c(\bar\gamma)$ does not depend on T, f, and ϵ.

By Lemma 2.2 in [40] there exists a linear transformation $F(x) = Bx + b$ such that $F(K_1) = T$ or $F(K_2) = T$ and $\|B\|$, $\|B^{-1}\| \leqslant C(\bar\gamma)$, where tetrahedra K_1 and K_2 are as in Fig. 6. Thus without loss we can consider only tetrahedra of these types.

Let T_0 be a tetrahedron of type K_1 or K_2 with $h_1 = h_2 = h_3 = 1$ and A be the diagonal matrix with the elements h_1, h_2, h_3. Let $f_0(\boldsymbol{r}) \in W_2^1(T_0)$ be such that $f_0(\boldsymbol{r}) = f(A\boldsymbol{r})$ holds for almost all $\boldsymbol{r} \in T_0$. By the Hölder inequality there holds

$$|f_0|_{0,p(\epsilon),T_0} \leqslant (|f_0|_{0,2,T_0})^{1-\epsilon} (|f_0|_{0,6,T_0})^{\epsilon}.$$

By the Sobolev imbedding theorem [43] we get $|f_0|_{0,6,T_0} \leqslant c(|f_0|_{0,2,T_0} + |f_0|_{1,2,T_0})$, thus

$$|f_0|_{0,p(\epsilon),T_0} \leqslant c \left(|f_0|_{0,2,T_0} + (|f_0|_{0,2,T_0})^{1-\epsilon} (|f_0|_{1,2,T_0})^{\epsilon} \right). \tag{28}$$

By the linear coordinate transformation we get

$$|f|_{0,p(\epsilon),T} = (6V_T)^{1/p(\epsilon)} |f_0|_{0,p(\epsilon),T_0}, \quad |f_0|_{0,2,T_0} = (6V_T)^{-1/2} |f|_{0,2,T},$$

$$|f_0|_{1,2,T_0} \leqslant (6V_T)^{-1/2} \|A\| \, |f|_{1,2,T} \leqslant (6V_T)^{-1/2} h |f|_{1,2,T}$$

(here we used $V_{T_0} = 1/6$). Combining this with (28) and noting that $1/p(\epsilon) - 1/2 = -\epsilon/3$, we get (27). □

Lemma 13. *For each* $0 < \epsilon < 1$ *and each* $f \in \mathcal{H}_3$ *there holds*

$$\|\nabla(f - \Pi f)\| \leqslant c_{\bar{\gamma},\epsilon}\, h \left(\|\triangle f\| + \|\triangle f\|^{1-\epsilon}\|h\nabla\triangle f\|^{\epsilon}\right). \tag{29}$$

By (16) for each tetrahedron T there holds

$$\|\nabla(f - \Pi f)\|_{p(\epsilon),T} \leqslant c_{\bar{\gamma},p(\epsilon)} h |f|_{2,p(\epsilon),T}.$$

Applying (27) to the Hessian of f, we get

$$|f|_{2,p(\epsilon),T} \leqslant c(\bar{\gamma}) V_T^{-\epsilon/3} \left(|f|_{2,2,T} + \left(|f|_{2,2,T}\right)^{1-\epsilon}\left(h|f|_{3,2,T}\right)^{\epsilon}\right),$$

where V_T is the volume of T. Besides,

$$\|\nabla(f - \Pi f)\|_{2,T} \leqslant \|\nabla(f - \Pi f)\|_{p(\epsilon),T} V_T^{1/2 - 1/p(\epsilon)}.$$

Combining these inequalities yields

$$\|\nabla(f - \Pi f)\|_{2,T} \leqslant c_{\bar{\gamma},\epsilon} h \left(|f|_{2,2,T} + \left(|f|_{2,2,T}\right)^{1-\epsilon}\left(h|f|_{3,2,T}\right)^{\epsilon}\right).$$

Hence,

$$\|\nabla(f - \Pi f)\|^2 = \sum_{e\in E}\frac{1}{12}\sum_{\omega=1,\dots,12}\sum_{T\subset e(\omega)} \|\nabla(f - \Pi f)\|_{2,T}^2 \leqslant$$

$$\leqslant 2c_{\bar{\gamma},\epsilon}^2 h^2 \sum_{e\in E}\frac{1}{12}\sum_{\omega=1,\dots,12}\sum_{T\subset e(\omega)} |f|_{2,2,T}^2 +$$

$$+2c_{\bar{\gamma},\epsilon}^2 h^2 \sum_{e\in E}\frac{1}{12}\sum_{\omega=1,\dots,12}\sum_{T\subset e(\omega)} \left(|f|_{2,2,T}\right)^{2(1-\epsilon)}\left(h|f|_{3,2,T}\right)^{2\epsilon}.$$

Applying the Hölder inequality to the last term we get

$$\|\nabla(f - \Pi f)\|^2 \leqslant 2c_{\bar{\gamma},\epsilon}^2\, h^2 \left(\|\nabla^2 f\|^2 + \|\nabla^2 f\|^{2(1-\epsilon)}\, \|h\nabla^3 f\|^{2\epsilon}\right). \tag{30}$$

Using (15) we get

$$\|\nabla^2 f\|^2 = \sum_{j,k\in\{x,y,z\}} \|\nabla^j\nabla^k f\|^2 = \sum_{j,k\in\{x,y,z\}} (\nabla^j\nabla^k f, \nabla^j\nabla^k f) =$$

$$= -\sum_{j,k\in\{x,y,z\}} (\nabla^k f, \nabla^j\nabla^j\nabla^k f) = \sum_{j,k\in\{x,y,z\}} (\nabla^k\nabla^k f, \nabla^j\nabla^j f) = (\triangle f, \triangle f) = \|\triangle f\|^2$$

and

$$\|\nabla^3 f\|^2 = \sum_{j,k,l\in\{x,y,z\}} \|\nabla^j\nabla^k\nabla^l f\|^2 = \sum_{j,k,l\in\{x,y,z\}} (\nabla^j\nabla^k\nabla^l f, \nabla^j\nabla^k\nabla^l f) =$$
$$= \sum_{j,k,l\in\{x,y,z\}} (\nabla^l\nabla^k\nabla^k f, \nabla^l\nabla^j\nabla^j f) = \|\nabla\triangle f\|^2.$$

Substituting these equalities into (30) we get (29). □

Next we use the Nitsche trick to obtain an estimate for the approximation error.

Lemma 14. *Let the conditions of Theorem 2 hold. Then for each* $\varepsilon > 0$ *and each* $f \in W^2_{\infty,per}(\bar\Omega)$ *there holds*

$$\|Pf - f\| \leqslant C_{\bar\gamma,\epsilon} h^{2-\epsilon} |f|_{2,\infty}. \tag{31}$$

Denote $e = Pf - f$. Let $z \in \mathcal{H}_1$ be a solution (unique up to an additive constant) of the auxiliary problem $-\triangle z = e$ in Ω with the periodic boundary conditions, i. e.

$$(\nabla z, \nabla v) = (e, v), \quad v \in \mathcal{H}_1. \tag{32}$$

According to (12), $\mathcal{H}_1$ consists of 12 functions having no mutual dependency. Thus (32) is equivalent to 12 classical problems $-\triangle z_j = e_j$, $e_j \in W^1_{2,per}(\Omega)$. By standard regularity theory we get $z_j \in W^3_{2,per}(\Omega)$ and thus $z \in \mathcal{H}_3$. Note that establishing $z \in \mathcal{H}_3$ is the only place where we use the additional assumption of Theorem 2.

