SANDIA REPORT

SAND2011-5393
Unlimited Release
Printed June, 2011

Control Volume Finite Element Method with Multidimensional Edge Element Scharfetter-Gummel upwinding. Part 2. Computational Study

Pavel Bochev and Kara Peterson

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone:	(865) 576-8401
Facsimile:	(865) 576-5728
E-Mail:	reports@adonis.osti.gov
Online ordering:	http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone:	(800) 553-6847
Facsimile:	(703) 605-6900
E-Mail:	orders@ntis.fedworld.gov
Online ordering:	http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2011-5393
Unlimited Release
Printed June, 2011

Control Volume Finite Element Method with Multidimensional Edge Element Scharfetter-Gummel upwinding. Part 2. Computational Study

Pavel Bochev
Numerical Analysis and Applications, MS-1320
Sandia National Laboratories, Albuquerque, NM 87185-1320
pbboche@sandia.gov

Kara Peterson
Numerical Analysis and Applications, MS-1320
Sandia National Laboratories, Albuquerque, NM 87185-1320
kjpeter@sandia.gov

Abstract

In [3] we proposed a new Control Volume Finite Element Method with multi-dimensional, edge-based Scharfetter-Gummel upwinding (CVFEM-MDEU). This report follows up with a detailed computational study of the method.

The study compares the CVFEM-MDEU method with other CVFEM and FEM formulations for a set of standard scalar advection-diffusion test problems in two dimensions. The first two CVFEM formulations are derived from the CVFEM-MDEU by simplifying the computation of the flux integrals on the sides of the control volumes, the third is the nodal CVFEM [2] without upwinding, and the fourth is the streamline upwind version of CVFEM [10]. The finite elements in our study are the standard Galerkin, SUPG and artificial diffusion methods. All studies employ logically Cartesian partitions of the unit square into quadrilateral elements. Both uniform and non-uniform grids are considered.

Our results demonstrate that CVFEM-MDEU and its simplified versions perform equally well on rectangular or nearly rectangular grids. However, performance of the simplified versions significantly degrades on non-affine grids, whereas the CVFEM-MDEU remains stable and accurate over a wide range of mesh Peclet numbers and non-affine grids. Compared to FEM formulations the CVFEM-MDEU appears to be slightly more dissipative than the SUPG, but has much less local overshoots and undershoots.

Acknowledgment

The authors acknowledge funding by the Advanced Simulation & Computing (ASC) program of the National Nuclear Security Administration (NNSA), and DOE's Office of Science Advanced Scientific Computing Research Program.

Contents

List of Figures

Nomenclature

Ω **-** computational domain in two dimensions.
$K_h(\Omega)$ **-** finite element mesh
K_s **-** finite element
$\boldsymbol{b}_s$ **-** barycenter of element K_s
$\boldsymbol{v}_i$ **-** a mesh vertex
$\boldsymbol{e}_{ij}$ **-** an edge with endpoints $\boldsymbol{v}_i$ and $\boldsymbol{v}_j$
$\boldsymbol{m}_{ij}$ **-** midpoint of edge $\boldsymbol{e}_{ij}$
$K(\boldsymbol{v}_i)$ **-** the set of all elements K_s having a common vertex $\boldsymbol{v}_i$.
$K(\boldsymbol{e}_{ij})$ **-** the set of all elements K_s having a common edge $\boldsymbol{e}_{ij}$.
$V(K_s)$ **-** the vertices of element K_s.
$E(K_s)$ **-** the edges (sides) of element K_s
$E(\boldsymbol{v}_i)$ **-** the set of all edges having $\boldsymbol{v}_i$ as a vertex
C_i **-** control volume associated with vertex $\boldsymbol{v}_i$
∂C_{ij}^s **-** the side of C_i contained in element K_s which intersects edge $\boldsymbol{e}_{ij}$
$\boldsymbol{m}_{ij}^s$ **-** the midpoint of side ∂C_{ij}^s
∂C_{ij} **-** the union of the two sides of C_i which intersect edge $\boldsymbol{e}_{ij}$, i.e.,
$\partial C_{ij} = \partial C_{ij}^s \cup \partial C_{ij}^t$; $K_s, K_t \in K(\boldsymbol{e}_{ij})$
$\mathbf{G}_h(\Omega)$ $H^1(\Omega)$-conforming finite element space (C^0, nodal elements)
$\mathbf{C}_h(\Omega)$ $H(curl, \Omega)$-conforming finite element space (edge elements)
N_i **-** C^0 Lagrangian (nodal) basis function associated with vertex $\boldsymbol{v}_i$
$\vec{W}_{ij}$ **-** Lowest-order Nedelec (edge element) basis function associated with edge $\boldsymbol{e}_{ij}$

0.1 Introduction

In this report we study computationally the new Control Volume Finite Element Method with multi-dimensional Scharfetter-Gummel upwinding (CVFEM-MDEU), proposed in [3] (referred henceforth as Part 1). To facilitate comparison with other published algorithms, we present results for the generic scalar advection-diffusion equation

$$\begin{cases} -\nabla\cdot(\varepsilon\nabla\phi - \vec{\mathbf{u}}\phi) = f & \text{in } \Omega \\ \phi = g & \text{on } \Gamma\,. \end{cases} \tag{1}$$

We assume that $\nabla\cdot\vec{\mathbf{u}} = 0$. Our study compares the CVFEM-MDEU scheme [3], specialized for (1), with several CVFEM and FEM formulations. The first two CVFEM formulations are derived from the CVFEM-MDEU by simplifying the computation of the flux integrals on the sides of the control volumes, the third is the nodal CVFEM [2] without upwinding, and the fourth is the streamline upwind version of CVFEM [10]. The finite element methods in our study are the standard Galerkin, SUPG [6] and artificial diffusion methods for (1).

The report is organized as follows. Section 0.2 states the CVFEM and FEM formulations studied in this report. Section 0.3 specifies the test problems (1) and Section 0.4 describes the grids used in the computational study. Section 0.5 presents the numerical results obtained with the different methods. Sections 0.6 summarizes our findings.

0.1.1 Notation

We follow the notation established in Part 1. In all examples Ω is the unit square $[0,1]$ and $K_h(\Omega)$ is a conforming, logically Cartesian, but not necessarily uniform, finite element partition of Ω into quadrilateral elements K_s. $\mathbf{G}_h(\Omega)$ and $\mathbf{C}_h(\Omega)$ are the lowest-order $H^1(\Omega)$ and $H(curl,\Omega)$-conforming spaces on $K_h(\Omega)$. Thus, $\mathbf{G}_h(\Omega)$ is the C^0 piecewise bilinear finite element space (nodal, or Lagrangian elements) and $\mathbf{C}_h(\Omega)$ contains piecewise polynomial vector fields whose tangential component is continuous along the element edges (edge, or Nedelec elements) [9]. The basis of $\mathbf{G}_h(\Omega)$ is $\{N_i\}$, $\boldsymbol{v}_i \in \dot{\Omega}$ and $\{\vec{W}_{ij}\}$, $\boldsymbol{e}_{ij} \in \dot{\Omega}$ is basis for the Nedelec edge elements.

The order of the edge vertices induces the orientation σ_{ij} of $\boldsymbol{e}_{ij}$:

$$\sigma_{ij} = \begin{cases} -1 & \text{if the vertex order is } \boldsymbol{v}_i \to \boldsymbol{v}_j \\ 1 & \text{if the vertex order is } \boldsymbol{v}_j \leftarrow \boldsymbol{v}_i \end{cases} \tag{2}$$

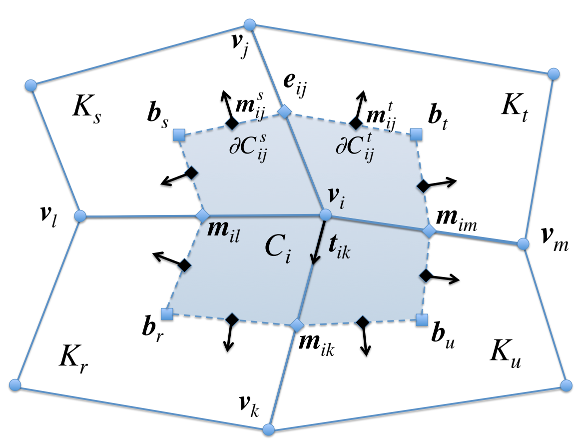


Figure 1: The dual control volume C_i for a primal vertex $\boldsymbol{v}_i$ on a logically Cartesian quadrilateral grid is generally a non-convex octagon. If all elements are rectangles then C_i is also a rectangle.

The oriented unit tangent on $\boldsymbol{e}_{ij}$ always points towards the *second* vertex of the edge:

$$\vec{\boldsymbol{t}}_{ij} = \sigma_{ij} \frac{\boldsymbol{v}_i - \boldsymbol{v}_j}{|\boldsymbol{v}_i - \boldsymbol{v}_j|}\,.$$

To simplify notation, throughout the paper we assume that edge basis functions $\vec{W}_{ij}$ are oriented along the direction of the edges $\boldsymbol{e}_{ij}$, i.e., $\vec{W}_{ij} \cdot \vec{\boldsymbol{t}}_{ij} > 0$.

For quadrilateral grids the control volume C_i corresponding to vertex $\boldsymbol{v}_i$ is constructed as follows. For every element K_r which has $\boldsymbol{v}_i$ as a vertex, i.e., for every $K_r \in K(\boldsymbol{v}_i)$, we connect its barycenter $\boldsymbol{b}_r$ with the midpoints $\boldsymbol{m}_{ik}$ and $\boldsymbol{m}_{il}$ of the two edges coming out of $\boldsymbol{v}_i$; see Fig. 1. This construction guarantees that $\boldsymbol{v}_i \in C_i$ whenever the grid is comprised of convex but not necessarily uniform quadrilaterals.

0.2 Summary of the discretization methods

In this report we consider $H^1(\Omega)$-conforming CVFEM and FEM formulations, which seek approximate solutions ϕ_h of (1) in the nodal space $\mathbf{G}_h(\Omega)$. As a result, all methods use identical representations of the discrete solution in terms of the nodal basis:

$$\phi_h = \sum_{j \in \dot{\Omega}} n_j N_j + \sum_{j \in \Gamma} g(\boldsymbol{x}_j) N_j\,. \tag{3}$$

The CVFEM and FEM methods use different approaches to construct the matrix problem

$$\mathbb{K}\vec{n} = \vec{f}$$

for the unknown vector of nodal coefficients $\vec{n} = (n_1, \ldots, n_N)$. We briefly review the methods compared in this report.

0.2.1 CVFEM formulations

The basic CVFEM method for (1) is defined by the "weak" equation

$$-\int_{\partial C_i} \mathbf{J}_h \cdot \vec{\boldsymbol{n}}\, dS = \int_{C_i} f dV \quad \forall i \in \dot{\Omega} \tag{4}$$

where $\mathbf{J}_h$ is approximation of the total flux $\mathbf{J} = \varepsilon\nabla\phi - \vec{\mathbf{u}}\phi$. Different choices of $\mathbf{J}_h$ lead to different CVFEM formulations. In the nodal CVFEM the finite element solution (3) defines the discrete nodal flux

$$\mathbf{J}_h^N = \varepsilon\nabla\phi_h - \vec{\mathbf{u}}\phi_h\,. \tag{5}$$

We term this formulation CVFEM-N. The stiffness matrix $\mathbb{K}$ in the CVFEM-N has element

$$\mathbb{K}_{ij} = -\int_{\partial C_i} (\varepsilon\nabla N_j - \vec{\mathbf{u}} N_j) \cdot \vec{\boldsymbol{n}}\, dS \quad \forall i, j \in \dot{\Omega}\,. \tag{6}$$

The streamline upwind version [10] of CVFEM-N employs the upwind flux

$$\mathbf{J}_h^{SU} = \varepsilon\nabla\phi_h - \vec{\mathbf{u}}\phi_h + \tau_{cv}\vec{\mathbf{u}}\nabla \cdot (\vec{\mathbf{u}}\phi_h) \tag{7}$$

We term this method CVFEM-NSU. In (7) we use the same stabilization parameter as in [10]:

$$\tau_{cv} = \left(\coth Pe - \frac{1}{Pe}\right) \frac{h_K}{2|\vec{\mathbf{u}}|}\,,$$

where h_K is the element size and

$$Pe = \frac{|\vec{\mathbf{u}}| h_K}{2\varepsilon} \tag{8}$$

is the element Peclet number.

