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

IMPROVING THE EFFICIENCY OF THE SUB-STEPPING VELOCITY SPLITTING SCHEME WITHIN THE NEKTAR++ FRAMEWORK

Alexandra I. Liosi*[1,2], Adam Swift[2], Athanasios Chatzopoulos[2], Francesco Bottone[2], Masashi Horikoshi[2], Julien Hoessler[2] and Spencer J. Sherwin[1]

[1] Department of Aeronautics, Imperial College London, SW7 2BX, a.liosi22@imperial.ac.uk, s.sherwin@imperial.ac.uk

[2] McLaren Racing Ltd, Woking GU21 4YH

Key words: Spectral h/p element method, Semi lagrangian schemes, high-performance computing, efficient time-stepping techniques

Summary. In this work, the computational performance of the Nektar++ framework for solving the incompressible Navier-Stokes (NS) equations using Spectral h/p elements is assessed and improved for industry-relevant geometries at high Reynolds numbers. There is an increasing need for simulating complex geometries with greater accuracy at a reduced computational cost. These simulations involve multiple spatial/temporal scales, and they should model or resolve complex transient phenomena such as turbulence transition, separation, and vortex system evolution. Standard explicit techniques require very low time steps, which are not feasible within an industrial environment, hence the need for efficient implicit time-stepping techniques. This study uses the implicit Sub-Stepping Velocity Splitting Scheme to solve the incompressible NS equations numerically in a segregated manner. This method is distinguished by its use of a mixed discretization scheme. Specifically, it employs Discontinuous Galerkin discretization to solve an unsteady advection equation during the Advection step while using Continuous Galerkin discretization for the Pressure and Diffusion steps. First, the performance of the proposed method is evaluated in terms of its numerical efficiency for both serial and parallel computations. Then, the Advection, Pressure, and Diffusion steps are explored based on the fundamental kernel efficiency to identify the most time-consuming components, aiming at reducing the memory footprint and moving them closer to the CPU-bound of the roofline model. We will present a systematic performance analysis, the challenges, and the advancements for mixed discretization time-stepping techniques relevant to any available Finite Element code aiming at solving exascale industrial problems.

1 INTRODUCTION

Unsteady numerical solvers need efficient time-marching techniques to simulate flows around complex geometries at high Reynolds (Re) numbers with reasonable computation cost. The computational cost is directly proportional to the timestep size and the number of simulated timesteps. For industry-relevant geometries involving multiple spatial and temporal scales, the Courant-Friedrichs-Lewy (CFL) criterion restricts the maximum timestep size for explicit time-stepping techniques, significantly increasing their cost. This can be mitigated by using implicit time-stepping techniques that allow timesteps significantly above the CFL limit. The considered unsteady flows are described by the incompressible Navier-Stokes (NS) equations, which are discretized using a continuous Galerkin projection, while the pressure and velocity fields are approximated by Spectral/hp Finite Elements (FE).

Maday et al.[1] were the first to introduce the framework, "Operator-Integration-Factor" scheme (OIFS), for developing time-stepping techniques for the spectral/hp element community. According to that, any time-dependent ordinary or partial differential equation can be split into an ensemble of sub-problems equal to the number of operators in the initial equation. An alternative method to develop implicit time-stepping schemes is using the semi-Lagrangian formulation of the mass and momentum equations. Xiu et al.[2] applied this approach to homogenous turbulent flows and achieved a similar level of accuracy with the Eulerian formulation of these problems at a reduced cost. Xu et al.[3] extended the previous study by investigating a 3D channel-flow. They reported errors comparable to those of their Eulerian counterpart; however, the performance needed further improvements. Sherwin[4] combined the semi-Lagrangian formulation with the OIFS and demonstrated unconditional stability for 2D flow problems. Xiu et al.[5] compared the schemes proposed in [2] and [4] for canonical flow problems and summarized the conditions that favored each alternative. Patel et al.[6] leveraged the OIFS to discretize the NS equations for a moving frame of reference to simulate the flow inside an internal combustion engine.