Then we have

$$\begin{aligned}(e, e) = (\nabla e, \nabla z) = a(e, z) = a(e, z - Pz) \leqslant \\ \leqslant (a(e, e))^{1/2}(a(z - Pz, z - Pz))^{1/2}.\end{aligned} \tag{33}$$

The first identity in this chain is by (32), the second one is by (20), the third one is by (22), and finally we use the Schwarz inequality. By the definition of the Ritz projection, we have

$$a(z - Pz, z - Pz) \leqslant a(z - \Pi z, z - \Pi z) = \|\nabla(f - \Pi f)\|^2.$$

Using (29) to estimate the latter and (18) to estimate $a(e, e)$, from (33) we get

$$(e, e) \leqslant c_{\bar\gamma,p(\epsilon)} c_{\bar\gamma,\epsilon} h^2\ |f|_{2,p(\epsilon)} (\|\triangle z\| + \|\triangle z\|^{1-\epsilon} \|h\nabla\triangle z\|^{\epsilon}) \leqslant$$
$$\leqslant \tilde{C}_{\bar\gamma,\epsilon} h^2\ |f|_{2,p(\epsilon)} (\|e\| + \|e\|^{1-\epsilon} \|h\nabla e\|^{\epsilon}).$$

In the last inequality we use $\Delta z = -e$. Then, using (18), $|f|_{2,p(\epsilon)} \leqslant |f|_{2,\infty}$, and dividing by $\|e\|$ we get

$$\|e\| \leqslant \hat{C}_{\bar{\gamma},\epsilon} h^2 |f|_{2,\infty} \left(1 + \|e\|^{-\epsilon} (h|f|_{2,\infty})^{\epsilon}\right). \tag{34}$$

We claim that (31) holds with

$$C_{\bar{\gamma},\epsilon} = \hat{C}_{\bar{\gamma},\epsilon}(1 + \hat{C}_{\bar{\gamma},\epsilon}^{-\epsilon}). \tag{35}$$

Indeed, if $\|e\| \leqslant \hat{C}_{\bar{\gamma},\epsilon} h^2 |f|_{2,\infty}$ then (31), (35) is obvious. Otherwise (31), (35) is by substitution of the inequality $\|e\| > \hat{C}_{\bar{\gamma},\epsilon} h^2 |f|_{2,\infty}$ into the last term of (34). □

Lemma 15. *Let $f, g \in S$. For the bilinear form $\Delta(f,g) = [f,g] - (f,g)$ there holds*

$$|\Delta(f,g)| \leqslant Ch^2(|\nabla f|, |\nabla g|) \leqslant a(f,f) + C^2 h^4 a(g,g). \tag{36}$$

By definition, there holds

$$\Delta(f,g) = \sum_j f_j g_j V_j - \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \int_{e(\omega)} f_{e,\omega} g_{e,\omega} dV.$$

Using the expression (8) for V_j and regrouping the sum, we get

$$\Delta(f,g) = \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T\subset e(\omega)} \int_T \left[\left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) f_j g_j \right) - \right.$$
$$\left. - \left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) f_j \right) \left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) g_j \right) \right] dV.$$

Here $T \subset e(\omega)$ are tetrahedra used in the splitting ω. For $f = const$, the integrals in this sum are zero, and the same holds for $g = const$. Thus, we can replace f_j by $f_j - f_T$, where f_T is the value in the barycenter of T. This leads to

$$\Delta(f,g) = \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\ldots,12} \sum_{T\subset e(\omega)} \int_T \left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) (\nabla f)_T \cdot (\boldsymbol{r}_j - \boldsymbol{r}_T)(\nabla g)_T \cdot (\boldsymbol{r}_j - \boldsymbol{r}_T) \right) -$$
$$- \left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) (\nabla f)_T \cdot (\boldsymbol{r}_j - \boldsymbol{r}_T) \right) \left(\sum_{j\in e} \phi_j^{(e,\omega)}(\boldsymbol{r}) (\nabla g)_T \cdot (\boldsymbol{r}_j - \boldsymbol{r}_T) \right) dV$$

and

$$|\Delta(f,g)| \leqslant Ch^2 \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\dots,12} \int\limits_{e(\omega)} |\nabla f|\,|\nabla g| dV \leqslant$$

$$\leqslant \sum_{e\in E} \frac{1}{12} \sum_{\omega=1,\dots,12} \int\limits_{e(\omega)} \left(|\nabla f|^2 + (Ch^2)^2|\nabla g|^2\right) dV.$$

Thus we get (36). □

Proof of Theorems 1 and 2. Consider the heat equation (1) with $\mu(\boldsymbol{r}) \equiv 1$. Let $u \in C^1([0,\infty); C^2_{per}(\Omega))$ be the solution of (1); let $v \in \mathcal{H}_1$. Integrating (1) with the weight $v_{e,\omega}$, we get

$$\int\limits_{e(\omega)} \frac{\partial u}{\partial t} v_{e,\omega} dV = \int\limits_{e(\omega)} v_{e,\omega} \Delta u dV.$$

Taking the average over splittings and the sum over elements, we get

$$\left(\frac{\partial u}{\partial t}, v\right) + (v, \Delta u) = 0.$$

By (15) we have $(v, \Delta u) = -(\nabla u, \nabla v) = -a(u,v)$. Thus the weak formulation of (1) is: find a function $u \in C^1([0,\infty); \mathcal{H}_1)$ such that $u|_{t=0} = v_0(\boldsymbol{r})$ and

$$\left(\frac{\partial u}{\partial t}, v\right) + a(u,v) = 0 \quad \forall v \in \mathcal{H}_1. \tag{37}$$

Since the solution of this problem is unique, it coincides with the solution of (1).

Now we consider the following numerical scheme: find $u^h \in S$ such that $u^h|_{t=0} = \Pi v_0$ and

$$\left[\frac{\partial u^h}{\partial t}, v\right] + a(u^h, v) = 0 \quad \forall v \in S. \tag{38}$$

Clearly, the solution of (38) is unique and its nodal values satisfy (6). Thus (38) is a weak formulation of (6).

Substracting (38) from (37) with $v = e \equiv Pu - u^h$, we get

$$\left(\frac{\partial u}{\partial t}, e\right) - \left[\frac{\partial u^h}{\partial t}, e\right] + a(u - u^h, e) = 0.$$

By (22),

$$\left[\frac{\partial e}{\partial t}, e\right] + \left(\frac{\partial u}{\partial t} - P\frac{\partial u}{\partial t}, e\right) + a(e,e) + \left(P\frac{\partial u}{\partial t}, e\right) - \left[P\frac{\partial u}{\partial t}, e\right] = 0.$$

Using the identity

$$\left[\frac{\partial e}{\partial t}, e\right] = \frac{1}{2}\frac{\partial}{\partial t}[e,e] = \frac{1}{2}\frac{\partial}{\partial t}\|e\|_L^2 = \|e\|_L\frac{\partial\|e\|_L}{\partial t},$$

we get

$$\|e\|_L\frac{\partial\|e\|_L}{\partial t} \leqslant \left\|\frac{\partial u}{\partial t} - P\frac{\partial u}{\partial t}\right\| \|e\| + \left|\Delta\left(e, P\frac{\partial u}{\partial t}\right)\right| - a(e,e).$$

Applying (36), we obtain

$$\|e\|_L\frac{\partial\|e\|_L}{\partial t} \leqslant \left\|\frac{\partial u}{\partial t} - P\frac{\partial u}{\partial t}\right\| \|e\| + Ch^4 a\left(P\frac{\partial u}{\partial t}, P\frac{\partial u}{\partial t}\right).$$

The first term on the right-hand side is estimated using (26) (in the proof of Theorem 1) or (31) (in the proof of Theorem 2). For the second term by (22) we have

$$a\left(P\frac{\partial u}{\partial t}, P\frac{\partial u}{\partial t}\right) = a\left(\frac{\partial u}{\partial t}, \frac{\partial u}{\partial t}\right) - a\left(P\frac{\partial u}{\partial t} - \frac{\partial u}{\partial t}, P\frac{\partial u}{\partial t} - \frac{\partial u}{\partial t}\right). \tag{39}$$

The first term on the right-hand side of (39) does not depend on the mesh, and the second one is bounded by (18) since

$$a(Pf - f, Pf - f) \leqslant a(\Pi f - f, \Pi f - f) = \|\nabla(\Pi f - f)\|^2.$$

Thus, replacing h^4 by $h^{2A-2\epsilon}$, where $A = 1$ for Theorem 1 and $A = 2$ for Theorem 2,

$$\|e\|_L\frac{\partial\|e\|_L}{\partial t} \leqslant C\left(h^{A-\epsilon}\|e\| + h^{2(A-\epsilon)}\right) \leqslant C\left(h^{A-\epsilon}\|e\|_L + h^{2(A-\epsilon)}\right).$$

Clearly, there holds $\|e(t)\|_L \leqslant h^{A-\epsilon}y(tC)$, where y is the solution of

$$y\frac{dy}{d\tau} = y + 1; \quad y(0) = \frac{\|e(0)\|}{h^{A-\epsilon}}.$$

The solution of this equation is $y - \ln(y+1) = \tau + const$, so $y(\tau) \leqslant 2\tau + c$, where constant is defined by the initial condions, so it is well-defined for $0 < \tau < \infty$. Recall that $e \equiv Pu - u^h$, then $e|_{t=0} = Pv_0 - \Pi v_0$ and

$$\|e(0)\|_L \leqslant \frac{1}{\sigma}\|e(0)\| \leqslant \frac{1}{\sigma}\left(\|Pv_0 - v_0\| + \|\Pi v_0 - v_0\|\right) \leqslant Ch^{A-\epsilon}.$$

Here σ is the norms equivalence constant given by Lemma 4. In the last inequality we have used (26) (in the proof of Theorem 1) or (31) (in the proof of Theorem 2) to estimate the first term and (17) for the second term. This proves that $\|e(t)\|_L \leqslant C(t+1)h^{A-\epsilon}$.

