Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 226 (2007) 1291–1309
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A stable and efficient hybrid scheme for viscous problems in complex geometries

Jing Gong [a,*], Jan Nordström [a,b,c]

[a] Department of Information Technology, Scientific Computing, Uppsala University, SE-75105 Uppsala, Sweden
[b] Department of Computational Physics, The Swedish Defence Research Agency (FOI), SE-16490 Stockholm, Sweden
[c] Department of Aeronautical and Vehicle Engineering, The Royal Institute of Technology (KTH), SE-10044 Stockholm, Sweden

Received 9 January 2007; received in revised form 12 May 2007; accepted 21 May 2007
Available online 8 June 2007

Abstract

In this paper, we present a stable hybrid scheme for viscous problems. The hybrid method combines the unstructured finite volume method with high-order finite difference methods on complex geometries. The coupling procedure between the two numerical methods is based on energy estimates and stable interface conditions are constructed. Numerical calculations show that the hybrid method is efficient and accurate.

Keywords: Viscous problems; Hybrid methods; Finite difference; Finite volume; Coupling procedure; Stability; Efficiency

1. Introduction

High-order finite difference methods (HOFDM) provide an efficient approach when high resolution is essential in a calculation. It is also clear that the node-centered unstructured finite volume method (UFVM) is widely used for problems with complex geometries and non-linear phenomena. In computational physics, the computational domain is often for efficiency and mesh generation reasons divided into multiple blocks, where either HOFDM or UFVM can be used. If a stable and accurate coupling at the block interfaces is achieved we can construct a very flexible and efficient computational method.

Attempts to combine structured and unstructured mesh types have been considered before. For instance, in [13] a method which uses both the finite difference method and the finite element method is developed. The finite difference domain and the finite element domain are patched together using overlapping meshes. In [5], the calculation of unsteady flow in Turbo-machinery was done using a mixture of quadrilateral and triangular cells for added flexibility. In [9], a two-dimensional zonal interactive scheme for Euler equations was developed for computing flows around complex geometries. Although many type of hybrid methods have been

* Corresponding author. Tel.: +46 184712975.
E-mail addresses: jing.gong@it.uu.se (J. Gong), Jan.Nordstrom@foi.se (J. Nordström).

0021-9991/$ - see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2007.05.018

developed to improve the accuracy and efficiency of calculations around complex geometries, very few deal with the essential stability issue.

In [3,15,16,18,6], stable interface treatment between multiple domains for HOFDM were presented. The technique is based on the so-called summation-by-parts (SBP) operators and impose the boundary and interface conditions weakly, see [2]. The weak imposition of boundary and interface conditions is necessary for stability since it preserves the SBP character of the difference operators. In [3], the authors developed stable and conservative interface and boundary conditions treatments of arbitrary spatial accuracy for the linear advection–diffusion equation. In [15], boundary and interface conditions for the constant coefficient Euler and Navier–Stokes equations were developed. The interface conditions are stable and conservative even if the finite difference operators and mesh sizes vary from domain to domain. The method is applied to multidimensional linear problems in curvilinear coordinates in [16]. In [18], it was shown that the method is suitable for aeroacoustic sound generation and propagation while various versions of interface procedures for viscous problems in one dimension were investigated in [6].

In a parallel development it was shown in [17,22,21] that the UFVM approximation of the first derivative [17] and the Laplacian [22,21] is an SBP formulation. In a similar manner as for the HOFDM, it was also shown that a correct weak imposition of boundary conditions lead to stability.

In [19], it was shown how to couple the UFVM and HOFDM in a stable way for hyperbolic problems. The energy method and a modification of the dual mesh in the UFVM lead to stability. The present paper continues the study of stable interface treatment by considering hybrid schemes for viscous problems. We also add the additional complexity of a curvilinear mesh in the HOFDM region. The technique derived in this paper makes it straight forward to apply the hybrid technique to the full Navier–Stokes equation.

The rest of the paper is organized as follows. In the next section, we derive stable boundary conditions for the continuous problem. Section 3 presents the two numerical methods on a single domain. In Section 4, we derive the stable coupling procedure. In Section 5, numerical experiments are performed. Conclusions are drawn in Section 6.

2. The continuous problem

Consider the model problem

$$u_t + au_x + bu_y = \varepsilon(u_{xx} + u_{yy}), \quad x, y \in \Omega,\ t > 0, \tag{1a}$$

$$u(x, y, 0) = f(x, y), \quad x, y \in \Omega, \tag{1b}$$

$$\alpha u + \beta \frac{\partial u}{\partial n} = g(x, y, t), \quad x, y \in \partial\Omega,\ t > 0. \tag{1c}$$

The coefficients a, b and ε are constants. In general, the coefficients α and β depend on x, y and t.

Let the inner product for real valued functions $u, v \in \Omega$ be defined by $(u, v) = \int\int_\Omega uv\,\mathrm{d}x\,\mathrm{d}y$ and the corresponding norm $\|u\|^2 = (u, u)$. Applying the energy method to (1a) yields,

$$\|u\|_t^2 + 2\varepsilon(\|u_x\|^2 + \|u_y\|^2) = -\oint_{\partial\Omega} \left(\bar{c}u^2 - 2\varepsilon u \frac{\partial u}{\partial n}\right) \mathrm{d}s. \tag{2}$$

where

$$n = \frac{(\mathrm{d}y, -\mathrm{d}x)}{\mathrm{d}s}, \quad \mathrm{d}s = \sqrt{\mathrm{d}x^2 + \mathrm{d}y^2}, \quad \bar{c} = (a, b) \cdot n, \quad \frac{\partial u}{\partial n} = (u_x, u_y) \cdot n.$$

Substituting the boundary conditions (1c) into (2) we obtain

$$\begin{aligned} \|u\|_t^2 + 2\varepsilon(\|u_x\|^2 + \|u_y\|^2) &= \oint_{\partial\Omega} \left[-\left(\bar{c} + \frac{2\alpha}{\beta}\varepsilon\right)u^2 + \frac{2\varepsilon}{\beta} ug\right] \mathrm{d}s \\ &= -\oint_{\partial\Omega} \left(\bar{c} + \frac{2\alpha}{\beta}\varepsilon\right)\left(u - \frac{\varepsilon}{\beta}\frac{1}{\bar{c} + \frac{2\alpha}{\beta}\varepsilon} g\right)^2 \mathrm{d}s + \oint_{\partial\Omega} \left(\frac{\varepsilon}{\beta}\right)^2 \left(\frac{1}{\bar{c} + \frac{2\alpha}{\beta}\varepsilon}\right) g^2\,\mathrm{d}s. \end{aligned} \tag{3}$$

This leads to immediately to:

Proposition 2.1. *The continuous problem (1) is strongly well posed if*

$$\bar{c} + \frac{2\alpha}{\beta}\varepsilon \geqslant 0 \quad \text{on } \partial\Omega. \tag{4}$$

Remark. When the solution can be estimated in terms of all types of data, the problem (1a) is called strongly well posed, see [7] for more details.

3. The discrete single domain problem

3.1. The finite volume method

The so-called *edge-based* finite volume method is used in this paper (see [4,8,11,12,17,24] for more details). The computational domain consists of non-overlapping elements and the variables are stored at the nodes of the mesh. For each node, the control volume that constitutes the dual grid is defined as a polygon with its vertices at the centers of gravity of the surrounding triangles (or quadrilaterals) and at the midpoints of the sides, see Fig. 1.

In the finite volume method the unknown variable u in Eq. (1a) is discretized by the vector $\mathbf{u} = [u_0, u_1, \ldots, u_N]$. $\mathbf{u}_x$, $\mathbf{u}_{xx}$, $\mathbf{u}_y$ and $\mathbf{u}_{yy}$ denote the approximations of u_x, u_{xx}, u_y and u_{yy}, respectively. We define

$$\mathbf{u}_x \approx D_x\mathbf{u} = (P)^{-1}Q_x\mathbf{u}, \quad \mathbf{u}_{xx} \approx D_xD_x\mathbf{u} = D_x^2 = (P)^{-1}Q_x(P)^{-1}Q_x\mathbf{u},$$
$$\mathbf{u}_y \approx D_y\mathbf{u} = (P)^{-1}Q_y\mathbf{u}, \quad \mathbf{u}_{yy} \approx D_yD_y\mathbf{u} = D_y^2 = (P)^{-1}Q_y(P)^{-1}Q_y\mathbf{u},$$

where P is a positive diagonal matrix with the control volumes Ω_i on the diagonal. In [17,19], it was shown by using the Green–Gauss theorem (see also [1]) that the matrices Q_x and Q_y have the components,

$$(Q_x)_{ij} = \frac{\mathrm{d}y_j}{2} = -(Q_x)_{ji}, \quad (Q_x)_{ii\notin\partial\Omega} = 0, \quad (Q_x)_{ii\in\partial\Omega} = \frac{\mathrm{d}y_i}{2}, \tag{5}$$

$$(Q_y)_{ij} = -\frac{\mathrm{d}x_j}{2} = -(Q_y)_{ji}, \quad (Q_y)_{ii\notin\partial\Omega} = 0, \quad (Q_y)_{ii\in\partial\Omega} = -\frac{\mathrm{d}x_i}{2}. \tag{6}$$

For the definition of $\mathrm{d}x_j$ and $\mathrm{d}y_j$, see Fig. 1. Moreover, Eqs. (5) and (6) imply that Q_x and Q_y satisfy

$$Q_x + (Q_x)^{\mathrm{T}} = Y, \quad Q_y + (Q_y)^{\mathrm{T}} = X, \tag{7}$$

where the non-zero elements in Y and X are Δy_i, $-\Delta x_i$ and correspond to the boundary points. Formulas 5 and 6 show that D_x and D_y are summation-by-parts (SBP) operators, see [17,19] for more details.

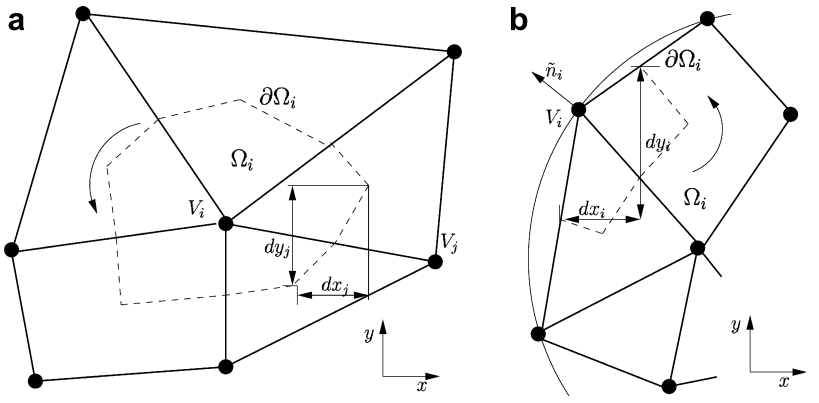

Fig. 1. The grid (solid lines) and the dual grid (dashed lines): (a) in the interior and (b) on the boundary.

On a grid point i at the boundary $\partial\Omega$, let $(D_n\mathbf{u})_i$ denote an approximation of $\partial u/\partial n$, i.e.,

$$\left(\frac{\partial u}{\partial n}\right)_i = [(u_x, u_y)\cdot n]_i \approx [(\mathbf{u}_x, \mathbf{u}_y)\cdot \tilde{n}]_i \approx ((D_x\mathbf{u})_i, (D_y\mathbf{u})_i)\cdot \tilde{n}_i = (D_n\mathbf{u})_i, \tag{8}$$

where $\tilde{n}_i$ is the outward pointing normal defined by,

$$\tilde{n}_i = \frac{(\mathrm{d}y_i, -\mathrm{d}x_i)}{\mathrm{d}s_i}, \quad \mathrm{d}s_i = \sqrt{\mathrm{d}x_i^2 + \mathrm{d}y_i^2}, \tag{9}$$

when proceeding counter-clockwise around the domain, see Fig. 1.

