Recent Developments in the CHApel Multi-Physics Simulation Software

Frédéric Plante
Michael Gagnon
Eric Laurendeau
frederic.plante@polymtl.ca
michael.gagnon@polymtl.ca
eric.laurendeau@polymtl.ca
Polytechnique Montréal
Montréal, Québec, Canada

ABSTRACT

This paper presents the recent developments to the CHApel Multi-Physics Simulation (CHAMPS) software developed at Polytechnique Montréal. This software is at its core a 2D and 3D flow simulation software relying on the Unsteady Reynolds-Averaged Navier-Stokes (URANS) equations towards application to aircraft aerodynamics. Other physics such as ice accretion and structural mechanics allow to perform aero-icing and aero-elastic simulations. New features added this year include methods to handle overset grids and to perform linear stability analyses of fluid simulations. These features are presented on concrete aircraft applications.

CCS CONCEPTS

• **Applied computing** → **Aerospace**.

KEYWORDS

Computational Fluid Dynamics, Stability Analysis, Transonic Buffet, Overset Grids

ACM Reference Format:
Frédéric Plante, Michael Gagnon, and Eric Laurendeau. 2022. Recent Developments in the CHApel Multi-Physics Simulation Software. In *Chapel Implementers and Users Workshop 2022, June 10, 2022, Virtual format*. 7 pages.

1 INTRODUCTION

The aerospace industry relies more and more on numerical simulations for the design and analysis of new products. These simulations can be used to evaluate aircraft performance (drag and lift prediction) or to evaluate the edges of the flight envelope (stall, buffet, icing, etc.). With the development of the numerical methods and the increased availability of the computational resources, the complexity of the problems also increases. The inclusion of multiple physics like the coupling between the fluid and the structure or the computation of ice accretion can be challenging for the numerical methods. Additionally, the trend to model unsteady phenomena greatly increases the computational cost of the simulations. The problems can also be made more complex by the geometry. As the modeled geometries become more complex, the generation of grids for the solution of partial differential equations becomes challenging.

CHAMPS is a research code used to develop and test novel simulation techniques. Since its first working version in September 2019 as an inviscid flow solver, the size and capabilities of this software have largely increased. The figure 1 shows the evolution of the number of code lines from the version 1.0 to the current version 2.0.2. Previous presentations discussed the capabilities of the flow solver [11], the water droplet solver and the thermodynamic model. Together, they allow to perform ice accretion simulations [10].

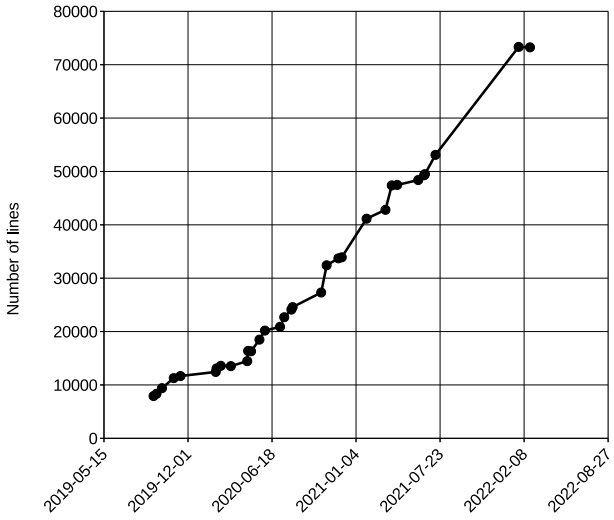


Figure 1: Number of code lines in champs since the first version

This paper shows new capabilities added to this software. In order to reduce the computational cost to predict unsteady phenomena, a module to perform global linear stability analysis [15, 17–19] was added. These analyses allow to predict instabilities by an analysis of the Jacobian matrix of the governing equations. A module

CHIUW 2022, June 10, 2022, Online

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00

to alleviate the grid generation process is also added in the form of the overset grid technique. This method allows to mesh parts of the domain with overlapping grids and to assemble them to have a valid computational domain. Interpolation methods are then used to exchange information between overlapping grids.

The flow solver is first introduced. Then the implementation of the linear stability module is described and verified with an application to a wing-body buffet case. The overset grid method is finally presented with an application to a full-aircraft configuration.

