

A Stabilised Face-Upwinded High-Order Method for Incompressible Flows

Yulong Pan*

University of California, Berkeley, Berkeley, CA 94720-3840, U.S.A.

Per-Olof Persson†

University of California, Berkeley, Berkeley, CA 94720-3840, U.S.A.

We have applied the FUSE method (Face-Upwinded Spectral Elements) to incompressible flow problems. This approach utilizes a continuous solution field on unstructured meshes of quadrilateral elements, with stabilisation achieved through upwinding only on the element faces. By selecting optimal node locations, the method provides a provably convergent scheme for linear convection at any polynomial degree. This technique offers significant computational advantages due to its use of continuous spaces and line-based connectivities. Our study extends these results to the incompressible Navier-Stokes equations, including the upwind-downwind discretisation of the divergence and gradient operators. Notably, the scheme exhibits inf-sup stability, even for equal approximation degrees of velocity and pressure, unlike standard continuous Galerkin discretisations. To evaluate the method’s properties, we solve two standard model problems. These results have implications for the development of efficient and stable numerical methods for fluid dynamics.

I. Introduction

Over the last few decades, significant research has been dedicated to the development of stable, high-order accurate numerical methods for convection-dominated flow problems. While the continuous Galerkin finite element method is a popular approach, it requires specialised stabilisation techniques. Methods based on artificial diffusion such as SUPG, VMS, and spectrally vanishing viscosity [1, 2] have been proposed, but for various reasons, they are not widely used by the high-order computational fluid dynamics community. Consequently, researchers have developed several methods based on discontinuous solution fields, with the discontinuous Galerkin (DG) method [3–5] being the most popular. It offers provable linear stability for any polynomial degree and element shape. Related methods include DG-SEM [6, 7], spectral difference and spectral volume [8–10], and flux reconstruction [11], which can be shown to be identical in some special cases [12], but in general they have different properties.

Our Face-Upwinded Spectral Element (FUSE) [13] method aims to combine the ideas from the discontinuous methods to obtain new stabilization on continuous solution fields. The approach is straightforward: we employ a standard spectral differentiation technique for all interior nodes, while an upwinded high-order stencil is utilised only for the nodes on the element faces. This methodology, along with a unique set of node locations, results in a linearly stable scheme for any degree. The primary motivation behind this method is the reduced cost, both due to the fewer degrees of freedom and the sparse Jacobian matrices. However, we observe other advantages, such as superior CFL conditions and the potential for using better solves like static condensation, which is less clear how to apply to the discontinuous methods.

In this study, we extend the FUSE method for solving the unsteady incompressible Navier-Stokes equations. We use an upwind-downwind split form inspired by the Local Discontinuous Galerkin (LDG) method [14] to discretise the gradient and divergence operators. Our approach provides inf-sup stability for all test problems, even when equal degrees are used to approximate the velocities and pressures. To demonstrate the stability and accuracy of the method, we solve two standard model problems: the lid-driven cavity and flow around a cylinder.

*Graduate Student, Department of Mathematics, University of California, Berkeley, CA, AIAA Student Member. Email: yllpan@berkeley.edu.

†Professor, Department of Mathematics, University of California, Berkeley, CA, AIAA Senior Member. Email: persson@berkeley.edu.

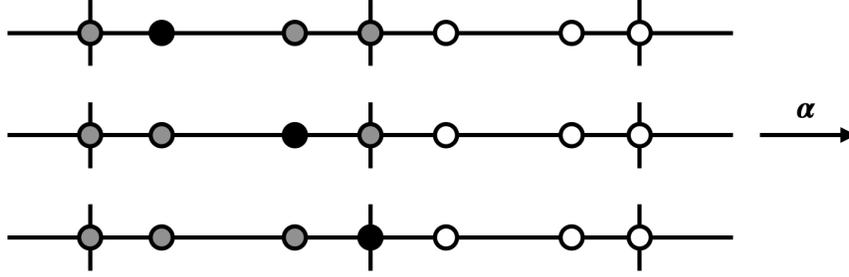


Fig. 1 FUSE stencil selection in 1D for $p = 3$. Velocity is assumed to be positive and shaded points are used to form first derivatives at points shown in black. For non-endpoint nodes inside an element the points of that element are always used. For endpoints nodes the upwind element is chosen, in this case the left one.

II. Face-Upwinded Spectral Elements

We briefly describe the Face-Upwinded Spectral Element (FUSE) method and refer the reader to [13] for more details. To describe the method we first consider the advection equation in 1D

$$\frac{\partial c}{\partial t} + \alpha \frac{\partial c}{\partial x} = 0 \quad (1)$$

The problem is defined over domain $\Omega = [0, 1]$ with periodic boundary conditions. Under this equation the initial condition $c_0(x)$ is simply advected with velocity $|\alpha|$ either to the right or left depending on whether the sign of α is positive or negative respectively.

The FUSE method first discretises the domain Ω into distinct elements $\{E_i, i = 0, \dots, n-1\}$ on which given a polynomial degree p a set of $p+1$ nodes $\{s_i, i = 0, \dots, p\}$ is distributed. The solution $c(x)$ is discretised using these nodes as a piecewise polynomial of degree p . To ensure continuity two nodes s_0, s_p are always placed on the endpoints of each element and unlike the Discontinuous Galerkin method nodes on endpoints of adjacent elements are not duplicated. This results in a node distribution that is consistent with that of Finite Element methods.