A semi-discrete approximation of Eqs. (1b) and (1c) can be written,

$$\mathbf{u}_t + aD_x\mathbf{u} + bD_y\mathbf{u} = \varepsilon(D_x^2\mathbf{u} + D_y^2\mathbf{u}) + P^{-1}(E_B)^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B + \tilde{\beta}(D_n\mathbf{u})_B - g] \tag{10}$$

where $\mathbf{u}_B$ represents $\mathbf{u}$ on the boundary $\partial\Omega$. E_B is a projection matrix which maps the values on the computational domain Ω to the outer boundary $\partial\Omega$, that is, $\mathbf{u}_B = E_B\mathbf{u}$ and $(D_n\mathbf{u})_B = E_B(D_n\mathbf{u})$. $\tilde{\alpha}$ and $\tilde{\beta}$ are diagonal matrices where the discrete value of the coefficients α and β are injected on the diagonal, respectively. Γ is a penalty matrix that will be determined below by stability requirements (see [17,19]).

By multiplying (10) with $\mathbf{u}^{\mathrm{T}}P$ and using,

$$\begin{aligned} &a\mathbf{u}^{\mathrm{T}}Q_x\mathbf{u} + b\mathbf{u}^{\mathrm{T}}Q_y\mathbf{u} = \frac{1}{2}\mathbf{u}_B^{\mathrm{T}}\Lambda_B\mathbf{u}_B,\\ &\mathbf{u}^{\mathrm{T}}Q_xD_x\mathbf{u} + \mathbf{u}^{\mathrm{T}}Q_yD_y\mathbf{u} = -(D_x\mathbf{u})^{\mathrm{T}}P(D_x\mathbf{u}) - (D_y\mathbf{u})^{\mathrm{T}}P(D_y\mathbf{u}) + \mathbf{u}_B^{\mathrm{T}}S_B(D_n\mathbf{u})_B, \end{aligned} \tag{11}$$

we obtain,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|_P^2 + 2\varepsilon\|D_x\mathbf{u}\|_P^2 + 2\varepsilon\|D_y\mathbf{u}\|_P^2 &= -\mathbf{u}_B^{\mathrm{T}}\Lambda_B\mathbf{u}_B + 2\varepsilon\mathbf{u}_B^{\mathrm{T}}S_BD_n\mathbf{u}_B + 2\mathbf{u}_B^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B + \tilde{\beta}(D_n\mathbf{u})_B - g]\\ &= -\mathbf{u}_B^{\mathrm{T}}(\Lambda_B - 2\Gamma\tilde{\alpha})\mathbf{u}_B + \underbrace{\mathbf{u}_B^{\mathrm{T}}(2\varepsilon S_B + 2\Gamma\tilde{\beta})(D_n\mathbf{u})_B}_{(\mathrm{IL})} - 2\mathbf{u}_B^{\mathrm{T}}\Gamma g. \end{aligned} \tag{12}$$

In (11) and (12), we have introduced (see Eqs. (8) and (9))

$$\bar{c}_i = (a,b)\cdot\tilde{n}_i, \quad \Lambda_B = \mathrm{diag}(\bar{c}_i\,\mathrm{d}s_i), \quad S_B = \mathrm{diag}(\mathrm{d}s_i), \quad i\in\partial\Omega.$$

To cancel the term (IL) in (12), we require $\Gamma = -\varepsilon S_B\tilde{\beta}^{-1}$. Applying the condition to (12) yields

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|_P^2 + 2\varepsilon\|D_x\mathbf{u}\|_P^2 + 2\varepsilon\|D_y\mathbf{u}\|_P^2 &= -\mathbf{u}_B^{\mathrm{T}}(\Lambda_B + 2\varepsilon S_B\tilde{\beta}^{-1}\tilde{\alpha})\mathbf{u}_B + 2\varepsilon\mathbf{u}_B^{\mathrm{T}}S_B\tilde{\beta}^{-1}g\\ &= \sum_{i\in\partial\Omega}\left[-\left(\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon\right)u_i^2 + \frac{2\varepsilon}{\tilde{\beta}_{i,i}}u_ig_i\right]\mathrm{d}s_i\\ &= -\sum_{i\in\partial\Omega}\left(\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon\right)\left(u_i - \frac{\varepsilon}{\tilde{\beta}_{i,i}}\frac{1}{\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon}g_i\right)^2\mathrm{d}s_i\\ &\quad + \sum_{i\in\partial\Omega}\left(\frac{\varepsilon}{\tilde{\beta}_{i,i}}\right)^2\left(\frac{1}{\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon}\right)g_i^2\,\mathrm{d}s_i. \end{aligned} \tag{13}$$

We have proved the following proposition.

Proposition 3.1. *If* $\Gamma = -\varepsilon S_B\tilde{\beta}^{-1}$ *and*

$$\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon \geqslant 0, \quad i\in\partial\Omega, \tag{14}$$

are satisfied. The problem (10) *is strongly stable.*

Remark. When the solution can be estimated in terms of all types of data, the problem is called strongly stable, see [7] for more details.

Remark. The estimate (13) is completely similar to the continuous estimate (3). Condition (14) is completely similar to (4).

3.2. The finite difference method

For the finite difference approximations, the physical domain must be possible to smoothly transform to a rectangular computational domain (see Fig. 2). We start by transforming Eq. (1a) to curvilinear form. Note that $u_x = \xi_x u_\xi + \eta_x u_\eta$ and $u_y = \xi_y u_\xi + \eta_y u_\eta$, where we have introduced the transformation $x = x(\xi,\eta)$ and $y = y(\xi,\eta)$ and the metric relations,

$$J\xi_x = y_\eta, \quad J\xi_y = -x_\eta, \quad J\eta_x = -y_\xi, \quad J\eta_y = x_\xi, \quad J = x_\xi y_\eta - x_\eta y_\xi = (\xi_x\eta_y - \xi_y\eta_x)^{-1} \neq 0.$$

For simplicity we also introduce the notations,

$$\tilde{a} = aJ\xi_x + bJ\xi_y, \quad \tilde{b} = aJ\eta_x + bJ\eta_y, \quad \tilde{f} = J(\xi_x u_x + \xi_y u_y) = J(\nabla u \cdot \nabla\xi), \quad \tilde{g} = J(\eta_x u_x + \eta_y u_y) = J(\nabla u \cdot \nabla\eta).$$

It follows that $J(au_x + bu_y) = (\tilde{a}u)_\xi + (\tilde{b}u)_\eta$ and $J(u_{xx} + u_{yy}) = \tilde{f}_\xi + \tilde{g}_\eta$ since $\tilde{a}_\xi + \tilde{b}_\eta = 0$. Eq. (1a) transforms into

$$Ju_t + (\tilde{a}u)_\xi + (\tilde{b}u)_\eta = \varepsilon(\tilde{f}_\xi + \tilde{g}_\eta). \tag{15}$$

For reasons that will become obvious later, we split the terms $(\tilde{a}u)_\xi$ and $(\tilde{b}u)_\eta$ in (15) as (see [14])

$$(\tilde{a}u)_\xi = \frac{1}{2}[(\tilde{a}u)_\xi + \tilde{a}u_\xi + \tilde{a}_\xi u], \quad (\tilde{b}u)_\eta = \frac{1}{2}[(\tilde{b}u)_\eta + \tilde{b}u_\eta + \tilde{b}_\eta u].$$

The difference operators in the ξ and η directions on the right subdomain are denoted by $D_\xi = (P_\xi)^{-1}Q_\xi \otimes I_\eta$ and $D_\eta = I_\xi \otimes (P_\eta)^{-1}Q_\eta$, respectively. Note that the operators $(P_\xi)^{-1}Q_\xi$ and $(P_\eta)^{-1}Q_\eta$ are SBP operators since the matrices P_ξ and P_η are symmetric and positive definite and,

$$\begin{aligned} Q_\xi + (Q_\xi)^{\mathrm{T}} &= B_\xi = \mathrm{diag}([-1,0,\ldots 0,1]),\\ Q_\eta + (Q_\eta)^{\mathrm{T}} &= B_\eta = \mathrm{diag}([-1,0,\ldots 0,1]). \end{aligned} \tag{16}$$

In matrix formulation we have

$$\tilde{\xi}_x = \mathrm{diag}((\xi_x)_i), \quad \tilde{\xi}_y = \mathrm{diag}((\xi_y)_i), \quad \tilde{\eta}_x = \mathrm{diag}((\eta_x)_i), \quad \tilde{\eta}_y = \mathrm{diag}((\eta_y)_i), \quad \widetilde{A} = \mathrm{diag}(\tilde{a}_i),$$
$$\widetilde{B} = \mathrm{diag}(\tilde{b}_i), \quad \widetilde{F} = \mathrm{diag}(\tilde{f}_i), \quad \widetilde{G} = \mathrm{diag}(\tilde{g}_i), \quad \widetilde{J} = \mathrm{diag}(J_i).$$

In the curvilinear coordinate system, the finite difference approximation of u at the grid point (ξ_i,η_j) is a vector denoted $\mathbf{u}_{ij}$. We organize the solution in the global vector $\mathbf{u} = [\mathbf{u}_{11},\ldots,\mathbf{u}_{1l},\mathbf{u}_{21},\ldots,\mathbf{u}_{2l},\ldots,\mathbf{u}_{n1},\ldots,\mathbf{u}_{nl}]^{\mathrm{T}}$. $\mathbf{u}_\xi$, $\mathbf{u}_\eta$ are approximations of u_ξ, u_η and are approximated using the high-order accurate SBP operators for the first

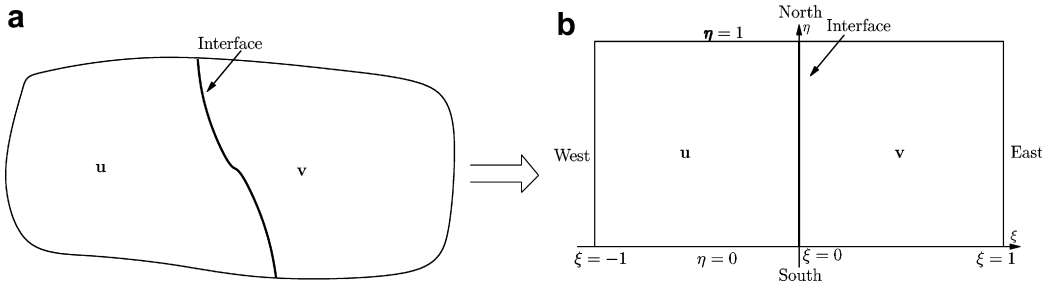

Fig. 2. (a) The physical domain and (b) the computational domain.

derivative that were constructed in [3,10]. Moreover, on the boundary we define $D_n\mathbf{u}$ to be the approximation of

$$\begin{aligned}\left(\frac{\partial u}{\partial n}\right)_i &= [(u_x,u_y)\cdot n]_i = [(\xi_x u_\xi+\eta_x u_\eta,\xi_y u_\xi+\eta_y u_\eta)\cdot n]_i\\ &\approx ((\tilde{\xi}_x D_\xi\mathbf{u}+\tilde{\eta}_x D_\eta\mathbf{u})_i,(\tilde{\xi}_y D_\xi\mathbf{u}+\tilde{\eta}_y D_\eta\mathbf{u})_i)\cdot\tilde{n}_i=(D_n\mathbf{u})_i,\quad i\in\partial\Omega,\end{aligned}\tag{17}$$

with

$$\begin{aligned}\tilde{n}_i &= \frac{(\mathrm{d}y_i,-\mathrm{d}x_i)}{\mathrm{d}s_i}=\frac{((y_\xi\,\mathrm{d}\xi+y_\eta\,\mathrm{d}\eta)_i,-(x_\xi\,\mathrm{d}\xi+x_\eta\,\mathrm{d}\eta)_i)}{\mathrm{d}s_i},\\ \mathrm{d}s_i &= \sqrt{\mathrm{d}x_i^2+\mathrm{d}y_i^2}=\sqrt{(x_\xi\,\mathrm{d}\xi+x_\eta\,\mathrm{d}\eta)_i^2+(y_\xi\,\mathrm{d}\xi+y_\eta\,\mathrm{d}_\eta)_i^2}.\end{aligned}\tag{18}$$

By using the notation above, a semi-discrete approximation of (1a) can be written,

$$\begin{aligned}&\widetilde{J}\mathbf{u}_t+\frac{1}{2}[D_\xi(\widetilde{A}\mathbf{u})+\widetilde{A}D_\xi\mathbf{u}+(I_\xi\otimes I_\eta)\widetilde{A}_\xi\mathbf{u}]+\frac{1}{2}[D_\eta(\widetilde{B}\mathbf{u})+\widetilde{B}D_\eta\mathbf{u}+(I_\xi\otimes I_\eta)\widetilde{B}_\eta\mathbf{u}]\\ &\quad=\varepsilon(D_\xi\widetilde{F}+D_\eta\widetilde{G})+[(P_\xi)^{-1}\otimes(P_\eta)^{-1}](E_B)^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B+\tilde{\beta}(D_n\mathbf{u})_B-g].\end{aligned}\tag{19}$$

Here E_B is a projection matrix which maps the values on the computational domain to the outer boundary, that is, $\mathbf{u}_B=E_B\mathbf{u}$ and $(D_n\mathbf{u})_B=E_B(D_n\mathbf{u})$. The boundary conditions have been introduced by using the penalty technique SAT, see [2,17,19].