By the triangle inequality

$$\|u^h(t) - \Pi u(t)\|_L \leqslant \|e(t)\|_L + \|Pu(t) - u(t)\|_L + \|\Pi u(t) - u(t)\|_L \leqslant$$
$$\leqslant C(t+1)h^{A-\epsilon} + Ch^{A-\epsilon} + Ch^2 \leqslant C(t+3)h^{A-\epsilon}.$$

This concludes the proof of Theorems 1 and 2.

5. Implementation for Navier – Stokes solvers

Now we move to the approximation of the Navier – Stokes system

$$\frac{\partial Q}{\partial t} + \nabla \cdot \mathcal{F}(Q) = \nabla \cdot \mathcal{F}^V(Q, \nabla Q), \tag{40}$$

where $Q = (\rho, \rho\boldsymbol{u}, E)^T$, $E = \rho\boldsymbol{u}^2/2 + p/(\gamma - 1)$, and

$$\mathcal{F} = \begin{pmatrix} \rho\boldsymbol{u} \\ \rho\boldsymbol{u}\boldsymbol{u} + p\boldsymbol{I} \\ (E+p)\boldsymbol{u} \end{pmatrix}, \quad \mathcal{F}^V(Q, \nabla Q) = \begin{pmatrix} 0 \\ \boldsymbol{\tau} \\ \boldsymbol{\tau}\cdot\boldsymbol{u} - \boldsymbol{q} \end{pmatrix}. \tag{41}$$

The tension tensor $\boldsymbol{\tau} = \{\tau_{\alpha\beta}\}$ and the heat flux $\boldsymbol{q} = \{q_\alpha\}$ are defined as

$$\tau_{\alpha\beta} = \mu\left(\nabla_\alpha u_\beta + \nabla_\beta u_\alpha - \frac{2}{3}\delta_{\alpha\beta}\nabla_\xi u_\xi\right), \quad q_\alpha = -\frac{\gamma\mu}{\mathrm{Pr}}\nabla_\alpha\left(\frac{p}{(\gamma-1)\rho}\right). \tag{42}$$

The diffusion terms in Navier – Stokes equations can be represented in a general form as $D_{\alpha\beta}u = \nabla_\alpha(\varkappa\nabla_\beta u)$, where $\varkappa$ can be dynamic viscosity, or dynamic viscosity times velocity, or heat conductivity, and u can be velocity or temperature. Turbulence modeling in RANS framework can add one or more diffusive terms of this form, depending on the specific model in use.

Let Q_j be the set of conservative variables in node j, and $Q = \{Q_j, j \in N\}$. The general form of a semidiscrete scheme is

$$\frac{dQ_j}{dt} + \frac{1}{V_j}\Phi_j(Q) = \frac{1}{V_j}\Phi_j^V(Q), \tag{43}$$

where Φ_j and Φ_j^V are some approximations of convective and diffusive terms of (40). As in most of the vertex-centered schemes, we use a finite-volume discretization of convective fluxes and a finite-element discretization of viscosity and heat terms.

Following finite-volume approach, we write the convective terms of the numerical scheme in the flux form

$$\Phi_j(Q) = \sum_{k \in \hat{N}_1(j)} F_{jk}(Q).$$

Here $\hat{N}_1(j)$ is the set of nodes connected to j by edge or by element, depending on the construction of control volumes, and $F_{jk}(Q)$ are convective fluxes between the control volumes associated with the nodes j and k.

Unless specifically stated, we use "direct" control volumes [15], see also [44] for a similar approach. Then the control volumes corresponding to edge-connected nodes have intersections of nonzero 2D measure, and the ones of the diagonal-connected nodes have not. Then $\hat{N}_1(j)$ is the set of nodes connected to j by edge. In the case of semitransparent control volumes [15], which result from the application of local element splittings to convective terms, $\hat{N}_1(j)$ is the set of elementwise-connected nodes. The semitransparent control volumes preserve 1-exactness of edge-based schemes on arbitrary mixed-element meshes, but they are not suitable for highly-anisotropic meshes used for the simulation of high Reynolds number flows. A specific scheme for convective fluxes, F_{jk}, is of no importance in the framework of this paper; we will use a suitable one for each test.

The approximation of viscous and heat fluxes $\Phi^V_j(Q)$ is given by

$$\Phi^V_j(Q) = \begin{pmatrix} 0 \\ [\nabla\cdot\boldsymbol{\tau}]_j \\ [\nabla\cdot(\boldsymbol{\tau}\cdot\boldsymbol{u})]_j - [\nabla\cdot\boldsymbol{q}]_j \end{pmatrix}, \tag{44}$$

$$([\nabla\cdot\boldsymbol{\tau}]_j)_\alpha = \sum_{k\in N} G_{jk,\beta\alpha}[\mu](u_k)_\beta + \sum_{k\in N} G_{jk,\beta\beta}[\mu](u_k)_\alpha - \frac{2}{3}\sum_{k\in N} G_{jk,\alpha\gamma}[\mu](u_k)_\gamma,$$

$$[\nabla\cdot(\boldsymbol{\tau}\cdot\boldsymbol{u})]_j = \sum_{k\in N} G_{jk,\beta\alpha}[\mu u_\alpha](u_k)_\beta + \sum_{k\in N} G_{jk,\beta\beta}[\mu u_\alpha](u_k)_\alpha - \frac{2}{3}\sum_{k\in N} G_{jk,\alpha\gamma}[\mu u_\alpha](u_k)_\gamma,$$

$$[\nabla\cdot\boldsymbol{q}]_j = -\frac{\gamma}{\mathrm{Pr}}\sum_{k\in N} G_{jk,\beta\beta}[\mu]\left(\frac{p_k}{(\gamma-1)\rho_k}\right)$$

(the sum over repeating indices is assumed). The coefficients $G_{jk,\alpha\beta}[\varkappa]$ and V_j are given by (4) and $V_j = \int \phi_j dV$ for the P1-Galerkin method and by (7) and (8) for the method of local element splittings. Note that in practice the definition of V_j does not matter; one can use its value given by Galerkin method, or by (8), or calculate the volume of the dual cell used for the approximation of convective fluxes.

There are several terms of the form $D_{\alpha\beta}u = \nabla_\alpha(\varkappa\nabla_\beta)u$, where $\varkappa$ are dynamic viscosity, or dynamic viscosity times velocity, or heat conductivity. Clearly, $\varkappa$ are not constant in space: for dynamic viscosity times velocity this holds unless some trivial cases, and for dynamic viscosity this holds for RANS models, which replace dynamic viscosity by its sum with a turbulent viscosity. Then the coefficients $G_{jk,\alpha\beta}[\varkappa]$ given by (4) or (7) should be approximated. The linear interpolation between nodes seems to be correct:

$$G_{jk,\alpha\beta}[\varkappa] = -\sum_{e\in E(j)\cap E(k)} \int_e (\nabla_\alpha \phi_j)(\nabla_\beta \phi_k) \sum_{l\in e} \phi_l \varkappa_l dV \qquad (45)$$

for the Galerkin method and

$$G_{jk,\alpha\beta}[\varkappa] = -\sum_{e\in E(j)\cap E(k)} \frac{1}{12} \sum_{\omega=1,\ldots,12} \int_{e(\omega)} (\nabla_\alpha \phi_j^{(e,\omega)})(\nabla_\beta \phi_k^{(e,\omega)}) \sum_{l\in e} \phi_l^{(e,\omega)} \varkappa dV \qquad (46)$$

for the method of local element splittings.

If $\varkappa$ were constant in time, we could compute $G_{jk,\alpha\beta}[\varkappa]$ before the time integration. But this is not the case. In (45) and (46), the coefficients $G_{jk,\alpha\beta}[\varkappa]$ are linear functions of $\{\varkappa_m\}$, where m runs over all nodes of all elements contacting the nodes j and k, i. e.

$$G_{jk,\alpha\beta}[\varkappa] = \sum_m G_{jk,\alpha\beta,m} \frac{dG_{jk,\alpha\beta}[\varkappa]}{d\varkappa_m}. \qquad (47)$$

The coefficients $dG_{jk,\alpha\beta}[\varkappa]/d\varkappa_m$ can be calculated before the time integration and stored in memory. We use another expression, namely,

$$G_{jk,\alpha\beta}[\varkappa] = -\sum_{e\in E(j)\cap E(k)} G_{jk,\alpha\beta,e}\varkappa_e, \qquad (48)$$

where $\varkappa_e$ is the average over all nodes of element e and

$$G_{jk,\alpha\beta,e} = \frac{1}{12} \sum_{\omega=1,\ldots,12} \int_{e(\omega)} (\nabla_\alpha \phi_j^{(e,\omega)})(\nabla_\beta \phi_k^{(e,\omega)}) dV$$

(note that integrand is constant). This refers to a piecewise-constant approximation of $\varkappa$. In theory, this may be less accurate, but we have never seen a significant difference.