Despite various implementations of implicit time-stepping schemes, only Patel et al.[6] have discussed their scheme's accuracy and computational performance for an industry-relevant problem. There is a clear need to expand and generalize this conversation about stability, accuracy and performance for other industrial flows around complex geometries at high Re numbers while utilizing these schemes. To address this issue, we have used the "auxiliary" form of the semi-Lagrangian formulation, or the "Sub-Stepping Scheme"[4] and simulated the flow around a multi-element wing at $Re = 208,045$ as shown in Figure 1. The auxiliary form was selected over the strong form to avoid the computational cost of the global interpolation during particle tracking to identify the location and the velocity at each departure point.

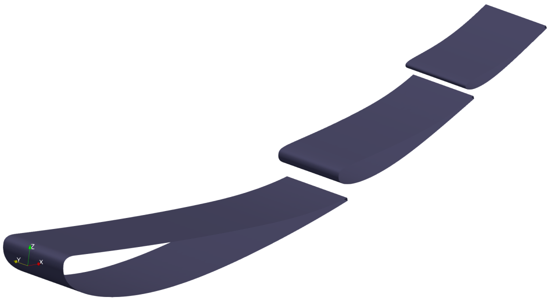

Figure 1: Extruded Imperial Front Wing

The Sub-Stepping scheme has been proven to extend the stability margin of the investigated multi-element wing and allowed for a maximum speed-up by a factor of 4.5 in total computation time while maintaining accuracy similar to that of the Eulerian formulation. In the following sections, the mathematical formulation of the Sub-Stepping scheme is described, and the system of equations is then discretized in space. The flow around the multi-element wing is provided, and the results of the Eulerian form and the Sub-Stepping scheme are compared. A range of different timesteps have been investigated, and the computational performance of the scheme has been analyzed.

2 METHODS

The semi-Lagrangian formulation of the NS equations is expressed as

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} = -\nabla p + \nu\nabla^2\mathbf{u} \tag{1}$$

where the material derivative is expressed in the Lagrangian frame of reference, but the pressure and the viscous terms are in the Eulerian frame of reference. The variables denoted with **bold** correspond to vectors, while the rest are scalars. Following the "High-Order Velocity Correction Scheme"[7], the unsteady term is firstly discretized in time. For simplicity, the Backward Difference Formula with first-order accuracy has been chosen for the following.

$$\frac{D\mathbf{u}}{Dt} = \frac{\mathbf{u}^{n+1} - \mathbf{u}_d^n}{\Delta t} \tag{2}$$

where the $\mathbf{u}_d^n$ corresponds to the velocity at the departure point $\mathbf{x_d} = (x_d, y_d, z_d)$ and at time instance t^n.

It is necessary to obtain the velocity field at the departure point. For this reason, the unsteady advection equation is solved in an auxiliary pseudo-time $t^{n+1-q} < \tau < t^{n+1}$ where q stands for the total number of iterations executed in pseudo-time.

$$\frac{\partial \hat{\mathbf{u}}}{\partial \tau} + \mathbf{u}\cdot\nabla\hat{\mathbf{u}} = 0 \tag{3}$$

where $\hat{\mathbf{u}}$ is a complementary velocity field that represents the influence of convection on the examined flow, u is the advection divergence-free velocity which is extrapolated from the previous time step to the next one.

The second step of the splitting scheme is to equate the pressure term with the material derivative of velocity

$$\frac{\tilde{\mathbf{u}} - \mathbf{u}_d^n}{\Delta t} = -\nabla p^{n+1} \tag{4}$$

where $\tilde{\mathbf{u}}$ is a divergence-free auxiliary velocity field, p^{n+1} is the pressure field at the next time step. Again, using the divergence operator to extract a unique expression for the pressure field at the new time-step

$$\nabla^2 p^{n+1} = \frac{1}{\Delta t}\nabla\cdot\mathbf{u}_d^n \tag{5}$$

Furthermore, $\hat{\mathbf{u}}(x, t^{n+1-q}) = \mathbf{u}(x, t^{n+1-q})$ along the characteristic of the hyperbolic, unsteady advection equation. Based on this relation, the velocity at the departure point is found.

The final step is to correct the velocity field by the pressure at the current timestep.

$$\frac{\mathbf{u}^{n+1} - \tilde{\mathbf{u}}}{\Delta t} = \nu\nabla^2\mathbf{u}^{n+1} \tag{6}$$

$$\tilde{\mathbf{u}} = -\Delta t\nabla p^{n+1} + \mathbf{u}_d^n \tag{7}$$

The velocity Helmholtz equation, or "diffusion" step becomes

$$\nabla^2\mathbf{u}^{n+1} - \frac{1}{\nu\Delta t}\mathbf{u}^{n+1} = \frac{1}{\nu}\nabla p^{n+1} - \frac{1}{\nu\Delta t}\mathbf{u}_d^n \tag{8}$$

Therefore, the final system of equations required to obtain the pressure and velocity field at each timestep is composed by the Eq. 3, 5, 8 and is summarized as a flow-chart in Figure 2.