The energy method leads to

$$\begin{aligned}&\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{J}\mathbf{u}_t+\frac{1}{2}[\mathbf{u}^{\mathrm{T}}(Q_\xi\otimes P_\eta)\widetilde{A}\mathbf{u}+\mathbf{u}^{\mathrm{T}}\widetilde{A}(Q_\xi\otimes P_\eta)\mathbf{u}+\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{A}_\xi\mathbf{u}]+\frac{1}{2}[\mathbf{u}^{\mathrm{T}}(P_\xi\otimes Q_\eta)\widetilde{B}\mathbf{u}\\ &\quad+\mathbf{u}^{\mathrm{T}}\widetilde{B}(P_\xi\otimes Q_\eta)\mathbf{u}+\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{B}_\eta\mathbf{u}]\\ &\quad=\varepsilon\mathbf{u}^{\mathrm{T}}(Q_\xi\otimes P_\eta)\widetilde{F}+\varepsilon\mathbf{u}^{\mathrm{T}}(P_\xi\otimes Q_\eta)\widetilde{G}+\mathbf{u}_B^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B+\tilde{\beta}(D_n\mathbf{u})_B-g].\end{aligned}\tag{20}$$

Remark. Notice that $(P_\xi\otimes P_\eta)\tilde{J}$ is a norm if P_ξ and P_η are diagonal, see Lemma 1 in [16].

Now we can make use of the splitting technique to obtain,

$$\frac{1}{2}[\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{A}_\xi\mathbf{u}+\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{B}_\eta\mathbf{u}]=\frac{1}{2}\mathbf{u}^{\mathrm{T}}(P_\xi\otimes P_\eta)(\widetilde{A}_\xi+\widetilde{B}_\eta)\mathbf{u}=0,$$

since $\widetilde{A}_\xi+\widetilde{B}_\eta=\mathrm{diag}((\tilde{a}_\xi+\tilde{b}_\eta)_i)=0$. We also need,

$$\begin{aligned}&\frac{1}{2}[\mathbf{u}^{\mathrm{T}}(Q_\xi\otimes P_\eta)\widetilde{A}\mathbf{u}+\mathbf{u}^{\mathrm{T}}\widetilde{A}(Q_\xi\otimes P_\eta)\mathbf{u}]=\frac{1}{2}\mathbf{u}^{\mathrm{T}}(B_\xi\otimes P_\eta)\widetilde{A}\mathbf{u},\\ &\frac{1}{2}[\mathbf{u}^{\mathrm{T}}(P_\xi\otimes Q_\eta)\widetilde{B}\mathbf{u}+\mathbf{u}^{\mathrm{T}}\widetilde{B}(P_\xi\otimes Q_\eta)\mathbf{u}]=\frac{1}{2}\mathbf{u}^{\mathrm{T}}(P_\xi\otimes B_\eta)\widetilde{B}\mathbf{u}.\end{aligned}$$

The viscous terms becomes,

$$\begin{aligned}\mathbf{u}^{\mathrm{T}}(Q_\xi\otimes P_\eta)\widetilde{F}+\mathbf{u}^{\mathrm{T}}(P_\xi\otimes Q_\eta)\widetilde{G}=&-\underbrace{[(D_\xi\mathbf{u})^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{F}+(D_\eta\mathbf{u})^{\mathrm{T}}(P_\xi\otimes P_\eta)\widetilde{G}]}_{(\mathrm{Diss})}+\mathbf{u}^{\mathrm{T}}(B_\xi\otimes P_\eta)\widetilde{F}\\ &+\mathbf{u}^{\mathrm{T}}(P_\xi\otimes B_\eta)\widetilde{G}.\end{aligned}\tag{21}$$

As was shown above we have,

$$\begin{aligned}\widetilde{F}&=\mathrm{diag}(\tilde{f})=\mathrm{diag}([J(u_x\xi_x+u_y\xi_y)]_i)=\widetilde{J}[(\tilde{\xi}_x^2+\tilde{\xi}_y^2)D_\xi\mathbf{u}+(\tilde{\eta}_x\tilde{\xi}_x+\tilde{\eta}_y\tilde{\xi}_y)D_\eta\mathbf{u}],\\ \widetilde{G}&=\mathrm{diag}(\tilde{g})=\mathrm{diag}([J(u_x\eta_x+u_y\eta_y)]_i)=\widetilde{J}[(\tilde{\xi}_x\tilde{\eta}_x+\tilde{\xi}_y\tilde{\eta}_y)D_\xi\mathbf{u}+(\tilde{\eta}_x^2+\tilde{\eta}_y^2)D_\eta\mathbf{u}].\end{aligned}$$

This implies that (Diss) in (21) becomes,

$$\begin{aligned}(\text{Diss}) &= (D_\xi \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{J}[(\tilde{\xi}_x^2+\tilde{\xi}_y^2)D_\xi \mathbf{u} + (\tilde{\eta}_x\tilde{\xi}_x+\tilde{\eta}_y\tilde{\xi}_y)D_\eta \mathbf{u}] + (D_\xi \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{J}[(\tilde{\xi}_x\tilde{\eta}_y+\tilde{\xi}_y\tilde{\eta}_y)D_\xi \mathbf{u}\\ &+ (\tilde{\eta}_x^2+\tilde{\eta}_x^2)D_\eta \mathbf{u}] = \underbrace{\begin{bmatrix} D_\xi \mathbf{u} \\ D_\eta \mathbf{u}\end{bmatrix}^{\mathrm{T}}}_{\mathbf{w}^{\mathrm{T}}} \underbrace{\begin{bmatrix} (P_\xi \otimes P_\eta)\widetilde{J}(\tilde{\xi}_x^2+\tilde{\xi}_y^2) & (P_\xi \otimes P_\eta)\widetilde{J}(\tilde{\xi}_x\tilde{\eta}_x+\tilde{\xi}_y\tilde{\eta}_y) \\ (P_\xi \otimes P_\eta)\widetilde{J}(\tilde{\xi}_x\tilde{\eta}_x+\tilde{\xi}_y\tilde{\eta}_y) & (P_\xi \otimes P_\eta)\widetilde{J}(\tilde{\eta}_x^2+\tilde{\eta}_y^2)\end{bmatrix}}_{H} \underbrace{\begin{bmatrix} D_\xi \mathbf{u} \\ D_\eta \mathbf{u}\end{bmatrix}}_{\mathbf{w}}. \end{aligned} \tag{22}$$

The following Lemma is proved in the Appendix.

Lemma 3.2. *The term* (Diss) *in Eqs.* (21) *and* (23) *is positive semi-definite.*

Via the previous analysis, Eq. (20) is rewritten as

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|^2_{P_\xi \otimes P_\eta \widetilde{J}} + 2\varepsilon(D_\xi \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{F} + 2\varepsilon(D_\eta \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{G}\\ &\quad = -\mathbf{u}^{\mathrm{T}}(B_\xi \otimes P_\eta)\widetilde{A}\mathbf{u} - \mathbf{u}^{\mathrm{T}}(P_\xi \otimes B_\eta)\widetilde{B}\mathbf{u} + 2\varepsilon \mathbf{u}^{\mathrm{T}}(B_\xi \otimes P_\eta)\widetilde{F} + 2\varepsilon \mathbf{u}^{\mathrm{T}}(P_\xi \otimes B_\eta)\widetilde{G} + 2\mathbf{u}_B^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B + \tilde{\beta}(D_n \mathbf{u})_B - g].\end{aligned} \tag{23}$$

Note that $\xi = \text{const.}$ at the West and East boundaries and that $\eta = \text{const.}$ at the South and North boundaries (see Fig. 2). We have

$$\begin{aligned}\tilde{n}_i &= \frac{((y_\eta)_i, -(x_\eta)_i)\mathrm{d}\eta}{\mathrm{d}s_i}, \quad \mathrm{d}s_i = \sqrt{(x_\eta)_i^2 + (y_\eta)_i^2}\,\mathrm{d}\eta, \quad i \in \text{West},\\ \tilde{n}_i &= \frac{((y_\xi)_i, -(x_\xi)_i)\mathrm{d}\xi}{\mathrm{d}s_i}, \quad \mathrm{d}s_i = \sqrt{(x_\xi)_i^2 + (y_\xi)_i^2}\,\mathrm{d}\xi, \quad i \in \text{South},\\ \tilde{n}_i &= \frac{((y_\eta)_i, -(x_\eta)_i)\mathrm{d}\eta}{\mathrm{d}s_i}, \quad \mathrm{d}s_i = \sqrt{(x_\eta)_i^2 + (y_\eta)_i^2}\,\mathrm{d}\eta, \quad i \in \text{East},\\ \tilde{n}_i &= \frac{((y_\xi)_i, -(x_\xi)_i)\mathrm{d}\xi}{\mathrm{d}s_i}, \quad \mathrm{d}s_i = \sqrt{(x_\xi)_i^2 + (y_\xi)_i^2}\,\mathrm{d}\xi, \quad i \in \text{North}.\end{aligned} \tag{24}$$

Consequently, the right-hand-side of (23) can be rewritten as

$$\mathbf{u}^{\mathrm{T}}(B_\xi \otimes P_\eta)\widetilde{A}\mathbf{u} + \mathbf{u}^{\mathrm{T}}(P_\xi \otimes B_\eta)\widetilde{B}\mathbf{u} = \mathbf{u}_B^{\mathrm{T}}\Lambda_B \mathbf{u}_B, \quad \mathbf{u}^{\mathrm{T}}(B_\xi \otimes P_\eta)\widetilde{F} + \mathbf{u}^{\mathrm{T}}(P_\xi \otimes B_\eta)\widetilde{G} = \mathbf{u}_B^{\mathrm{T}} S_B (D_n \mathbf{u})_B, \tag{25}$$

where $\Lambda_B = \mathrm{diag}(\bar{c}_i \overline{\mathrm{d}l}_i)$, $S_B = \mathrm{diag}(\overline{\mathrm{d}l}_i)$, $\bar{c}_i = (a, b)\cdot \tilde{n}_i$, and

$$\overline{\mathrm{d}l}_i = \begin{cases} -p_i^{\mathrm{W}}\,\mathrm{d}s_i, & \mathrm{d}\eta p_i^{\mathrm{W}} = \mathrm{diag}(P_\eta)_i, \quad i \in \text{West},\\ -p_i^{\mathrm{S}}\,\mathrm{d}s_i, & \mathrm{d}\xi p_i^{\mathrm{S}} = \mathrm{diag}(P_\xi)_i, \quad i \in \text{South},\\ p_i^{\mathrm{E}}\,\mathrm{d}s_i, & \mathrm{d}\eta p_i^{\mathrm{E}} = \mathrm{diag}(P_\eta)_i, \quad i \in \text{East},\\ p_i^{\mathrm{N}}\,\mathrm{d}s_i, & \mathrm{d}\xi p_i^{\mathrm{N}} = \mathrm{diag}(P_\xi)_i, \quad i \in \text{North}.\end{cases} \tag{26}$$

The relations (24)–(26) inserted in (23) yields

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|^2_{P_\xi \otimes P_\eta \widetilde{J}} + 2\varepsilon(D_\xi \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{F} + 2\varepsilon(D_\eta \mathbf{u})^{\mathrm{T}}(P_\xi \otimes P_\eta)\widetilde{G}\\ &\quad = \mathbf{u}_B^{\mathrm{T}}\Lambda_B \mathbf{u}_B + 2\varepsilon \mathbf{u}_B^{\mathrm{T}} S_B (D_n \mathbf{u})_B + 2\mathbf{u}_B^{\mathrm{T}}\Gamma[\tilde{\alpha}\mathbf{u}_B + \tilde{\beta}(D_n \mathbf{u})_B - g]\\ &\quad = \mathbf{u}_B^{\mathrm{T}}(\Lambda_B - 2\Gamma\tilde{\alpha})\mathbf{u}_B + \underbrace{\mathbf{u}_B^{\mathrm{T}}(2\varepsilon S_B + 2\Gamma\tilde{\beta})(D_n \mathbf{u})_B}_{\text{IR}} - 2\mathbf{u}_B^{\mathrm{T}}\Gamma g.\end{aligned} \tag{27}$$

To cancel the term (IR) in Eq. (27), we require $\Gamma = -\varepsilon S_B \tilde{\beta}^{-1}$. By employing the same technique as in Section 3.1, we prove the following proposition.