Inside each element the interior $p-1$ nodes $\{s_1, \dots, s_{p-1}\}$ are placed at the standard Gauss-Legendre quadrature points. As is discussed in [13] this is to ensure stability of the resulting first derivative operators. Under this construction the lowest possible order for which the method is defined is then for $p=1$, where the only points in each element are the two endpoints.

Given an arbitrary function $f(x)$, the function is approximated with this method as a continuous piecewise polynomial $\tilde{f}(x)$ of degree p . One possible way to represent this polynomial on an element E_k is to define a set of interpolatory polynomials $\{\phi_i(x), i = 0, 1, \dots, p\}$ on E_k , such that $\phi_i(x(s_j)) = \delta_{ij}$ where δ_{ij} is the Kronecker delta. The approximation $\tilde{f}(x)$ on E_k is given as a linear combination of these basis functions $\tilde{f}(x) = \sum_j f_j \phi_j(x)$, where $f_j = f(x(s_j))$ is the value of the function $f(x)$ at the node s_j .

The first derivative $\frac{\partial}{\partial x}$ is discretised at each node using exactly one element E_k to which the node belongs. For a node on the interior of an element E_k that is not an endpoint, the element to which the node belongs is used always to form the derivative. For endpoint nodes on the boundary between two elements, the upwind element is the one used to form the derivative: for $\alpha \geq 0$ the left element is chosen where is $\alpha < 0$ the right element is instead the one chosen. The upwinding on element boundaries plus specific node location choice stabilises the first derivative operator and is shown in Fig. 1 for $p=3$.

There are two possible ways to discretise the first derivative of a given function $\frac{\partial f}{\partial x}$ with this method on an element E_k . The first is to simply differentiate the polynomial approximation of the function $f'(x) \approx \tilde{f}'(x)$. The second considers the equation $q(x) = \frac{\partial f}{\partial x}$ to which we multiply a test function $w(x)$ to each side and integrates over the element E_k

$$\int_{E_k} q(x)w(x)dx = \int_{E_k} \frac{\partial f(x)}{\partial x} w(x)dx \quad (2)$$

As this integral is performed inside a single element E_k , the test functions can be chosen to be each of the basis functions $\phi_i(x)$ and the derivative represented as a linear combination of the basis functions $q(x) = \sum_j q_j \phi_j(x)$. This gives a set

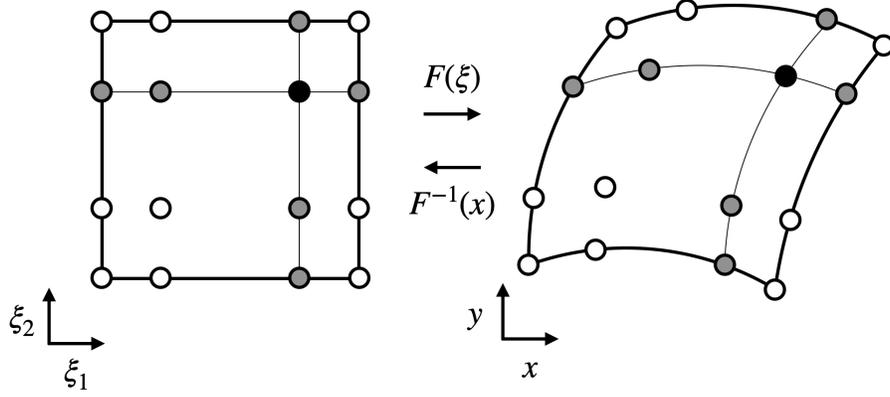


Fig. 2 Schematic of line-based derivative discretisation for point shown in black. Elements are mapped from physical space to reference space $\xi = F^{-1}(x)$ where stencils are constructed using the two 1D lines defined by the tensor product node structure.

of linear equations

$$\sum_j \int_{E_k} \phi_i(x) \phi_j(x) dx q_j = \int_{E_k} \frac{\partial f(x)}{\partial x} \phi_i(x) dx, \quad \forall i = 0, \dots, p \quad (3)$$

which can be solved for the unknowns q_j . Both methods are equivalent for a linear function $f(x)$ with the first being the more efficient. For nonlinear $f(x)$ however the second while more involved may be necessary to avoid issues such as aliasing.

A. Higher dimensions

The FUSE method has thus far been developed only for quad-based discretisations in higher dimensions, that is quadrilateral meshes in 2D and hexahedral meshes in 3D. The model problem considered in higher dimensions is as before the advection equation

$$\frac{\partial c}{\partial t} + \alpha \cdot \nabla c = 0 \quad (4)$$

on a domain $\Omega = [0, 1]^2$ with periodic boundary conditions and α the velocity. The domain is as in 1D divided up into distinct elements and given a polynomial degree p on each element a set of $(p+1)^d$ nodes is introduced, which is chosen simply to be the tensor product of $p+1$ the 1D nodes. Nodes on element boundaries are not duplicated resulting in a Finite Element like distribution of nodes.

