The 9th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2024
3–7 June 2024, Lisboa, Portugal

IMMERSED BOUNDARY METHODS FOR HIGH-ORDER DISCRETIZATION OF THE COMPRESSIBLE REYNOLDS AVERAGED NAVIER-STOKES EQUATIONS

DIEGO LODARES[1,2], JUAN MANZANERO[1,2], ESTEBAN FERRER[1,3] AND EUSEBIO VALERO[1,3]

[1] ETSIAE-UPM - School of Aeronautics
Universidad Politécnica de Madrid
Plaza Cardenal Cisneros 3, 28040 Madrid, Spain
e-mail: diego.lodares@airbus.com, juan.manzanero@upm.es, esteban.ferrer@upm.es, eusebio.valero@upm.es

[2] Airbus Defence and Space
Avenida de John Lennon s/n, 28906 Getafe, Spain
e-mail: diego.lodares@airbus.com, juan.manzanero@airbus.com

[3] Center for Computational Simulation
Universidad Politécnica de Madrid
Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain

Key words: Immersed boundary methods, DGSEM, RANS, Fluid Mechanics, Aerodynamics

Summary. The Immersed Boundary Method (IBM) presents clear advantages for CFD simulation of compressible flows around complex geometries. In contrast to the standard body-fitted approach, in which meshes are designed to conform to geometries, the IBM treats solid obstacles via local modification of the governing equations. Popular modifications rest on adding volumetric penalization terms to those mesh cells that are covered by immersed bodies [1] or on imposing special boundary conditions on mesh faces surrounding them [2]. In the context of the nodal Discontinuous Galerkin Spectral Element Method (DGSEM), one can also apply subcell-based limiting strategies to further discretize the immersed mesh cells employing a compatible low-order method [3]. In this paper, we present a comparison between these three techniques in a high-order setting to solve compressible flows around 2D geometries using the RANS equations with the Spalart-Allmaras one-equation turbulence model. Our results show that introducing wall model-based terms is necessary for IBM formulations to yield correct RANS flow fields, and suggest that subcell-based limiting in the context of IBM can be advantageous in terms of convergence while maintaining solution accuracy.

1 INTRODUCTION

The Immersed Boundary Method (IBM) [4] atempts to reproduce the effect of boundary conditions in the flow without requiring body-fitted meshes. The method can reduce considerably the effort of meshing around complex geometries, making it generally regarded as an acceptable compromise between the quality of the solutions it generates and the speed at which they can be obtained. In this work we will study three different ways of treating immersed boundaries when

solving the Reynolds Averaged Navier Stokes (RANS) equations with the Spallart-Allmaras (SA) one-equation turbulence model [5] at a high Reynolds number, using the high-order Discontinuous Galerkin Spectral Element Method (DGSEM) [6] for meshes of hexahedral elements.

2 REYNOLDS AVERAGED NAVIER-STOKES EQUATIONS

The main five state equations of the RANS–SA system are the Favre–averaged compressible Navier–Stokes equations for an ideal gas,

$$\begin{aligned} \rho_t + \vec{\nabla}\cdot(\rho\vec{u}) &= 0, \\ (\rho\vec{u})_t + \vec{\nabla}\cdot\left(\rho\vec{u}\vec{u}^T\right) &= -\vec{\nabla}p + \vec{\nabla}\cdot\tau, \\ (\rho e)_t + \vec{\nabla}\cdot(\rho\vec{u}h) &= \vec{\nabla}\cdot(\tau\cdot\vec{u}) + \vec{\nabla}\cdot\vec{q}, \end{aligned} \tag{1}$$

where ρ is the density, $\vec{u} = (u, v, w)^T$ is the velocity, p is the pressure, e the total specific energy, and $h = e + p/\rho$ the total specific enthalpy. τ and $\vec{q}$ are the viscous tensor and heat flux vector,

$$\tau = (\mu + \mu_t)\left(2\mathsf{S} - \frac{2}{3}\vec{\nabla}\cdot\vec{u}\underline{I}_3\right), \quad \vec{q} = (\kappa + \kappa_t)\,\vec{\nabla}T, \quad \mathrm{Pr} = \frac{\mu c_p}{\kappa}, \quad \mathrm{Pr}_t = \frac{\mu_t c_p}{\kappa_t}, \tag{2}$$