Proposition 3.3. *If* $\Gamma = -\varepsilon S_B \tilde{\beta}^{-1}$ *and*

$$\bar{c}_i + \frac{2\tilde{\alpha}_{i,i}}{\tilde{\beta}_{i,i}}\varepsilon \geqslant 0, \quad i \in \partial\Omega, \tag{28}$$

are satisfied. The problem (19) *is strongly stable.*

Remark. By inserting $\Gamma = -\varepsilon S_B \tilde{\beta}^{-1}$ and (28) into (27), we obtain an estimate that is completely similar to (3) and (13). Note also that (28) is completely similar to (4).

4. Multiple domains and interface conditions

Without loss of generality, we consider a computational domain which consists of two subdomains. The unknown on the left subdomain is denoted by u and on the right subdomain by v, respectively. The same technique described in the previous section is used here to discretize both u and v. The superscripts $^{\mathrm{L}}$ and $^{\mathrm{R}}$ are added in order to identify the left and right subdomains.

Since the outer boundary treatment has been already discussed, we will only focus on the interface treatment. The coupling of $\mathbf{u}$ and $\mathbf{v}$ as well as the first derivatives $D_1^{\mathrm{L}}\mathbf{u}$ and $D_1^{\mathrm{R}}\mathbf{v}$ at the interface will be treated by using the various forms of the SAT technique.

4.1. The finite volume method

A semi-discrete approximation of (1) on the left part of the computational domain can be written,

$$\begin{aligned}\mathbf{u}_t + aD_x^{\mathrm{L}}\mathbf{u} + bD_y^{\mathrm{L}}\mathbf{u} = \varepsilon\left(D_x^{\mathrm{L}}D_x^{\mathrm{L}}\mathbf{u} + D_y^{\mathrm{L}}D_y^{\mathrm{L}}\mathbf{u}\right) + \left(P^{\mathrm{L}}\right)^{-1}\left(E_I^{\mathrm{L}}\right)^{\mathrm{T}}F_1^{\mathrm{L}}(\mathbf{u}_I - \mathbf{v}_I) + \left(P^{\mathrm{L}}\right)^{-1}\left(E_I^{\mathrm{L}}\right)^{\mathrm{T}}F_2^{\mathrm{L}}\left[\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I\right.\\ \left.+\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I\right] + \left(P^{\mathrm{L}}\right)^{-1}\left(D_n^{\mathrm{L}}\right)^{\mathrm{T}}\left(E_I^{\mathrm{L}}\right)^{\mathrm{T}}F_3^{\mathrm{L}}(\mathbf{u}_I - \mathbf{v}_I) + \mathrm{Pen}_1^{\mathrm{L}},\end{aligned} \tag{29}$$

where $\mathrm{Pen}_1^{\mathrm{L}}$ is the penalty term that imposes the outer boundary conditions weakly. The other three penalty terms on the right-hand-side will be used to couple the left subdomain calculation to the right subdomain calculation. Note that $\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I + \left(D_n^{\mathrm{R}}\mathbf{v}\right)_I$ is small and proportional to the truncation error. $\mathbf{u}_I$ and $\mathbf{v}_I$ are vectors which represent $\mathbf{u}$ and $\mathbf{v}$ ($\mathbf{v}$ is the discrete finite difference solution that will be presented below) on the interface, respectively. E_I^{L} is a projection matrix which maps the values on the left computational domain to the interface, that is, $\mathbf{u}_I = E_I^{\mathrm{L}}\mathbf{u}$ and $\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I = E_I^{\mathrm{L}}\left(D_n^{\mathrm{L}}\mathbf{u}\right)$. F_1^{L}, F_2^{L} and F_3^{L} are penalty matrices that will be determined below by stability requirements. $D_n^{\mathrm{R}}\mathbf{v}$ is an approximation of $\partial v/\partial n$ which will be derived in the next section.

By multiplying (29) with $\mathbf{u}^{\mathrm{T}}P^{\mathrm{L}}$ we obtain,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|_{P^{\mathrm{L}}}^2 + 2\varepsilon\left(\|D_x^{\mathrm{L}}\mathbf{u}\|_{P^{\mathrm{L}}}^2 + \|D_y^{\mathrm{L}}\mathbf{u}\|_{P^{\mathrm{L}}}^2\right) = -\mathbf{u}_I^{\mathrm{T}}\Lambda_I^{\mathrm{L}}\mathbf{u}_I + 2\varepsilon\mathbf{u}_I^{\mathrm{T}}S_I^{\mathrm{L}}\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I + 2\mathbf{u}_I^{\mathrm{T}}F_1^{\mathrm{L}}(\mathbf{u}_I - \mathbf{v}_I)\\ + 2\mathbf{u}_I^{\mathrm{T}}F_2^{\mathrm{L}}\left[\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I + \left(D_n^{\mathrm{R}}\mathbf{v}\right)_I\right] + 2\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I^{\mathrm{T}}F_3^{\mathrm{L}}(\mathbf{u}_I - \mathbf{v}_I) + \mathrm{BT}^{\mathrm{L}}.\end{aligned} \tag{30}$$

where BT^{L} collects the outer boundary terms (see Section 3.1) and

$$\Lambda_I^{\mathrm{L}} = \mathrm{diag}[(a,b)\cdot\tilde{n}_i^{\mathrm{L}}\,\mathrm{d}s_i^{\mathrm{L}}], \quad S_I^{\mathrm{L}} = \mathrm{diag}(\mathrm{d}s_i^{\mathrm{L}}), \quad i \in \text{Interface}.$$

4.2. The finite difference method

A semi-discrete approximation of (1) on the right subdomain can be written,

$$\begin{aligned}\widetilde{J}\mathbf{v}_t + \frac{1}{2}\left[D_\xi^{\mathrm{R}}(\widetilde{A}\mathbf{v}) + \widetilde{A}D_\xi^{\mathrm{R}}\mathbf{v} + (I_\xi\otimes I_\eta)\widetilde{A}_\xi\mathbf{v}\right] + \frac{1}{2}\left[D_\eta^{\mathrm{R}}(\widetilde{B}\mathbf{v}) + \widetilde{B}D_\eta^{\mathrm{R}}\mathbf{v} + (I_\xi\otimes I_\eta)\widetilde{B}_\eta\mathbf{v}\right]\\ = \varepsilon D_\xi^{\mathrm{R}}\widetilde{F} + \varepsilon D_\eta^{\mathrm{R}}\widetilde{G} + \left[\left(P_\xi^{\mathrm{R}}\right)^{-1}\otimes\left(P_\eta^{\mathrm{R}}\right)^{-1}\right]\left(E_I^{\mathrm{R}}\right)^{\mathrm{T}}F_1^{\mathrm{R}}(\mathbf{v}_I - \mathbf{u}_I) + \left[\left(P_\xi^{\mathrm{R}}\right)^{-1}\otimes\left(P_\eta^{\mathrm{R}}\right)^{-1}\right]\left(E_I^{\mathrm{R}}\right)^{\mathrm{T}}F_2^{\mathrm{R}}\left[\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I\right.\\ \left.+\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I\right] + \left[\left(P_\xi^{\mathrm{R}}\right)^{-1}\otimes\left(P_\eta^{\mathrm{R}}\right)^{-1}\right]\left(D_n^{\mathrm{R}}\right)^{\mathrm{T}}\left(E_I^{\mathrm{R}}\right)^{\mathrm{T}}F_3^{\mathrm{R}}(\mathbf{v}_I - \mathbf{u}_I) + \mathrm{Pen}^{\mathrm{R}}.\end{aligned} \tag{31}$$

Here $\mathrm{Pen}^{\mathrm{R}}$ is the penalty term for the outer boundary conditions on the right part of the computational domain. E_I^{R} is a projection matrix which maps the values on the right computational domain to the interface, that is, $\mathbf{v}_I = E_I^{\mathrm{R}}\mathbf{v}$ and $\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I = E_I^{\mathrm{R}}\left(D_n^{\mathrm{R}}\mathbf{v}\right)$.

The energy method leads to

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}\|^2_{P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\widetilde{J}} + 2\varepsilon\left(D_\xi^{\mathrm{R}}\mathbf{v}\right)^{\mathrm{T}}\left(P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\right)\widetilde{F} + 2\varepsilon\left(D_\eta^{\mathrm{R}}\mathbf{v}\right)^{\mathrm{T}}\left(P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\right)\widetilde{G}\\ &= \mathbf{v}_I^{\mathrm{T}}P_\eta^{\mathrm{R}}\left(\widetilde{A}\mathbf{v}\right) + 2\varepsilon\mathbf{v}_I^{\mathrm{T}}P_\eta^{\mathrm{R}}S_I^{\mathrm{R}}\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I + 2\mathbf{v}_I^{\mathrm{T}}F_1^{\mathrm{R}}(\mathbf{v}_I-\mathbf{u}_I) + 2\mathbf{v}_I^{\mathrm{T}}F_2^{\mathrm{R}}\left[\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I + \left(D_n^{\mathrm{L}}\mathbf{u}\right)_I\right] + 2F_3^{\mathrm{R}}\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I^{\mathrm{T}}(\mathbf{v}_I-\mathbf{u}_I) + \mathrm{BT}^{\mathrm{R}},\end{aligned} \tag{32}$$

where BT^{R} collects the outer boundary terms (for details see Section 3.2) and

$$\Lambda_I^{\mathrm{R}} = \mathrm{diag}[(p_i^{\mathrm{R}}\,\mathrm{d}\eta)\tilde{a}_i], \quad S_I^{\mathrm{R}} = \mathrm{diag}(p_i^{\mathrm{R}}\,\mathrm{d}s_i^{\mathrm{R}}), \quad p_i^{\mathrm{R}}\,\mathrm{d}\eta = \mathrm{diag}\left(P_\eta^{\mathrm{R}}\right)_i, \quad i \in \text{Interface}.$$