There are two possible implementations of these methods for the calculation of diffusion terms: edge-based and elementwise. The conventional approach for simplicial meshes is an elementwise implementation, i. e. computing these terms in a loop over mesh elements. For each element e, the values $|e|\nabla_\alpha \phi_j$ and $|e|\nabla_\beta \phi_k$ are constant inside the element e, the values of them being equal to linear (2D) or quadratic (3D) functions of nodal coordinates. So this procedure is very efficient and there is no need to store additional data of geometric nature.

The situation differs on a mixed-element mesh. We have not implemented an elementwise approach for the P1-Galerkin method, but for the method of local element

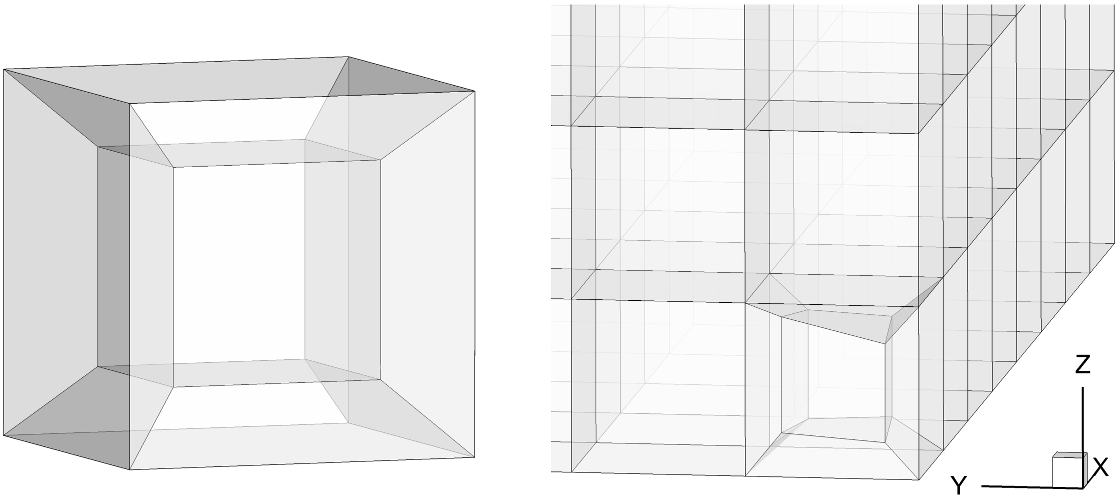

Figure 7. Left: mesh of 7 hexahedrons with no possible tetrahedral splittings without extra nodes. Right: mesh used for the heat equation

splittings, the edge-based implementation with the storage of $G_{jk,\alpha\beta,e}$ is preferable. Note that $G_{jk,\alpha\beta,e} = G_{kj,\beta\alpha,e}$, so for each element with ν vertices, we need to store $9\nu(\nu-1)/2$ coefficients. In the case of double-precision floating-point arithmetic, it consumes at most 2KB of memory per element, which is admissible. The use of (47) with the storage of $G_{jk,\alpha\beta,m}$ consumes significantly more memory.

6. Verification

To verify the method of local element splittings, we consider a problem for the heat equation followed by a low Reynolds number problem and a convection-dominated problem.

Consider the initial problem for the heat equation (1) in $\Omega = (0,1)^3$ with $\mu \equiv 1$, $v_0(\boldsymbol{r}) = \sin(2\pi x + 1)\sin(2\pi y + 2)\sin(2\pi z + 2.5)$, and the periodic boundary conditions. The exact solution of this problem is $u(t, \boldsymbol{r}) = v_0(\boldsymbol{r})\exp(-12\pi^2 t^2)$. In Section 4 we proved the convergence with the order $2 - \varepsilon$ provided that the statement of Lemma 3 holds. It is interesting to study the opposite case when it is not possible to construct simplicial splittings without additional mesh nodes. A well-known mesh fragment satisfying this condition is shown in Fig. 7 at the left.

Consider a structured cubic mesh with edge length h, $1/h \in \mathbb{N}$, and split one of these elements into 7 hexahedrons as shown in Fig. 7 at the right. The radius-vectors for 6 additional nodes are chosen randomly provided that the resulting mesh satisfies the conditions of Theorem 1, and scaled with the factor h when generating a mesh with another h.

The norms of the numerical error at $t = t_{max} = (\ln 2)/(12\pi^2)$, namely,

$$\varepsilon_{u,2} = \|u^h(t) - \Pi u(t,\,\cdot\,)\|/\sqrt{|\Omega|}, \quad \varepsilon_{u,\infty} = \max_{j \in N} |u_j(t) - u(t, \boldsymbol{r}_j)|, \tag{49}$$

Table 1. The numerical error for the heat equation

h	P1-Galerkin		Local elem. splitting		
	$\varepsilon_{u,2}$	$\varepsilon_{u,\infty}$	$\varepsilon_{u,2}$	$\varepsilon_{u,\infty}$	$\varepsilon_{u,\infty}^{P1-Gal.}/\varepsilon_{u,\infty}^{MLES}$
1/8	$2.38 \cdot 10^{-2}$	$6.12 \cdot 10^{-2}$	$6.27 \cdot 10^{-3}$	$1.61 \cdot 10^{-2}$	3.80
1/16	$6.22 \cdot 10^{-3}$	$1.71 \cdot 10^{-2}$	$1.57 \cdot 10^{-3}$	$4.41 \cdot 10^{-3}$	3.88
1/32	$1.57 \cdot 10^{-3}$	$4.43 \cdot 10^{-3}$	$3.94 \cdot 10^{-4}$	$1.22 \cdot 10^{-3}$	3.63
1/64	$3.93 \cdot 10^{-4}$	$1.11 \cdot 10^{-3}$	$9.84 \cdot 10^{-5}$	$3.20 \cdot 10^{-4}$	3.47
1/128	$9.84 \cdot 10^{-5}$	$2.78 \cdot 10^{-4}$	$2.46 \cdot 10^{-5}$	$8.17 \cdot 10^{-5}$	3.40

are collected in Table 1. Both schemes are of the second order of accuracy on uniform Cartesian meshes. The defect in the mesh structure is local and thus it not strong enough to deteriorate the overall convergence in the L_2-norm, so the result $\varepsilon_{u,2} \sim h^2$ is as expected. The method of local element splittings yields 4 times smaller numerical error due to a more compact stencil.

We can also compare the method of the local element splittings with the P1-Galerkin method by the value of $\varepsilon_{u,\infty}$. For the method of local element splittings, the local maximum of the error is located near the splitted elements (except for $h = 1/8$). In contrast, for the P1-Galerkin method, the local maximum of the error is localed near one of the local extrema of the solution, The ratio of the numerical errors obtained by the P1-Galerkin method and by the method of local element splittings is shown in the right column of the table. This ratio becomes smaller as the mesh is refined. This means that the error of method of local element splittings does depend on the mesh quality more than the error of the P1-Galerkin method. However, in this test, the order of vanishing of the numerical error is greater than one even in L_∞-norm.

Now consider the Navier – Stokes system with no heat conduction linearized on the steady uniform field $\bar{\rho} = 1$, $\bar{\boldsymbol{u}} = 0$, $\bar{p} = 1/\gamma$, namely,

$$\frac{\partial Q'}{\partial t} + \nabla \cdot \mathcal{F}_{lin}(Q') = \nabla \cdot \mathcal{F}_{lin}^{V}(Q, \nabla Q),$$

where $Q' = (\rho', \boldsymbol{u}', p'/(\gamma - 1))^T$,

$$\mathcal{F}_{lin} = \begin{pmatrix} \boldsymbol{u}' \\ p'\boldsymbol{I} \\ \boldsymbol{u}'/(\gamma - 1) \end{pmatrix}, \quad \mathcal{F}_{lin}^{V}(Q, \nabla Q) = \begin{pmatrix} 0 \\ \boldsymbol{\tau} \\ 0 \end{pmatrix},$$

and the stress tensor $\boldsymbol{\tau}$ is given by (42). We put $\mu = 1$; the computational domain is the cube 25x25x25 with edges aligned with mesh axes and the periodical boundary conditions. A general solution of these equations is a linear combination of acoustical, vortex, and entropy waves. We consider the solution

$$Q' = \exp(-\mu \boldsymbol{k}^2 t + i\boldsymbol{k}\cdot\boldsymbol{r}) \begin{pmatrix} 0 \\ [\boldsymbol{c} \times \boldsymbol{k}] \\ 0 \end{pmatrix} + A\exp(i\omega t + i\boldsymbol{k}\cdot\boldsymbol{r}) \begin{pmatrix} -\boldsymbol{k}^2/\omega \\ \boldsymbol{k} \\ -\boldsymbol{k}^2/(\omega(\gamma-1)) \end{pmatrix},$$

where ω is given by the dispersion relation

$$\omega = \frac{2}{3} i\mu \boldsymbol{k}^2 \pm \left(\boldsymbol{k}^2 - \left(\frac{2}{3}\mu \boldsymbol{k}^2 \right)^2 \right)^{1/2}.$$

We put $\boldsymbol{k} = (2\pi/25, 2\pi/25, 0)^T$, $A = \sqrt{2}$, $\boldsymbol{c} = (0{,}0{,}1/(2\pi))^T$, so the exact solution is independent of z and $u'_z = 0$.