2 FLOW SOLVER

CHAMPS solves the Reynolds-Averaged Navier-Stokes equations with a finite volume scheme. These equations can be written in semi-discrete integral form as:

$$\boldsymbol{V}\frac{\partial \boldsymbol{W}}{\partial t^*} + \boldsymbol{V}\frac{\partial \boldsymbol{W}}{\partial t} + \boldsymbol{R}(\boldsymbol{W}) = 0 \tag{1}$$

where $\boldsymbol{W} = [\rho, \rho u, \rho v, \rho w, \rho e, \tilde{\nu}]^t$ is the conservative variables vector, $\boldsymbol{V}$ is the cell volume diagonal matrix and $\boldsymbol{R}$ is the summation of the source terms and the integral of the convective and viscous fluxes. In this case the field variable of the Spalart-Allmaras turbulence model [16] $\tilde{\nu}$ is included in the system, resulting in a fully coupled formulation. This system of equations can also be solved using a segregated approach where the flow and the turbulence are treated separately, in which case any turbulence model can be used, such as the 2 equations $K-\omega$ model [7]. In equation 1 t denotes the physical time used for time accurate simulations with a dual time stepping approach and t^* is a pseudo-time used to march towards a steady state. The physical time step are treated like a steady-state solution. The time step is local to each cells and selected to accelerate the convergence. In the case of a steady solution, the pseudo-time derivatives are null. In the dual time stepping approach the time derivatives are discretized with a second order backward finite difference:

$$\frac{\partial \boldsymbol{W}}{\partial t} = \frac{3\boldsymbol{W}^n - 4.0\boldsymbol{W}^{n-1} + \boldsymbol{W}^{n-2}}{2\Delta t} \tag{2}$$

The pseudo-time derivatives are discretized with a backward Euler scheme:

$$\left(\frac{\boldsymbol{V}}{\Delta t} + \left.\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right|_{W_n}\right)\Delta \boldsymbol{W} = -\boldsymbol{R}(\boldsymbol{W}_n) \tag{3}$$

If we neglect the Jacobian matrix $\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}$, a Runge-Kutta scheme can be used for the resulting explicit scheme. Otherwise, the linear system of equations is solved using a Block Gauss-Seidel method or a Generalised Minimized Residual (GMRES) [13] algorithm. An approximated version of the Jacobian matrix based on first order fluxes is used in most cases. The GMRES method can also be used in a Jacobian-free manner where the product of the Jacobian by a vector is evaluated using a second-order finite difference with no simplification to the fluxes evaluation:

$$\boldsymbol{A}\boldsymbol{q} = \frac{\boldsymbol{R}(\boldsymbol{W}_0 + \epsilon \boldsymbol{q}) - \boldsymbol{R}(\boldsymbol{W}_0 - \epsilon \boldsymbol{q})}{2\epsilon} \tag{4}$$

where ϵ is a small parameter, typically of the order of 10^{-8}.

The solver uses unstructured grids on which the conservative values are stored at the center of the grid cells and the fluxes are evaluated at the facets of these cells. The convective fluxes can be computed using the Roe or AUSM schemes and second order accuracy is achieved by using a piecewise linear reconstruction or the UMUSCL scheme [2]. The gradients can be computed using the Green-Gauss or the Weighted Least Square approach. A first or second order upwind scheme is used for the convective fluxes of the turbulence model. This software was verified for a set of canonical and application test cases [11].

3 LINEAR STABILITY ANALYSIS

A module to perform the linear stability analysis of URANS steady flow solutions $\boldsymbol{W}_0$ is implemented. This is done by linearizing the URANS equations around the fully converged solution $\boldsymbol{W}_0$:

$$\boldsymbol{V}\frac{\partial(\boldsymbol{W}_0 + \boldsymbol{W}')}{\partial t} + \boldsymbol{R}(\boldsymbol{W}_0 + \boldsymbol{W}') = 0 \tag{5}$$

$$\boldsymbol{V}\frac{\partial \boldsymbol{W}_0}{\partial t} + \boldsymbol{R}(\boldsymbol{W}_0) + \boldsymbol{V}\frac{\partial \boldsymbol{W}'}{\partial t} + \left.\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right|_{\boldsymbol{W}_0}\boldsymbol{W}' = 0 \tag{6}$$

$$\frac{\partial \boldsymbol{W}'}{\partial t} = -\boldsymbol{V}^{-1}\left.\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right|_{\boldsymbol{W}_0}\boldsymbol{W}' \tag{7}$$