where $\mathsf{S} = (\vec{\nabla}\vec{u} + \vec{\nabla}\vec{u}^T)/2$ is the symmetric strain tensor, T is the temperature, μ and μ_t are the laminar and turbulent viscosities, κ and κ_t are the laminar and turbulent thermal conductivities, Pr and Pr_t are the Prandtl and turbulent Prandtl numbers that relate viscosities and conductivities. Laminar viscosity follows Sutherland's law

$$\mu = \mu_{\mathrm{ref}}\left(\frac{T}{T_{\mathrm{ref}}}\right)^{3/2}\left(\frac{T_{\mathrm{ref}} + S}{T + S}\right), \tag{3}$$

where S is Sutherland's constant, T_{ref} a reference temperature, and μ_{ref} is the viscosity value at it. Kinematic laminar viscosity is defined as $\nu = \mu/\rho$.

Turbulent viscosity μ_t is a result of the Spalart–Allmaras one–equation turbulent closure [5] with the *SA-neg* model extension [7], an additional advection-diffusion equation that introduces SA state variable $\rho\tilde{\nu}$

$$(\rho\tilde{\nu})_t + \vec{\nabla}\cdot(\rho\tilde{\nu}\vec{u}) = \rho(P - D) + \frac{1}{\sigma}\left(\vec{\nabla}\cdot\left(\rho\hat{\sigma}\vec{\nabla}\tilde{\nu}\right) + c_{b2}\rho|\vec{\nabla}\tilde{\nu}|^2\right), \tag{4}$$

where P and D are production and destruction terms, $\hat{\sigma}$ is a sum of turbulent and laminar kinematic viscosities, and c_{b2} is a constant. Turbulent viscosity can be calculated from the SA state variable using

$$\mu_t = \max(\rho\tilde{\nu}, 0)f_{v1}, \quad f_{v1} = 1 - \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\tilde{\nu}}{\nu}, \quad c_{v1} = 7.1. \tag{5}$$

Detailed expressions for all the terms in this model can be found at [8], for example.

3 DGSEM METHOD AND ENTROPY-STABLE SPATIAL DISCRETIZATIONS

The Discontinuous Galerkin Spectral Element Method (DGSEM) [6] is a particular nodal formulation of the high–order Discontinuous Galerkin (DG) method that is notably efficient on meshes of hexahedral elements. In it, the state vector is approximated by the tensor product of polynomials of degree N within a cubic *reference element* E with coordinates $(\xi, \eta, \zeta) \in [-1, 1]^3$. The whole domain is discretized into non–overlapping hexahedral elements which are mapped to E using a series of geometrical transformations [9].

$$\mathbf{q}\big|_e \approx \mathbf{Q}\left(\vec{\xi}, t\right) = \sum_{i,j,k=0}^{N} \mathbf{Q}_{ijk}\left(t\right) l_i\left(\xi\right) l_j\left(\eta\right) l_k\left(\zeta\right), \quad \mathbf{Q} \in \mathcal{P}^N, \tag{6}$$

With DGSEM, the Lagrange basis polynomials $l_i\left(\xi\right), l_j\left(\eta\right), l_k\left(\zeta\right)$ can employ Gauss–Lobatto points to satisfy the summation–by–parts simultaneous–approximation–term (SBP–SAT) property [10, 11], which is key when developing provably stable semi-discrete (continuous in time, discrete in space) numerical discretizations. Entropy stable approximations belong to this type of robust schemes, as they ensure that a given *mathematical entropy* (which will depend on the scheme, e.g. kinetic energy, or thermodinamic entropy) remains bounded. In this work, we choose the Kinetic Energy Preserving (KEP) scheme developed in [12] to discretize the RANS-SA system, which uses Lax Friedrix's numerical flux [13] and Pirozzoli's split form [14] to treat inviscid fluxes, and the Bassi-Rebay 1 [15] scheme for the viscous part.