For this test, we use four artificial meshes. The first one is shown in Fig. 8, and the other three meshes are generated by successive refining of the first one. These meshes contain all four types of mesh elements. For the discretization of convective fluxes we use semitransparent control volumes [15]; we present results for the basic approximation $F_{jk}(Q) = (1/2)(\mathcal{F}(Q_j) + \mathcal{F}(Q_k)) \cdot \boldsymbol{n}_{jk}$, where $\boldsymbol{n}_{jk}$ is the geometric coefficient associated with the pair of nodes j and k, for the EBR3 [28] scheme and for the Flux Correction method [20]. In the last scheme we used gradient approximation using third order interpolation polynomials and the pointwise treatment of the source/time derivative term. This scheme is 2-exact on arbitrary unstructured meshes.

We will look at the norms of the numerical errors $\varepsilon_{f,l}$ (see (49)) for $f = p'$ (pressure pulsation) and $f = u'_y$ (y-component of the velocity pulsation) for $l = 2$ and $l = \infty$ at $t = t_{max} = 1$. They are collected in Table 2 for the basic approximation, in Table 3 for the EBR3 scheme and in Table 4 for the Flux Correction method.

The Table 4 shows that in the square norm the numerical solution converges to the exact solution approximately with the second order, while for maximal norm the numerical order is lower. The results given in Tables 2 and 3 also shows lower convergence rate due to the lower order of convective terms approximation. For the sake of this paper, the main result of this test is that there is no significant difference between the P1-Galerkin method and the method of local element splittings.

Now we move to a high Reynolds number problem. In convection-dominated problems the numerical error is mostly determined by the choice of convective terms approximation, so we consider only one case.

Consider a flow around a blade with the profile NACA0012 [45]. The chord length is unit; Mach number of the background flow is $M_\infty = 0.15$, Reynolds number is Re $= 6 \cdot 10^6$. Molecular viscosity is defined by the Sutherland law with

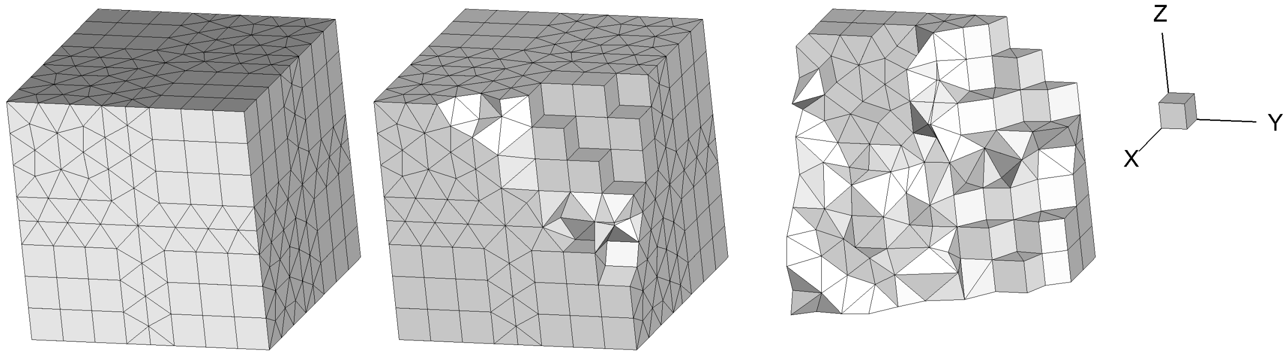

Figure 8. Coarse mesh for the linear waves dissipation test

Table 2. The numerical error for the linear waves dissipation test. Basic approximation for convective terms

h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$	$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$
6.221	0.279	$1.66 \cdot 10^{-1}$	$6.43 \cdot 10^{-2}$	$1.65 \cdot 10^{-1}$	$6.48 \cdot 10^{-2}$
3.508	0.140	$1.13 \cdot 10^{-1}$	$2.30 \cdot 10^{-2}$	$1.13 \cdot 10^{-1}$	$2.32 \cdot 10^{-2}$
1.917	0.070	$5.53 \cdot 10^{-2}$	$8.04 \cdot 10^{-3}$	$5.53 \cdot 10^{-2}$	$8.09 \cdot 10^{-3}$
0.975	0.035	$2.70 \cdot 10^{-2}$	$2.83 \cdot 10^{-3}$	$2.70 \cdot 10^{-2}$	$2.84 \cdot 10^{-3}$
h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$	$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$
6.221	0.279	$1.02 \cdot 10^{-1}$	$4.11 \cdot 10^{-2}$	$1.02 \cdot 10^{-1}$	$4.05 \cdot 10^{-2}$
3.508	0.140	$5.32 \cdot 10^{-2}$	$1.16 \cdot 10^{-3}$	$5.28 \cdot 10^{-2}$	$1.14 \cdot 10^{-2}$
1.917	0.070	$1.58 \cdot 10^{-2}$	$2.93 \cdot 10^{-3}$	$1.58 \cdot 10^{-2}$	$2.89 \cdot 10^{-3}$
0.975	0.035	$4.58 \cdot 10^{-3}$	$7.34 \cdot 10^{-4}$	$4.54 \cdot 10^{-3}$	$7.24 \cdot 10^{-4}$

$T_\infty = 300K$. The angle of attack is set at 15 degrees, to make the setup more sensitive to the scheme. We solve Reynolds-averaged Navier – Stokes equations with the Spalart – Allmaras turbulence model. For this test we use two structured quadrilateral meshes (coarse and fine) and a second-order finite volume scheme with no slope limitation. The number of nodes on the profile is 162 for the coarse mesh and 246 for the fine mesh; the first mesh step normal to the blade is equal to 10^{-6} for both meshes.

The results for the drag and lift coefficients are presented in Table 5. We compare them with the result obtained on a very fine mesh (897 nodes on the blade). We see again that the difference between the P1-Galerkin method and the method of local

Table 3. The numerical error for the linear waves dissipation test. EBR3 scheme for convective terms

h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$	$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$
6.221	0.279	$1.05 \cdot 10^{-1}$	$3.24 \cdot 10^{-2}$	$1.05 \cdot 10^{-1}$	$3.28 \cdot 10^{-2}$
3.508	0.140	$4.08 \cdot 10^{-2}$	$8.15 \cdot 10^{-3}$	$4.09 \cdot 10^{-2}$	$8.15 \cdot 10^{-3}$
1.917	0.070	$1.57 \cdot 10^{-2}$	$2.31 \cdot 10^{-3}$	$1.57 \cdot 10^{-2}$	$2.36 \cdot 10^{-3}$
0.975	0.035	$6.48 \cdot 10^{-3}$	$6.70 \cdot 10^{-4}$	$6.49 \cdot 10^{-3}$	$6.81 \cdot 10^{-4}$

h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$	$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$
6.221	0.279	$6.71 \cdot 10^{-2}$	$1.90 \cdot 10^{-2}$	$6.09 \cdot 10^{-2}$	$1.85 \cdot 10^{-2}$
3.508	0.140	$3.59 \cdot 10^{-2}$	$4.87 \cdot 10^{-3}$	$3.59 \cdot 10^{-2}$	$4.86 \cdot 10^{-3}$
1.917	0.070	$1.29 \cdot 10^{-2}$	$1.21 \cdot 10^{-3}$	$1.29 \cdot 10^{-2}$	$1.18 \cdot 10^{-3}$
0.975	0.035	$4.72 \cdot 10^{-3}$	$3.06 \cdot 10^{-4}$	$4.69 \cdot 10^{-3}$	$2.92 \cdot 10^{-4}$