Edge element based CVFEM formulations. The edge-based CVFEM, proposed in Part 1, uses a discrete flux defined in terms of the edge element basis $\{\vec{W}\}$

$$\mathbf{J}_h^E = \sum_{\boldsymbol{e}_{ij}} \frac{u_{ij}}{2} \Big[n_i(\coth \alpha_{ij} - 1) - n_j(\coth \alpha_{ij} + 1) \Big] \vec{W}_{ij} \,, \tag{9}$$

where n_i, n_j are nodal values at the endpoints of $\boldsymbol{e}_{ij}$, and

$$u_{ij} = \mathbf{u} \cdot \vec{\boldsymbol{t}}_{ij} \quad \text{and} \quad \alpha_{ij} = \frac{u_{ij}|\boldsymbol{e}_{ij}|}{2\varepsilon} \tag{10}$$

are the edge velocity and the edge Peclet number for edge $\boldsymbol{e}_{ij}$, respectively. The coefficients of the edge basis functions in (9) are derived by applying the Scharfetter-Gummel formula for (1) to the edges of the mesh; see Part 1. The elements of the resulting edge-based CVFEM stiffness matrix $\mathbb{K}$ are

$$\mathbb{K}_{ij} = \; - \sum_{\boldsymbol{e}_{nj} \in E(\boldsymbol{v}_j)} \sigma_{nj} \frac{u_{nj}}{2} (\coth \alpha_{nj} - \sigma_{nj}) \int_{\partial C_i} \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \, dS \quad \forall i,j \in \dot{\Omega} \,, \tag{11}$$

where σ_{nj} is the orientation of edge $\boldsymbol{e}_{nj}$ defined in (2).

Formula (11) involves integrals of $\vec{W}_{nj} \cdot \vec{\boldsymbol{n}}$ on the sides of the control volume, which must be approximated by quadrature rules. We consider three versions of the edge-based CVFEM, which correspond to three different ways to compute these integrals. However, regardless of the particular choice of a CVFEM formulation, assembly of the stiffness matrix can be accomplished in a completely standard manner by computing and scattering contributions from individual elements. The contribution $\mathbb{K}_{ij}^r$ from an element K_r to the global stiffness matrix is

$$\mathbb{K}_{ij}^r = \; - \sum_{\boldsymbol{e}_{nj} \in E(\boldsymbol{v}_j) \cap E(K_r)} \left(\sigma_{nj} \frac{u_{nj}}{2} (\coth \alpha_{nj} - \sigma_{nj}) \right) \left(\int_{\partial C_{ij}^r} \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \, dS + \int_{\partial C_{ik}^r} \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \, dS \right) \tag{12}$$

where ∂C_{ij}^r and ∂C_{ik}^r are the sides of the control volume fraction $C_i \cap K_r$; see Fig. 2. We recall that $E(\boldsymbol{v}_j)$ is the set of all edges connected to vertex $\boldsymbol{v}_j$ and $E(K_r)$ is the set of all edges in element K_r. Therefore, in two-dimensions, the intersection $E(\boldsymbol{v}_j) \cap E(K_r)$ contains exactly two edges.

The CVFEM-MDEU formulation The CVFEM with Multi-Dimensional Edge-based Upwinding (CVFEM-MDEU) formulation corresponds to computation of the integrals on ∂C_i in (11) by using the midpoint rule on each side. Specifically, the CVFEM-MDEU method approximates the element contribution $\mathbb{K}_{ij}^r$ in (12) by using the midpoint rule for the integrals on ∂C_{ij}^r and ∂C_{ik}^r:

$$\mathbb{K}_{ij}^r \approx \; - \sum_{\boldsymbol{e}_{nj} \in E(\boldsymbol{v}_j) \cap E(K_r)} \left(\sigma_{nj} \frac{u_{nj}}{2} (\coth \alpha_{nj} - \sigma_{nj}) \right) \left(|\partial C_{ij}^r| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ij}^r + |\partial C_{ik}^r| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ik}^r \right) . \tag{13}$$

In (11) $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ij}^r$ and $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ik}^r$ are the normal components of the edge basis function $\vec{W}_{nj}$, evaluated at the midpoints $\boldsymbol{m}_{ij}^r$ and $\boldsymbol{m}_{ik}^r$ of the control volume fraction sides ∂C_{ij}^r and ∂C_{ik}^r, respectively; see the left pane in Fig. 2.

The CVFEM-EPEU formulation. The CVFEM with End-Point Edge-based Upwinding corresponds to computation of the integrals on the control volume sides in (11) by a one-point quadrature rule. The integration point for side ∂C_{ij}^r is the midpoint $\boldsymbol{m}_{ij}$ of the edge $\boldsymbol{e}_{ij}$, which intersects the side, instead of the midpoint $\boldsymbol{m}_{ij}^r$ of the side itself. As a result, the CVFEM-EPEU formulation approximates the element contribution $\mathbb{K}_{ij}^r$ in (12) using the formula

$$\mathbb{K}_{ij}^r \approx \; - \sum_{\boldsymbol{e}_{nj} \in E(\boldsymbol{v}_j) \cap E(K_r)} \left(\sigma_{nj} \frac{u_{nj}}{2} (\coth \alpha_{nj} - \sigma_{nj}) \right) \left(|\partial C_{ij}^r| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ij} + |\partial C_{ik}^r| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle_{ik} \right) . \tag{14}$$

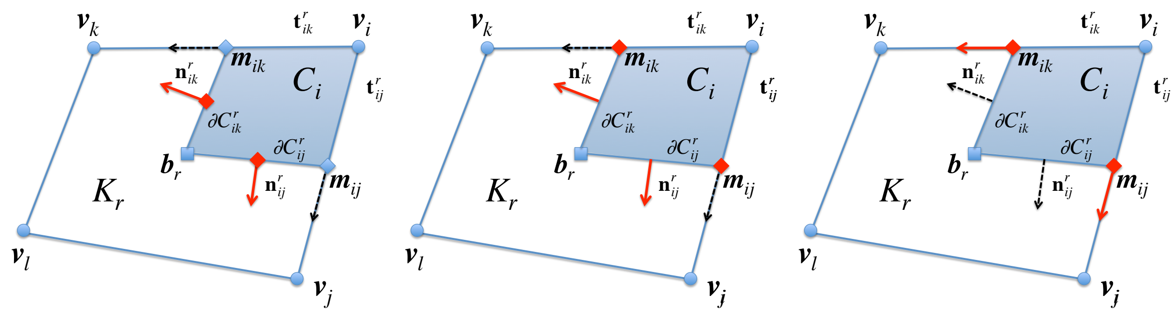


Figure 2: The CVFEM-MDEU formulation (left pane), and its simplified versions CVFEM-EPEU (middle pane) and CVFEM-DSEU (right pane) approximate the integral of the normal flux on the control volume sides ∂C^r_{ij} and ∂C^r_{ik} using different evaluation points (the red diamonds) and unit vectors (the red arrows). CVFEM-MDEU uses the midpoints $\boldsymbol{m}^r_{ij}$ and $\boldsymbol{m}^r_{ik}$ of ∂C^r_{ij} and ∂C^r_{ik} and the unit normals to these sides. The CVFEM-EPEU (middle pane) uses the midpoints $\boldsymbol{m}_{ij}$ and $\boldsymbol{m}_{ik}$ of the element edges which intersect ∂C^r_{ij} and ∂C^r_{ik}, but retains the unit normals to the these sides. The second simplification of the CVFEM-MDEU, the CVFEM-DSEU scheme, uses the edge midpoints $\boldsymbol{m}_{ij}$ and $\boldsymbol{m}_{ik}$ *and* the edge unit tangents $\vec{\boldsymbol{t}}_{ij}$, $\vec{\boldsymbol{t}}_{ik}$.

In contrast to (13), in this formula $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle^r_{ij}$ and $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{n}} \rangle^r_{ik}$ are the normal components of the edge basis function $\vec{W}_{nj}$, evaluated at the midpoints $\boldsymbol{m}_{ij}$ and $\boldsymbol{m}_{ik}$ of the edges $\boldsymbol{e}_{ij}$ and $\boldsymbol{e}_{ik}$, respectively; see the midlle pane in Fig. 2.

The CVFEM-DSEU formulation. The CVFEM with Dimension-Split Edge-based Upwinding approximates the integrals on the control volume sides in (11) using the formula

$$\mathbb{K}^r_{ij} \approx -\sum_{\boldsymbol{e}_{nj} \in E(\boldsymbol{v}_j) \cap E(K_r)} \left(\sigma_{nj} \frac{u_{nj}}{2} (\coth \alpha_{nj} - \sigma_{nj}) \right) \left(|\partial C^r_{ij}| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{t}} \rangle_{ij} + |\partial C^r_{ik}| \langle \vec{W}_{nj} \cdot \vec{\boldsymbol{t}} \rangle_{ik} \right) . \tag{15}$$

In this formula $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{t}} \rangle^r_{ij}$ and $\langle \vec{W}_{nj} \cdot \vec{\boldsymbol{t}} \rangle^r_{ik}$ are the *tangential* components of the edge basis function $\vec{W}_{nj}$, evaluated at the midpoints $\boldsymbol{m}_{ij}$ and $\boldsymbol{m}_{ik}$ of the edges $\boldsymbol{e}_{ij}$ and $\boldsymbol{e}_{ik}$, respectively; see the right pane in Fig. 2. Therefore, the only difference between the CVFEM-DSEU and the CVFEM-EPEU is in the component of the edge basis function they employ. We can interpret CVFEM-DSEU as an approximation of CVFEM-EPEU in which the unit normal to a control volume side is replaced by the unit tangent to the edge which intersects this side. The CVFEM-DSEU and CVFEM-EPEU are identical on rectangular grids where the element edge tangent is collinear with the control volume side normal.

0.2.2 Finite element methods

The standard Galerkin method for (1) is usually defined by the weak equation

$$\int_\Omega \varepsilon \nabla \phi_h \cdot \nabla N_i + \vec{\mathbf{u}} \cdot \nabla \phi_h N_i \, dV = \int_\Omega f N_i dV \quad \forall i \in \dot{\Omega} . \tag{16}$$

Weak form (16) relies on the assumption that $\vec{\mathbf{u}}$ is divergence free so that $\nabla \cdot (\vec{\mathbf{u}} \phi_h) = \vec{\mathbf{u}} \cdot \nabla \phi_h$.

The artificial diffusion version of the Galerkin method adds to (16) the weak form of the dissipation term $-h\Delta$, where h is the mesh parameter. The corresponding variational problem is

$$\int_\Omega (\varepsilon + h) \nabla \phi_h \cdot \nabla N_i + \vec{\mathbf{u}} \cdot \nabla \phi_h N_i \, dV = \int_\Omega f N_i dV \quad \forall i \in \dot{\Omega} . \tag{17}$$

The additional dissipation term helps to stabilize the finite element solution when $\varepsilon \ll h$.