In d -dimensions, exactly d lines passing through a point are needed to discretise the first derivative at the point. To do this each element can be mapped from reference space $\xi = [0, 1]^d$ to physical space x via a diffeomorphic map $F(\xi) = x$. On each element exactly d lines made up of $p+1$ points are uniquely defined by the tensor product structure of the node distributions as shown in Fig. 2. Derivatives can then be formed on each of these lines in reference space $\xi_i, i = 1, \dots, d$ and mapped back to physical space for instance in 2-dimensions via the mapping

$$\begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} = \underbrace{\begin{pmatrix} \frac{\partial \xi_1}{\partial x} & \frac{\partial \xi_2}{\partial x} \\ \frac{\partial \xi_1}{\partial y} & \frac{\partial \xi_2}{\partial y} \end{pmatrix}}_{=(\nabla F)^{-1}} \begin{pmatrix} \frac{\partial}{\partial \xi_1} \\ \frac{\partial}{\partial \xi_2} \end{pmatrix} \quad (5)$$

Derivatives are formed at each node as with the 1D case using the points of exactly one element to which the node belongs. For a node interior to an element E_k that is not on the element boundary, the element E_k is always the one used to form the derivative. For a node on an element boundary an iterative element selection procedure is performed using the pointwise velocity α at the node. Enumerating all the lines originating from the point as $\{l_i\}$, the line most upwind to the pointwise velocity is chosen by minimising the projection of the velocity onto the lines

$$k_0 = \operatorname{argmin}_i \frac{\alpha \cdot l_i}{\|\alpha\| \|l_i\|} \quad (6)$$

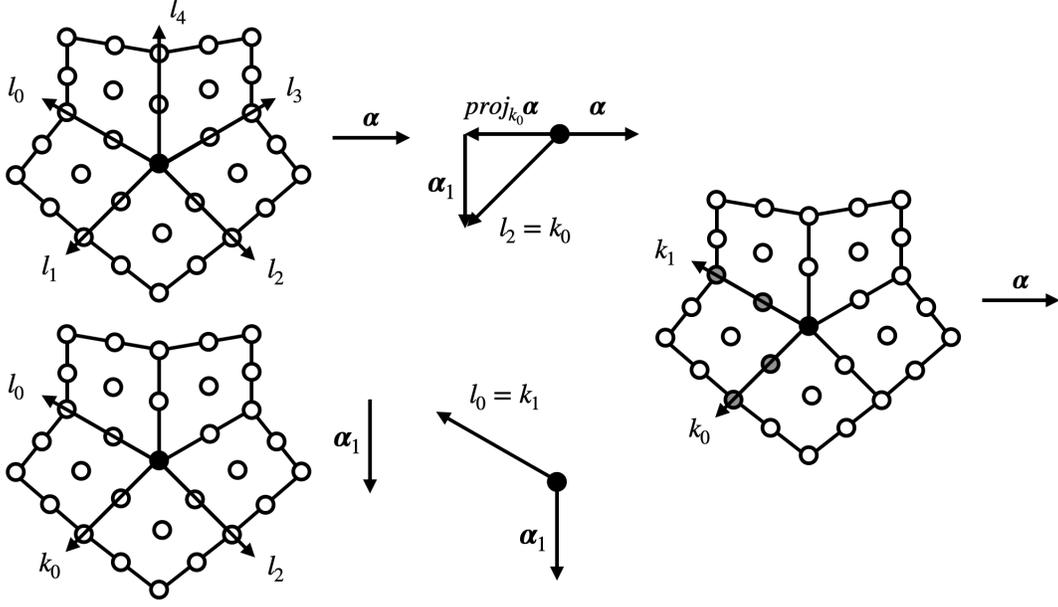


Fig. 3 Upwind element selection procedure in 2D with the point of interest is shown in black. 1) All possible lines $\{l_i, i = 0, \dots, 4\}$ are enumerated and the most upwind line k_0 to the velocity α is found as l_2 . The projection $\text{proj}_{k_0} \alpha$ is then subtracted from the original velocity to give intermediate velocity α_1 as shown in the top left. 2) The lines l_0, l_2 that belong to the same elements as k_0 are enumerated and the most upwind line k_1 to the velocity α_1 is found as l_0 as shown in the bottom left. 3) The two lines k_0, k_1 with points shaded are used to discretise the derivative at the point of interest as shown on the right.

The projection of the velocity onto this line $\text{proj}_{k_0} \alpha$ is then subtracted from the original velocity to give an intermediate velocity of $\alpha_1 = \alpha - \text{proj}_{k_0} \alpha$. Next, the lines $\{l_i\}$ that belong to the same elements as k_0 are searched over to find the line most upwind to α_1

$$k_1 = \underset{l_i \neq k_0, l_i \in \text{same elements as } k_0}{\text{argmin}} \frac{\alpha_1 \cdot l_i}{\|\alpha_1\| \|l_i\|} \quad (7)$$

This process is then repeated until exactly d lines are chosen. This ensures that lines from exactly one element are chosen and two instances of the procedure is shown in Figs. 3 and 4.

B. Second derivatives

As noted in [13], there exists more than one way to discretise second derivative operators using this method. One could take advantage of the fact that the FUSE grid has an identical distribution to that of Finite Elements, and as such use Finite Elements to discretise second derivatives on the same grid. For this paper however we adopt the approach of the Local Discontinuous Galerkin inspired split form construction [14]. Considering the model second derivative equation

$$-\nabla^2 c = f \quad (8)$$

on a domain $\Omega = [0, 1]^2$ with periodic boundary conditions, this can be rewritten into a system of first derivative equations via introducing a new variable $\mathbf{q} = \nabla c$

$$\mathbf{q} = \nabla c \quad (9)$$

$$-\nabla \cdot \mathbf{q} = f \quad (10)$$

Each of these equations can be discretised using the first derivative techniques described previously in the section. For the upwinding, an arbitrary constant non-zero velocity field α can be used for the first equation Eq. 9, as long as the opposite constant velocity field $-\alpha$ is used for the second equation Eq. 10. This upwind-downwind construction of the second derivative operator ensures that for each node between element boundaries different elements are chosen for the two Eqs. 9-10 which is necessary for stability of the resulting Laplace operator.