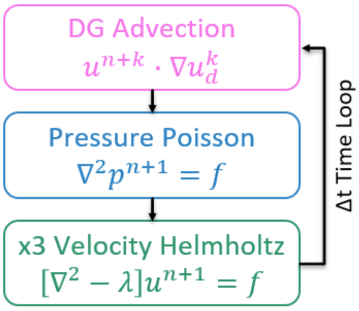

Figure 2: Flow-Chart for the Sub-Stepping Scheme

2.1 Discretization in space

The final system of equations involves solving an unsteady advection equation, a Poisson equation for the pressure field and a Helmholtz equation for the velocity vector field. The unsteady advection equation is discretized in space using a Discontinuous Galerkin (DG) projection[8]. The pressure and velocity equations are discretized in space with a Continuous Galerkin projection[9]. For hyperbolic problems such as the unsteady advection, where the solution is not smooth, it is beneficial to use a DG projection for two main reasons. Firstly, flow discontinuities can be captured more easily without posing stability issues. Secondly, when solving the discretized system of equations, assembling the global mass matrix that includes the influence of all elements is unnecessary. It is possible to solve the system of equations locally within a single element and then progress to the next one until the entire domain is computed. The localized solution of the system, along with efficient programming techniques, can remarkably reduce the computational cost of the advection step. This cost is important because its relationship to the diffusion step is the limiting condition for when it is beneficial to use the Sub-Stepping scheme.

2.2 Notes on the boundary conditions

To close the system of equations, an appropriate boundary conditions is necessary for the pressure field which is obtained by the inner product of Eq. 1 multiplied by the normal vector to the surface.

$$\frac{\partial p^{n+1}}{\partial n} = \mathbf{n} \cdot \left[\nu\nabla^2\mathbf{u}^{n+1} - \frac{\mathbf{u}^{n+1} - \mathbf{u}_d^n}{\Delta t}\right] = -\mathbf{n} \cdot \left[\nu\nabla \times \nabla\mathbf{u}^{n+1} + \frac{\mathbf{u}^{n+1} - \mathbf{u}_d^n}{\Delta t}\right] \tag{9}$$

Given that $\nabla^2\mathbf{u} = \nabla(\nabla\cdot\mathbf{u} - \nabla\times\nabla\mathbf{u})$ and $\mathbf{u}$ is divergence-free. The viscous term was expressed by the solenoidal component of velocity as it is more stable in this form[7]. The velocity at the current timestep can be found via extrapolation from the previous timesteps.

3 PROBLEM DESCRIPTION

The examined multi-element wing is a simplified configuration from the "Imperial Front Wing" (IFW), which is a Formula 1 front-wing geometry that was released as open-source in [10]. Specifically, the IFW was cut at a spanwise location of y = -250 mm. This cross-section was then extended by 50 mm in the spanwise direction to create the "Extruded IFW", the focus of this study, as shown in Figure 1. The characteristic length of the geometry is the same as the chord length of the main element, which is 0.25 m. The freestream velocity is $12.5\frac{m}{s}$, resulting in a $Re = 208,045$. The Extruded IFW is simulated inside free-stream air. The free-stream air enters the domain and is imposed as a uniform Dirichlet boundary condition at the inlet. Likewise, the ceiling and the ground have the same boundary conditions as the inlet. The side walls are treated as slip walls where the gradient of the in-planar velocity components is zero, while the normal velocity component is set to zero to prevent cross-flow development. A high-order stable form of zero Neumann boundary condition at the outlet is used for velocity, and the pressure is fixed to the ambient pressure.