Here $\boldsymbol{W}'$ is a small perturbation and we note that $\frac{\partial \boldsymbol{W}_0}{\partial t} + \boldsymbol{R}(\boldsymbol{W}_0) = 0$. The physical time derivatives are null as well. The pseudo-time derivatives are dropped since they are a numerical artifacts introduced for the iterative scheme, to solve the nonlinear equations. For convenience, the Jacobian matrix $-\boldsymbol{V}^{-1}\left.\frac{\partial \boldsymbol{R}}{\partial \boldsymbol{W}}\right|_{\boldsymbol{W}_0}$ will be defined as $\boldsymbol{A}$. By assuming a solution in the form of a normal mode:

$$\boldsymbol{W}' = \hat{\boldsymbol{W}}e^{\lambda t} \tag{8}$$

the problem reduces to solving the eigenvalues and eigenvectors of the matrix $\boldsymbol{A}$:

$$\lambda \hat{\boldsymbol{W}}e^{\lambda t} = A\hat{\boldsymbol{W}}e^{\lambda t} \tag{9}$$

$$\lambda \hat{\boldsymbol{W}} = A\hat{\boldsymbol{W}} \tag{10}$$

where $\lambda = \sigma + \omega j$ gives the growth rate σ and the frequency ω, and the eigenvectors are the spatial shape of the modes. Here $j = \sqrt{-1}$.

In CHAMPS, the eigenvalues are solved using an Arnoldi iteration with a shift-and-invert spectral transformation [3]. This allows to choose a point η in the complex plane around which we want to obtain the eigenvalues. The eigenproblem is then:

$$(\boldsymbol{A} - \eta \boldsymbol{I})^{-1}\boldsymbol{x} = \boldsymbol{x}\kappa \tag{11}$$

where the eigenvalue $\kappa = 1/(\lambda - \eta)$. This method requires solving linear systems of equations, with the matrix $\boldsymbol{A} - \eta \boldsymbol{I}$, for which we use the iterative GMRES method. In this method we must compute the product of $\boldsymbol{A} - \eta \boldsymbol{I}$ by a set of vectors to iteratively compute the solution of the linear system. The Jacobian-free implementation allows to treat the system of equations as a black box and to introduce no simplification assumption by using the finite difference to compute the matrix vector products (eq: 4). This has the advantage of more precisely representing the dynamics of the system than an approximated Jacobian.

The particularity of this implementation is that the system of equations is now complex because the shift η can be complex. In that case, we convert the complex problem into a real value problem of doubled the size and we call the finite difference twice to have the

product of the matrix $\boldsymbol{A}$ (which is real) by the real and imaginary parts of the vector $\boldsymbol{q}$ separately. Then the product of $\eta \boldsymbol{I}$ by the vector $\boldsymbol{q}$ can be directly added. Note that in the event that the shift is taken as a real value the algorithm is simplified to limit the memory usage and computational cost. This is important since the memory consumption of this algorithm is a limiting factor to the size of the problems to be solved.

The implementation of this method has been made easier by using object oriented features. In Champs, the linear solvers and now the eigenvalue solver are implemented to work with any type of partial difference equations. For instance, any turbulence model and flow solver contain the functions to compute the matrix vector product with a finite difference. Hence, the GMRES method can be readily used for any of these equations. We also have the possibility to do fully coupled analysis where this finite difference is performed for the flow equations and the turbulence model at the same time. This is required for the stability of turbulent flow problems. This also means that if new physics are added to Champs we could be able to reuse the same stability analysis. The same concept works well in Champs to apply the same linear solvers to all the systems of equations.

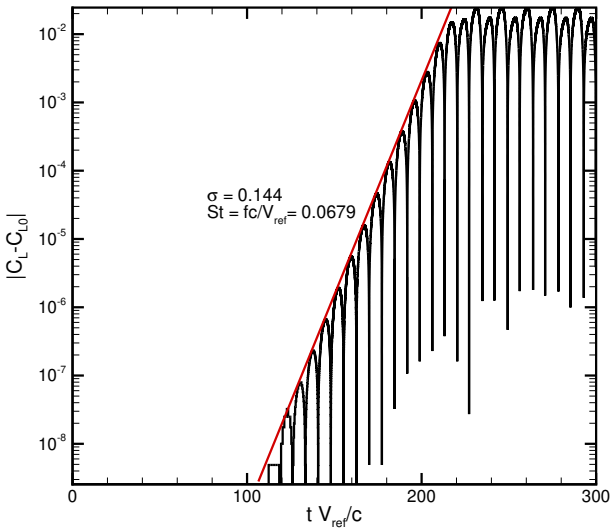


Figure 2: Bifurcation of the OAT15A URANS simulation.