4 TIME INTEGRATION

The ODE that the KEP DGSEM scheme described above yields is integrated in the time domain using the linearized implicit Euler steady state solver [16]

$$\begin{gathered} \mathbf{Q}_t - \mathbf{R}(\mathbf{Q}) = \mathbf{0}, \\ \left(\left.\frac{\partial \mathbf{R}}{\partial \mathbf{Q}}\right|^n - \mathbf{I}/\Delta t_n \right) \cdot \Delta \mathbf{Q} = -\mathbf{R}(\mathbf{Q}^n), \\ \mathbf{Q}^{n+1} = \mathbf{Q}^n + \epsilon \Delta \mathbf{Q}, \end{gathered} \tag{7}$$

where the Jacobian matrix is obtained via second–order accurate numerical perturbation of residual $\mathbf{R}$. Linear system (7) is solved iteratively up to a certain relative tolerance (typically 10^{-5}) using PETSc's [17] GMRES linear solver with the additive Schwarz preconditioner, and the ϵ parameter is a constant in (0, 1] that makes the formula tend to the damped Newton method. The time step Δt evolves with two CFL numbers, an advective and a diffusive one, from which the most restrictive time step size is taken at each iteration. These time steps, whose expressions can be found in [12], are computed via heuristics that take into account the smallest mesh spacing, the velocity-related eigenvalues of the inviscid flux and the laminar and eddy viscosities.

5 IMMERSED BOUNDARY METHODS

In this work we test three different Immersed Boundary methods in the described RANS-DGSEM configuration. All of them substitute the process of meshing around a geometry by a

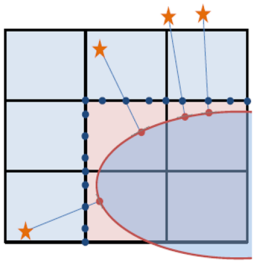


Figure 1: Face-based IB method: the blue dots represent high-order nodes of discretization element faces that become *boundary points* at which a wall model boundary condition is placed. The red domain is composed of blanked elements that the solid occupies, and no longer require to be solved. The red dots are the projections of the boundary points over the immersed wall, and are used to solve the friction velocity with the wall model. The stars are the *donor points*, whose velocity is an input to the wall model, and are placed at a high enough wall distance by prolongating the wall normal that joins each boundary and wall point pair. This distance can be initially estimated with e.g. known flat-plate analytical solutions, and then tuned as the results for a particular case are obtained and analyzed. The donor point must also satisfy not being inside a blanked element, and in our DGSEM setting, the state variables at them can be readily obtained via Lagrange interpolation within their corresponding mesh elements.

modification the underlying equations, to account for a body that is now *immersed* in the fluid domain.

5.1 Face-based IB method

This method is presented in [2], where the authors solve the RANS equations with a Finite Volume (FV) discretization around a series of aerodynamic geometries. In it, the immersed boundaries become boundary conditions that enforce a wall model. Any wall model can be chosen, e.g. free-slip linear, no-slip linear, logarithmic, Musker's, Allmaras's, etc. In our case we've chosen Allmaras's wall function [18],

$$u^+(y^+) = \bar{B} + c_1 \log((y^+ + a_1)^2 + b_1^2) - c_2 \log((y^+ + a_2)^2 + b_2^2) - c_3 \mathrm{atan2}(y^+ + a_1, b_1) - c_4 \mathrm{atan2}(y^+ + a_2, b_2), \tag{8}$$

in which $\bar{B}, a_1, a_2, b_1, b_2, c_1, c_2, c_3$ and c_4 are numerical constants, and u^+ and y^+ are the wall coordinates common to all wall models ($u^+ = u/u_\tau$, $y^+ = u_\tau d/\nu$) from which one can calculate friction velocity u_τ at the wall and finally the magnitude of the tangential flow velocity u at the boundary point with wall distance d, a process that involves solving a non-linear equation with one unknown (u_τ) and is fed with the flow velocity at a far enough *donor point*, as shown in Fig. 1. For RANS, one also has to reconstruct the SA variable $\tilde{\nu}$ at the boundary, which can be obtained as a root of a quartic equation explained in [2].

5.2 Immersed Boundary Volume Penalization (IBVP) method

The IBVP method [1, 19, 20, 21] is arguably the most straightforward approach to solve immersed boundaries, as it consists in adding penalization source terms at those mesh element points that lie inside the solid, which drive the flow velocity and eddy viscosity to zero. Within

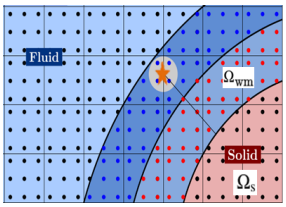


Figure 2: IBVP method (image from [21]): the black dots represent the solid domain in which the penalty term scaled by η applies, while the red ones are in the wall model penalization layer surrounding the body in which the η_{wm} term is introduced. Each point in Ω_{wm} requires a donor point (represented by the star marker), generated in the same way as for the face-based IB method, to compute a wall model reference state at it.