All simulations were performed using *Nektar++*[11], an open-source framework for solving unsteady partial differential equations using the spectral h/p element method. High-order polynomials were used to approximate the flow variables. *NekMesh*[12], the mesh-generation utility of Nektar++, was used to create an unstructured, conformal high-order mesh, which included only prisms and tetrahedrons. A fourth-order polynomial was used for the velocity, and a third-order polynomial was used for the pressure, following a Taylor-Hood approximation to ensure better stabilization properties for the pressure Poisson equation. Polynomial de-aliasing was employed via over-intergration. A second-order accurate time-integration scheme with a constant time-step was utilized. The simulation was initialized from an existing velocity field obtained via RANS modeling, and the relative pressure was set to 0. The discretized system of equations for velocity and pressure was solved numerically using an iterative method, the Conjugate Gradient (CG) algorithm. It involved applying static condensation to reduce the number of algebraic degrees of freedom. To ensure algebraic convergence, an absolute tolerance of $1e^{-4}$ was set for all the flow variables.

4 DISCUSSION ON ACCURACY, STABILITY & PERFORMANCE

The flow around the Extruded IFW was simulated for a physical time of 2.5 seconds, using a range of timesteps from $dt = 1e^{-5}$ sec up to $dt = 4e^{-4}$ sec. The flow results and the total computation time from the standard semi-implicit velocity-correction scheme[7] are treated as reference for the following comparisons. The maximum timestep used for the reference simulation was $dt = 1e^{-5}$ sec. All the simulations were performed using twelve Intel Xeon Platinum 8358 (Ice Lake), 2.60GHz, 32-core, compute nodes, fully populated. Each node includes 64 cores and 512 GB RAM.

4.1 Exploring stability in time

The computational performance of the Sub-Stepping scheme is assessed based on total computation time. Additionally, the duration of its components is monitored to identify potential areas for improvement. The first key performance indicator (KPI) examined is the “Speed-Up” which is defined as the ratio of the reference case’s computation time divided by the sub-stepping scheme’s computation time with a certain discrete timestep. It represents the performance benefits of utilizing the Sub-Stepping scheme compared to the reference. The second KPI of interest is the computational cost to solve one simulation timestep. The variation of the “Speed-Up” with the examined timesteps is illustrated on the left of Figure 3, while the variation of computation time per simulation timestep is on the right side of this figure. The “Speed-Up” increases rapidly as the discretization timestep is increased until $dt = 1e^{-4}$ sec. Beyond this timestep, the scheme's performance saturates, and the reduction rate for the total computation time is lower than before. The maximum "Speed-Up" achieved for this configuration is **3.72**. Similar trends are observed in the computational cost per simulation timestep. Specifically, the cost per timestep slowly increases until $dt = 1e^{-4}$. After this point, there is a sharp increase in the computational cost, highlighting the modest performance benefits.

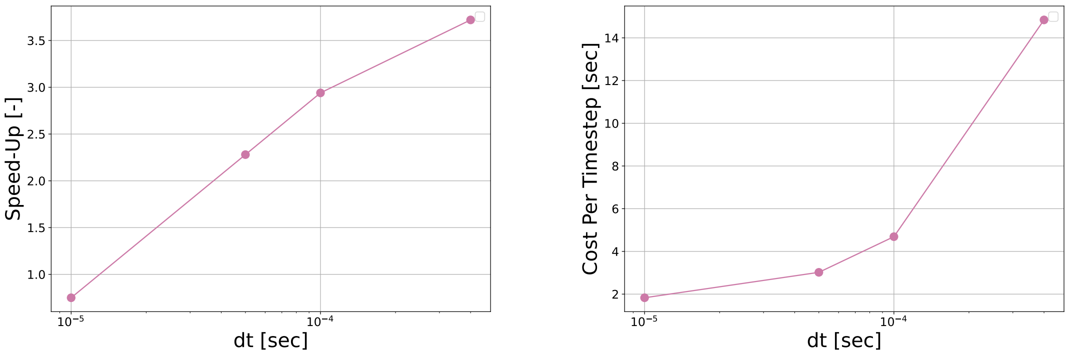

Figure 3: Speed Up (Left) and Cost Per Time Step (Right) for each examined timestep

Overall, the "Speed-Up" is above one for the entire range of the investigated discretization timesteps, except for $dt = 1e^{-5}$ sec. It is beneficial to use the Sub-Stepping scheme across the entire range of examined discrete timesteps as the total computation time of the simulation decreases. Furthermore, the stability margin of the configuration is increased, as it is possible to simulate the flow around the Extruded IFW by using a discrete timestep forty times larger than the reference timestep.