This method is now used to predict the transonic buffet instability. The test case of the ONERA OAT15A airfoil at a Mach number of 0.73, a Reynolds number of 3 million and an angle of attack of 4° is used [14]. A 2D structured grid of 512 by 128 cells is used. The first step of the analysis is to obtain a steady-state flow solution. The Selective Frequency Damping [1] method is used to force the convergence to this steady state in conditions where the flow could be unstable. Then, the dual time stepping approach is used to compute a time-accurate solution of the development of the instability around the base flow $\boldsymbol{W_0}$. This allows to have a reference for the physics that should be captured by the stability analysis. In the early stage of the unsteady simulation, if there is an unstable mode, the simulation should bifurcate with a linear mode which is the same as the result of the stability analysis. In a second time we carry out the stability analysis of the same base flow and we compare the results. Figure 2 shows the absolute value of the lift coefficient subtracted of the steady-state solution lift coefficient on a log scale. Hence, the growth rate is the slope of this curve (red line) and the frequency is extracted from the time between 3 local maximum (this corresponds to one buffet cycle because the graph is in absolute value and the lift coefficient oscillates around the steady-state value). A non-dimensionalized growth rate $\sigma/V_{ref} = 0.144$ and a Strouhal number $St = fc/V_{ref} = 0.0679$ are found. Here, c is the chord length of the airfoil, $f = \omega/2\pi$ and V_{ref} is the free-stream velocity. Figure 3 shows the eigenvalues of the Jacobian matrix of this case. There is one unstable eigenvalue (with a positive growth rate) and its value is the one extracted from the URANS simulation. Hence, the stability analysis allows to predict the onset of transonic buffet without carrying a costly time-accurate simulation.

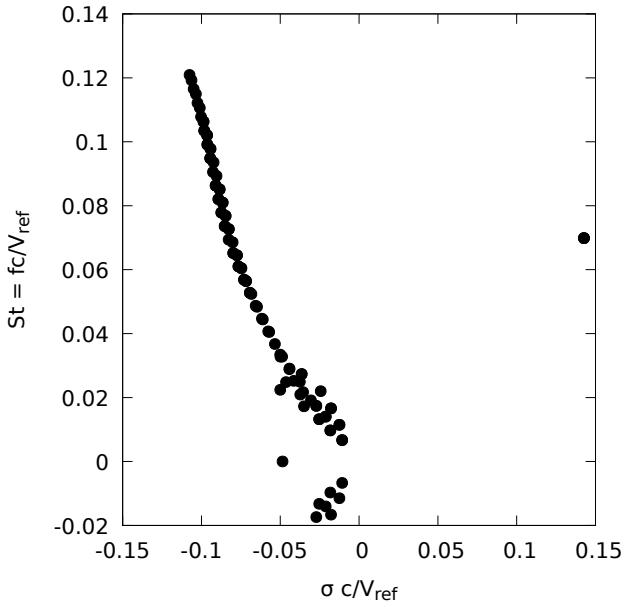
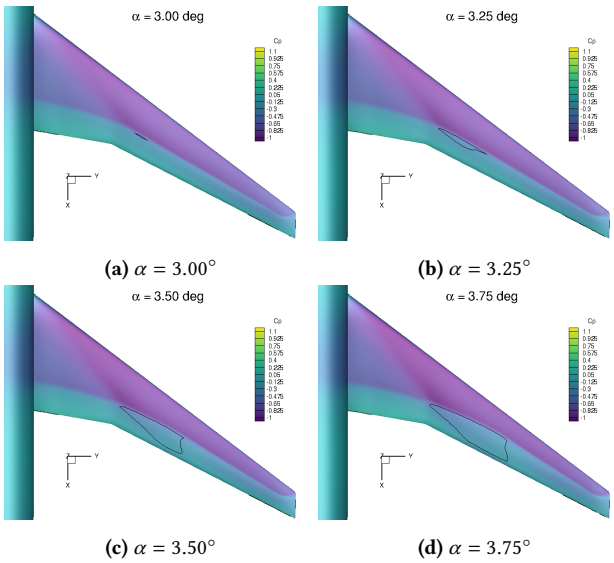