4.3. Stable interface treatment

Combining (30) and (32) we have

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{u}\|^2_{P^{\mathrm{L}}} + \frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}\|^2_{P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\widetilde{J}} + 2\varepsilon\|D_x^{\mathrm{L}}\mathbf{u}\|^2_{P_L} + 2\varepsilon\|D_y^{\mathrm{L}}\mathbf{u}\|^2_{P_L} + 2\varepsilon\left(D_\xi^{\mathrm{R}}\mathbf{v}\right)^{\mathrm{T}}\left(P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\right)\widetilde{F} + 2\varepsilon\left(D_\eta^{\mathrm{R}}\mathbf{v}\right)^{\mathrm{T}}\left(P_\xi^{\mathrm{R}}\otimes P_\eta^{\mathrm{R}}\right)\widetilde{G}\\ &\quad= -\mathbf{u}_I^{\mathrm{T}}\Lambda_I^{\mathrm{L}}\mathbf{u}_I + 2\varepsilon\mathbf{u}_I^{\mathrm{T}}S_I^{\mathrm{L}}\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I + 2\mathbf{u}_I^{\mathrm{T}}F_1^{\mathrm{L}}(\mathbf{u}_I-\mathbf{v}_I) + 2\mathbf{u}_I^{\mathrm{T}}F_2^{\mathrm{L}}\left[\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I + \left(D_n^{\mathrm{R}}\mathbf{v}\right)_I\right] + 2\left(D_n^{\mathrm{L}}\mathbf{u}\right)_I^{\mathrm{T}}F_3^{\mathrm{L}}(\mathbf{u}_I-\mathbf{v}_I)\\ &\qquad+ \mathrm{BT}^{\mathrm{L}} + \mathbf{v}_I^{\mathrm{T}}\Lambda_I^{\mathrm{R}}\mathbf{v} + 2\varepsilon\mathbf{v}_I^{\mathrm{T}}S_I^{\mathrm{R}}\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I + 2\mathbf{v}_I^{\mathrm{T}}F_1^{\mathrm{R}}(\mathbf{v}_I-\mathbf{u}_I) + 2\mathbf{v}_I^{\mathrm{T}}F_2^{\mathrm{R}}\left[\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I + \left(D_n^{\mathrm{L}}\mathbf{u}\right)_I\right]\\ &\qquad+ 2\left(D_n^{\mathrm{R}}\mathbf{v}\right)_I^{\mathrm{T}}F_3^{\mathrm{R}}(\mathbf{v}_I-\mathbf{u}_I) + \mathrm{BT}^{\mathrm{R}} = \mathbf{x}_I^{\mathrm{T}}M_I\mathbf{x}_I + \mathrm{BT}^{\mathrm{L}} + \mathrm{BT}^{\mathrm{R}},\end{aligned} \tag{33}$$

where $M_I = M_I' + M_I''$ and

$$\mathbf{x}_I = \begin{bmatrix}\mathbf{u}_I\\ \mathbf{v}_I\\ \left(D_n^{\mathrm{L}}\mathbf{u}\right)_I\\ \left(D_n^{\mathrm{R}}\mathbf{v}\right)_I\end{bmatrix}, \quad M_I' = \begin{bmatrix}-\Lambda_I^{\mathrm{L}} + 2F_1^{\mathrm{L}} & -F_1^{\mathrm{L}} - F_1^{\mathrm{R}} & 0 & 0\\ -F_1^{\mathrm{L}} - F_1^{\mathrm{R}} & \Lambda_I^{\mathrm{R}} + 2F_1^{\mathrm{R}} & 0 & 0\\ 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0\end{bmatrix},$$

$$M_I'' = \begin{bmatrix}0 & 0 & F_2^{\mathrm{L}} + F_3^{\mathrm{L}} + \varepsilon S_I^{\mathrm{L}} & F_2^{\mathrm{L}} - F_3^{\mathrm{R}}\\ 0 & 0 & -F_3^{\mathrm{L}} + F_2^{\mathrm{R}} & F_2^{\mathrm{R}} + F_3^{\mathrm{R}} + \varepsilon S_I^{\mathrm{R}}\\ F_2^{\mathrm{L}} + F_3^{\mathrm{L}} + \varepsilon S_I^{\mathrm{L}} & -F_3^{\mathrm{L}} + F_2^{\mathrm{R}} & 0 & 0\\ F_2^{\mathrm{L}} - F_3^{\mathrm{R}} & F_2^{\mathrm{R}} + F_3^{\mathrm{R}} + \varepsilon S_I^{\mathrm{R}} & 0 & 0\end{bmatrix}.$$

A sufficient condition for M_I to be a negative semi-definite matrix is that M_I' is a negative semi-definite matrix and $M_I'' = 0$. If the conditions

$$\Lambda_I^{\mathrm{L}} = \Lambda_I^{\mathrm{R}}, \quad F_1^{\mathrm{L}} \leqslant \Lambda_I^{\mathrm{R}}/2; \quad F_1^{\mathrm{R}} = F_1^{\mathrm{L}} - \Lambda_I^{\mathrm{R}}, \tag{34}$$

are satisfied, M_I' is a negative semi-definite matrix. Moreover, the relations

$$S_I^{\mathrm{L}} = S_I^{\mathrm{R}}, \quad F_2^{\mathrm{L}} + F_2^{\mathrm{R}} + \varepsilon S_I^{\mathrm{L}} = 0, \quad F_3^{\mathrm{L}} = F_2^{\mathrm{R}}, \quad F_2^{\mathrm{L}} = F_3^{\mathrm{R}}, \tag{35}$$

lead to $M_I'' = 0$.

We have now proved the main proposition of this paper.

Proposition 4.1. *The hybrid method (29) and (31) have a stable interface treatment if the conditions (34) and (35) hold.*

Remark. It can be shown that the interface treatment is conservative if the conditions

$$\Lambda_I^{\mathrm{L}} = \Lambda_I^{\mathrm{R}}, \quad F_1^{\mathrm{R}} = F_1^{\mathrm{L}} - \Lambda_I^{\mathrm{R}}, \tag{36}$$

$$S_I^{\mathrm{L}} = S_I^{\mathrm{R}}, \quad F_2^{\mathrm{L}} + F_2^{\mathrm{R}} + \varepsilon S_I^{\mathrm{L}} = 0, \tag{37}$$

$$F_3^{\mathrm{L}} = F_3^{\mathrm{R}}, \tag{38}$$

holds. Note that conditions (36) and (37) are a subset of the stability conditions (34) and (35) while (38) is a new condition. If (38) is used, we get $F_2^{\mathrm{L}} = F_3^{\mathrm{L}} = F_2^{\mathrm{R}} = F_3^{\mathrm{R}} = -\varepsilon S_I^{\mathrm{L}}/2 = -\varepsilon S_I^{\mathrm{R}}/2$.

4.4. Realizing the stability conditions

The specific SBP operators that are based on diagonal norms are given in [10,20]. The standard second-, fourth- and sixth-order diagonal norm are

$$\mathrm{d}\eta \cdot \mathrm{diag}\left(\frac{1}{2}, 1, 1, \ldots,\right), \tag{39}$$

$$\mathrm{d}\eta \cdot \mathrm{diag}\left(\frac{17}{48}, \frac{59}{48}, \frac{43}{48}, \frac{49}{48}, 1, 1, \ldots,\right), \tag{40}$$

$$\mathrm{d}\eta \cdot \mathrm{diag}\left(\frac{13,649}{43,200}, \frac{12,013}{8640}, \frac{2711}{4320}, \frac{5359}{4320}, \frac{7877}{8640}, \frac{43,801}{43,200}, 1, 1, \ldots,\right), \tag{41}$$

respectively.

Recall that for $i \in$ Interface we have,

$$\Lambda_I^{\mathrm{L}} = \mathrm{diag}[(a,b)\cdot(\mathrm{d}y_i, -\mathrm{d}x_i)], \quad \Lambda_I^{\mathrm{R}} = \mathrm{diag}[p_i^{\mathrm{R}}\,\mathrm{d}\eta(a,b)\cdot((y_\eta)_i, -(x_\eta)_i)], \tag{42}$$

$$S_I^{\mathrm{L}} = \mathrm{diag}\left(\sqrt{\mathrm{d}x_i^2 + \mathrm{d}y_i^2}\right), \quad S_I^{\mathrm{R}} = \mathrm{diag}\left(p_i^{\mathrm{R}}\,\mathrm{d}\eta\sqrt{(x_\eta)_i^2 + (y_\eta)_i^2}\right). \tag{43}$$

A sufficient condition for obtaining $\Lambda_I^{\mathrm{L}} = \Lambda_I^{\mathrm{R}}$ and $S_I^{\mathrm{L}} = S_I^{\mathrm{R}}$ at the interface is

$$\mathrm{d}x_i = p_i^{\mathrm{R}}\,\mathrm{d}\eta(x_\eta)_i, \quad \mathrm{d}y_i = p_i^{\mathrm{R}}\,\mathrm{d}\eta(y_\eta)_i, \quad i \in \text{Interface}. \tag{44}$$

Denote the discretization points at the interface by $0, 1, \ldots, N$, where 0 is the start point and N is the end point. Since the vertices of each old dual grid close to the interface consist of the center of the triangles and the mid-point of the edge at the interface, we have

$$\mathrm{d}x_i = \begin{cases} \frac{x_1 - x_0}{2} & i = 0, \\ \frac{x_N - x_{N-1}}{2} & i = N, \\ \frac{x_{i+1} - x_{i-1}}{2} & \text{otherwise}, \end{cases} \qquad \mathrm{d}y_i = \begin{cases} \frac{y_1 - y_0}{2} & i = 0, \\ \frac{y_N - y_{N-1}}{2} & i = N, \\ \frac{y_{i+1} - y_{i-1}}{2} & \text{otherwise}, \end{cases} \tag{45}$$

When we use the second-order accurate finite difference method to compute the metric coefficients on the right subdomain we get,

$$(x_\eta)_i = \begin{cases} \frac{x_1 - x_0}{\mathrm{d}\eta} & i = 0, \\ \frac{x_N - x_{N-1}}{\mathrm{d}\eta} & i = N, \\ \frac{x_{i+1} - x_{i-1}}{2\mathrm{d}\eta} & \text{otherwise}, \end{cases} \qquad (y_\eta)_i = \begin{cases} \frac{y_1 - y_0}{\mathrm{d}\eta} & i = 0, \\ \frac{y_N - y_{N-1}}{\mathrm{d}\eta} & i = N, \\ \frac{y_{i+1} - y_{i-1}}{2\mathrm{d}\eta} & \text{otherwise}. \end{cases} \tag{46}$$

By multiplying (46) with the standard second order diagonal norm we exactly obtain (45) and consequently (44) is satisfied automatically.

However, the relations $\Lambda_I^{\mathrm{L}} = \Lambda_I^{\mathrm{R}}$ and $S_I^{\mathrm{L}} = S_I^{\mathrm{R}}$ are not automatically satisfied when curvilinear interfaces or high-order SBP operators are used. We need to modify the control volume for the UFVM to guarantee the conditions (34) and (35). To do this, we must move the position of the vertex $i \in$ interface at the interface, which is determined by (44). Relation (44) should be understood as follows: adjust the left-hand-side (that produces the dual grid) to the given value of the right-hand-side.

We take the following example and show how to deal with the interface for high-order SBP operators. Let us choose a curved interface (see Fig. 3) of the form

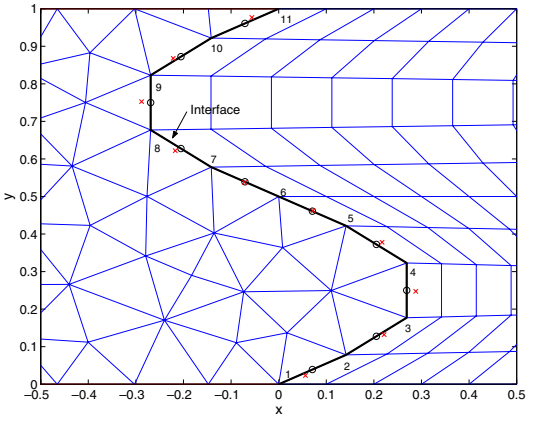

Fig. 3. The black 'o' represents the midpoints of the interface and the red 'x' represents the new modified dual grid points.

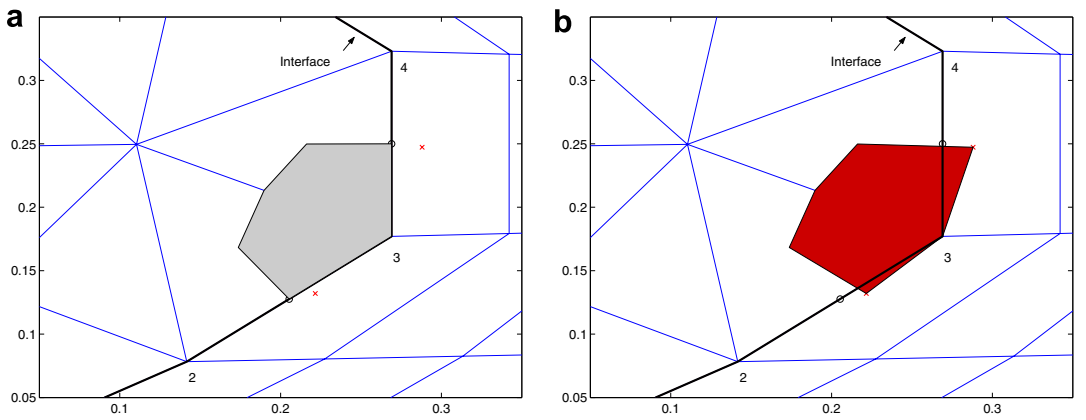

Fig. 4. The control volume connected to point 3 at the interface: (a) the original control volume and (b) the modified control volume.

$$x = x(0,\eta) = 0.3\sin(2\pi\eta), \quad y = y(0,\eta) = \eta, \quad 0 \leqslant \eta \leqslant 1.$$

The interface is discretized using 11 points and each line segment has equal length. If we use a fourth-order accurate SBP operator to approximate the first derivative in Eq. (44), the new modified dual grid points will be located as in Fig. 3. Instead of the original control volumes (see Fig. 4), the new control volumes (see Fig. 4) should be used in order to guarantee the stability of the hybrid scheme.