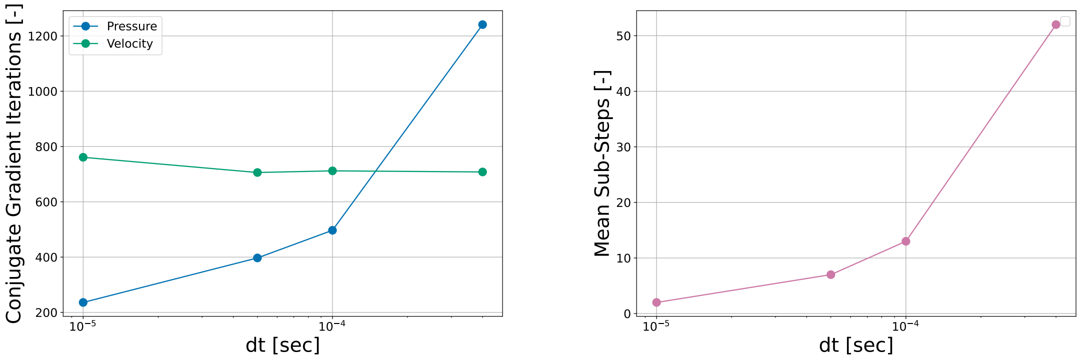

Figure 4: Conjugate Gradient Iterations per timestep (Left) and Mean Sub-Steps per timestep (Right)

It is interesting to understand how the computational cost is distributed when varying the timestep, as it can help unlock the scheme's performance potential. To assess the variation in cost for the pressure and velocity steps, the average pressure and velocity iterations of the conjugate gradient solver are plotted against each discrete timestep in Figure 4. The mean velocity iterations remain nearly constant for each timestep, indicating that the cost of the viscous step remains the same for all simulations. The pressure iterations gradually increase until $dt = 1e^{-4}$ sec. Past this timestep, the pressure iterations rapidly grow, indicating an increase in the cost of the pressure step due to the algebraic system becoming stiffer and harder to converge. For the advection step, the mean Sub-Steps, which represent the pseudo-steps needed to solve the unsteady advection equation, are illustrated on the right side of this figure. The cost of the advection step increases in proportion to the number of Sub-Steps required.

The maximum tested timestep is $dt = 5e^{-4}$ sec and the simulation did not complete, because the pseudo-timestep became smaller than $5e^{-7}$ sec.

4.2 Impact in accuracy

The influence of the timestep increase in the accuracy of the flow results is reviewed based on the integral values such as the coefficient of lift and drag. The variation of the integral forces over time is shown in Figure 5. To aid visualization, the time-averaged signal of the coefficient of lift and drag are presented separately. While the oscillations have minimal impact on the lift coefficient, with an amplitude of only 2% from the mean value, they are more pronounced for the drag coefficient, ranging from 10% to 15% of the mean value. This could be attributed to under-resolved flow features within the wing's boundary layer. The oscillations are primarily driven by pressure contributions to the forces. The simulations utilized a Jacobi preconditioner for the pressure system of algebraic equations. It is anticipated that implementing an improved preconditioner could help reduce these oscillations.

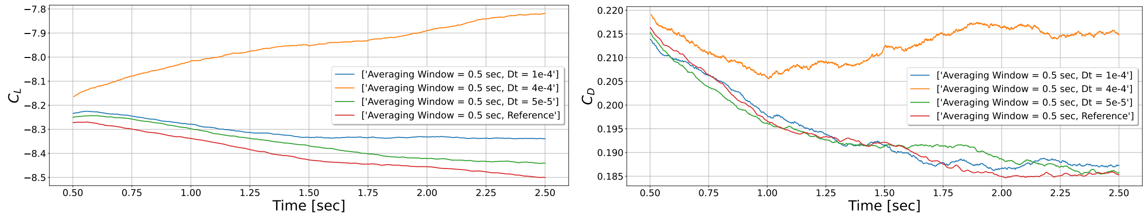

Figure 5: Evolution of integral forces with time

Table 1: Integral forces difference with reference

dt, sec	C_L	$C_{L,diff}$	C_D	$C_{D,diff}$
$1e^{-5}$	-8.39	N/A	0.20	N/A
$5e^{-5}$	-8.35	0.04	0.20	0.0
$1e^{-4}$	-8.30	0.09	0.20	0.0
$4e^{-4}$	-7.98	0.41	0.21	0.01