a layer surrounding the body, an additional wall model penalty is introduced, so that the flow better approximates what should be the desirable solution as the wall distance tends to zero. The reference wall state for this latter penalty contribution can be computed using (8) exactly as in the face-based IB method described before, again requiring a donor point generated along the wall normal line that passes through each penalized point. The expressions of these penalty terms for a static immersed boundary are the following

$$
\begin{aligned}
\mathbf{s}_{\text{IBVP}} &= \begin{pmatrix} \mathbf{s}_{\text{NS}} \\ s_{\text{SA}} \end{pmatrix}, \\
\mathbf{s}_{\text{NS}} &= -\frac{\chi}{\eta}\begin{pmatrix} 0 \\ \rho\vec{u} \\ \frac{\rho|\vec{u}|^2}{2} \end{pmatrix} + \frac{\chi_{\text{wm}}}{\eta_{\text{wm}}}\begin{pmatrix} 0 \\ (\rho\vec{u})_{\text{wm}} - \rho\vec{u} \\ \frac{1}{2}(\rho|\vec{u}|^2)_{\text{wm}} - \frac{\rho|\vec{u}|^2}{2} \end{pmatrix}, \\
s_{\text{SA}} &= -\frac{\chi}{\eta}\rho\tilde{\nu} + \frac{\chi_{\text{wm}}}{\eta_{\text{wm}}}\left((\rho\tilde{\nu})_{\text{wm}} - \rho\tilde{\nu}\right), \\
\chi(x,y,z) = \begin{cases} 1 & \text{if } (x,y,z) \in \Omega_s, \\ 0 & \text{otherwise,} \end{cases} &\qquad \chi_{\text{wm}}(x,y,z) = \begin{cases} 1 & \text{if } (x,y,z) \in \Omega_{\text{wm}}, \\ 0 & \text{otherwise,} \end{cases}
\end{aligned}
\tag{9}
$$

in which Ω_s is the domain occupied by the immersed body, Ω_{wm} the wall model penalization layer around it, χ and χ_{wm} define the *hard masks* that toggle the penalization terms in these two differentiated domains, ρ_{wm}, $\vec{u}_{\text{wm}}$ and $\tilde{\nu}_{\text{wm}}$ represent the reference wall model variables calculated per discretization point in Ω_{wm}, and η, η_{wm} are the scales of these penalization terms, taken uniform accross the mesh. Fig. 3 illustrates the main features of this method for a high-order discretization.

5.3 Subcell IB method

Subcell methods have been proven as an effective strategy to construct robust high-order methods that will guarantee additional properties, such as bounds on physical quantities and/or entropy dissipation [22]. When using this method for IB, the mesh elements with at least one high-order point inside the immersed body are re-gridded using a compatible low-order method (FV) generating *subcells*, and the time derivative of the state at these points is a linear blending between the ones that the low-order and the DGSEM methods yield

$$
\mathbf{q}_t = (1-\alpha)\mathbf{q}_t^{\text{DGSEM}} + \alpha\mathbf{q}_t^{\text{FV}}, \tag{10}
$$

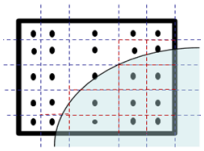


Figure 3: Subcell IB method: a high-order element touched by the solid is re-gridded into subcells in which a parallel low-order solution is obtained. For this low-order scheme, no-slip wall boundary conditions are applied at those faces that interfere with the body (indicated by the red dashed lines). Blending formula (10) makes these boundary conditions influence the target DGSEM result, achieving zero velocity within the body.

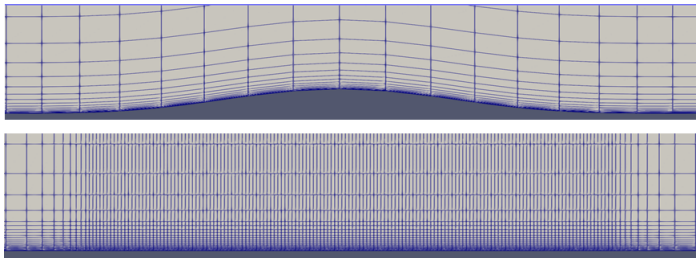


Figure 4: 2D bump test case: body-fitted (above) and IB (below) meshes.