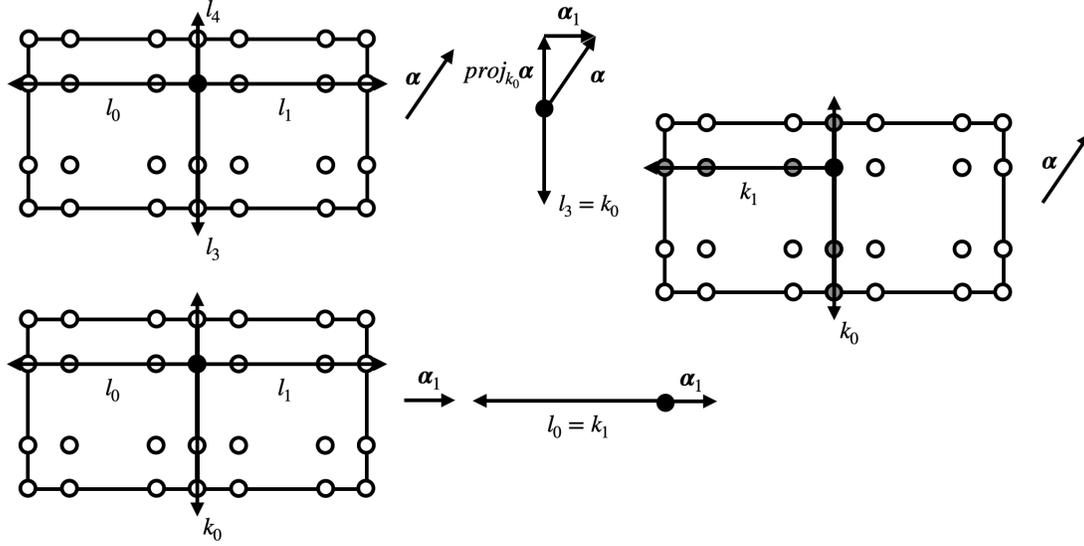


Fig. 4 Upwind element selection procedure in 2D with the point of interest is shown in black. 1) All possible lines $\{l_i, i = 0, \dots, 3\}$ are enumerated and the most upwind line k_0 to the velocity α is found as l_3 . In this case the entire line l_3 and l_4 are in fact chosen as they belong to the same overall line on the element. The projection $\text{proj}_{k_0} \alpha$ is then subtracted from the original velocity to give intermediate velocity α_1 as shown in the top left. 2) The lines l_0, l_1 that belong to the same elements as k_0 are enumerated and the most upwind line k_1 to the velocity α_1 is found as l_0 as shown in the bottom left. 3) The two lines k_0, k_1 with points shaded are used to discretise the derivative at the point of interest as shown on the right.

III. Governing Equations and Discretisation

We focus on solving the incompressible Navier-Stokes equations in 2-dimensions, with the variables \mathbf{u}, p denoting the velocity and pressure. The equations are written in convective form to clearly show that the velocity field \mathbf{u} is used to form the upwind gradient operators described in the previous section

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} - \mu \nabla^2 \mathbf{u} + \nabla p = \mathbf{0} \quad (11)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (12)$$

Here the density has been assumed to be a constant $\rho = 1$ and μ is the viscosity. The equations can be non-dimensionalised by introducing characteristic length and velocity scales for the flow L, U respectively such that

$$\begin{aligned} \mathbf{u}^* &= \frac{\mathbf{u}}{U} \\ \nabla^* &= L \nabla \\ t^* &= \frac{t}{L/U} \end{aligned}$$

giving the non-dimensional time-dependent incompressible Navier-Stokes equations

$$\frac{\partial \mathbf{u}^*}{\partial t^*} + (\mathbf{u}^* \cdot \nabla^*) \mathbf{u}^* - \frac{1}{\text{Re}} \nabla^{*2} \mathbf{u}^* + \nabla^* p^* = \mathbf{0} \quad (13)$$

$$\nabla^* \cdot \mathbf{u}^* = 0 \quad (14)$$

where $\text{Re} = \frac{UL}{\mu}$ denotes the non-dimensional Reynolds number of the system measuring the ratio of convection to diffusion in the problem. For high Reynolds numbers the problem is convection-dominated which require stabilised methods in order to be resolved numerically in a stable manner.

A. FUSE discretisation

To discretise Eq. 11 the discrete gradient operator ∇_h is formed using the upwind procedure outlined above with an arbitrary velocity field α used to determine the choice of upwind element. The discrete divergence operator $\nabla_h \cdot$ is formed in a similar fashion but instead $-\alpha$ is used to determine the choice of upwind element. The discrete Laplace operator is then formed using the LDG inspired methodology by composing the discrete divergence and gradient $\nabla_h^2 = \nabla_h \cdot \nabla_h$. In this paper we make the specific choice of $\alpha = \mathbf{u}$ for our examples as this allows us to use the same discrete gradient and divergence operators for all the terms in Eqs. 11-12.