For bilinear elements the second-order terms in the classical SUPG formulation [6] vanish and the SUPG weak equation assumes the form

$$\int_\Omega \varepsilon \nabla \phi_h \cdot \nabla N_i + \vec{\mathbf{u}} \cdot \nabla \phi_h (N_i + \tau \vec{\mathbf{u}} \cdot \nabla N_i)\, dV = \int_\Omega f(N_i + \tau \vec{\mathbf{u}} \cdot \nabla N_i) dV \quad \forall i \in \dot{\Omega}\,, \tag{18}$$

In our study we define the stabilization parameter τ on element K as in [5]

$$\tau(\boldsymbol{x}) = \begin{cases} \dfrac{h_K}{2|\vec{\mathbf{u}}(\boldsymbol{x})|} & \text{if } Pe > 3 \\ h_K^2 & \text{if } Pe \leq 3 \end{cases} \tag{19}$$

where h_K is the element size and Pe is the element Peclet number defined in (8)

0.3 Specification of the example problems

For each example problem we specify the advective velocity $\vec{\mathbf{u}}$, the Dirichlet boundary data g, and the forcing term f. To increase or decrease the Peclet number we vary the value of the diffusion coefficient ε. The boundary $\Gamma = \Gamma_B \cup \Gamma_T \cup \Gamma_L \cup \Gamma_R$, where

$$\Gamma_B = \{(x, y)\,|\, 0 \leq x \leq 1; y = 0\}; \quad \Gamma_T = \{(x, y)\,|\, 0 \leq x \leq 1; y = 1\}$$

are the bottom and top sides of Ω and

$$\Gamma_L = \{(x, y)\,|\, 0 \leq y \leq 1; x = 0\}; \quad \Gamma_R = \{(x, y)\,|\, 0 \leq y \leq 1; x = 1\}$$

are the left and the right sides of Ω, respectively

Example 1: Linear solution. This is a manufactured solution problem with an exact solution $\phi = x + y$. Substitution of the exact solution into the PDE (1) defines the boundary data and the forcing term.

Example 2: Pseudo one-dimensional problem. In this problem

$$\vec{\mathbf{u}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad f = 0, \quad g = \begin{cases} 1 & \text{on } \Gamma_L \cup \Gamma_B \cup \Gamma_T \\ 0 & \text{on } \Gamma_R \end{cases} \tag{20}$$

Solution of this example problem develops an exponential boundary layer at the right side Γ_R.

Example 3: Constant advection. This Example specializes [4, Example 3.1.3, p.118] to the unit square:

$$\vec{\mathbf{u}} = \begin{pmatrix} -\sin \pi/6 \\ \cos \pi/6 \end{pmatrix}; \quad f = 0; \quad g = \begin{cases} 0 & \text{on } \Gamma_L \cup \Gamma_T \cup (\Gamma_B \cap \{x \leq 0.5\}) \\ 1 & \text{on } \Gamma_R \cup (\Gamma_B \cap \{x > 0.5\}) \end{cases} . \tag{21}$$

Discontinuity in the boundary data leads to an internal layer of width $O(\sqrt{\varepsilon})$. Near Γ_T the solution of (21) develops an exponential boundary layer to match the prescribed boundary data on Γ_T.

Example 4: Double glazing. This example specializes [4, Example 3.1.4, p.119] to the unit square:

$$\vec{\mathbf{u}} = \begin{pmatrix} 2(2y-1)(1-(2x-1)^2) \\ -2(2x-1)(1-(2y-1)^2) \end{pmatrix}; \quad f=0; \quad g = \begin{cases} 1 & \text{on } \Gamma_R \\ 0 & \text{on } \Gamma_B \cup \Gamma_T \cup \Gamma_L \end{cases} . \tag{22}$$

Problem (22) is known as the *double-glazing* problem. It models temperature distribution in a cavity with a "hot" external wall (Γ_R).The discontinuities at the two corners of the hot wall create boundary layers near its corners.

0.4 Specification of the computational grids

Our numerical study employs uniform grids and three different nonuniform structured quadrilateral grids. All grids are defined by moving the nodes of an initial uniform grid, i.e., the mesh node positions are specified by

$$x_{ij} = x(\xi_i, \eta_j, \gamma), \quad y_{ij} = y(\xi_i, \eta_j, \gamma), \qquad 0 \le i \le N_x, \quad 0 \le j \le N_y \,, \tag{23}$$

where N_x and N_y are the numbers of cells in x and y direction, respectively, γ is real parameter, $x(\xi, \eta, \gamma)$ and $y(\xi, \eta, \gamma)$ are coordinate maps and

$$\xi_i = \frac{i}{N_x}, \quad i = 0, \ldots, N_x; \qquad \text{and} \qquad \eta_j = \frac{j}{N_y}, \quad j = 0, \ldots, N_y; \tag{24}$$

are the initial (uniform) grid coordinates, respectively. For uniform grids

$$x(\xi_i, \eta_j, \gamma) = \xi_i \quad \text{and} \quad y(\xi_i, \eta_j, \gamma) = \eta_j$$

Randomly perturbed grids. The coordinate maps for these grids are

$$x(\xi_i, \eta_j, \gamma) = \xi_i + 0.25h(r_x h^\gamma); \quad y(\xi_i, \eta_j, \gamma) = \eta_j + 0.25h(r_y h^\gamma) \tag{25}$$

where r_x, r_y are uniformly distributed random numbers in $[-1, 1]$, and $\gamma \ge 0$ is the strength of the perturbation. The nodes on the vertical sides are not allowed to move horizontally and the nodes on the horizontal sides are not allowed to move vertically.

We use $\gamma = 0, 1, 2$. If $\gamma = 0$, then the x and y coordinates of the mesh nodes can move up to 1/4 of the initial uniform element size along their respective coordinate axes. We refer to the resulting grids as $O(1)$ perturbations of the initial uniform grid; see Fig. 3. If $\gamma = 1$, the coordinate movement is limited to h times 1/4 of the element size, and if $\gamma = 2$ the motion is further restricted to h^2 times 1/4 of the element size. We refer to these cases as $O(h)$ and $O(h^2)$ perturbed uniform grids; see Fig. 3.

Tensor product grids. We use a coordinate map definition from [7, 8]:

$$x(\xi, \eta, \gamma) = (1-\alpha(\gamma))\xi + \alpha(\gamma)\xi^3; \quad y(\xi, \eta, \gamma) = (1-\alpha(\gamma))\eta + \alpha(\gamma)\eta^2; \quad \alpha(\gamma) = \frac{\sin(4\pi\gamma)}{2} \,, \tag{26}$$

where $0 \le \gamma \le 1$. The coordinate maps (26) generate a sequence of rectangular, affine tensor-product grids; see the left pane in Fig. 4.

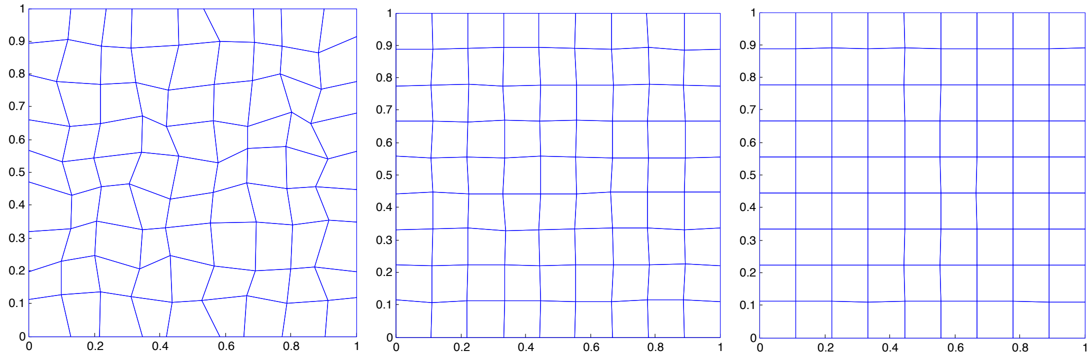


Figure 3: Randomly perturbed grids: $O(1)$ grid (left pane), $O(h)$ grid (middle pane) and $O(h^2)$ grid (right pane).

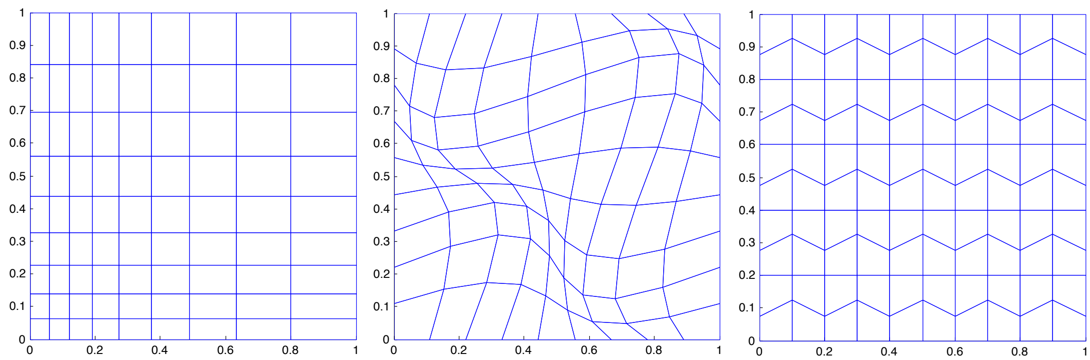


Figure 4: Structured nonuniform grids. The left pane is the tensor product grid (26) with $\gamma = 0.1$. The middle pane shows the smooth grid (27) with $\gamma = 0.5$. The right pane is 11×11 trapezoidal grid.

Smooth non-afine grids. The coordinate maps for this grid type are also from [7, 8]:

$$x(\xi, \eta, \gamma) = \xi + \alpha(\gamma) \sin(2\pi\xi) \sin(2\pi\eta); \quad y(\xi, \eta, \gamma) = \eta + \alpha(\gamma) \sin(2\pi\xi) \sin(2\pi\eta) \,. \tag{27}$$

Here, the function α is

$$\alpha(\gamma) = \begin{cases} \gamma/5 & \text{if } 0 \le \gamma \le 0.5 \\ (1-\gamma)/5 & \text{if } 0.5 < \gamma \le 1.0 \end{cases} \,.$$

The coordinate maps (27) define logically Cartesian but not rectangular grids; see the middle pane in Fig. 4. For any $0 \le t \le \gamma$ the grids generated by (26) and (27) are valid [8].

Structured non-affine grids. The "trapezoidal" grid from [1] is the third nonuniform grid type in the study. The coordinate maps for this grid are

$$x(\xi_i, \eta_j, \gamma) = \xi \quad y(\xi_i, \eta_j, \gamma) = \eta + \text{mod }(j, 2)(-1)^{1-\text{mod }(i,2)} 0.25h; \tag{28}$$

These grid functions require even numbers N_x and N_y of grid cells in the x and y directions. The grid functions only perturb the y-coordinates of the nodes on the *odd horizontal* grid lines[1], i.e., when $j = 2k+1$. If the x grid line is even, i.e., $i = 2l$, the y-coordinate shifts down by $1/4h$, otherwise, it moves up by the same amount; see the right pane in Fig. 4.

[1]Numbering of grid lines starts at zero; see (24).

0.5 Computational results

Example 1: Linear solution. This example tests how well the edge-based upwind CVFEM formulations approximate a function that belongs to the finite element space $\mathbf{G}_h$. The Galerkin and SUPG methods are weighted-residual formulations, which recover globally linear functions on any mesh and for any advective vector. The edge-based CVFEMs, the CVFEM-N and the CVSFEM-NSU are not residual-based formulations. In general they recover a globally linear function only on select grids and for constant advective fields.

To determine the boundary data and the right hand side we substitute the exact solution $\phi = x + y$ into (1). The diffusion coefficient is $\varepsilon = 0.01$ and the velocity field is from Test 3, i.e., $\vec{\mathbf{u}}$ is defined in (21). We solve (1) on uniform, $O(h^2)$, $O(h)$ and $O(1)$ randomly perturbed grids, tensor grid (26) with $\gamma = 0.1$, smooth grid (27) with $\gamma = 0.5$ and trapezoidal grids with 33×33 grid lines. The largest values of the edge Peclet number α_{ij} (10) are

- 1.35 for the $O(h^2)$ grid,
- 1.34 for the $O(h)$ grid,
- 2.19 for the $O(1)$ grid,
- 1.97 for the tensor grid (26) with $\gamma = 0.1$,
- 1.71 for the smooth grid,
- 1.69 for the trapezoidal grid

Therefore, in all cases the problem is diffusion dominated.

Table 1 and Figures 5-6 summarize the results from this test. As expected for constant advection fields, on uniform grids the edge-based and the nodal CVFEM formulations recover the exact solution of the test problem. On the rectangular (but not uniform) tensor product grid the three edge-based CVFEM formulations do not recover the linear solution but their accuracy is identical.

Calculations on the randomly perturbed grids reveal that only the CVFEM-MDEU performs well. As the strength of the grid perturbation increases, the accuracy of the CVFEM-EPEU and CVFEM-DSEU deteriorates to a point where on the $O(1)$ random grid their results are unusable; see the bottom row in Fig. 5.