where α represents the weight attributed to the low-order solution. The main effect of this approach is that it increments the number of mesh faces along the immersed boundary, specifically for the FV contribution: at the interfaces between subcells that have face or center points inside the solid, the method applies the numerical fluxes from a standard no-slip wall boundary condition, with a chosen Riemann solver (Roe's one in this work), and thus it is the low order term the one in charge of cancelling the flow velocity inside the body. As in the IBVP method, a wall model penalization source term is applied within a layer surrounding the solid, with its associated donor points providing the inputs to calculate a wall model reference state. When employing high-order meshes with curved faces, the definition of the low-order subcells requires special care, and their faces' normal vectors need to be derived from a high-order flux differencing formula that takes into account the host element's contravariant basis vectors to ensure an FV dicretization compatible with the high-order one [22].

6 NUMERICAL EXPERIMENTS

6.1 2D bump test case

We start by solving the flow over a two-dimensional bump at $\mathrm{M} = 0.15$, $\mathrm{Re} = 6{\cdot}10^6$, $\mathrm{Pr} = 0.72$ and Sutherland reference temperature $T_{\mathrm{ref}} = 300$ K, in order to assess the IBVP and subcell IB methods and the wall model penalization source term used in them. To generate a reference solution against which to compare the IBM results, we solve a body-fitted mesh with third-order curved faces, while for IBM we use a finer planar grid, as shown in Fig. 4. The bump's definition is the one from the *2D Bump-in-channel Verification Case* at [8].

Fig. 5 depicts the points at which the different source terms are applied in the IBVP method.

Fig. 6 shows that applying the wall model penalty appears fundamental, as unpenalized

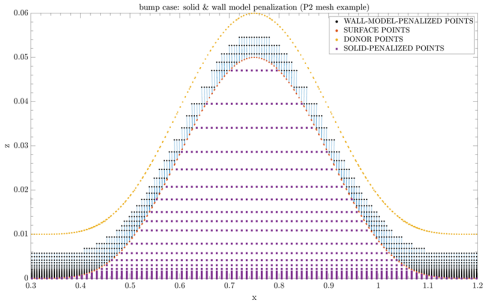


Figure 5: 2D bump test case: setup for the IBVP method, when solved with polynomial order 2. The subcell IB solution employs the same donor points and wall model penalization layer.

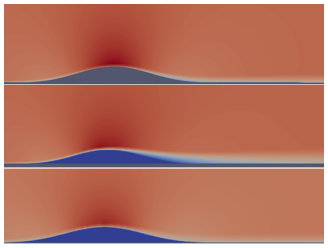


Figure 6: 2D bump test case: visual comparison of the horizontal velocity field for a body-fitted solution (above), an IBM solution with zero wall penalty (center), and wall-penalized IBM (below). The unpenalized solution is characterized by a clearly incorrect, magnified wake.

solutions exaggerate the bump's wake inadequately.

We compare the convergence of the IBVP and subcell IB methods when solving the case with polynomial order 3 and using that solution as the starting point for one of polynomial order 4. As can be seen in Fig. 7, the IBVP method appears to be much stiffer than the subcell IB method, taking much more implicit iterations to converge for very similar results.

Fig. 8 represents velocity and eddy viscosity profiles at different zones near the bump. Solutions without wall-model-penalization maintain an undesirable velocity deficit after the obstacle, while the wall-penalized ones respect the main features of the reference body-fitted results. The velocity profile over the bump suggest that the planar IB mesh should be made even finer, as increasing the polynomial order does not seem effective. We note that solutions for the wall-penalized IBVP and subcell IB methods are very similar.

6.2 2D NACA 0012 test case

Next, we simulate the flow around a NACA 0012 symmetric airfoil at angle of attack $\alpha = 10$ deg, $\mathrm{M} = 0.15$, $\mathrm{Re} = 6 \cdot 10^6$, $\mathrm{Pr} = 0.72$ and $T_{\mathrm{ref}} = 300$ R, with the face-based IB method and the subcell IB one. Fig. 9 depicts the elliptical grid with cubic polynomial faces we use for IBM, while the reference body fitted results are generated using an adaptation to piecewise quadratic edges of the coarsest grid in the *2D NACA 0012 Airfoil Validation Case* available at [8].