5. Numerical calculations

The model problem tested below is written

$$u_t + au_x + bu_y = \varepsilon(u_{xx} + u_{yy}) + F, \tag{47}$$

with suitable initial data and boundary data. F is the forcing function. In the test we use $a = 1$, $b = 1$ and $\varepsilon = 0.1$. In order to estimate the accuracy of the schemes, an exact solution $u = \sin(2\pi(x + y - 2t))$ has been chosen. The initial data, boundary data and the forcing function F are adjusted to fit the exact solution.

To test the efficiency and accuracy of these schemes, we define the rate of convergence, q, on the computational domain as

$$q = \frac{\log_{10}(\|u - v^{(1)}\|_2/\|u - v^{(2)}\|_2)}{\log_{10}\left(\sqrt{N^{(1)}}/\sqrt{N^{(2)}}\right)},$$

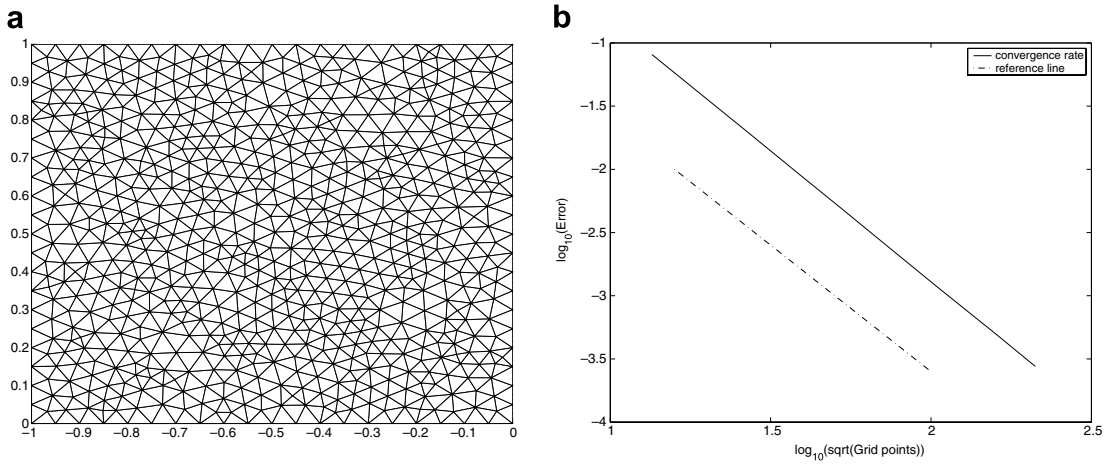

Fig. 5. (a) Unstructured mesh, 704 nodes and (b) convergence rate.

Table 1
Convergence rates of approximations to $u_t + u_x + u_y = 0.1(u_{xx} + u_{yy})$ on a single domain $[0,0] \times [1,1]$ by using the UFVM and HOFDMs

Points	UFVM		HOFDM (2nd)		HOFDM (3rd)		HOFDM (4th)	
	$\log_{10}$-Err	q	$\log_{10}$-Err	q	$\log_{10}$-Err	q	$\log_{10}$-Err	q
9×9	−0.99	–	−0.99	–	−1.48	–	–	–
17×17	−1.57	2.10	−1.57	2.10	−2.30	2.97	−2.25	–
33×33	−2.17	2.06	−2.17	2.06	−3.16	2.98	−3.34	3.78
65×65	−2.76	2.03	−2.76	2.03	−4.04	3.01	−4.47	3.86
129×129	−3.36	2.02	−3.36	2.02	−4.94	3.01	−5.64	3.93
257×257	−3.97	2.00	−3.97	2.00	−5.83	2.99	−6.83	3.96

Cartesian meshes are used.

where u is the exact solution. $v^{(1)}$ and $v^{(2)}$ are the corresponding numerical solutions on meshes with $N^{(1)}$ and $N^{(2)}$ nodes (including boundary nodes), respectively. We use the classical fourth-order Runge–Kutta method for the time integration. A small time-step is used to minimize the temporal error. We measure the error at $t = 1.0$.

5.1. Single domains and basic accuracy

We start by studying the accuracy of the UFVM on unstructured triangulated meshes (see Fig. 5). The convergence rates are presented in Fig. 5. Due to symmetry of the triangle meshes, second order accuracy is obtained.

Table 1 shows the convergence rates for both UFVM and HOFDM on a Cartesian meshes. The nodes of the Cartesian meshes (see Fig. 5) are refined from 81 to 66,049. The convergence rates for the schemes with interior accuracy of order 2, 4, 6 and boundary accuracy of order 1, 2, 3 are 2, 3, 4th order as shown in [23]. Note that the same error are obtained by using the UFVM and the second order HOFDM. This shows that the UFVM and the second order HOFDM are identical schemes on Cartesian meshes. Other test, not shown here, confirm that the correct convergence rate for the HOFDM on stretched and curvilinear meshes is obtained.

5.2. Multiple domains

In this section, we will illustrate the stability and efficiency of the hybrid scheme on multiple domains. The testing is processed as follows:

(1) Applying the UFVM on an unstructured mesh in all subdomains.

(2) Using the UFVM on the same mesh in a subdomain and the HOFDM on structured mesh(es) in the other subdomain(s).
(3) Adjusting the number of grid points in the subdomain until we obtain a similar L_2-error in all subdomains.

First we calculate on two subdomains with a linear interface at $x = 0$ (see Fig. 6). Table 2 shows the convergence rate of UFVM and second and fourth order accurate HOFDM. The convergence rate for the UFVM is 2 on unstructured symmetrical meshes. The $\log_{10}L_2$-error is -3.30 for the UFVM on the finest mesh with 50,138 points. We only need a mesh with 28,852 points for the hybrid method (UFVM + HOFDM (4th)) to obtain the same error level.

Next, we test the hybrid method on two subdomains with a smooth curved interface (see Fig. 6). In the result shown in Table 3, we see that in the fourth order case the hybrid scheme is efficient since only one fifth of the nodes are required for the HOFDM. The error levels are almost same as with a linear interface. The solution and the error are presented in Fig. 7. The wave propagates from left to the right via the curved interface without reflection. We conclude that the curved interface does not introduce more error and reflections compared with the linear interface.

Next we will test the hybrid schemes on a computational domain $[-1,1]\times[-1,1]$ with four subdomains (see Fig. 8). On the subdomain $[-1,0]\times[-1,0]$ excluding an ellipse, the UFVM was used. On the three other subdomains, the HOFDM was used. The finite difference and the finite volume solutions are co-located at the interfaces $y = 0$ and $x = 0$. Table 4 shows the convergence rate by using the hybrid scheme. The solution and error are shown in Fig. 9. The efficiency of the hybrid scheme with the fourth order HOFDM is clearly seen.

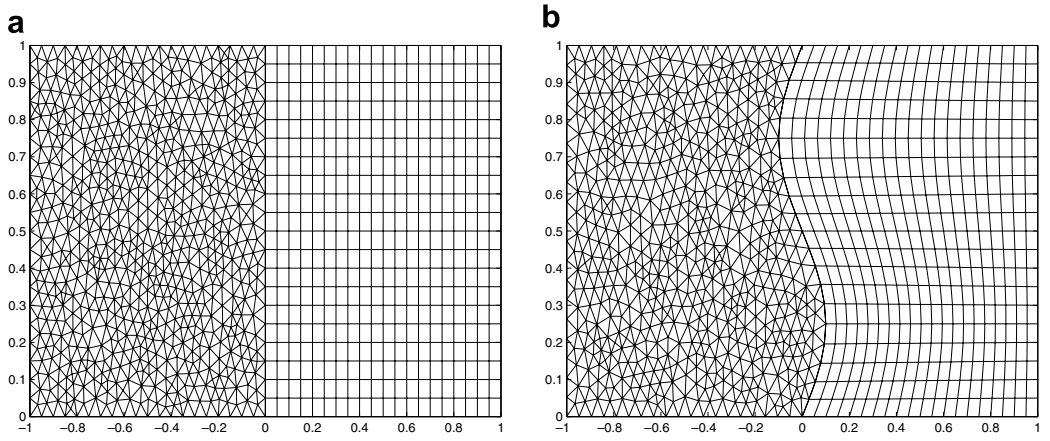

Fig. 6. Hybrid mesh with two subdomains: (a) with a linear interface and (b) with a curved interface.

Table 2
Convergence rates of approximations to $u_t + u_x + u_y = 0.1(u_{xx} + u_{yy})$ on two subdomains with a linear interface

UFVM (whole domain)			Hybrid (UFVM + HOFDM (2nd))			Hybrid (UFVM + HOFDM (4th))		
Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q
360	−1.09	–	292 (182 + 110)	−1.09	–	–	–	–
1425	−1.70	2.06	1124 (704 + 420)	−1.69	2.05	977 (704 + 273)	−1.73	–
3133	−2.07	2.07	2537 (1607 + 930)	−2.07	2.16	2072 (1607 + 465)	−2.07	2.06
5588	−2.32	2.06	4447 (2807 + 1640)	−2.32	2.05	3545 (1807 + 738)	−2.32	2.13
8779	−2.52	2.09	6907 (4357 + 2550)	−2.52	2.08	5428 (4357 + 1071)	−2.53	2.26
22,389	−2.94	2.07	17,619 (11,139 + 6480)	−2.94	2.06	13,164 (11,139 + 1863)	−2.93	2.09
50,138	−3.30	2.03	39,621 (25,101 + 14,520)	−3.30	2.03	28,852 (25,101 + 3751)	−3.30	2.15

UFVM is used on the left domain and HOFDM is used on the right domain.

Table 3
Convergence rates of approximations to $u_t + u_x + u_y = 0.1(u_{xx} + u_{yy})$ on two subdomains with a curvilinear interface

UFVM			Hybrid (UFVM + HOFDM (2nd))			Hybrid (UFVM + HOFDM (4th))		
Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q
360	−1.09	–	271 (160 + 110)	−1.00	–	–	–	–
1425	−1.70	2.06	1017 (597 + 420)	−1.62	2.16	870 (597 + 273)	−1.64	–
3133	−2.07	2.07	2236 (1306 + 930)	−1.98	2.14	1771 (1306 + 465)	−1.99	2.26
5588	−2.32	2.06	3942 (2302 + 1640)	−2.24	2.10	3040 (2302 + 738)	−2.26	2.26
8779	−2.52	2.09	6217 (3667 + 2550)	−2.43	1.87	4738 (3667 + 1071)	−2.44	1.95
22,380	−2.94	2.07	15,709 (9229 + 6480)	−2.86	2.13	11,092 (9229 + 1863)	−2.85	2.18
50,138	−3.30	2.03	35,226 (20,706 + 14,520)	−3.21	2.03	24,457 (20,706 + 3751)	−3.22	2.19

UFVM is used on the left domain and HOFDM is used on the right domain.