The same holds true for CVFEM-EPEU and CVFEM-DSEU on the smooth and trapezoidal grids; see the last two rows in Fig. 6. On the trapezoidal grids CVFEM-EPEU and CVFEM-DSEU develop strong node to node oscillations, while on the smooth grid we see strong mesh imprinting in the results.

Example 2: Pseudo one-dimensional problem. This example reveals the amount of "pollution" caused by a nonuniform grid structure in a method. We solve (20) on three 33×33 nonuniform grids and $\varepsilon = 1/2000$. The grids and the corresponding maximum element Peclet numbers are

- 45.6 for the $O(1)$ grid,
- 50.1 for the smooth grid (27) with $\gamma = 0.5$,
- 60.6 for the tensor grid (26) with $\gamma = 0.1$.

With $\gamma = 0.1$ the tensor grid is de-refined along the right side Γ_R where the solution develops a boundary layer. This choice is intentional to make the test more challenging. Figures 7–9 present the computational results. The plots in these figures show that the CVFEM-MDEU introduces the

Table 1: Approximation of a globally linear function by CVFEM and FEM on 33×33 randomly perturbed grids and structured nonuniform grids. Constant velocity (21) and $\varepsilon = 0.01$

Grid	error	CVFEM-MDEU	CVFEM-EPEU	CVFEM-DSEU	CVFEM-N	CVFEM-NSU
Uniform	L^2	0.1980529E-14	0.3009857E-14	0.3009857E-14	0.2280905E-14	0.2459688E-14
	H^1	0.2520868E-13	0.3385221E-13	0.3385221E-13	0.3205941E-13	0.3584957E-13
$O(h^2)$	L^2	0.3080467E-05	0.2516427E-03	0.2516357E-03	0.4258521E-14	0.4424319E-06
	H^1	0.2569498E-03	0.1922736E-01	0.1922749E-01	0.1792076E-12	0.3356442E-04
$O(h)$	L^2	0.7225208E-04	0.5817959E-02	0.5812317E-02	0.3907673E-14	0.1015684E-04
	H^1	0.6034698E-02	0.4496073E+00	0.4496437E+00	0.1739823E-12	0.7741027E-03
$O(1)$	L^2	0.1604478E-02	0.1270834E+00	0.1403228E+00	0.4192892E-14	0.2161388E-03
	H^1	0.1360415E+00	0.1076107E+02	0.1062789E+02	0.1832044E-12	0.1685313E-01
Tensor	L^2	0.8256768E-02	0.8238724E-02	0.8238724E-02	0.3775538E-14	0.1750204E-03
	H^1	0.7696003E-01	0.7667471E-01	0.7667471E-01	0.1754775E-12	0.1996121E-02
Smooth	L^2	0.1866377E-02	0.2717521E+00	0.3040389E+00	0.5685555E-14	0.8652625E-03
	H^1	0.2075523E-01	0.4026375E+01	0.5855023E+01	0.2720250E-12	0.1000046E-01
Trapezoidal	L^2	0.2190952E-02	0.2969728E+00	0.2780892E+00	0.2171915E-14	0.1994121E-14
	H^1	0.1882770E+00	0.2169223E+02	0.2072906E+02	0.3942341E-13	0.3158541E-13

least amount of "pollution" from the non-uniform grid. Interestingly, the pollution effect in the CVFEM-NSU and SUPG is accentuated especially strongly on the rectangular tensor grid, which is affine, and not on the $O(1)$ and the smooth grids, which are non-affine. CVFEM-MDEU preserves perfectly the one-dimensional character of the flow whereas the other two formulations do not.

Example 3: Constant advection. We solve (21) on 33×33 nonuniform grids and $\varepsilon = 1/2000$. The grids and the corresponding maximum element Peclet numbers are

- 27.1 for the $O(h^2)$ grid,
- 45.6 for the $O(1)$ grid,
- 39.5 for the tensor grid (26) with $\gamma = 0.1$,
- 34.2 for the smooth grid (27) with $\gamma = 0.5$.

Figures 10–13 summarize the results. The figures show that qualitatively the CVFEM-NSU and the SUPG are very similar. The SUPG is the least diffusive method, but has the largest overshoots and undershoots near the layers. CVFEM-MDEU is the most diffusive and the least-oscillatory scheme among the three.

While CVFEM-MDEU is more dissipative than CVFEM-NSU and SUPG it is significantly less dissipative than the standard artificial diffusion method. Results in Figures 14-15 clearly demonstrate this property of the new scheme.

Example 4: Double glazing. We solve (22) on 33×33 nonuniform grids and $\varepsilon = 1/2000$. The grids and the corresponding maximum element Peclet numbers are

- 62.1 for the $O(h^2)$ grid,
- 68.9 for the smooth (27) grid with $\gamma = 0.5$,

Figures 16–17 present the results. As before, the CVSEM-NSU and the SUPG perform similarly. Likewise, the CVFEM-MDEU is the most dissipative and the SUPG is the least dissipative from the three formulations. However, as the plots in Figures 18–19 show, the CVFEM-MDEU is significantly less dissipative than the classical artificial diffusion. These figures show that artificial diffusion completely destroys the distinguishing characteristics of the double glazing example, whereas the CVFEM-MDEU does not.

0.6 Conclusions

We presented an extensive computational study of the new CVFEM-MDEU formulation on a series of standard test problems for scalar advection-diffusion equations. Our study compares CVFEM-MDEU with two simplified versions of the method, nodal CVFEM with and without upwinding, SUPG and artificial diffusion stabilized Galerkin.

The objectives of our study are to 1) demonstrate that CVFEM-MDEU is indeed a truly multi-dimensional extension of the classical Scharfetter-Gummel upwinding, and 2) compare the new formulation with published algorithms.

The computational study in this report clearly demonstrates that CVFEM-MDEU performs well over a wide range of non-uniform grid, and so, it is indeed a proper multi-dimensional extension of Scharfetter-Gummel to arbitrary grids. The new formulation introduces more dissipation than SUPG but significantly less dissipation than an artificial diffusion approach. Most importantly, tests such as the double glazing problem demonstrate that the CVFEM-MDEU formulation does preserve important qualitative features of the solution that are completely destroyed by artificial diffusion.

Furthermore, in many cases the CVFEM-MDEU delivers nearly monotone solutions whose overshoots and undershoots are significantly smaller than those in the SUPG and the CVFEM-NSU solutions.

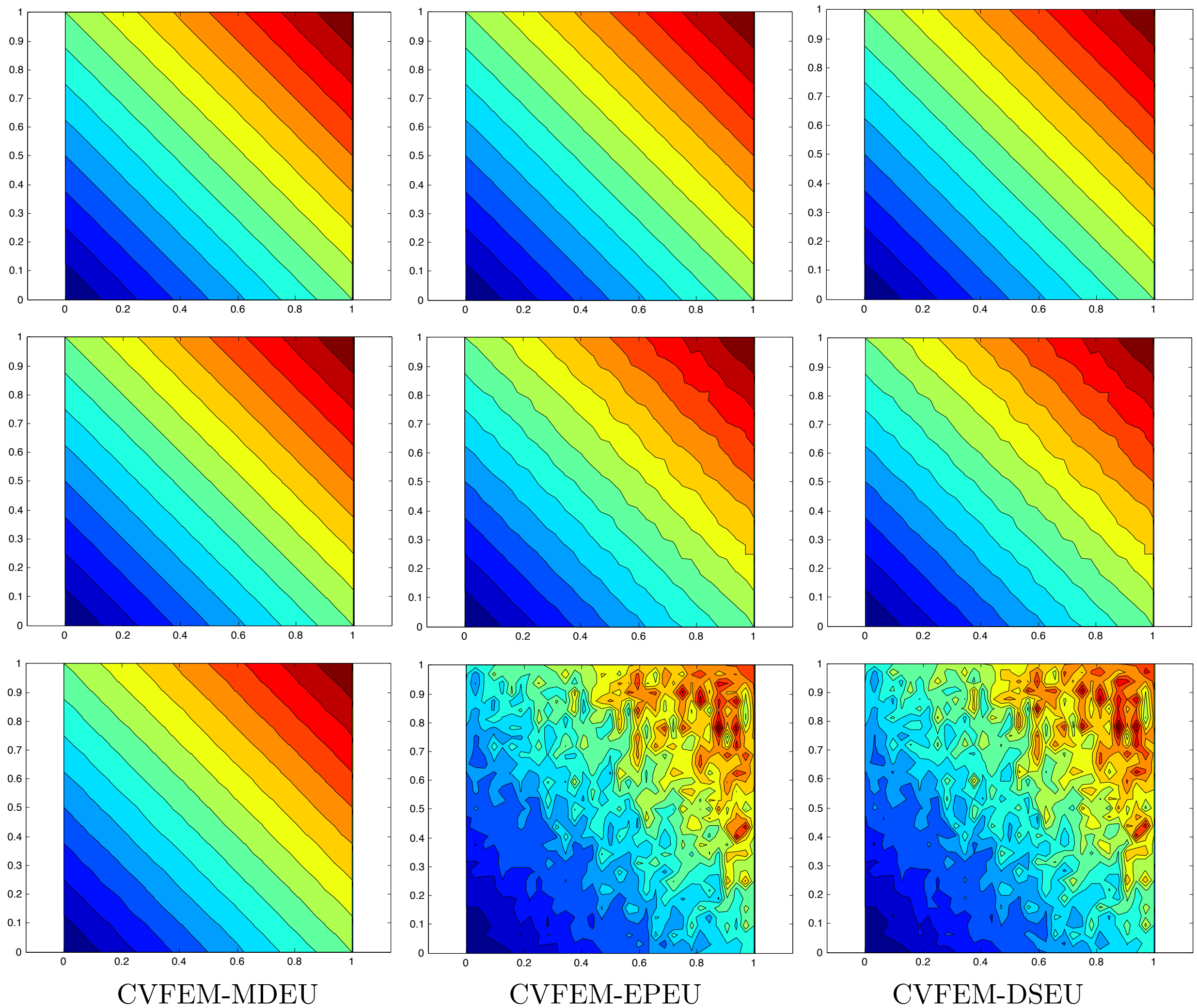


Figure 5: Approximation of a globally linear function by CVFEM on 33×33 randomly perturbed grids. Row 1: $O(h^2)$ grid; row 2: $O(h)$ grid; row 3: $O(1)$ grid. Column 1: CVFEM-MDEU; column 2: CVFEM-EPEU; column 3: CVFEM-DSEU.