In Figure 5, the evolution of the coefficient of drag with time is very similar for all examined timestep, except the maximum $dt = 4e^{-4}$ sec. This difference is reflected in the time-averaged value of this coefficient, as seen in Table 1. Despite the variation in the drag force evolution, the difference from the reference case in the time-averaged value is only 5%. A similar trend is observed for the lift coefficient, with the downforce (negative lift) developing to reseble the reference case for all examined timesteps except the maximum one. For the intermediate timesteps, it is clear that the simulations need to be extended further until they reach the same load as in the reference case. In the case of the maximum timestep, the Extruded IFW is losing load, suggesting that the flow is evovling differently than in the reference case. These simulations need to be extended further until convergence in time is achieved and the flow mechanics are assessed between the reference and each timestep.

4.3 Advection performance

It is crucial to efficiently solve the unsteady advection equation, as the cost of the advection step rises rapidly when the discretization timestep nears its stability limit. This can be done in two ways: by reducing the overall number of sub-steps, or by lowering the computational cost for a single sub-step.

The main challenge of the scheme is determining the pseudo-timestep for solving the unsteady advection equation numerically. The pseudo-timestep can be controlled using a user-defined safety factor or by specifying the minimum required number of Sub-Steps. Currently, the pseudo-timestep is determined using the CFL number per simulation timestep, which is adjusted based on the safety factor. Therefore, adjusting the value of the safety factor can change both the pseudo-timestep and the number of Sub-Steps. During the advection step, the unsteady advection equation is solved in pseudo-time using a second-order accurate Runge-Kutta time-integration scheme.

Table 2: Computational Cost Per Timestep and Sub-Step for each discrete timestep

dt, sec	Mean CPU Time Per Timestep, sec	Timer Per Sub-Step, sec
$1e^{-5}$	1.83	0.24
$5e^{-5}$	3.02	0.24
$1e^{-4}$	4.69	0.25
$4e^{-4}$	14.89	0.25

The average computational cost per timestep and Sub-Step for all simulations are summarized in Table 2. In these tests, a safety factor of 0.5 was used, which is a conservative value. The cost per Sub-Step remains the same for all examined discretization timesteps. Therefore, the performance of the advection step is primarily determined by the required number of Sub-Steps.

To explore the influence of the safety factor, its value was varied from 0.1 to 1 while using a timestep of $dt = 1e^{-4}$ sec. The impact of the safety factor on the “Speed-Up” of the simulation is shown in Figure 6. It is evident from the graph that the performance of the scheme is constrained when the safety factor is restricted, leading to a decrease in “Speed-Up”. On the contrary, relaxing the safety factor further reduces the simulation cost without causing any stability issues. The maximum ”Speed-Up” achieved without a safety factor is **4.5**. It is possible to raise the safety factor above one and amplify the estimated CFL value, in order to increase the Sub-Step timestep, thereby lowering the required Sub-Steps. This remark raises the question regarding the adequacy of the current CFL definition for hyperbolic problems that are discretized using DG Galerkin. Alternative CFL definitions for automatically specifying the

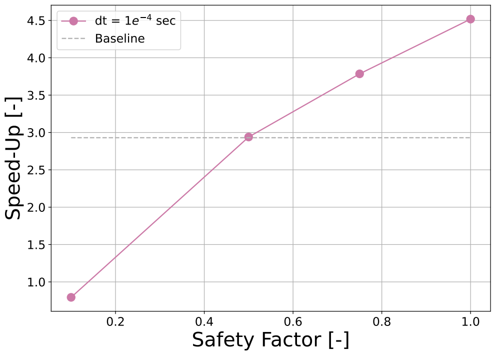

Figure 6: Variation of Speed-Up for $dt = 1e^{-4}$ sec with different safety factors

pseudo-timestep will be explored in future work.

4.4 Parallel performance

Industrial geometries typically involve complex geometrical features that demand many degrees of freedom to capture and resolve the surrounding flow phenomena. These simulations typically take place in a high-performance computing environment using hundreds or even thousands of processing units. For this reason, the parallel performance of the scheme was assessed within such an environment.