Special care must be taken when using this choice of upwind velocity $\alpha = \mathbf{u}$ for forming the discrete Laplace operator at nodes on the boundary of an element where the pointwise velocity is equal to zero. In this case it is possible in this case to pick the same upwind element at that point for both the discrete gradient and discrete divergence operators using the aforementioned upwind element selection procedure. To deal with this an arbitrary random pointwise velocity $\alpha \neq \mathbf{0}$ can be introduced at these points for the upwind element selection procedure for the discrete gradient. For the discrete divergence this nonzero velocity $-\alpha$ can then be used for upwind element selection to ensure that opposite upwind-downwind elements are chosen for the two operators. This is necessary to give stability of the discrete Laplace operator whilst being safe to do as it does not affect the stability of the convection term $(\mathbf{u} \cdot \nabla_h)\mathbf{u}$ since the velocity at these points is equal to zero.

B. Inf-sup condition

The inf-sup condition is a necessary and sufficient condition that ensures existence and uniqueness of solutions to mixed problems and must be satisfied discretely to avoid certain instabilities in the solution, known as inf-sup instability. Our choice of using the LDG inspired construction for the discrete Laplacian is motivated by the result stated in [15] and proved in [16] on this condition. The result states that for stabilised mixed methods, the classical inf-sup condition can be circumvented even if equal degree p polynomials are used for both the velocity and pressure. This is in contrast to Finite Element methods which are not stabilised and it is well known that different polynomial orders for the velocity and pressure are required to satisfy the condition. In our numerical examples we verify that we do not observe the "checkerboard" instabilities arising from not satisfying the inf-sup condition despite using equal p for both the velocity and pressure fields.

IV. Numerical Examples

We consider two examples in 2D at moderate to high Reynolds numbers to validate the method for incompressible flow problems. For unsteady flows the time integration method used in each case is the Crank-Nicolson method was employed with timestep $\Delta t = 1/400$. Newton's method was used to solve the implicit problem at each timestep with the tolerance for the residual set at 10^{-8} . We verify in both examples that inf-sup instabilities are not triggered and that smooth fields in both velocities and pressure are obtained.

A. Flow around a cylinder

For our first example we consider the problem of flow around a cylinder with the setup shown in Fig. 5. The cylinder is centred at the origin $(0, 0)$ with a radius of 0.05. The bottom of the domain is set at 0.2 and the top at 0.21. Finally the left boundary is placed at a distance 0.2 away from $x = 0$, and the right boundary at $x = 2.0$.

The mesh used for this example at zero refinements is shown in Fig. 6. Extra resolution is added in around the cylinder to better resolve the flow at that boundary, which is represented by a cubic spline curve. For this example a polynomial degree of $p = 3$ was chosen.

No-slip boundary conditions are applied at the top and bottom of the domain, as well as at the walls of the cylinder boundary. The outflow pressure at the right of the domain is set to be equal to $p = 0$, with no condition on the velocity imposed. An inflow velocity is prescribed at the left of the domain and no condition on the pressure is applied there.

A parabolic profile is chosen for the inflow velocity such that at $y = 0$ the velocity is at its maximum value V , giving an average velocity of $U = \frac{2}{3}V$. For this problem the characteristic length scale of the flow is defined by the diameter of the cylinder $L = 1.0$. For the final parameter the viscosity here is set to $\mu = 10^{-3}$.

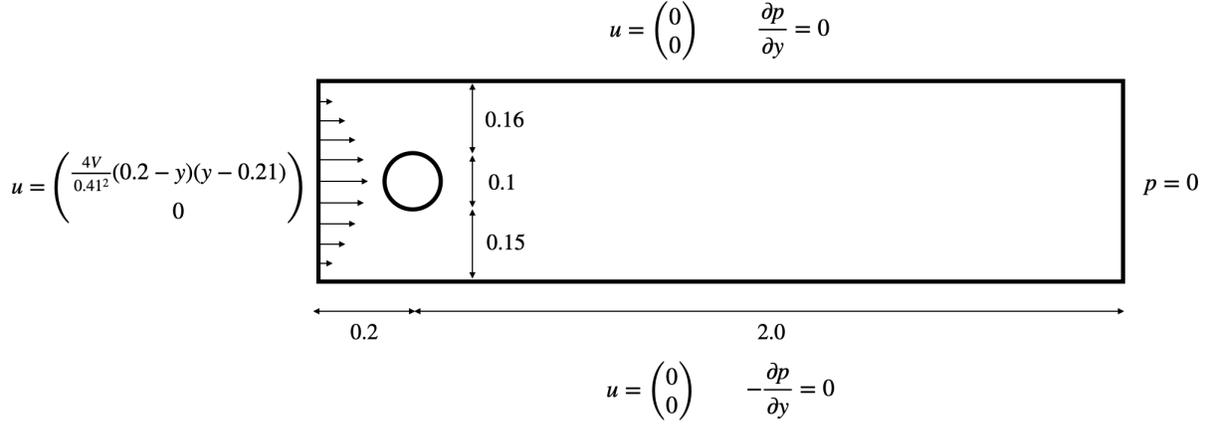


Fig. 5 Problem setup for flow around cylinder example.

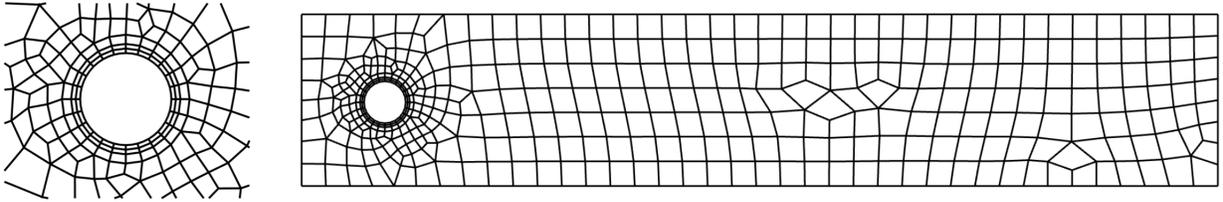


Fig. 6 Mesh for flow around cylinder at zero refinements. Left shows zoom in around cylinder.