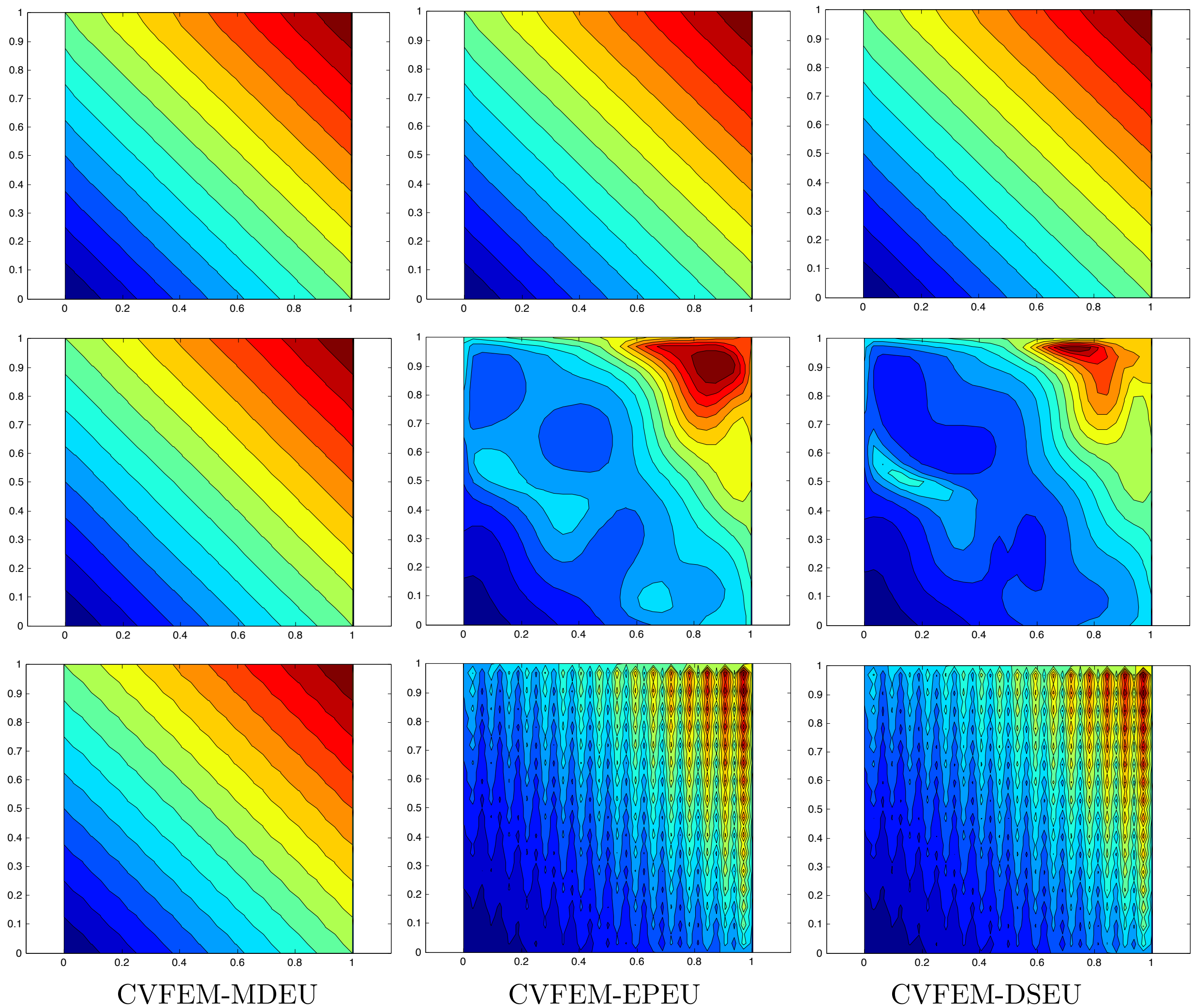
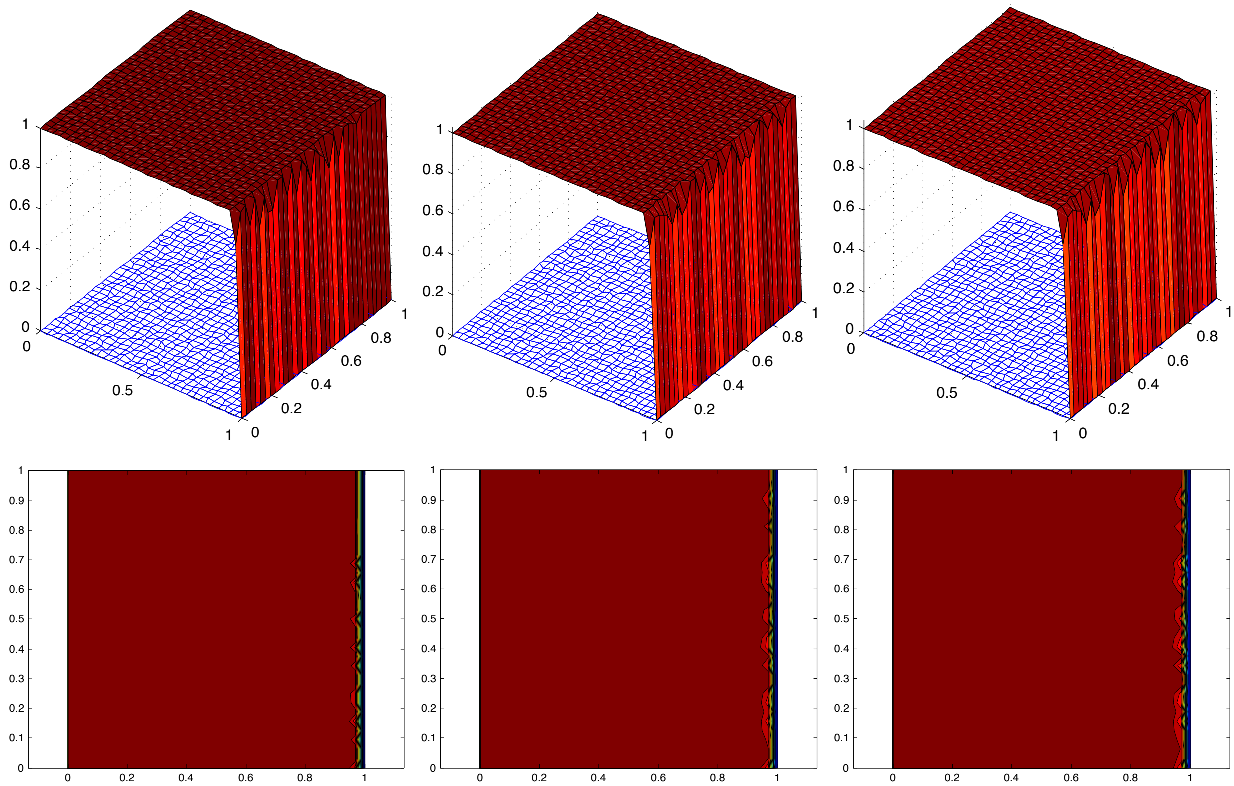
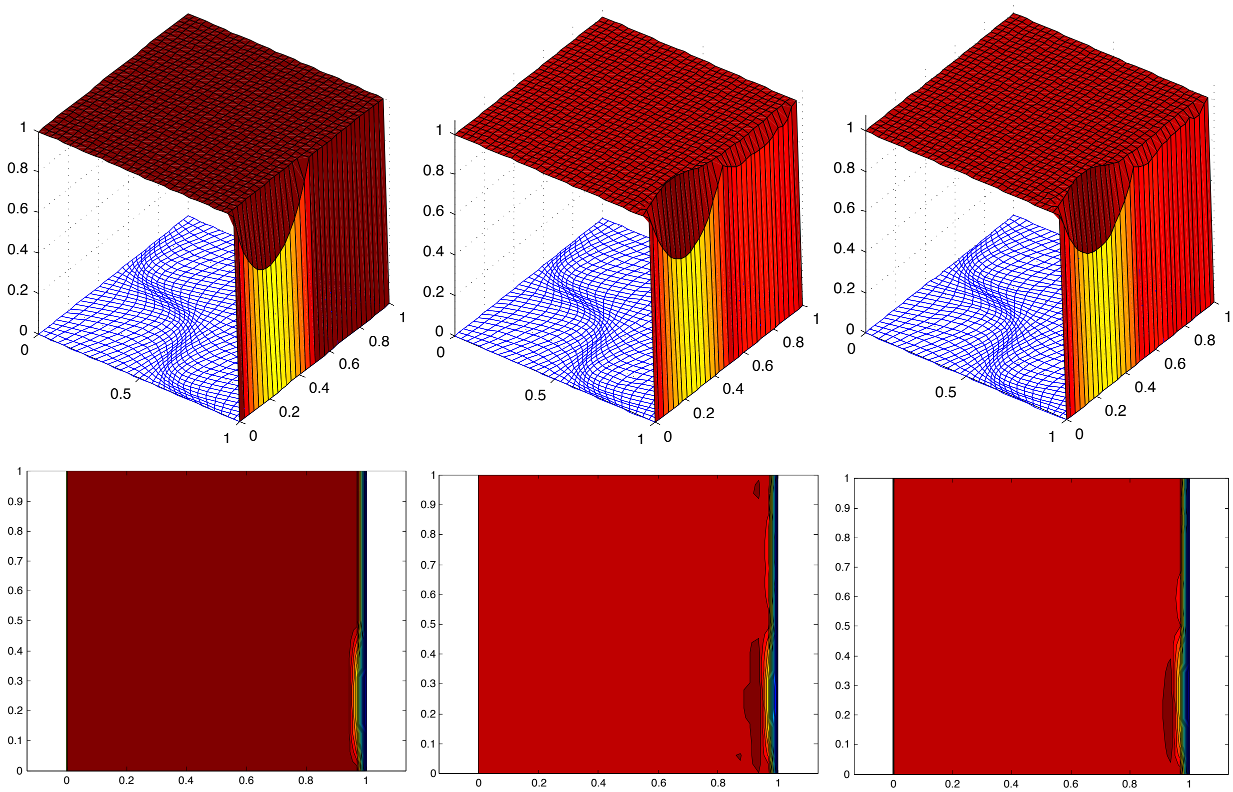


Figure 6: Approximation of a globally linear function by CVFEM on 33×33 structured nonuniform grids. Row 1: tensor product grid; row 2: smooth grid; row 3: trapezoidal grid. Column 1: CVFEM-MDEU; column 2: CVFEM-EPEU; column 3: CVFEM-DSEU.



CVFEM-MDEU CVFEM-NSU SUPG

Figure 7: Solution of the pseudo 1D test problem (20) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 $O(1)$ randomly perturbed grid.



CVFEM-MDEU CVFEM-NSU SUPG

Figure 8: Solution of the pseudo 1D test problem (20) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 smooth non affine grid (27).

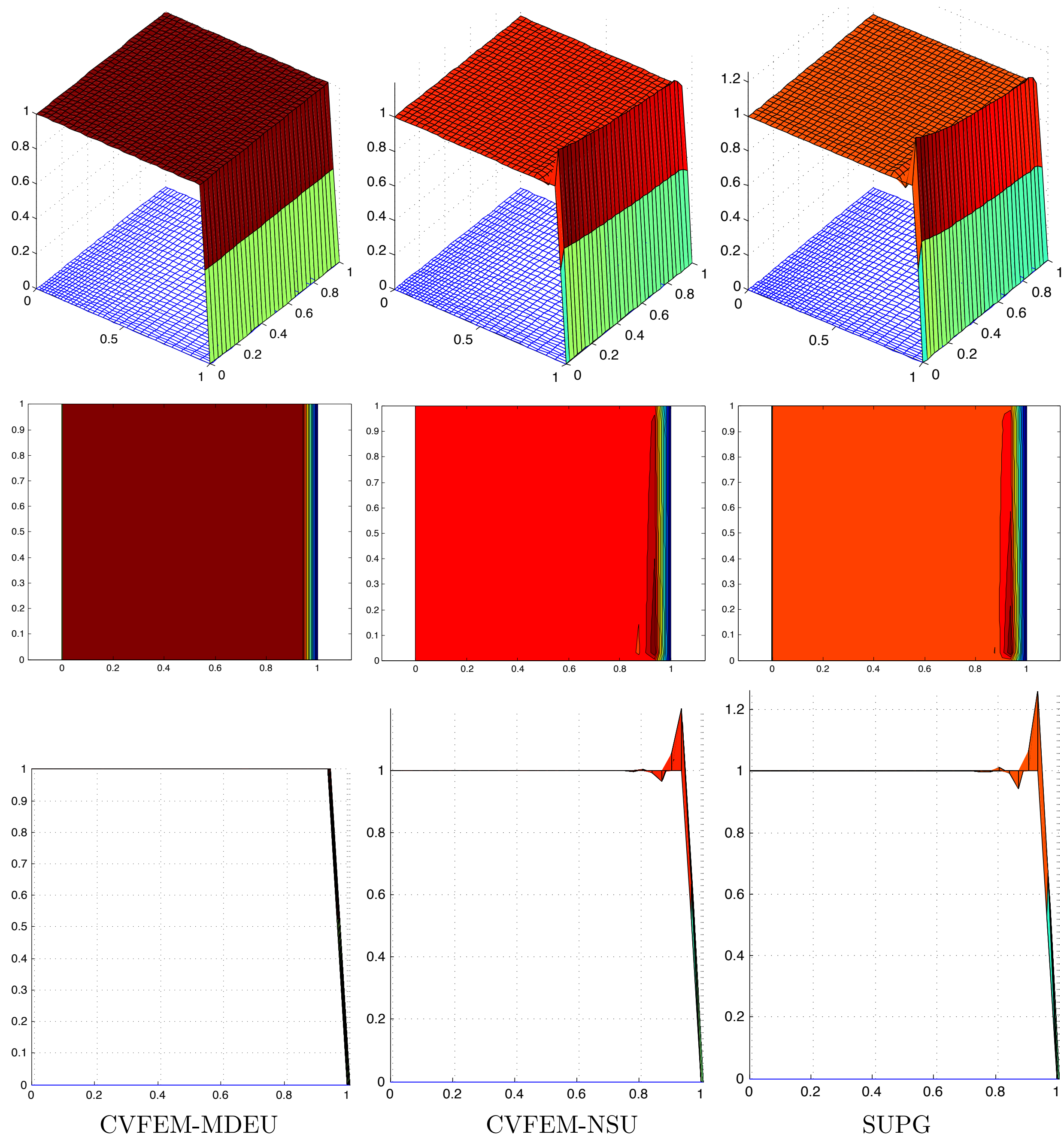


Figure 9: Solution of the pseudo 1D test problem (20) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 rectangular tensor grid (26) with $\gamma = 0.1$.