Figure 3: Eigenvalue spectrum for the OAT15A case

This method is also applied to the test case 2a of the 7th AIAA CFD Drag Prediction Workshop [9]. The geometry is the NASA Common Research Model with a fixed aero-elastic deflection which varies with the angle of attack at a Mach number of 0.85 and a Reynolds number of 20 million. First we compute a steady-state solution for several angles of attack. A structured grid of 5 million cells is used. For these cases, the Selective Frequency Damping is not required. Figure 4 shows the surface pressure with a black contour line showing the region where the friction coefficient in the x-direction is negative. This is an indication of potential flow separation. We observe that this region is negligible at an angle of attack of 3° and grows with α. When solving the eigenproblem based on the Jacobian matrix of the flow equations around these solutions we obtain the eigenvalues in Figure 5. The onset of a mode



(a) $\alpha = 3.00°$ **(b)** $\alpha = 3.25°$

(c) $\alpha = 3.50°$ **(d)** $\alpha = 3.75°$

Figure 4: Surface pressure coefficient and contour of x-direction friction coefficient equal to zero for the DPW7 case

with a positive growth rate is between $\alpha = 3.0°$ and $3.25°$, which is consistent with the presence of flow separation. The eigenvectors on the surface of the wing for the modes with the largest growth rate are shown in Figure 6. The mode form cells in the shock foot region. This is consistent with the results of Timme [20].

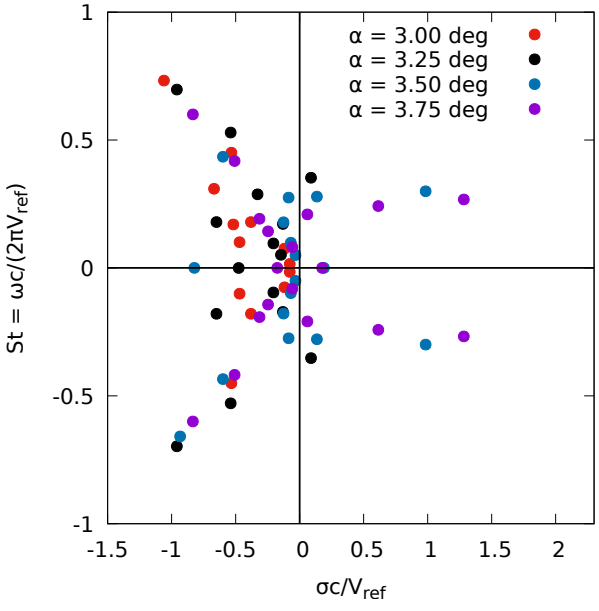
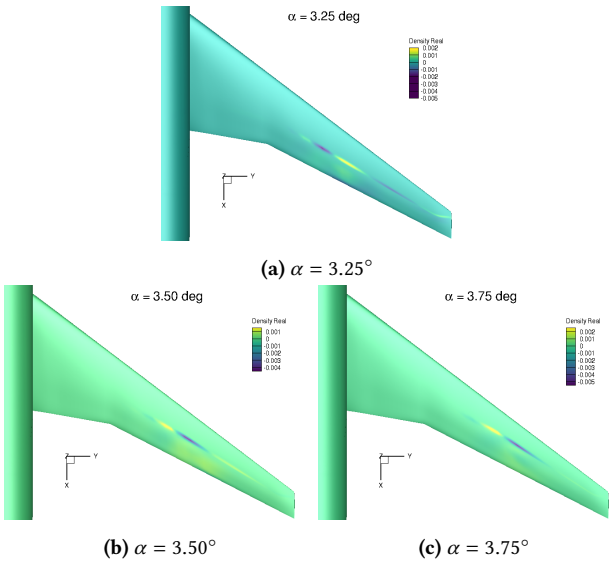


Figure 5: Eigenspectrum of the DPW7 case



(a) $\alpha = 3.25°$

(b) $\alpha = 3.50°$ **(c)** $\alpha = 3.75°$

Figure 6: Leading eigenmode for the DPW7 case

4 OVERSET GRID DEVELOPMENT

The quality of the grid(s) onto which a simulation is performed directly impacts the quality of the simulation results[5]. It was shown that two different discretizations of the same domain could lead to substantially different results. The grid generation process is thus a critical step, representing often around 50% of the overall time of the CFD process[21]. The overset approach decomposes a complex domain into simpler ones, which can be meshed separately. In contrast with the classic structured/unstructured hybrid grid approach, the overset grids do not require conformal matching frontiers. The connectivity is rather ensured by an arbitrary overlap between the grids. Each grid can then be tailored for its local discretization with minimum constraints regarding the inter-grid connectivity. This simplifies the grid generation process and increases the user's control on the grid characteristics. A complex geometry like a complete aircraft configuration can be subdivided as shown in Figure 7. Here, the wing and horizontal stabilizer are meshed separately from the fuselage.