1. Steady flow, $Re=20$

We first consider the problem with the maximum velocity on the inflow $V = 0.3$, with the average velocity then equal to $U = \frac{2}{3}V = 0.2$. The Reynolds number can then be calculated as $Re = \frac{0.2 \cdot 0.1}{10^{-3}} = 20$. The flow in this case reaches a steady state and we compare the difference in pressure Δp from the front of the cylinder to the back to the reference value $\Delta p = 0.11752016697$ found in [17] to verify the accuracy of the method.

Fig. 7 shows the x-velocity and pressure profiles obtained with the mesh refined once. The profiles appear to be smooth and well-developed for this problem. A difference in pressure from the front to the back of the cylinder of 1.4×10^{-4} was obtained at zero refinements and 1.1×10^{-5} at one refinement of the mesh, reflecting the accuracy of a high-order method.

2. Unsteady flow, $Re=100$

We also consider the same example but with an increased maximum velocity at the inflow $V = 1.5$, with the average velocity the equal to $U = \frac{2}{3}V = 1.0$. This gives a Reynolds number here of $Re = \frac{1.0 \cdot 0.05}{10^{-3}} = 100$. The flow is unsteady at this Reynolds number and no steady state solution can be achieved.

Fig. 8 shows the x-velocity and pressure of the system at time $T = 3.0$. The unsteady nature of the flow can be seen through the characteristic vortex shedding behaviour of the system displayed here. Stability issues arising from the convection term are not observed despite rapidly varying velocity fields in the system. Furthermore as with the previous example smooth velocity and pressure profiles are seen indicating inf-sup stability.

B. Lid-driven cavity

We also consider the problem of the lid-driven cavity with the setup shown in Fig. 9. The domain is chosen to be the square $[-1, 1]^2$, such that the length of each side is equal to $L = 2$. No-slip boundaries conditions are applied on all boundaries except for the one at the top of the domain, which is set to be sliding to the right with a constant velocity of $u_x = 1$.

The characteristic velocity of this problem is defined by the sliding top $U = 1$. The viscosity is set to be $\mu = 10^{-3}$ giving a Reynolds number of $Re = \frac{1 \cdot 2}{10^{-3}} = 2000$. For this example the mesh shown on the left of Fig. 9 was refined 3

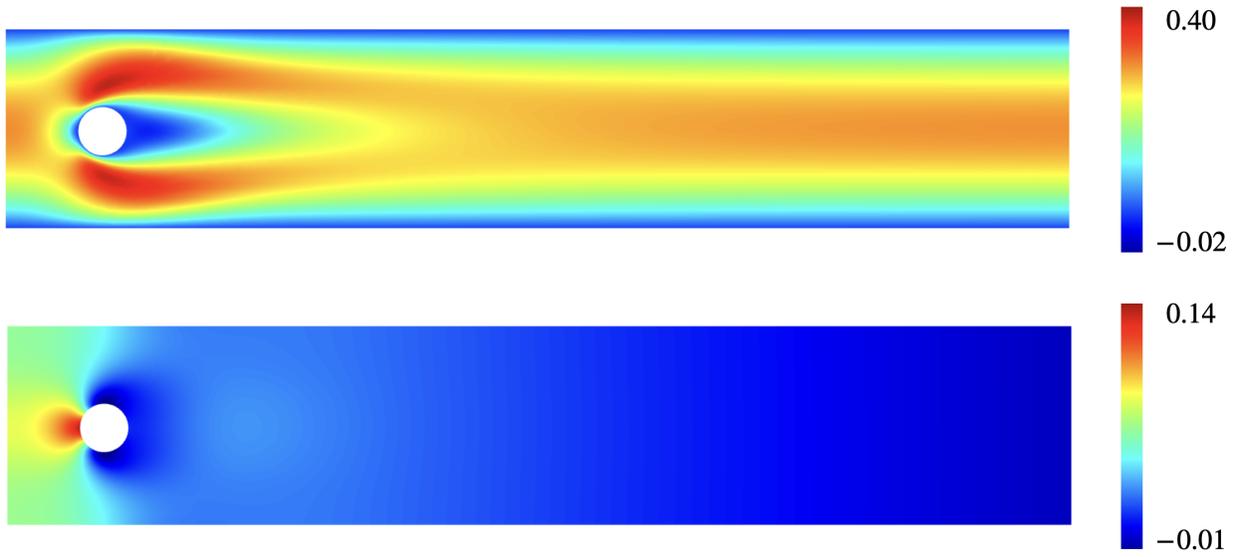


Fig. 7 Plots of x-velocity and pressure at steady state for flow around cylinder example for $Re = 20$.