Table 4. The numerical error for the linear waves dissipation test. Steady FC scheme for convective terms

h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$	$\varepsilon_{p',\infty}$	$\varepsilon_{p',2}$
6.221	0.279	$9.81 \cdot 10^{-2}$	$3.87 \cdot 10^{-2}$	$9.87 \cdot 10^{-2}$	$3.83 \cdot 10^{-2}$
3.508	0.140	$2.93 \cdot 10^{-2}$	$9.47 \cdot 10^{-3}$	$2.92 \cdot 10^{-2}$	$9.32 \cdot 10^{-3}$
1.917	0.070	$8.66 \cdot 10^{-3}$	$2.37 \cdot 10^{-3}$	$8.66 \cdot 10^{-3}$	$2.33 \cdot 10^{-3}$
0.975	0.035	$3.35 \cdot 10^{-3}$	$6.07 \cdot 10^{-4}$	$3.35 \cdot 10^{-3}$	$5.97 \cdot 10^{-4}$

h	h_{min}	P1-Galerkin		Local elem. splitting	
		$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$	$\varepsilon_{u'_y,\infty}$	$\varepsilon_{u'_y,2}$
6.221	0.279	$5.16 \cdot 10^{-2}$	$1.44 \cdot 10^{-2}$	$5.20 \cdot 10^{-2}$	$1.49 \cdot 10^{-2}$
3.508	0.140	$2.98 \cdot 10^{-2}$	$5.54 \cdot 10^{-3}$	$2.98 \cdot 10^{-2}$	$5.45 \cdot 10^{-3}$
1.917	0.070	$1.32 \cdot 10^{-2}$	$1.57 \cdot 10^{-3}$	$1.13 \cdot 10^{-2}$	$1.53 \cdot 10^{-3}$
0.975	0.035	$4.62 \cdot 10^{-3}$	$4.09 \cdot 10^{-4}$	$4.59 \cdot 10^{-3}$	$3.97 \cdot 10^{-4}$

element splittings in negligible.

Table 5. The drag and lift coefficients for NACA0012 profile

Scheme	Mesh	CL	CD
P1-Galerkin	coarse	1.505718	0.027731
	fine	1.533320	0.023143
Local elem. splitting	coarse	1.504825	0.027822
	fine	1.533145	0.023170
Local elem. splitting	very fine	1.543686	0.022019

7. Implicit time integration

Up to this point we have noticed no advantage of the method of local splittings compared to Galerkin method. They have approximately similar underlying theory, approximately the same accuracy, the same computational costs (in our implementation), and similar implementation complexity. However, implicit time discretization reveals a difference between them.

In this section, we assume the use of "direct" control volumes for the approximation of convective terms, then $\hat{N}_1(j)$ is the set of nodes connected to j by edge.

Apply the first-order backward difference formula (BDF1) to the scheme (43):

$$V_j \frac{Q_j^{n+1} - Q_j^n}{\Delta t} + \Phi_j(Q^{n+1}) = \Phi_j^V(Q^{n+1}). \tag{50}$$

To solve this nonlinear system, one can use an iterative process based on the Newton method: $Q_j^{(0)} = Q_j^n$,

$$\begin{gathered} Q_j^{(s+1)} = Q_j^{(s)} - M^{-1}\left(\text{diag}\{V_j\}\frac{Q^{(s)} - Q^n}{\Delta t} + \Phi(Q^{(s)}) - \Phi^V(Q^{(s)})\right), \\ M = \frac{1}{\Delta t}\text{diag}\{V_j\} - M^{conv} + M^{visc}, \\ M^{conv} \approx M^{conv}_{exact} = \frac{d\Phi}{dQ}(Q^{(s)}), \quad M^{visc} \approx M^{visc}_{exact} = \frac{d\Phi^V}{dQ}(Q^{(s)}). \end{gathered} \tag{51}$$

One needs to solve linear algebraic system with the matrix M at each iteration. We solve it using the preconditioned BiCGStab solver [46].

The true Newton method (i. e. with $M^{conv} = M^{conv}_{exact}$ and $M^{visc} = M^{visc}_{exact}$) is extremely inefficient in practice due to wide stencils of high-order finite volume schemes and big condition number of the resulting matrix M. Therefore a reduced variant of the matrix M^{conv} is commonly used (see [47] for example). We take the matrix M^{conv} in a reduced form as it be for the "first-order" finite-volume scheme and drop the derivatives of eigenvalues and eigenvectors, which are used in the Roe solver. Then the portrait of M^{conv} contains only the main diagonal and the cells (j, k) such that $j \in \hat{N}_1(k)$ (or, which is the same, $k \in \hat{N}_1(j)$), here we consider a 5x5 block related to the set of equations as one matrix element. On a structured hexahedral mesh, M^{conv} has 7 nonzero elements a row.

Since we use a simplified Jacobian for convective terms, the convergence of the iterations is far from quadratic, and there is no reason to use $M^{visc} = M^{visc}_{exact}$. For a structured hexahedral mesh each row of the matrix M^{visc}_{exact} consists of 27 nonzero elements. The idea is to keep only 7 elements per row for M^{visc}. We drop the elements jk such that the nodes jk are not connected by an edge and preserve zero column sum. Besides, we drop skew-symmetric part of the tensor $G_{jk,\alpha\beta}[\varkappa]$. To be precise, for $j \in \hat{N}_1(k)$, we put

$$M^{visc}_{jk} = \rho_k^{-1} \begin{pmatrix} 0 & 0 & 0 \\ -\boldsymbol{u}_k \cdot \boldsymbol{m} & \boldsymbol{m} & 0 \\ -\boldsymbol{e} \cdot \boldsymbol{u}_k - \sigma(E_k - \boldsymbol{u}_k^2/2) & \boldsymbol{e}^T - \sigma \boldsymbol{u}^T & \sigma \end{pmatrix},$$

where ρ_k, $\boldsymbol{u}_k$, E_k are density, velocity and total energy at node K, $\boldsymbol{m} = \{m_{\alpha\beta}[\mu]\}$, $\boldsymbol{e} = \{e_\alpha\}$,

$$m_{\alpha\alpha}[\varkappa] = \sum_{\delta=x,y,z} G_{jk,\delta\delta}[\varkappa] + \frac{1}{3} G_{jk,\alpha\alpha}[\varkappa], \quad m_{\alpha\beta}[\varkappa] = \frac{1}{6}(G_{jk,\alpha\beta}[\varkappa] + G_{jk,\beta\alpha}[\varkappa]),$$

$$e_\alpha = \sum_{\beta=x,y,z} m_{\alpha\beta}[\mu u_\beta], \quad \sigma = \frac{\gamma}{\mathrm{Pr}} \sum_{\delta=x,y,z} G_{jk,\delta\delta}[\mu].$$

Diagonal elements of the matrix M_{visc} are given by

$$M^{visc}_{jj} = - \sum_{k \in \hat{N}_1(j)} M^{visc}_{kj},$$

and $M^{visc}_{jk} = 0$ for $j \neq k$ and $j \notin \hat{N}_1(k)$.

The reduction of the martix portrait is almost 4 times, and it proportionally reduces the computational costs for solving the algebraic system. The memory usage also reduces significantly.

It turns out that this trick behaves differently for the P1-Galerkin method and for the method of local elements splitting. We demonstrate this behavior on the 2D flow around a plate. Consider Navier – Stokes equation in the domain $0 < x < 3$, $0 < y < 1$. On $y = 0$ we impose no-slip adiabatic boundary conditions, and on other boundaries we keep free-stream flow with Mach number 0.1 (namely, we put $\rho_\infty = 1$, $\boldsymbol{u}_\infty = (1{,}0{,}0)^T$, $p_\infty = 100/\gamma$). Viscosity coefficient is $\mu = \mathrm{Re}^{-1}$,