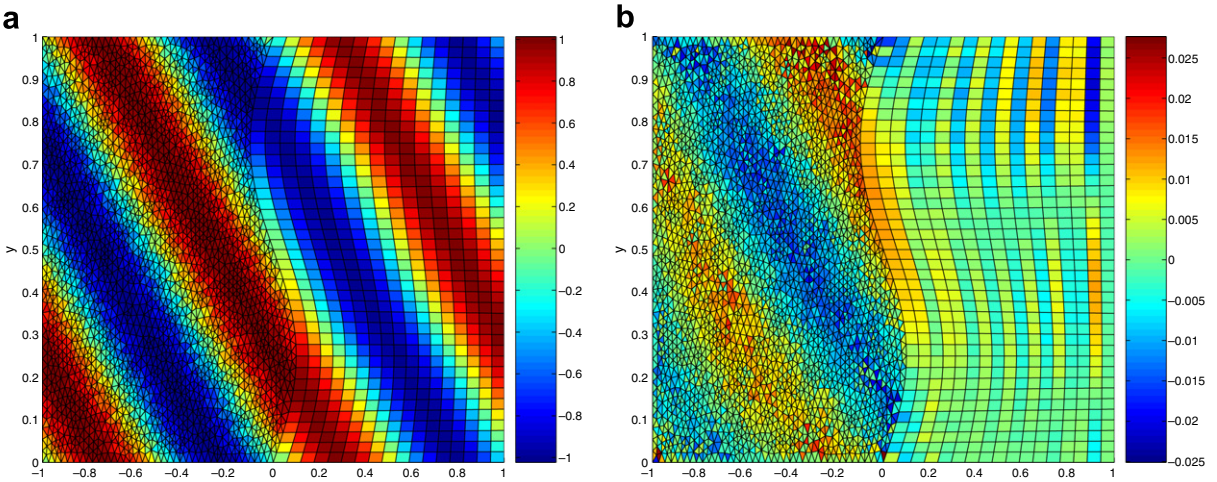

Fig. 7. $\log_{10}(L_2\text{-error}) = -2.22$ on the left subdomain with 2302 nodes and $\log_{10}(L_2\text{-error}) = -2.39$ with 738 nodes for $u_t + u_x + u_y = 0.1(u_{xx} + u_{yy})$. A curved interface is used: (a) solution at $T = 1.0$ and (b) error at $T = 1.0$.

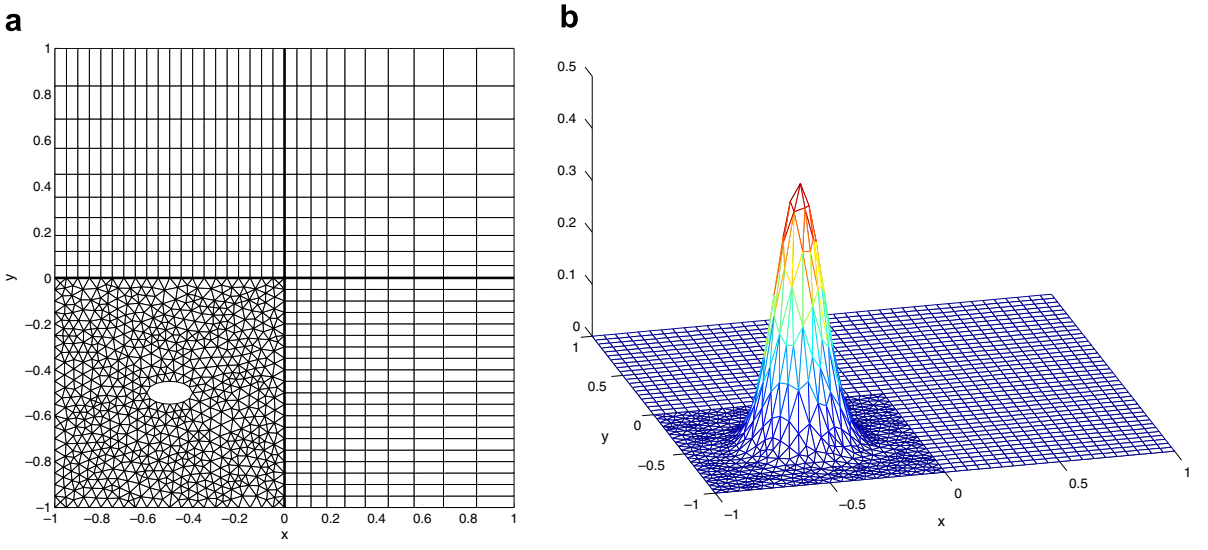

Fig. 8. (a) A hybrid mesh with four subdomains and (b) initial condition.

Table 4
Convergence rates of approximations to $u_t + u_x + u_y = 0.1(u_{xx} + u_{yy})$ on four subdomains

UFVM (whole domain)			Hybrid (UFVM + HOFDM (2nd))			Hybrid (UFVM + HOFDM (4th))		
Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q	Points	$\log_{10}$-Err	q
733	−1.09	–	550 (187 + 121 + 121 + 121)	−1.11	–	–	–	–
2809	−1.70	2.09	2020 (697 + 441 + 441 + 441)	−1.71	2.14	1412 (697 + 273 + 273 + 169)	−1.75	–
6389	−2.07	2.07	4451 (1568 + 961 + 961 + 961)	−2.08	2.15	2723 (1568 + 465 + 465 + 225)	−2.09	2.43
11,205	−2.32	2.05	7826 (2783 + 1681 + 1681 + 1681)	−2.34	2.07	4538 (2738 + 738 + 738 + 324)	−2.35	2.22
17,424	−2.52	2.08	12,156 (4353 + 2601 + 2601 + 2601)	−2.54	2.16	6936 (4353 + 1071 + 1071 + 441)	−2.56	2.43
44,447	−2.94	2.06	30,721 (11,038 + 6561 + 6561 + 6561)	−2.96	2.07	15,285 (11,030 + 1863 + 1863 + 529)	−2.96	2.21
99,923	−3.30	2.04	68,405 (24,482 + 14,641 + 14,641 + 14,641)	−3.32	2.07	32,945 (24,482 + 3751 + 3751 + 961)	−3.32	2.27

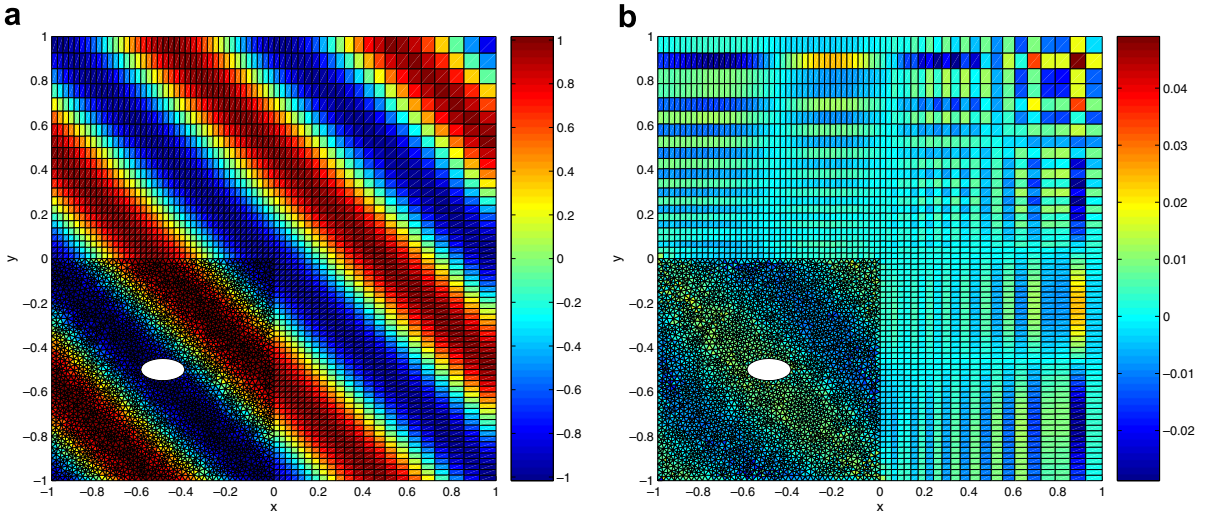

Fig. 9. A four subdomain case: (a) solution at $T = 1.0$ and (b) error at $T = 1.0$.

Another example illustrate the reflexion of error from the interfaces between multi-domains. The vortex is introduced into the computational domain $[-1, 1] \times [-1, 1]$ with four subdomains by using the analytic solution

$$u(x, y, t) = \kappa \exp(-\theta((x - c_1 t + b_1)^2 + (y - c_2 t + b_2)^2)),$$

as boundary and initial data (see Fig. 8). In the test we used $\kappa = 0.5$, $\theta = 50$, $c_1 = 1$, $b_1 = 0.5$, $c_2 = 1$ and $c_2 = 0.5$. Between $t = 0.3$ and 0.7 the vortex propagates close to the interfaces $y = 0$ and $x = 0$. No problems could be detected at the interfaces and the error/reflexion is very small (see Fig. 10).

5.3. An application to a non-linear problem

As an example of a non-linear problem we consider the two-dimensional viscous Burgers' equation

$$\begin{aligned} &u_t + \left(\frac{u^2}{2}\right)_x + u_y = \varepsilon u_{xx}, \quad 0 \leqslant x, y \leqslant 1, \\ &u(x, y, 0) = 1.5 - 2x, \\ &u(0, y, t) = 1.5, \quad u(1, y, t) = -0.5, \quad u(x, 0, t) = 1.5 - 2x. \end{aligned} \tag{48}$$

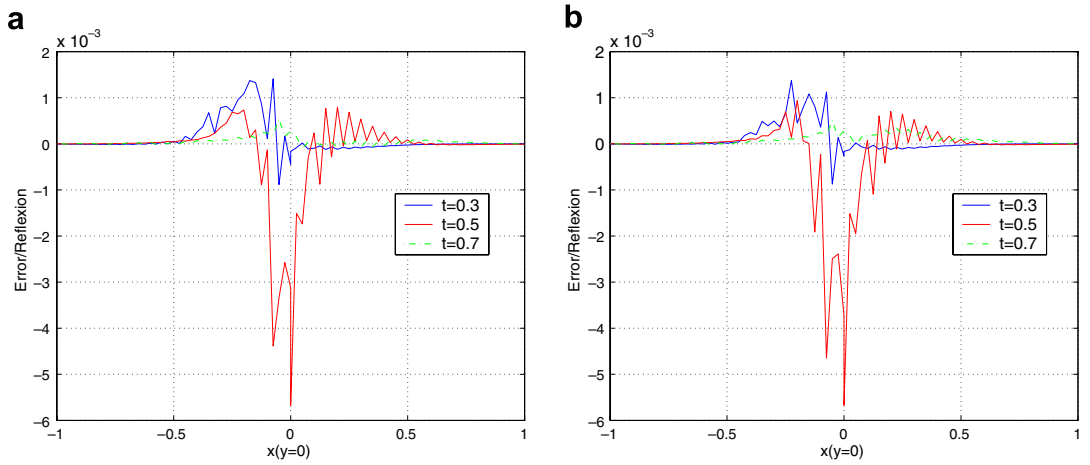

Fig. 10. The error on the interface between four subdomains. Subdomain 1: FVM on 2807 points; subdomain 2: 4th FDM on 41×41 points; subdomain 3: 4th FDM on 41×41 points; subdomain 4: 4th FDM on 41×41 points. (a) Interface $y = 0$ and (b) interface $x = 0$.

We have used $\varepsilon = 0.01$, 5647 grid points on the left unstructured domain and 81×41 grid points on the right structured domain. The result is shown in Fig. 11. We find that the interface does not destroy the shape of shock. The shock smoothly propagates through the interface.

5.4. Application to heat distribution around rods

Finally, we will exemplify our technique by computing the steady heat distribution around a set of rods. Consider the problem,

$$T_t + aT_x + bT_y = \varepsilon(T_{xx} + T_{yy}), \quad -1 \leqslant x, y \leqslant 1, \ t > 0, \tag{49}$$

with a initial condition $T = T_\infty$, and the boundary conditions

$$T = T_b, \quad (a, b) \cdot \hat{n} < 0, \quad \frac{\partial T}{\partial \hat{n}} = 0, \quad (a, b) \cdot \hat{n} \geqslant 0,$$

at the far-field boundary. $\hat{n}$ is the unit outward pointing normal. At the ith rod we specify the temperature $T = T_i$. For the temperatures we used $T_b = T_\infty = 1$, $T_1 = 2.0$, $T_2 = 0.1$, $T_3 = 1.5$ and $T_4 = 0.5$. In our test,

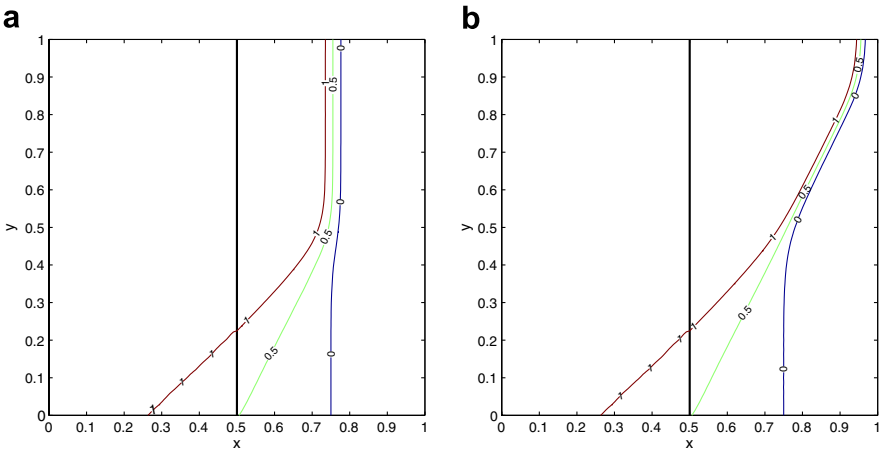

Fig. 11. The contour levels of the solution. The interface is located at $x = 0.5$. UFVM is used in the left domain and the third order HOFDM is used in the right domain: (a) $t = 0.5$ and (b) $t = 0.9$.