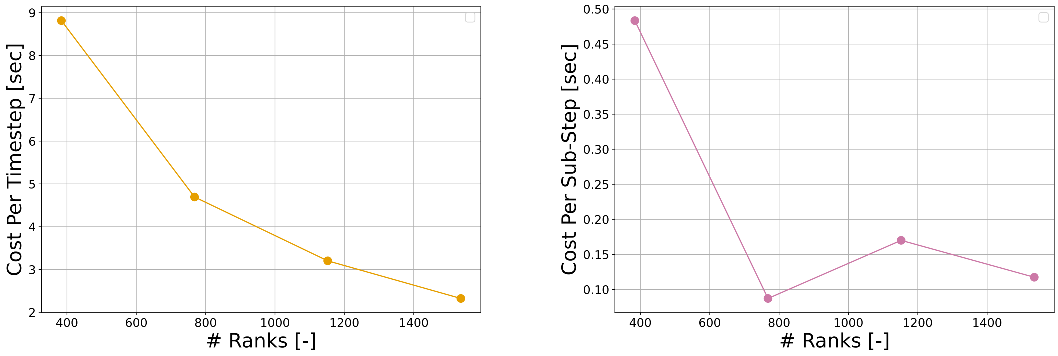

Figure 7: Computational cost per simulation timestep (Left) and Cost per Sub-Step (Right) for increasing ranks

The computational cost per simulation timestep and Sub-Step for increasing number of resources or ranks is shown in Figure 7. As the number of ranks increases while keeping the degrees of freedom constant, the computational cost per simulation timestep decreases. Similarly, the computational cost per sub-step decreases rapidly until 768 ranks. After this point, the rate of decrease slows down, raising concerns about the parallel performance of the advection step in isolation. This could be associated with a potential load imbalance in the advection step, given that it is discretized in space using a discontinuous projection, but the load is distributed between ranks according to the continuous projection.

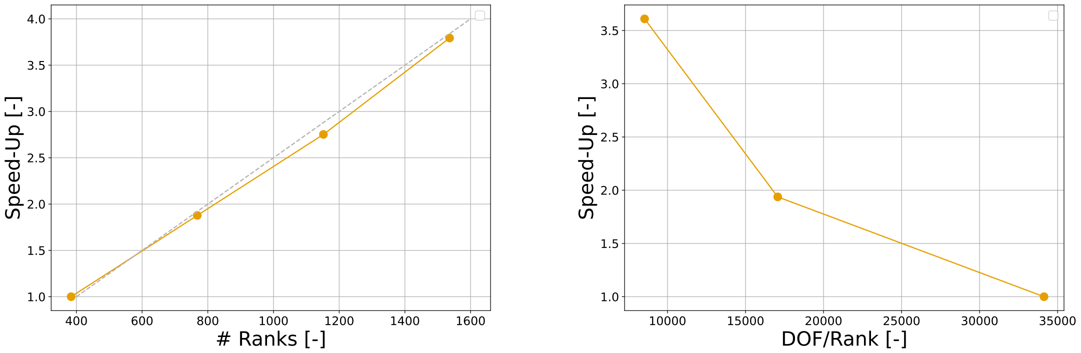

Figure 8: Strong Scaling

The strong and weak scaling of the Sub-Stepping scheme is shown in Figure 8, where the dotted, grey line expresses the ideal scaling and the Speed-Up is defined as

$$Speed\ Up = \frac{Total\ Computation\ Time\ @384\ ranks}{Total\ Computation\ Time\ @N\ ranks} \tag{10}$$

We observe ideal strong scaling for the sub-stepping scheme in Figure 8. To produce the weak scaling plot, the number of nodes was fixed to 12, and the allocation of cores was varied to 16, 32, and 64 successively to modify the distribution of the degrees of freedom (DOF) per rank. In this graph, the reference point is defined at the maximum DOF/rank distribution. As the DOF per rank decreases, the Speed-Up steadily increases until the 17065 threshold. Below this point, the Speed-Up increases rapidly at twice the previous rate. The limited increase in Speed-Up until the 17065 DOF per rank is linked to high memory transfers from the main memory to the cache. This suggests that the code is within its memory-bound limits. With lower DOF per rank, fewer memory transfers are required, and the code substantially benefits from its pure MPI parallelization.