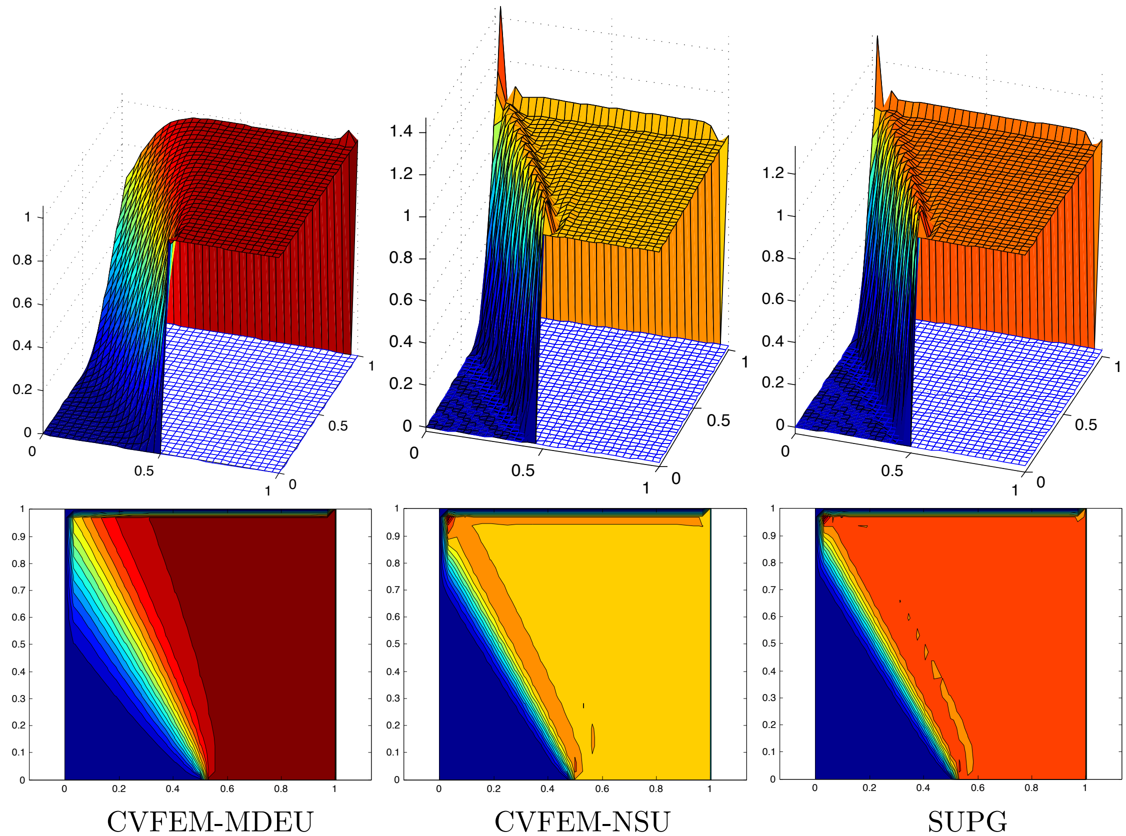


Figure 10: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 $O(h^2)$ randomly perturbed grid.

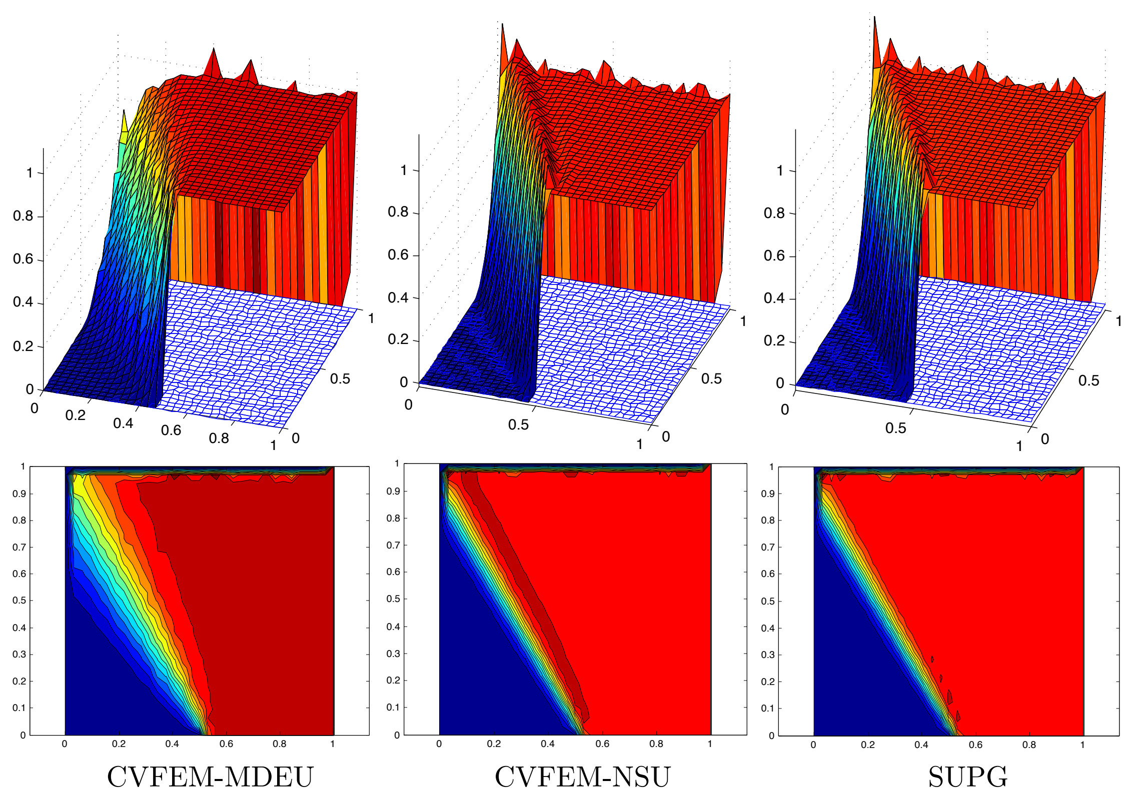


Figure 11: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 $O(1)$ randomly perturbed grid.

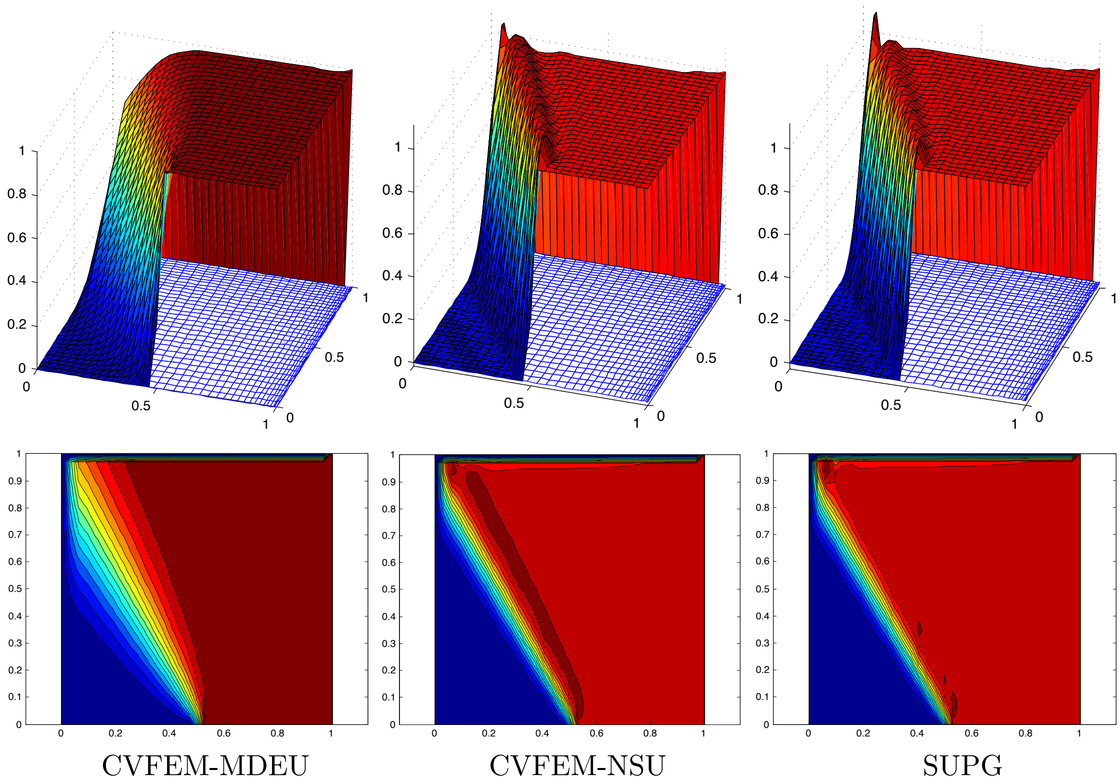


Figure 12: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 rectangular tensor grid (26) with $\gamma = 0.9$.