Figure 10 illustrates the donor points and wall model layer that we take in the subcell IB case. The donor points for the face-based IB method solutions are chosen at the same distance from the body.

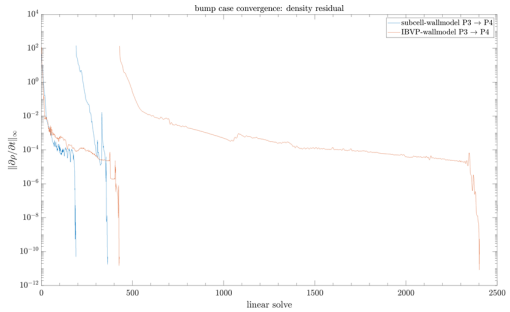


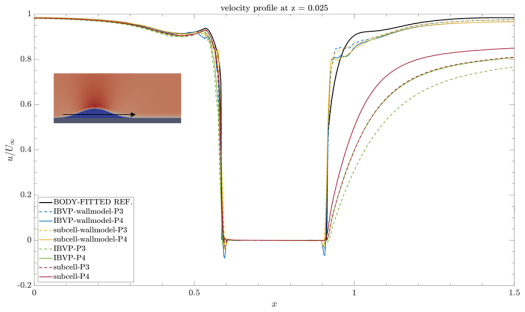
Figure 7: 2D bump test case: density residual history of IBVP and subcell IB solutions with polynomial degrees 3 and 4.

Fig. 11 compares the horizontal velocity flow field of the body-fitted and benchmarked immersed boundary methods. We can confidently confirm that using wall models (like the face-based IB method does, or through penalization in the subcell IB method) is clearly necessary to correctly solve the wake around immersed bodies with the RANS equations.

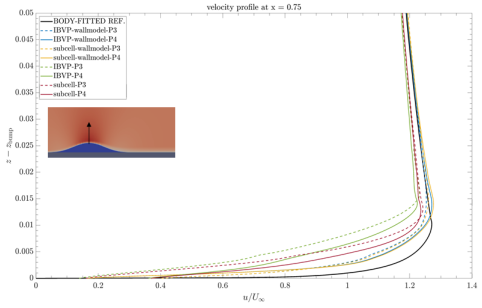
Finally, Fig. 12 represents velocity and eddy viscosity profiles along two different vertical behind the airfoil. All IBM solutions follow the general trend of the body-fitted reference results.

7 CONCLUSIONS

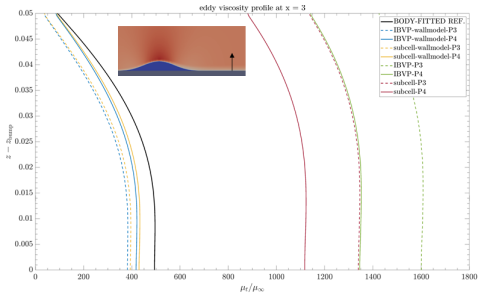
In this work, we have described the implementation of three different immersed boundary methods within a high–order nodal discontinuous Galerkin discretization that approximates the RANS–Spalart–Allmaras system. These methods can be applied to high–order meshes of hexahedral elements that are not required to conform to the geometry one wants to solve the flow around. Their convergence and numerical behaviour has been assessed in two popular 2D test cases, a bump and a NACA 0012 airfoil. Our results for both test cases demonstrate that employing a wall model, introduced either through penalization terms (like in the IBVP or the subcell IB methods) or via boundary conditions (like the face-based IB method does) is necessary to correctly solve the flow around immersed bodies with the RANS equations. In the 2D bump test case, the comparison between the IBVP and the subcell IB methods shows that subcell limiting in the context of IBM relaxes the stiffness of the system of equations (here treated implicitly) leading to faster convergence, while maintaining accuracy. The 2D NACA 0012 test case corroborates that the accuracy of this method is comparable to that of the face-based IB one.



(a) Velocity profile accross the bump.



(b) Velocity profile over the bump.



(c) Eddy viscosity profile in the wake of the bump.

Figure 8: 2D bump test case: simulation results.