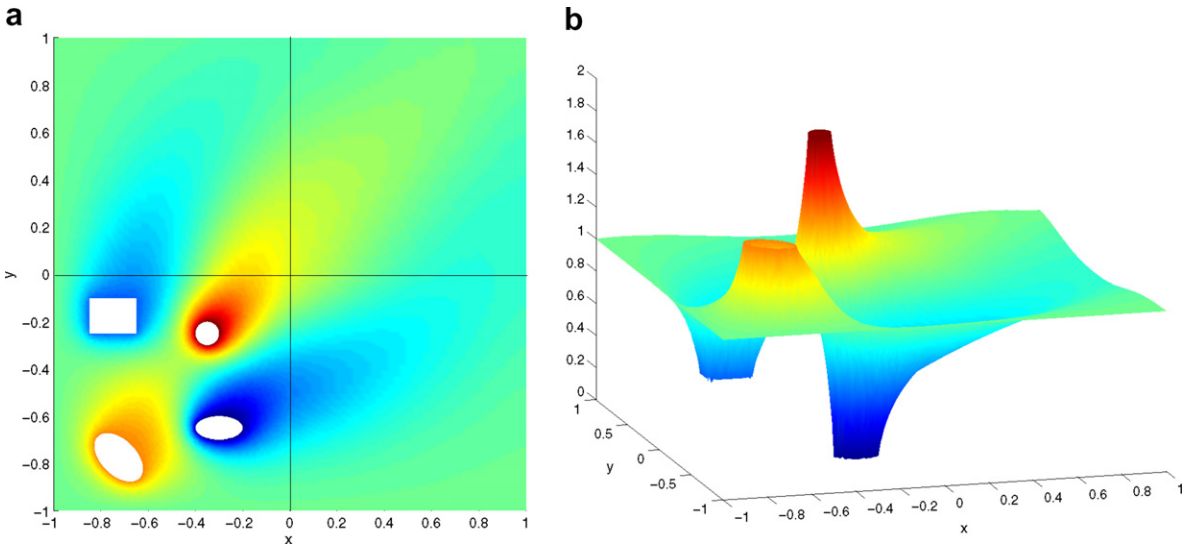

Fig. 12. The temperature distribution at $t = 3.0$ around four rods: (a) 2D and (b) 3D.

we used $a = 1, b = 1$ and $\varepsilon = 0.1$. The mesh topology is similar to the one in Fig. 8. An unstructured mesh with 10,300 nodes is used on the left-bottom subdomain. Structured meshes with 81×41, 41×81, 41×41 nodes are used on the left-top, right-bottom, right-top subdomains, respectively. The steady state solution is presented in Fig. 12. One can clearly see the advantage with this technique. The near-field around the rods is captured and the far-field part is efficiently handled.

6. Conclusions and future work

A conservative stable and efficient hybrid method for viscous problems that combines the unstructured finite volume method with the high-order finite difference method has been developed. The hybrid method can be applied to complex geometries with any type of interfaces. The calculations verify that the hybrid method is efficient, accurate, conservative and truly stable.

The technique developed in this paper makes it straight forward to apply the hybrid technique to the full Navier–Stokes equations. It also makes it possible to use two existing separate Navier–Stokes solver (one based on UFVM and one on HOFDM) and construct a significantly more efficient hybrid code suitable for aerodynamic and aeroacoustic source to signal type problems.

Appendix

Proof. The matrix H in Eq. (22) can be written in component form,

$$H = \left[\begin{array}{ccc|ccc} a_1 & & & b_1 & & \\ & \ddots & & & \ddots & \\ & & a_n & & & b_n \\ \hline b_1 & & & c_1 & & \\ & \ddots & & & \ddots & \\ & & b_n & & & c_n \end{array}\right] \tag{50}$$

where

$$\begin{aligned} a_i &= \left[\left(P_\xi^{\mathrm{R}} \otimes P_\eta^{\mathrm{R}}\right)\widetilde{J}\left(\tilde{\xi}_x^2 + \tilde{\xi}_y^2\right)\right]_{i,i} > 0, \\ b_i &= \left[\left(P_\xi^{\mathrm{R}} \otimes P_\eta^{\mathrm{R}}\right)\widetilde{J}\left(\tilde{\xi}_x\tilde{\eta}_x + \tilde{\xi}_y\tilde{\eta}_y\right)\right]_{i,i}, \quad i = 1,\ldots,n, \\ c_i &= \left[\left(P_\xi^{\mathrm{R}} \otimes P_\eta^{\mathrm{R}}\right)\widetilde{J}\left(\tilde{\eta}_x^2 + \tilde{\eta}_y^2\right)\right]_{i,i} > 0. \end{aligned} \tag{51}$$

For an arbitrary vector $\mathbf{x} = [x_1,\ldots,x_n,y_1,\ldots,y_n]^{\mathrm{T}}$, we have

$$\mathbf{x}^{\mathrm{T}}H\mathbf{x} = \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} a_1 & b_1 \\ b_1 & c_1 \end{bmatrix}\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + \cdots + \begin{bmatrix} x_i \\ y_i \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} a_i & b_i \\ b_i & c_i \end{bmatrix}\begin{bmatrix} x_i \\ y_i \end{bmatrix} + \cdots + \begin{bmatrix} x_n \\ y_n \end{bmatrix}^{\mathrm{T}}\begin{bmatrix} a_n & b_n \\ b_n & c_n \end{bmatrix}\begin{bmatrix} x_n \\ y_n \end{bmatrix}. \tag{52}$$

The eigenvalues of an arbitrary 2×2 matrix on the right-hand-side of (52) is

$$\lambda_{1,2}^i = \frac{a_i + c_i}{2} \pm \sqrt{\left(\frac{a_i + c_i}{2}\right)^2 - (a_ic_i - b_i^2)}, \quad ,i = 1,\ldots,n, \tag{53}$$

Since a_i, c_i are positive and

$$\begin{aligned} a_ic_i - b_i^2 &= \left[\left(P_\xi^{\mathrm{R}} \otimes P_\eta^{\mathrm{R}}\right)\widetilde{J}\right]_{i,i}\left[\left(\tilde{\xi}_x^2 + \tilde{\xi}_y^2\right)_{i,i}\left(\tilde{\eta}_x^2 + \tilde{\eta}_y^2\right)_{i,i} - \left(\tilde{\xi}_x\tilde{\eta}_x + \tilde{\xi}_y\tilde{\eta}_y\right)_{i,i}^2\right] \\ &= \left[\left(P_\xi^{\mathrm{R}} \otimes P_\eta^{\mathrm{R}}\right)\widetilde{J}\right]_{i,i}\left(\tilde{\xi}_y\tilde{\eta}_x - \tilde{\xi}_x\tilde{\eta}_y\right)_{i,i}^2 > 0, \end{aligned} \tag{54}$$

$\lambda_{1,2}^i$ is non-negative, which imply that H is a positive semi-definite matrix. We have proved the lemma. □

References

[1] T.J. Barth, P.O. Frederickson, Higher order solution of the euler equations on unstructured grids using quadratic reconstruction. AIAA Paper 90-0013, 1990.

[2] M.H. Carpenter, D. Gottlieb, S. Abarbanel, Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact schemes, Journal of Computational Physics 111 (2) (1994) 220–236.

[3] M.H. Carpenter, J. Nordström, D. Gottlieb, A stable and conservative interface treatment of arbitrary spatial accuracy, Journal of Computational Physics 148 (1999) 341–365.

[4] T. Gerhold, O. Friedrich, J. Evans, Calculation of complex three-dimensional configurations employing the DLR-τ-code, AIAA Paper 97-0167, 1997.

[5] M.B. Giles, UNSFLO: a numerical method the calculation of unsteady flow in turbomachinery, Report No. 205, Gas Turbine Laboratory, MIT, 1991.

[6] J. Gong, J. Nordström, Stable, accurate and efficient interface procedures for viscous problems, Technical Report 2006-19, Department of Information Technology, Uppsala University, Uppsala, Sweden, 2006.

[7] B. Gustafsson, H.-O. Kreiss, J. Oliger, Time Dependent Problems and Difference Methods, John Wiley and Sons Inc., 1995.

[8] A. Haselbacher, J.J. McGuirk, G.J. Page, Finite volume discretization aspects for viscous flows on mixed unstructured grids, AIAA Journal 37 (2) (1999).

[9] H. Hefazi, V. Chin, L.T. Chen, Two-dimensional zonal Euler method using composite structured and unstructured meshes, Journal of Aircraft 31 (1994) 651–658.

[10] K. Mattsson, J. Nordström, Summation by parts operators for finite difference approximations of second derivatives, Journal of Computational Physics 199 (2004) 503–540.

[11] D.J. Mavriplis, Accurate multigrid solution of the Euler equations on unstructured and adaptive meshes, AIAA Journal 28 (2) (1990).

[12] D.J. Mavriplis, V. Venkatakrishnan, A unified multigrid solver for the Navier–Stokes equations on mixed element meshes, Technical Report, Institute for Computer Applications in Science and Engineering, 1995.

[13] K. Nakahashi, S. Obayashi, FDM–FVM zonal approach for viscous flow computations over multiple bodies, AIAA Paper 87-0604, 1987.

[14] J. Nordström, Conservative finite difference formulations, variable coefficients, energy estimates and artificial dissipation, Journal of Scientific Computing 29 (3) (2006) 375–404.

[15] J. Nordström, M.H. Carpenter, Boundary and interface conditions for high order finite difference methods applied to the euler and Navier–Stokes equations, Journal of Computational Physics 148 (1999) 621–645.

[16] J. Nordström, M.H. Carpenter, High-order finite difference methods, multidimensional linear problems and curvilinear coordinates, Journal of Computational Physics 173 (2001) 149–174.

[17] J. Nordström, K. Forsberg, C. Adamsson, P. Eliasson, Finite volume methods, unstructured meshes and strict stability, Applied Numerical Mathematics 45 (2003) 453–473.
[18] J. Nordström, J. Gong, A stable and efficient hybrid method for aeroacoustic sound generation and propagation, Comptes Rendus Mecanique 333 (2005) 713–718.
[19] J. Nordström, J. Gong, A stable hybrid method for hyperbolic problems, Journal of Computational Physics 212 (2006) 436–453.
[20] B. Strand, Summation by parts for finite difference approximation for d/dx, Journal of Computational Physics 110 (1) (1994) 47–67.
[21] M. Svärd, J. Gong, J. Nordström, Stable artificial dissipation operators for finite volume schemes on unstructured grids, Applied Numerical Mathematics 56 (12) (2006) 1481–1490.
[22] M. Svärd, J. Nordström, Stability of finite volume approximations for the Laplacian operator on quadrilateral and triangular grids, Applied Numerical Mathematics 51 (2004) 101–125.
[23] M. Svärd, J. Nordström, On the order of accuracy for difference approximations of initial-boundary value problems, Journal of Computational Physics 218 (1) (2006) 333–352.
[24] J.M. Weiss, J.P. Maruszewski, W.A. Smith, Implicit solution of preconditioned Navier–Stokes equations using algebraic multigrid, AIAA Journal 37 (1) (1999).