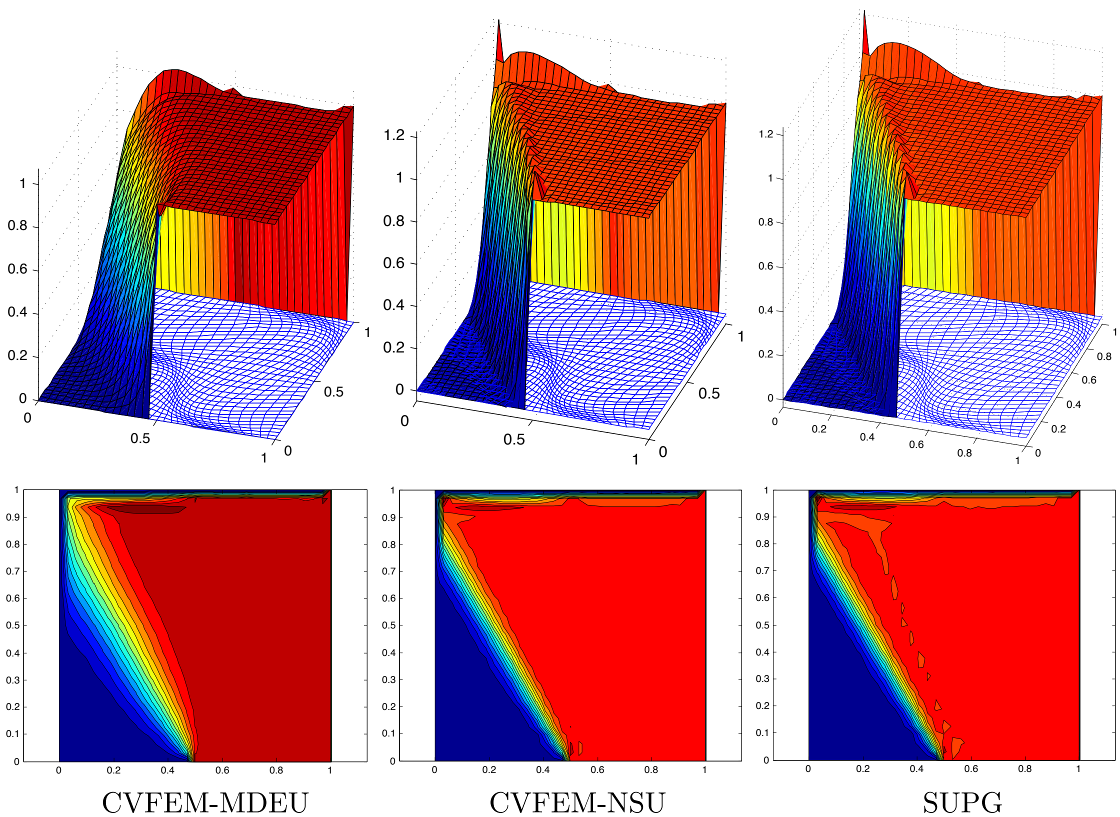
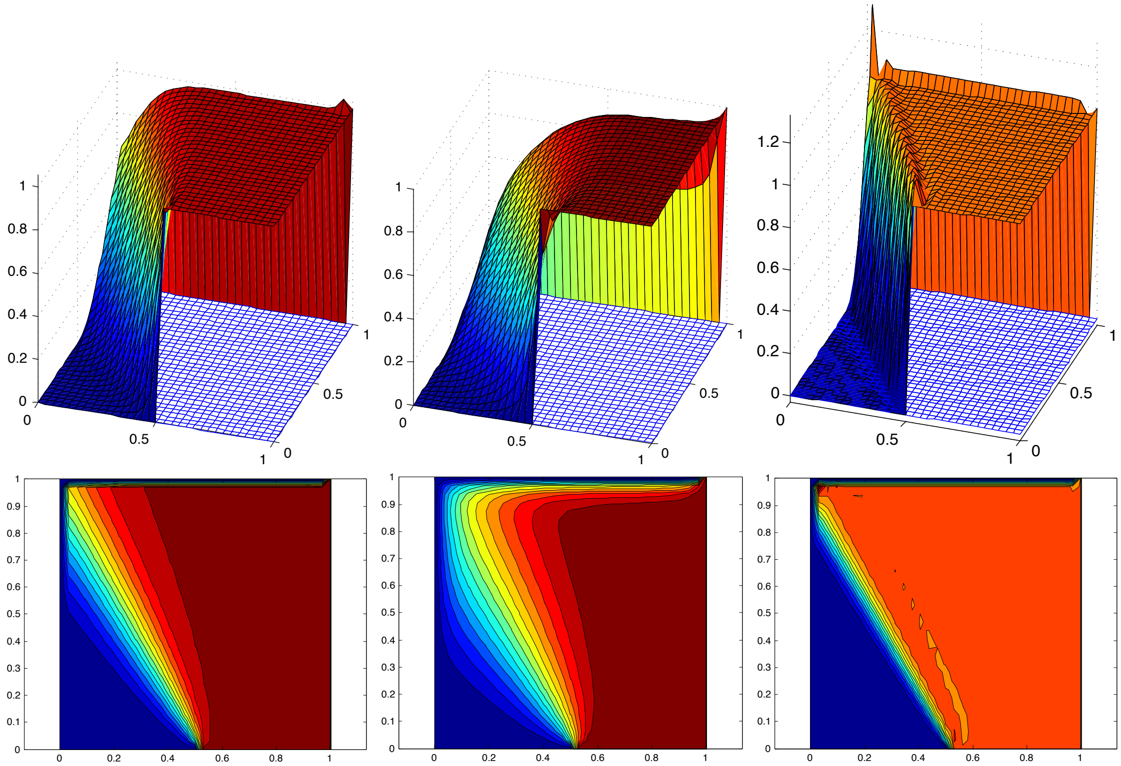
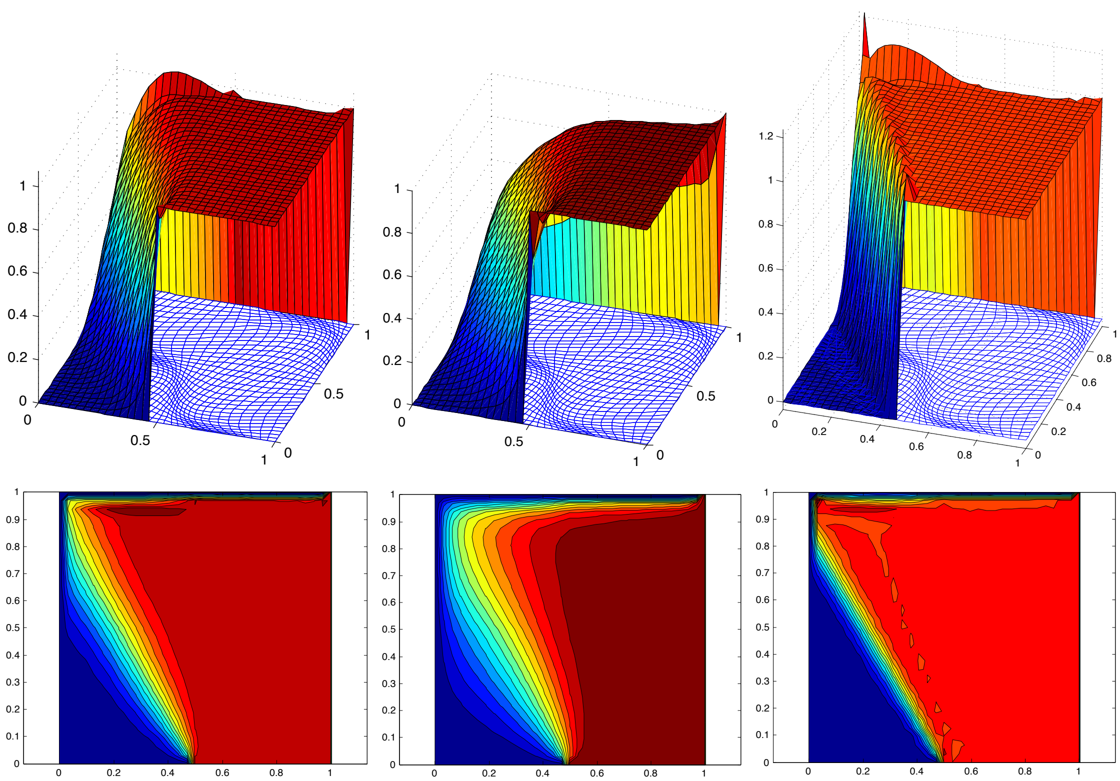


Figure 13: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 smooth non affine grid (27).



CVFEM-MDEU AD SUPG

Figure 14: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. Artificial Diffusion (middle) and SUPG (right) on a 33×33 $O(h^2)$ randomly perturbed grid.



CVFEM-MDEU AD SUPG

Figure 15: Solution of the constant advection test problem (21) by CVFEM-MDEU (left) vs. Artificial Diffusion (middle) and SUPG (right) on a 33×33 smooth non affine grid (27).

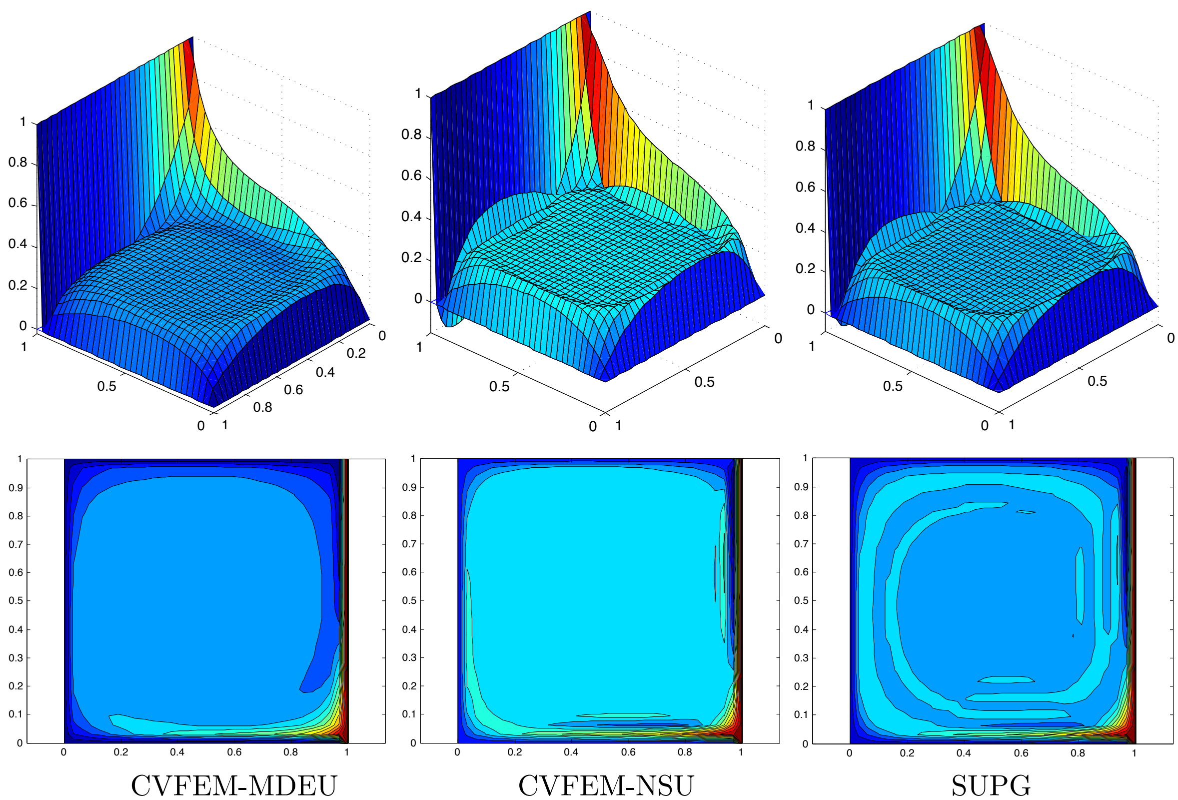


Figure 16: Solution of the double glazing test problem (22) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 $O(h^2)$ randomly perturbed grid.

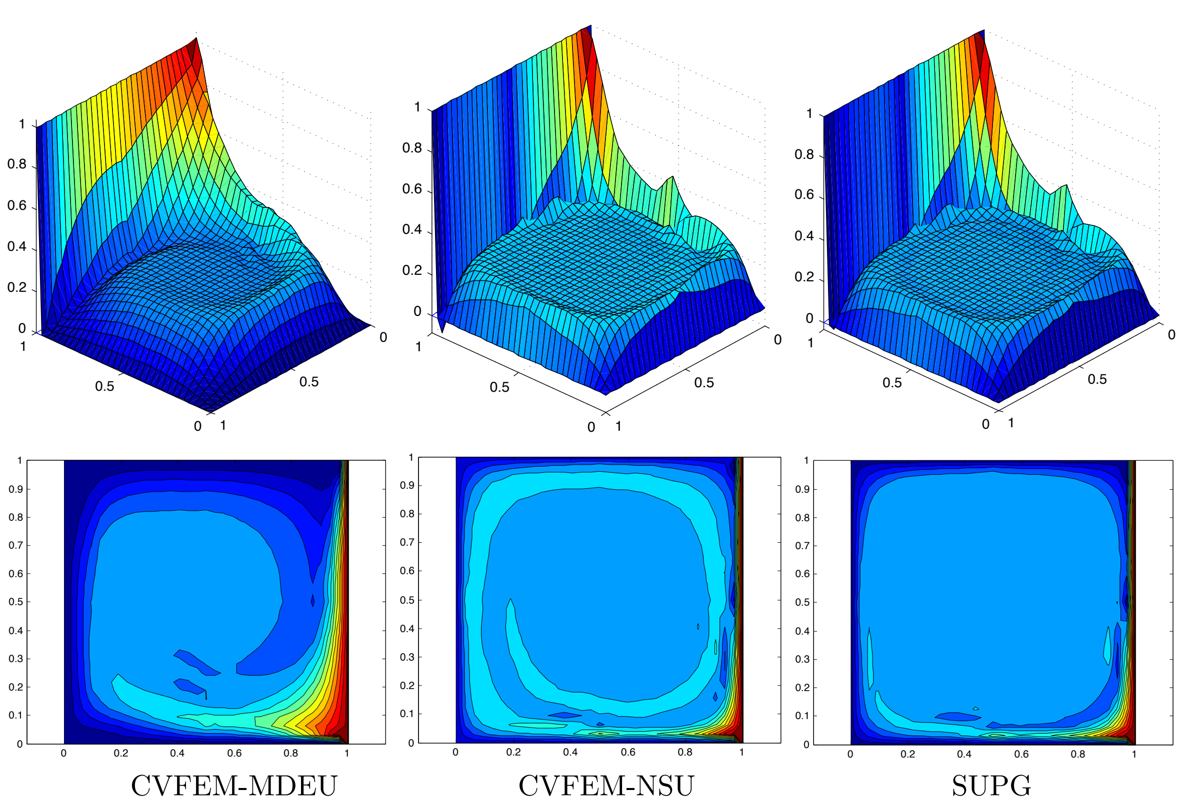


Figure 17: Solution of the double glazing test problem (22) by CVFEM-MDEU (left) vs. CVFEM-NSU (middle) and SUPG (right) on a 33×33 smooth non affine grid (27).