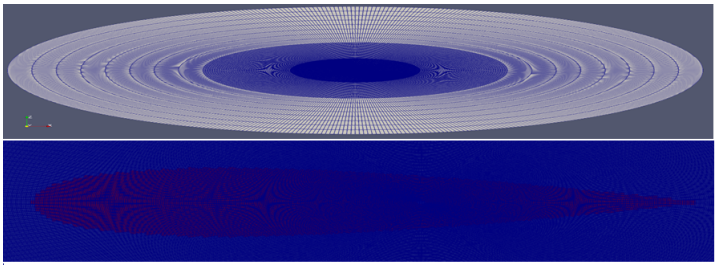


Figure 9: 2D NACA 0012 test case: elliptical mesh for IBM, with cubic faces (above), and blanked domain (in red) in the face-based IB method (below).

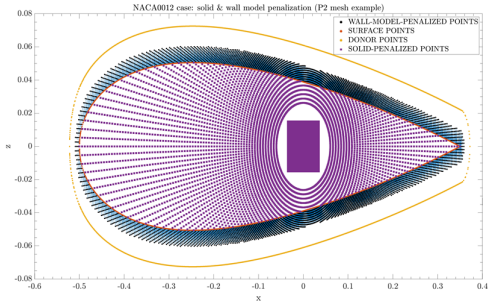


Figure 10: 2D NACA 0012 test case: donor points and wall penalization layer for a subcell IB solution, when solved with polynomial order 2. The face-based IB solutions employ donor points at the same distance from the airfoil.

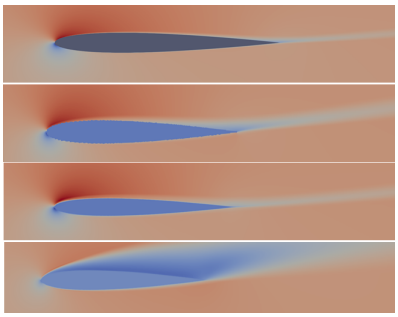
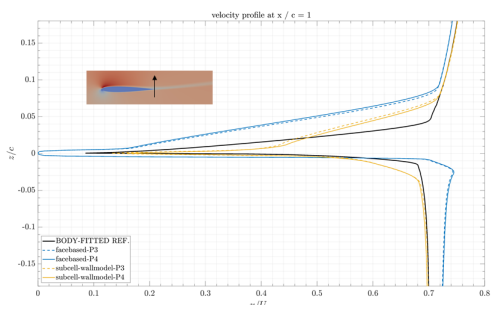
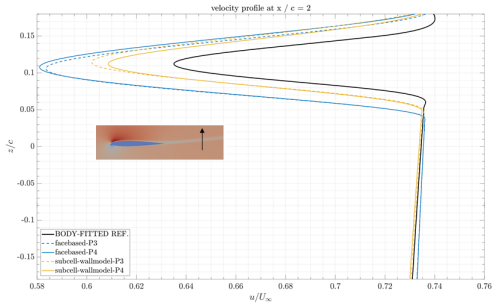


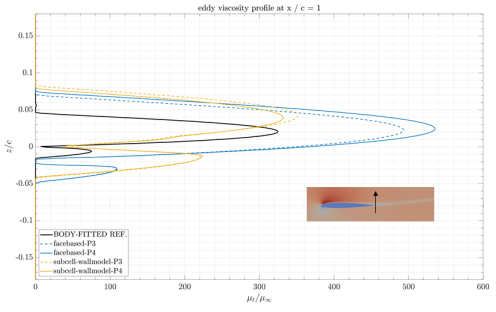
Figure 11: 2D NACA 0012 test case: visual comparison of the horizontal velocity field for a body-fitted solution (above), the face-based IB method (second), a subcell IB solution (third) and another subcell IB result, but this time with zero wall penalty (last). The unpenalized subcell IB solution exhibits a blatantly detached wake.



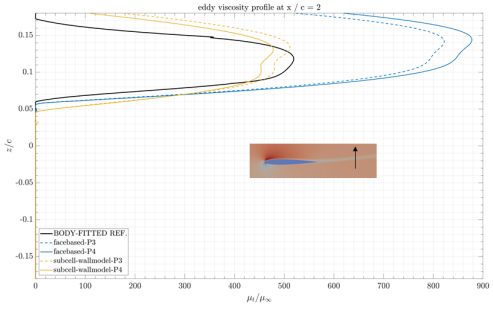
(a) Velocity profile at $x/c = 1$.



(b) Velocity profile at $x/c = 2$.



(c) Eddy viscosity profile at $x/c = 1$.



(d) Eddy viscosity profile at $x/c = 2$.

Figure 12: 2D NACA 0012 test case: simulation results.