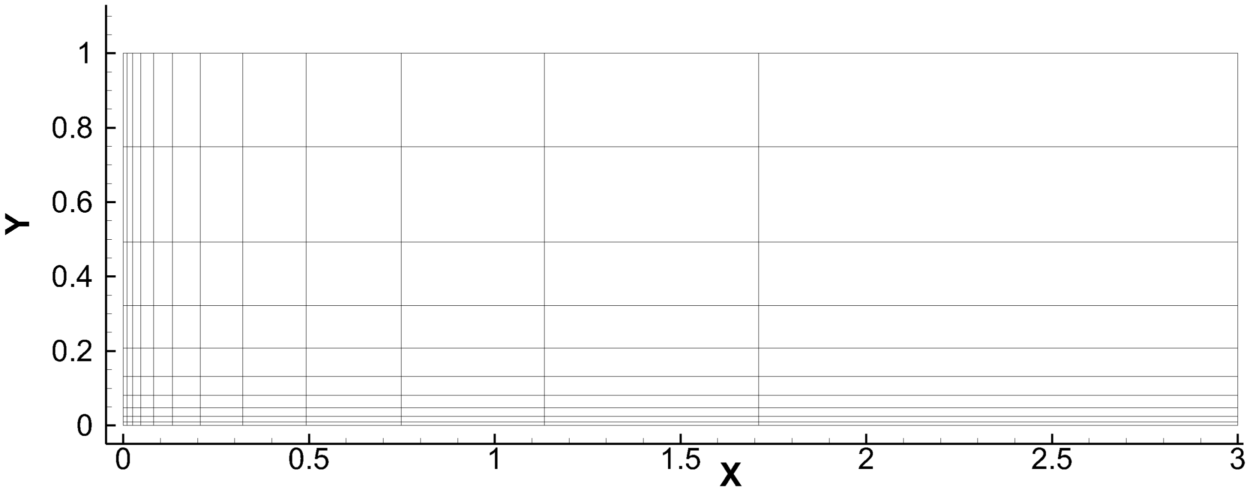

Figure 9. Mesh used for the flow simulation around a plate

where Re is the formal Reynolds number (i. e. based on the mesh length unit). For the convective terms approximation, we use the basic first-order finite-volume scheme with the Lax – Friedrichs – Rusanov flux. This allows us to use the exact flux Jacobian for convective terms.

We use Cartesian rectangular meshes with the nodes (x_j, y_j), where $x_0 = y_0 = 0$, $x_1 = y_1 = h_{\text{min}}$, and then the edge lengths grows progressively with the power 1.5. The mesh with $h_{\text{min}} = 0.01$ is shown on Fig. 9.

For these extremely small meshes we can store a flux Jacobian in the dense format and use Lapack package to analyze it. We apply block-diagonal preconditioner to the matrix, i. e. we analyze the matrix $\tilde{M} = M\Lambda$, where M is the flux Jacobian and diagonal blocks of Λ are the inverses of diagonal blocks of M. The condition numbers in l_2 of the matrices $\tilde{M}$ are collected in Table 6. Here we consider flux Jacobians calculated on the field $\rho_j = \rho_\infty$, $p_j = p_\infty$, and $\boldsymbol{u}_j = 0$ for the nodes on the no-slip boundary and $\boldsymbol{u}_j = \boldsymbol{u}_\infty$ otherwise; the flux Jacobians computed on steady solutions of the given problem exhibit similar behavior.

The condition numbers themselves do not provide any meaningful information about the possibility of solving the linear system. By the way, Table 6 shows that the truncation of the flux Jacobian significantly increases the condition number in the case of P1-Galerkin method. This is a reason to suspect that the truncation of flux Jacobian significantly corrupts it. In contrast, for the method of local element splittings the truncation does not significantly affect the condition number.

This suspicion is confirmed by the numerical experiments for high Reynolds number flows. The use of full Jacobian for diffusion terms makes it possible to use high CFL numbers, limited mainly by the convective terms. When using the P1-Galerkin method, the truncation of the flux Jacobian may either work properly,

Table 6. Condition numbers of preconditioned flux Jacobians for different schemes and case parameters

Scheme	Jacobian	$h_{\min}$	Re	CFL	cond. number
P1-Galerkin	full	10^{-2}	100	10	$7.90 \cdot 10^4$
	truncated	10^{-2}	100	10	$7.74 \cdot 10^4$
P1-Galerkin	full	10^{-2}	100	100	$8.61 \cdot 10^6$
	truncated	10^{-2}	100	100	$8.91 \cdot 10^6$
P1-Galerkin	full	10^{-2}	10	10	$5.69 \cdot 10^4$
	truncated	10^{-2}	10	10	$6.14 \cdot 10^6$
P1-Galerkin	full	10^{-2}	10	100	$2.10 \cdot 10^6$
	truncated	10^{-2}	10	100	$1.95 \cdot 10^7$
Local elem. splitting	full	10^{-2}	100	10	$8.34 \cdot 10^4$
	truncated	10^{-2}	100	10	$8.38 \cdot 10^4$
Local elem. splitting	full	10^{-2}	100	100	$8.87 \cdot 10^6$
	truncated	10^{-2}	100	100	$8.92 \cdot 10^6$
Local elem. splitting	full	10^{-2}	10	10	$5.66 \cdot 10^4$
	truncated	10^{-2}	10	10	$5.66 \cdot 10^4$
Local elem. splitting	full	10^{-2}	10	100	$2.60 \cdot 10^6$
	truncated	10^{-2}	10	100	$2.63 \cdot 10^6$
P1-Galerkin	full	10^{-3}	100	10	$5.60 \cdot 10^4$
	truncated	10^{-3}	100	10	$3.93 \cdot 10^6$
P1-Galerkin	full	10^{-3}	100	100	$2.10 \cdot 10^6$
	truncated	10^{-3}	100	100	$3.28 \cdot 10^7$
P1-Galerkin	full	10^{-3}	10	10	$2.04 \cdot 10^5$
	truncated	10^{-3}	10	10	$7.10 \cdot 10^6$
P1-Galerkin	full	10^{-3}	10	100	$7.87 \cdot 10^5$
	truncated	10^{-3}	10	100	$5.25 \cdot 10^7$
Local elem. splitting	full	10^{-3}	100	10	$5.66 \cdot 10^4$
	truncated	10^{-3}	100	10	$5.66 \cdot 10^4$
Local elem. splitting	full	10^{-3}	100	100	$2.60 \cdot 10^6$
	truncated	10^{-3}	100	100	$2.63 \cdot 10^6$
Local elem. splitting	full	10^{-3}	10	10	$1.75 \cdot 10^5$
	truncated	10^{-3}	10	10	$1.75 \cdot 10^5$
Local elem. splitting	full	10^{-3}	10	100	$7.32 \cdot 10^5$
	truncated	10^{-3}	10	100	$7.41 \cdot 10^5$

or lead to solver (BiCGStab) unconvergence, or lead to Newton process unconvergence, depending on the flow parameters, Courant number, and the mesh resolution of the boundary layer. In contrast, when using the method of local element splittings, the truncation of the flux Jacobian does not significantly change the behavior of the scheme.

For example, consider again the flow around NACA0012 airfoil. Using the full Jacobian, we can run the computations with the CFL number about $2 \cdot 10^4$ with both the P1-Galerkin method and the method of local element splittings. The increasing of the CFL number is possible up to $1 \cdot 10^6$ but this results in some spurious oscillations in time, and the further increase of the CFL number leads to the loss of convergence. This behavior remains the same if we use the truncated Jacobian and the method of local element splittings. However, if we use the truncated Jacobian and the P1-Galerkin method, we are unable to run the computations with CFL $\gg 1$. Note that on these meshes we have very well-resolved boundary layer; for poor-resolved boundary layers higher CFL numbers are possible, but still much less than when the full Jacobian and/or the method of local element splittings are in use.

8. Conclusion

The method of local element splittings is a novel finite-element method for the discretization of diffusion terms of the Navier – Stokes system on unstructured meshes. It is very close to the classical P1-Galerkin method; they coincide for simplicial meshes and have the same stencil. The new method is second-order accurate on structured meshes and on simplicial meshes, with possible degradation on unstructured mixed-element meshes. The difference from the P1-Galerkin method also appears in the approximation of the Laplace operator on Cartesian meshes. The P1-Galerkin method yields a combination of direct and skewed crosses, while the method of local element yields the direct cross (5-point in 2D and 7-point in 3D).

Although for the discretization of viscous terms the stencil of the method cannot be reduced to the set of edge-connected nodes, this approach may be applied to the approximated flux Jacobian used in implicit schemes, especially for convection-dominated flows. The numerical results show that for the method of local element splittings it significantly improves the computational costs and memory requrements without loss of convergence of a linear system solver and of the Newton method. This is not possible for the classical P1-Galerkin method.

The author thanks M. D. Surnachev and A. V. Gorobets for their useful remarks.

References

1. Jasak H., Weller H. G., Gosman A. D. High-resolution NVD differencing scheme for arbitrarily unstructured meshes // International Journal for Numerical Methods in Fluids. 1999. Vol. 31. P. 431–449.

2. Touze C., Murrone A., Guillard H. Multislope MUSCL method for general unstructured meshes // Journal of Computational Physics. 2015. P. 389–418.

3. Wolf W. R., Azevedo J. L. F. High-order ENO and WENO schemes for unstructured grids // International Journal for Numerical Methods in Fluids. 2007. Vol. 55, no. 10. P. 917–943.

4. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems / Dumbser M., Kaeser M., Titarev V. A. et al. // Journal of Computational Physics. 2007. Vol. 226. P. 204–243.