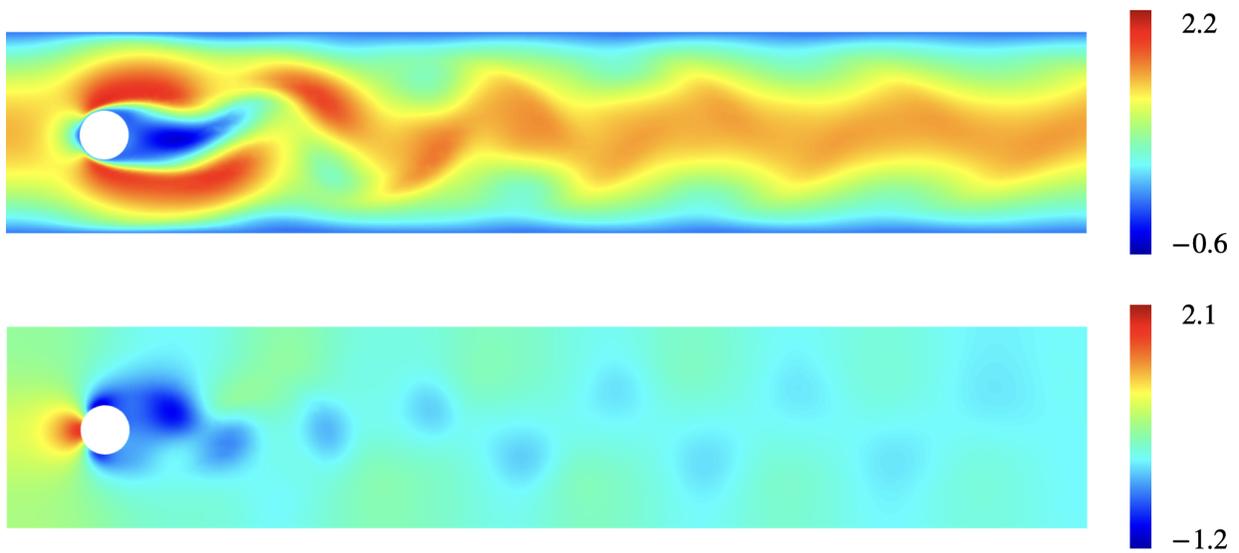


Fig. 8 Plots of x-velocity and pressure at $T = 3.0$ for flow around cylinder example for $Re = 100$.

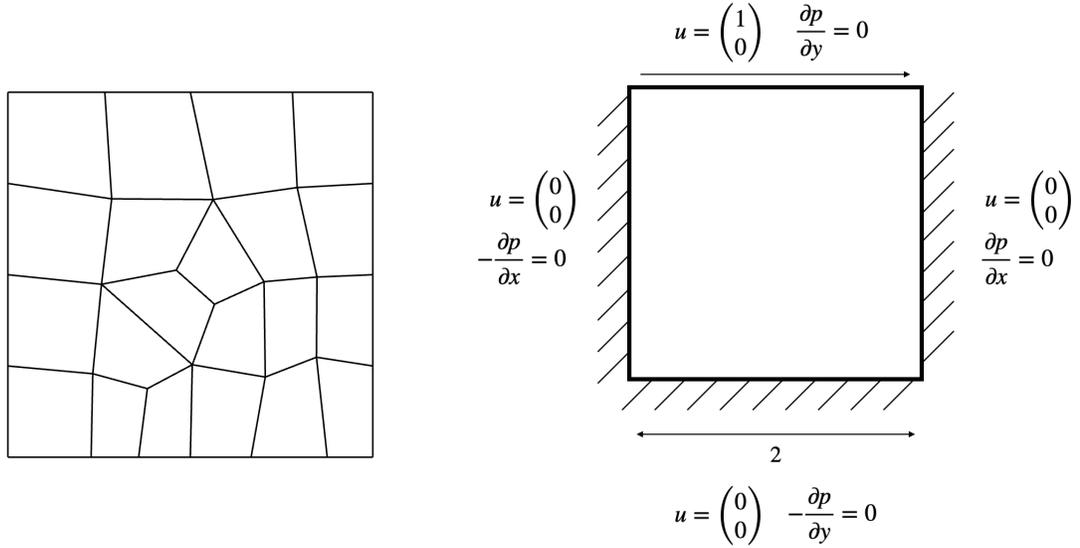


Fig. 9 Setup of lid-driven cavity example. Left shows mesh at zero refinements, right shows dimensions and boundary conditions of problem.

times from what is shown and polynomial degree $p = 3$ was used.

Fig. 10 shows the plots of the norm of the velocity and pressure at the time $T = 20.0$. The problem does not display any stability issues and the solution fields appear to smooth except for at the top corners. This can likely be attributed to the inconsistent no-slip boundary conditions applied at these corners, in addition to the lack of enhanced resolution at these corners of the mesh.

V. Conclusion

In this work we have demonstrated the use of the recently developed FUSE methods for incompressible flow problems at moderate to high Reynolds numbers. The convection terms in the Navier-Stokes equations are stabilised using a specific choice of node distribution in addition to a face-upwinded approach and the inf-sup instability is avoided via use of an LDG inspired upwind-downwind Laplace operator. The method is not limited by geometry and is applicable to a wide range of meshes. However, as the FUSE method does rely on a specific node distribution for stability including the boundary nodes, special care is required at corners where boundary conditions may be inconsistent and more geometry