REFERENCES

[1] J. Kou, S. Joshi, A. Hurtado-de-Mendoza, K. Puri, C. Hirsch, E. Ferrer, Immersed boundary method for high-order flux reconstruction based on volume penalization, Journal of Computational Physics, Volume 448, 110721, 2022.

[2] S. Péron, C. Benoit, T. Renaud, I. Mary, An immersed boundary method on Cartesian adaptive grids for the simulation of compressible flows around arbitrary geometries, Engineering with Computers 37, 2419–2437, 2021.

[3] A. Rueda-Ramírez, W. Pazner, G. Gassner, Subcell limiting strategies for discontinuous Galerkin spectral element methods, Computers & Fluids, Volume 247, 105627, 2022.

[4] Mittal, R. and Iaccarino G. Immersed Boundary Methods. Annual Review of Fluid Mechanics (2005) 37:239–261.

[5] P. Spalart, S. Allmaras, A one-equation turbulence model for aerodynamic flows, in: 30th aerospace sciences meeting and exhibit, 1992, p. 439.

[6] K. Black, A conservative spectral element method for the approximation of compressible fluid flow, Kybernetika 35 (1) (1999) 133–146.

[7] S. Allmaras, F. Johnson, P. Spalart, Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model, Seventh International Conference on Computational Fluid Dynamics (ICCFD7) (2012) 1–11.

[8] NASA's Langley Research Center Turbulence Modelling Resource, `https://turbmodels.larc.nasa.gov`.

[9] D. A. Kopriva, Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers, Springer Science & Business Media, 2009.

[10] T. C. Fisher, M. H. Carpenter, High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains, Journal of Computational Physics 252 (2013) 518–557.

[11] M. H. Carpenter, T. C. Fisher, E. J. Nielsen, S. H. Frankel, Entropy stable spectral collocation schemes for the Navier–Stokes equations: Discontinuous interfaces, SIAM Journal on Scientific Computing 36 (5) (2014) B835–B867.

[12] D. Lodares, J. Manzanero, E. Ferrer, E. Valero, An entropy–stable discontinuous Galerkin approximation of the Spalart–Allmaras turbulence model for the compressible Reynolds Averaged Navier–Stokes equations, Journal of Computational Physics, Vol. 455, 110998, 2022 (available online).

[13] P. Chandrashekar, Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations, Communications in Computational Physics 14 (5) (2013) 1252–1286.

[14] S. Pirozzoli, Numerical methods for high-speed flows, Annual review of fluid mechanics 43 (2011) 163–194.

[15] F. Bassi, S. Rebay, A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, Journal of computational physics 131 (2) (1997) 267–279.

[16] F. Witherden, A. Jameson, D. Zingg, Chapter 11 - the design of steady state schemes for computational aerodynamics, in: R. Abgrall, C.-W. Shu (Eds.), Handbook of Numerical Methods for Hyperbolic Problems, Vol. 18 of Handbook of Numerical Analysis, Elsevier, 2017, pp. 303 – 349.

[17] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Karpeyev, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, PETSc Web page, `https://www.mcs.anl.gov/petsc`.

[18] M. Berger, M. Atmosfis, Progress Towards a Cartesian Cut-Cell Method for Viscous Compressible Flow, 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition, `https://arc.aiaa.org/doi/abs/10.2514/6.2012-1301`.

[19] R. Abgrall, H. Beaugendre, C. Dobrzynski, An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques, Journal of Computational Physics, Vol. 257, 83-101, 2014.

[20] J. Kou, A. Hurtado-de-Mendoza, S. Joshi, S. Le Clainche, E. Ferrer, Eigensolution analysis of immersed boundary method based on volume penalization: Applications to high-order schemes, Journal of Computational Physics, Vol. 449, 110817, 2022.

[21] J. Nunez, D. Huergo, D. Lodares, S. Shrestha, J. Guerra, J. Florenciano, E. Ferrer and E. Valero, Implementation of Immersed Boundaries via Volume Penalization in the Industrial Aeronautical Computational Fluid Dynamics Solver CODA, 2024, submitted - under review, preprint available at `https://arxiv.org/abs/2404.16132`.

[22] A. M. Rueda-Ramírez, W. Pazner, G. J. Gassner, Subcell limiting strategies for discontinuous Galerkin spectral element methods, Computers & Fluids, Volume 247, 105627, 2022.