5. Tsoutsanis P., Titarev V. A., Drikakis D. WENO schemes on arbitrary mixed-element unstructured meshes in three space dimensions // Journal of Computational Physics. 2011. Vol. 230. P. 1585–1601.

6. Liu Y., Zhang Y.-T. A Robust Reconstruction for Unstructured WENO Schemes // Journal of Scientific Computing. 2013. Vol. 54. P. 603–621.

7. Tsoutsanis P., Antoniadis A. F., Drikakis D. WENO schemes on arbitrary unstructured meshes for laminar, transitional and turbulent flows // Journal of Computational Physics. 2014. Vol. 256. P. 254–276.

8. Zhou T., Li Y., Shu C.-W. Numerical comparison of WENO finite volume and Runge-Kutta discontinuous Galerkin methods // Journal of Scientific Computing. 2001. Vol. 16. P. 145–171.

9. Cockburn B., Shu C.-W. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems // Journal of Scientific Computing. 2001. Vol. 16, no. 3. P. 173–261.

10. Gassner G., Lorcher F., Munz C.-D. A discontinuous Galerkin scheme based on a space–time expansion ii. Viscous flow equations in multi dimensions // Journal of Scientific Computing. 2008. Vol. 34. P. 260–286.

11. Zhu J., Qui J. Runge–Kutta discontinuous Galerkin method using WENO-type limiters: Three-dimensional unstructured meshes // Commun. Comput. Phys. 2012. Vol. 11. P. 985–1005.

12. Huynh H. T., Wang Z. J., Vincent P. E. High-order methods for computational fluid dynamics: A brief review of compact differential formulations on unstructured grids // Computers and Fluids. 2012. Vol. 98. P. 209–220.

13. Roe, P. L. Error estimates for cell-vertex solutions of the compressible Euler equations: Tech. Rep.: : ICASE Report 87-6, 1987.

14. Barth T. J. Numerical aspects of computing high Reynolds number flows on unstructured meshes // AIAA Paper No. 91-0721. 1991.

15. Bakhvalov P. A., Kozubskaya T. K. Construction of Edge-Based 1-Exact Schemes for Solving the Euler Equations on Hybrid Unstructured Meshes // Comp. Math. Math. Phys. 2017. Vol. 57. P. 680–697.

16. Gorobets A. V. Parallel algorithm of the NOISEtte code for CFD and CAA simulations // Lobachevskii Journal of Mathematics. 2018. Vol. 39. P. 524–532.

17. Application of the FUN3D Solver to the 4th AIAA Drag Prediction Workshop / Lee-Rausch E. M., Hammond D. P., Nielsen E. J. et al. // Journal of Aircraft. 2014. Vol. 51. P. 680–697.

18. Rakowitz M., Schwamborn D., Sutcliffe M. Structured and Unstructured Computations on the DLR-F4 Wing-Body Configuration // Journal of Aircraft. 2003. Vol. 40. P. 1149–1160.

19. Eliasson P. EDGE, a Navier-Stokes solver for unstructured grids: Tech. Rep.: FOI-R–0298–SE. SE-172 90 STOCKHOLM: FOI Swedish defence research agency, Division of Aeronautics, FFA, 2001. December.

20. Katz Aaron, Sankaran Venkateswaran. An Efficient Correction Method to Obtain a Formally Third-Order Accurate Flow Solver for Node-Centered Unstructured Grids // J. Sci. Comput. New York, NY, USA, 2012. may. T. 51, № 2. C. 375–393.

21. Pincock B., Katz A. High-Order Flux Correction for Viscous Flows on Arbitrary Unstructured Grids // J. Sci. Comput. New York, NY, USA, 2014. nov. T. 61, № 2. C. 454–476.

22. Work C. D., Katz A. J. Aspects of the Flux Correction Method for Solving the Navier-Stokes Equations on Unstructured Meshes // AIAA paper No. 2015-0834. 2015.

23. High-Order strand grid methods for shock turbulence interaction / Tong O., Yanagita Y., Schaap R. et al. // AIAA paper No. 2015-2283. 2015.

24. Computation of unsteady flows with mixed finite volume/ finite element upwind methods / Debiez C., Dervieux A., Mer K. et al. // International Journal for Numerical Method in Fluids. 1998. Vol. 27. P. 193–206.

25. Debiez C., Dervieux A. Mixed-element-volume MUSCL methods with weak viscosity for steady and unsteady flow calculations // Computers and Fluids. 2000. Vol. 29, no. 1. P. 89–118.

26. Abalakin I., Dervieux A., Kozubskaya T. High Accuracy Finite Volume Method for Solving Nonlinear Aeroacoustics Problems on Unstructured Meshes // Chinese Journal of Aeronautics. 2006. Vol. 19. P. 97–104.

27. Koobus B., Alauzet F., Dervieux A. Computational Fluid Dynamics / Ed. by Magoules F. CRC Press, 2011. P. 131–204.

28. Abalakin I., Bakhvalov P., Kozubskaya T. Edge-based reconstruction schemes for unstructured tetrahedral meshes // International Journal for Numerical Methods in Fluids. 2016. Vol. 81. P. 331–356.

29. Bakhvalov P. A., Kozubskaya T. K. EBR-WENO scheme for solving gas dynamics problems with discontinuities on unstructured meshes // Computers and Fluids. 2017. Vol. 157. P. 312–324.

30. Nishikawa H. Alternative Formulations for First-, Second-, and Third-Order Hyperbolic Navier-Stokes Schemes // AIAA Paper No. 2015-2451. 2015.

31. Nakashima Y., Watanabe N., Nishikawa H. Hyperbolic Navier-Stokes Solver for Three-Dimensional Flows // AIAA Paper No. 2016-1101. 2016.

32. Mavriplis D. J., Venkatakrishnan V. A unified multigrid solver for the Navier–Stokes equations on mixed element meshes // International Journal for Computational Fluid Dynamics. 1997. Vol. 8. P. 247–263.

33. Acceleration of NOISEtte Code for Scale-resolving Supercomputer Simulations of Turbulent Flows / Gorobets A., Bakhvalov P., Duben A. et al. // Lobachevskii Journal of Mathematics. 2020. Vol. 41. P. 1463–1474.

34. Numerical Investigation of the Aerodynamic and Acoustical Properties of a Shrouded Rotor / Abalakin I., Anikin V., Bakhvalov P. et al. // Fluid Dynamics. 2016. Vol. 51. P. 419–433.

35. Dankov B., Duben A., Kozubskaya T. Numerical modeling of the self-oscillation onset near a three-dimensional backward-facing step in a transonic flow // Fluid Dynamics. 2016. Vol. 51. P. 534–543.

36. Dankov B., Duben A., Kozubskaya T. Numerical simulation of the transonic turbulent flow around a wedge-shaped body with a backward-facing step // Mathematical Models and Computer Simulations. 2016. Vol. 8. P. 274–284.

37. A tetrahedral-based superconvergent scheme for aeroacoustics: Paper: 5212 / Gourvitch N., Roge G., Abalakin I. et al. INRIA: INRIA Report, 2004.

38. Nishikawa H. Beyond Interface Gradient. A General Principle for Constructing Diffusion Schemes // AIAA Paper No. 2010-5093. 2010.

39. Křížek M. On the maximal angle condition for linear tetrahedral elements // SIAM J. Numer. Anal. 1992. P. 513–520.

40. Acosta G. Lagrange and average interpolation over 3D anisotropic elements // Journal of Computational and Applied Mathematics. 2001. Vol. 135. P. 91–109.

41. Al Shenk N. Uniform error estimates for certain narrow Lagrange finite elements // Mathematics of computation. 1994. Vol. 63. P. 105–119.

42. Kobayashi K., Tsuchiya T. Error analysis of Lagrange interpolation on tetrahedrons // Journal of Approximation Theory. 2020. Vol. 249.

43. Adams R. A. Sobolev spaces. AP, 1975.

44. Kallinderis Y. A 3-D finite-volume method for the Navier-Stokes equations with adaptive hybrid grids // Applied numerical mathematics. 1996. P. 378–406.

45. 2DN00: 2D NACA 0012 Airfoil Validation Case: Tech. Rep.: : NASA Langley Research Center. URL: https://turbmodels.larc.nasa.gov/naca0012_val.html.

46. Van der Vorst H. A. Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems // SIAM Journal on Scientific and Statistical Computing. 1992. Vol. 13. P. 631–644.

47. A multithreaded OpenMP implementation of the LU-SGS method using the multilevel decomposition of the unstructured computational mesh / Petrov M. N., Titarev V. A., Utyuzhnikov S. V. et al. // Computational Mathematics and Mathematical Physics. 2017. Vol. 57. P. 1856–1865.

Contents