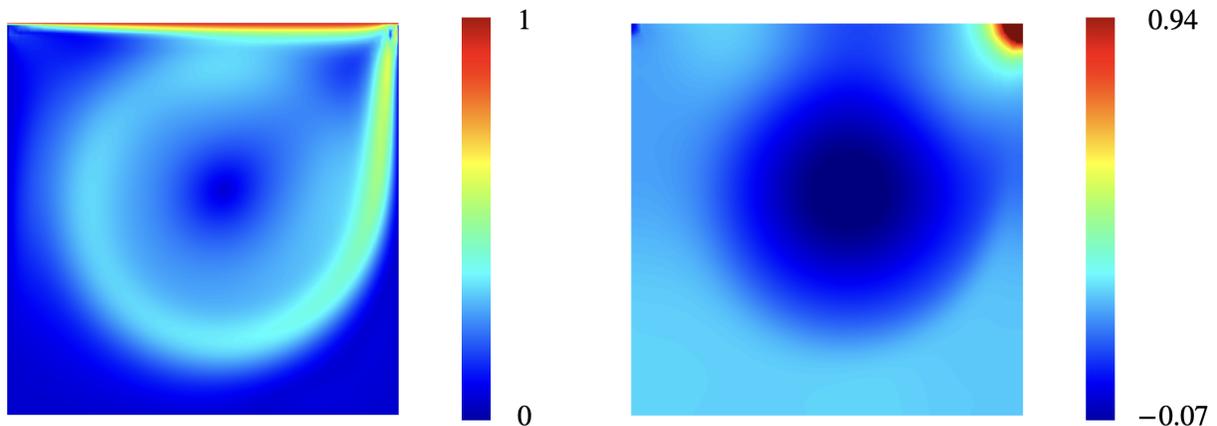


Fig. 10 Plots of magnitude of velocity and pressure at $T = 20.0$ for lid-driven cavity example.

resolution may be required to properly resolve the flow.

Acknowledgments

This work was supported in part by the Director, Office of Science, Office of Advanced Scientific Computing Research, U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

References

- [1] Tadmor, E., “Convergence of spectral methods for nonlinear conservation laws,” *SIAM J. Numer. Anal.*, Vol. 26, No. 1, 1989, pp. 30–44. <https://doi.org/10.1137/0726003>, URL <https://doi-org.libproxy.berkeley.edu/10.1137/0726003>.
- [2] Hughes, T. J., Scovazzi, G., and Franca, L. P., “Multiscale and stabilized methods,” *Encyclopedia of computational mechanics second edition*, 2018, pp. 1–64.
- [3] Reed, W. H., and Hill, T. R., “Triangular mesh methods for the neutron transport equation,” Tech. Rep. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
- [4] Cockburn, B., and Shu, C.-W., “Runge-Kutta discontinuous Galerkin methods for convection-dominated problems,” *J. Sci. Comput.*, Vol. 16, No. 3, 2001, pp. 173–261.
- [5] Hesthaven, J. S., and Warburton, T., *Nodal discontinuous Galerkin methods*, Texts in Applied Mathematics, Vol. 54, Springer, New York, 2008. Algorithms, analysis, and applications.
- [6] Kopriva, D. A., and Kalias, J. H., “A conservative staggered-grid Chebyshev multidomain method for compressible flows,” *J. Comput. Phys.*, Vol. 125, No. 1, 1996, pp. 244–261. <https://doi.org/10.1006/jcph.1996.0091>, URL <http://dx.doi.org/10.1006/jcph.1996.0091>.
- [7] Kopriva, D. A., and Gassner, G., “On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods,” *J. Sci. Comput.*, Vol. 44, No. 2, 2010, pp. 136–155. <https://doi.org/10.1007/s10915-010-9372-3>, URL <http://dx.doi.org/10.1007/s10915-010-9372-3>.
- [8] Wang, Z. J., “Spectral (finite) volume method for conservation laws on unstructured grids. Basic formulation,” *J. Comput. Phys.*, Vol. 178, No. 1, 2002, pp. 210–251.
- [9] Liu, Y., Vinokur, M., and Wang, Z. J., “Spectral difference method for unstructured grids. I. Basic formulation,” *J. Comput. Phys.*, Vol. 216, No. 2, 2006, pp. 780–801.
- [10] Wang, Z., Liu, Y., Lacor, C., and Azevedo, J., “Spectral volume and spectral difference methods,” *Handbook of Numerical Analysis*, Vol. 17, Elsevier, 2016, pp. 199–226.
- [11] Huynh, H. T., “A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods,” *18th AIAA computational fluid dynamics conference*, 2007, p. 4079.
- [12] Yu, M., Wang, Z. J., and Liu, Y., “On the accuracy and efficiency of discontinuous Galerkin, spectral difference and correction procedure via reconstruction methods,” *J. Comput. Phys.*, Vol. 259, 2014, pp. 70–95. <https://doi.org/10.1016/j.jcp.2013.11.023>, URL <https://doi-org.libproxy.berkeley.edu/10.1016/j.jcp.2013.11.023>.
- [13] Pan, Y., and Persson, P.-O., “A face-upwinded spectral element method on unstructured quadrilateral meshes,” , 2023. In review.
- [14] Cockburn, B., and Shu, C.-W., “The Local Discontinuous Galerkin Method for Time-Dependent Convection-Diffusion Systems,” *SIAM J. Numer. Anal.*, Vol. 35, No. 6, 1998, pp. 2440–2463.
- [15] Cockburn, B., Kanschat, G., and Schötzau, D., “The local discontinuous Galerkin method for linearized incompressible fluid flow: a review,” *Computers & fluids*, Vol. 34, No. 4-5, 2005, pp. 491–506.
- [16] Cockburn, B., Kanschat, G., Schötzau, D., and Schwab, C., “Local discontinuous Galerkin methods for the Stokes system,” *SIAM Journal on Numerical Analysis*, Vol. 40, No. 1, 2002, pp. 319–343.
- [17] John, V., and Matthies, G., “Higher-order finite element discretizations in a benchmark problem for incompressible flows,” *International Journal for Numerical Methods in Fluids*, Vol. 37, No. 8, 2001, pp. 885–903.