5 CONCLUSIONS

In this study, the performance, accuracy and stability of the Sub-Stepping scheme were investigated around an industry-relevant geometry at a high Reynolds number, specifically the Extruded IFW benchmark. The Sub-Stepping allowed the extension of the simulation stability by using a timestep that is forty times larger than the reference timestep. For this case, a maximum Speed-Up of 3.72 was achieved with a conservative numerical configuration for solving the advection step. The cost of the viscous step remained invariant for all the tested discretization timesteps, while the cost of the pressure step increased as the discretization timestep approached its stability limit. Upon analyzing the computational performance of the scheme, it was found that the total number of Sub-Steps needed to solve the unsteady advection equation is the limiting factor to its efficiency. The number of Sub-Steps can be reduced by relaxing the CFL number used to evaluate the pseudo-timestep of the unsteady advection equation. This way, a maximum Speed-Up of 4.5 was attained without stability issues. Although the overall scheme demonstrated ideal strong scaling, the parallel performance of the unsteady advection equation displayed a concerning trend that requires further investigation.

ACKNOWLEDGEMENTS

This project received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 955923.

The authors acknowledge computational resources and support provided by the Imperial College Research Computing Service (http://doi.org/10.14469/hpc/2232).

REFERENCES

[1] Maday Y., Patera A.T., Rønquist E.M., "An Operator-integration-factor splitting method for time-dependent problems: Application to incompressible fluid flow." *J Sci. Comput.*, Vol. **5**, pp. 263–292, 1990.

[2] Xiu D. and Karniadakis G. E. 2001. "A semi-Lagrangian high-order method for Navier-Stokes equations" Journal of Computational Physics, 658-684. https://doi.org/10.1006/jcph.2001.6847

[3] Xu J., Xiu D. and Karniadakis G. E., "A Semi-Lagrangian Method for Turbulence Simulations Using Mixed Spectral Discretizations" J Sci. Comput. Vol. **17**, pp. 585–597, 2002.

[4] Sherwin, S. J. 2003. "A Substepping Navier-Stokes Splitting Scheme for Spectral/hp Element Discretisations" J Parallel CFD 2002, 43-52. https://doi.org/10.1016/B978-044450680-1/50006-1

[5] Xiu D., Sherwin S. J., Dong S. and Karniadakis, G. E., "Strong and Auxiliary Forms of the Semi-Lagrangian Method for Incompressible Flows" Journal of Scientific Computing, 323-346. https://doi.org/10.1007/s10915-004-4647-1

[6] Patel S. S., Fischer P. F, Min M. and Tomboulides A. G., "An Operator-Integration-Factor Splitting (OIFS) method for Incompressible Flows in Moving Domains" (Report ANL/ALCF-17/8). Argonne Leadership Computing Facility. https://publications.anl.gov/anlpubs/2017/12/140626.pdf

[7] Karniadakis G. E., Orszag S. A., "High-Order Splitting Methods for the Incompressible Navier-Stokes Equations" J Computational Physics Vol. **97**, pp. 414–443, 1991.

[8] Hesthaven J. S. and Warburton T. 2008. "Nodal Discontinuous Galerkin Methods", Springer. https://link.springer.com/book/10.1007/978-0-387-72067-8

[9] Karniadakis G. E. and Sherwin S. J. 2005. "Spectral/hp Element Methods for Computational Fluid Dynamics", 2nd Ed. Oxford University Press. https://doi.org/10.1093/acprof:oso/9780198528692.001.0001

[10] Buscariolo F.F, Hoessler J., Moxey D., Jassim A., Gouder K., Basley J., Murai Y., Assi G. R. S. and Sherwin S. J. "Spectral/hp element simulation of flow past a Formula One front

wing: validation against experiments" J. Wind Eng. Ind. Aerodynamics, Vol. **221**, 104832 2022.

[11] Moxey D., Cantwell C. D., Bao Y., Cassinelli A., Castiglioni G., Chun S., Juda E., Kazemi E., Lackhove K., Marcon J., Mengaldo G., Serson D., Turner M., Xu H., Peiró J., Kirby R. M. and Sherwin S. J., "Nektar++: Enhancing the capability and application of high-fidelity spectral/hp element methods" Computer Physics Communications, Vol. **249**, 107–110, 2020.

[12] Kirilov K. S., Peiró J., Zhoum J., Green M. D. and Moxey D. "High-Order Curvilinear Mesh Generation From Third-Party Meshes" Proceedings of the 2024 International Meshing Roundtable (IMR), 93–105, 2024.