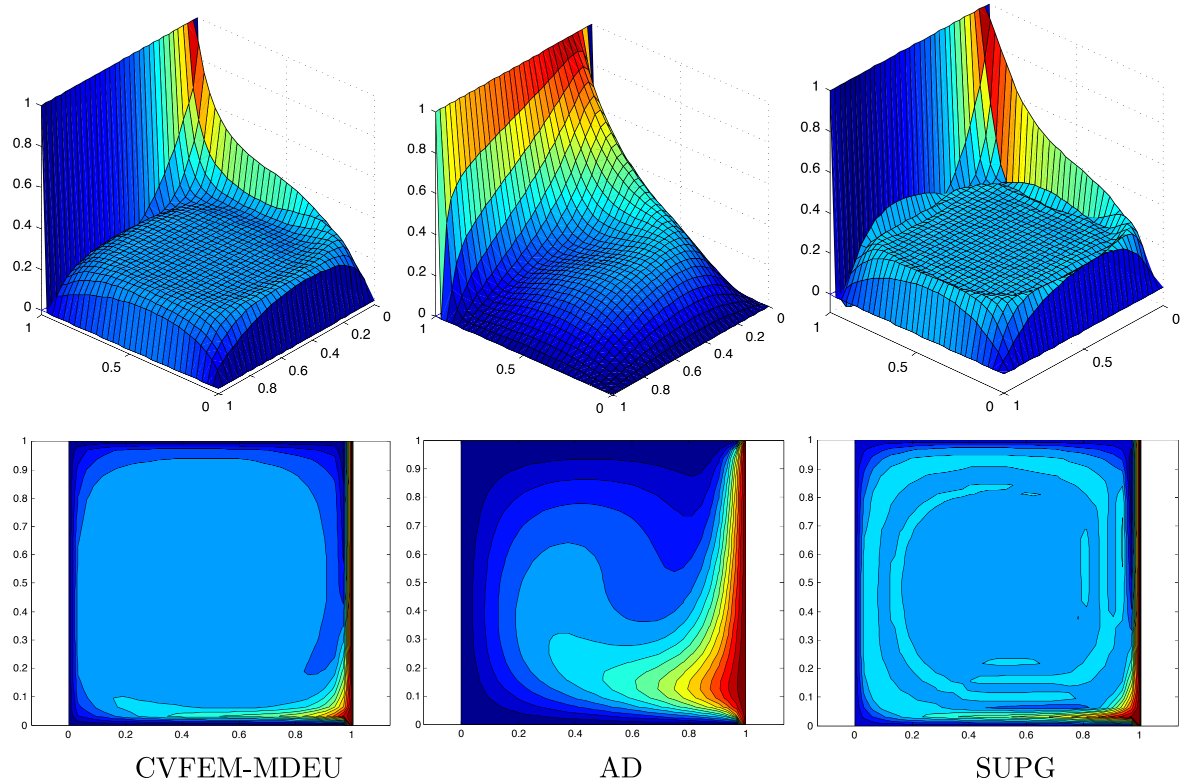


Figure 18: Solution of the double glazing test problem (22) by CVFEM-MDEU (left) vs. Artificial diffusion (middle) and SUPG (right) on a 33×33 $O(h^2)$ randomly perturbed grid.

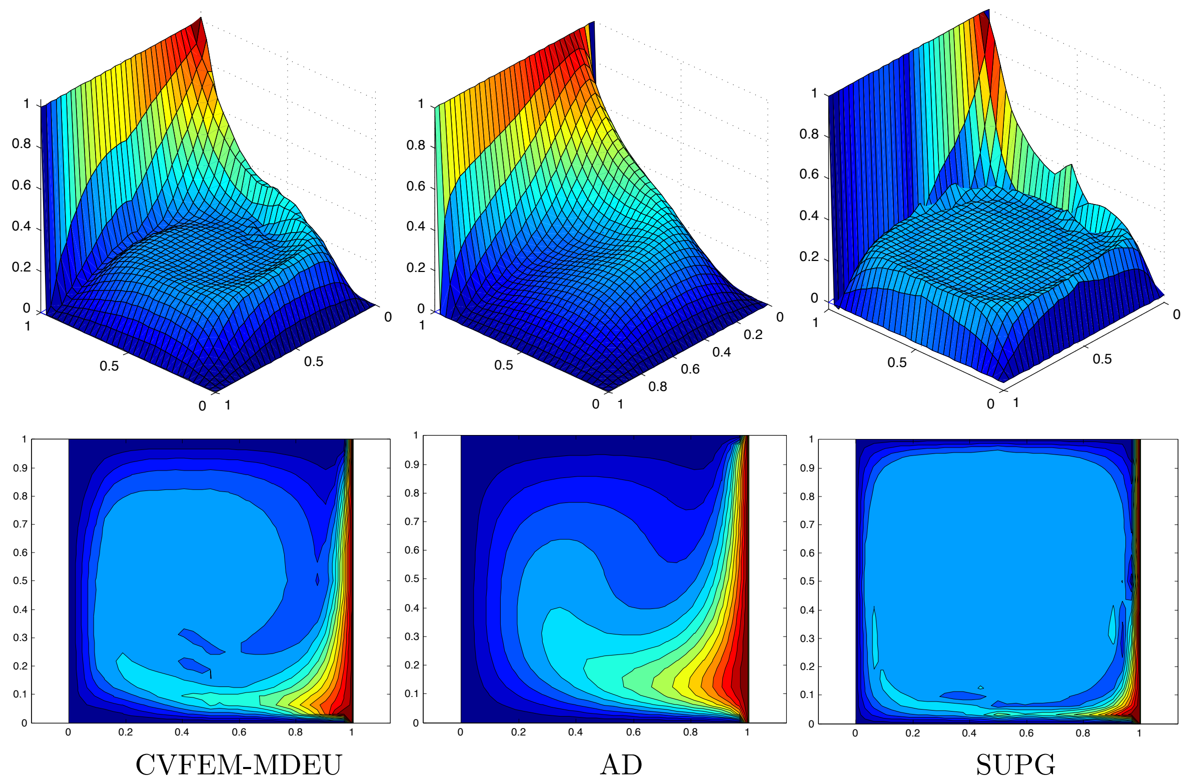


Figure 19: Solution of the double glazing test problem (22) by CVFEM-MDEU (left) vs. Artificial Diffusion (middle) and SUPG (right) on a 33×33 smooth non affine grid (27).

References

[1] D. N. Arnold, D. Boffi, and R. S. Falk. Quadrilateral $H(div)$ finite elements. *SIAM J. Numer. Anal.*, 42:2429–2451, 2005.

[2] B.R. Baliga and S.V Patankar. New finite element formulation for convection-diffusion problems. *Numerical Heat Transfer*, 3(4):393–409, October 1980.

[3] P. Bochev. Control volume finite element method with multidimensional edge element Scharfetter-Gummel upwinding. Part 1. formulation. Technical Report SAND 2011-3865, Sandia National Laboratories, Albuquerque, New Mexico, June 2011.

[4] H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics.* Numerical Mathematics and Scientific Computation. Oxford University Press, 2005.

[5] L. P. Franca, S. Frey, and T. J. R. Hughes. Stabilized finite element methods: I. Application to the advective–diffusive model. *Comput. Meth. Appl. Mech. Eng.*, 95:253–276, 1992.

[6] T. J. R. Hughes and A. Brooks. A theoretical framework for Petrov-Galerkin methods with discontinuous weighting functions: Application to the streamline-upwind procedure. In R. H. Gallagher et al, editor, *Finite Elements in Fluids*, volume 4, pages 47–65, New York, 1982. J. Wiley & Sons.

[7] Richard Liska, Mikhail Shashkov, Pavel Váchal, and Burton Wendroff. Optimization-based synchronized flux-corrected conservative interpolation (remapping) of mass and momentum for arbitrary Lagrangian-Eulerian methods. *J. Comput. Phys.*, 229:1467–1497, March 2010.

[8] L. G. Margolin and Mikhail Shashkov. Second-order sign-preserving conservative interpolation (remapping) on general grids. *Journal of Computational Physics*, 184(1):266 – 298, 2003.

[9] J. C. Nédélec. Mixed finite elements in $\mathbf{R}^3$. *Numerische Mathematik*, 35:315–341, 1980.

[10] C. R. Swaminathan, V. R. Voller, and S. V. Patankar. A streamline upwind control volume finite element method for modeling fluid flow and heat transfer problems. *Finite Elements in Analysis and Design*, 13(2-3):169 – 184, 1993.

DISTRIBUTION:

1 Gabriel Barrenechea
Department of Mathematics and Statistics
University of Strathclyde
26 Richmond Street
Glasgow, G1 1XH
Scotland, United Kingdom

1 Eric Burman
Department of Mathematics
University of Sussex
Falmer, Brighton
BN1 9QH United Kingdom

1 Max Gunzburger
Francis Eppes Distinguished Professor of Mathematics
School of Computational Science
400 Dirac Science Library
Florida State University Tallahassee, FL 32306-4120

1 Donald Estep
Department of Mathematics
Colorado State University
Fort Collins, Colorado 80523

1 Leo Franca
Department of Mathematics
University of Colorado at Denver
Denver, Colorado 80217-3364

1 Thomas Hughes
The University of Texas at Austin
Institute for Computational Engineering and Sciences
1 University Station C0200
Austin, TX 78712-0027

1 Arif Masud
Department of Civil and Environmental Engineering
University of Illinois at Urbana-Champaign
3110 Newmark Civil Engineering Laboratory
205 N. Mathews St., Urbana
Illinois 61801, USA

1 David Silvester
School of Mathematics
University of Manchester
Alan Turing Building
Manchester M13 9PL, United Kingdom

1 Garth Wells
Cambridge University
Engineering Department
Trumpington Street
Cambridge CB2 1PZ, United Kingdom

1	MS 1320	John Aidun, 1445
2	MS 1320	Pavel Bochev, 1442
1	MS 1320	Scott Collis, 1442
1	MS 1320	Eric Cyr, 1426
1	MS 0346	Clark Dohrman, 1523
1	MS 1541	Stefan Domino, 1541
1	MS 0316	Garry Hennigan, 1425
1	MS 0316	Scott Hutchinson, 1445
1	MS 1318	Rob Hoekstra, 1426
1	MS 0316	Larry Musson, 1425
1	MS 0836	Patrick Notz, 1541
1	MS 1320	John Shadid, 1444
1	MS 0316	Tom Smith, 1442
1	MS 1320	Roger Pawlowski, 1444
2	MS 1320	Kara Peterson, 1442
1	MS 0836	Rekha Rao, 1514
1	MS 1320	Denis Ridzal, 1441
1	MS 0378	Allen Robinson, 1443
1	MS 1321	John Shadid, 1444
1	MS 0836	Randall Schunk, 1516
1	MS 1318	Bart van Bloemen Waanders, 1442
1	MS 9409	Greg Wagner, 8365
1	MS 0380	David Womble, 1540
2	MS 0612	Review & Approval Desk, 4916
1	MS 0899	Technical Library, 9536 (electronic)

Sandia National Laboratories