A preprocessor is implemented in CHAMPS to interpret the overset grids and to create the inter-grids connectivity. Because of the arbitrary overlap between the grids, it is expected that some grid elements are to overlap solid bodies from other overlapping grids. Those grid elements are outside of the computational domain and must be located to properly define the inter-grid connectivity. The X-ray hole cutting algorithm [6] casts rays onto the solid components to create a Cartesian mapping based on the collision points. Figure 8 illustrates, from left to right and top to bottom, the hole cutting process using the X-ray method. For a certain overlapping grid points (e.g., P1 or P2), the data structure supporting ray distribution allows a quick identification of the surrounding rays. Their pierce points are then used to create bounding boxes which are used to determine the inside/outside status. To improve the robustness of the method, CHAMPS's user can specify to cast rays in X, Y and/or Z orientations.

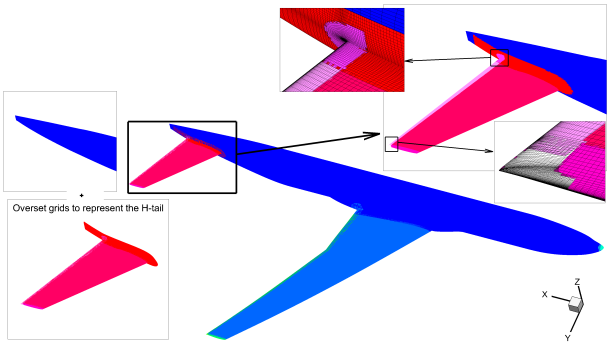


Figure 7: Representation of the overset grid approach applied to the NASA Common Research Model transonic Wing-Body-Tail. Grids were made available by Boeing as part of the 4th Drag Prediction Workshop [8].

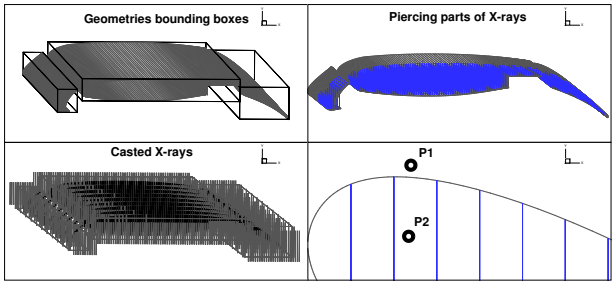


Figure 8: Hole cutting process using the X-ray method.

Different approaches exist to build the inter-grid connectivity. In CHAMPS, this connectivity is ensured by interpolations between the overlapping grids. Similar to the multi-block approach, information are communicated at each iteration of the solver. The hole cutting step identifies the preliminary position of the hole boundaries where inter-grid connectivity must be defined. The *hole cut* grid elements are not computed and must not impact the computed domain. Therefore, the grid elements neighboring those *hole cut* ones are defined as receivers. Furthermore, grid elements neighboring overlapped field boundaries are also identified as receivers. This first identification of the receiver elements ensure a minimum connectivity, but the final position of the communication interfaces must be revised to find the best donor/receiver pairs. To do so, a donor search is done on every overlapped grid elements. Using criteria characterizing the grid elements quality allows to first define the dominance between every overlapping grid element. This will also implicitly position the overset interfaces at an optimum position where overlapping grid elements are the most similar. This approach is commonly called the *implicit approach*. Because a low-resolution region cannot properly capture flow features that are formed in high resolution ones, communication between grid elements with different sizes will significantly impact the solution accuracy and speed of convergence of the simulation[4, 12].

To support the donor search algorithm, an oct-tree data structure is used [22]. This structure allows to decrease the number of *donor search tests* to complete : every grid element is stored in a *leaf node* which contains no more than a user-controlled number of elements. These *leaf nodes* are spatially defined as presented in Figure 9. Every potential donor element can be quickly filtered by the structure by verifying if it is contained in the node.

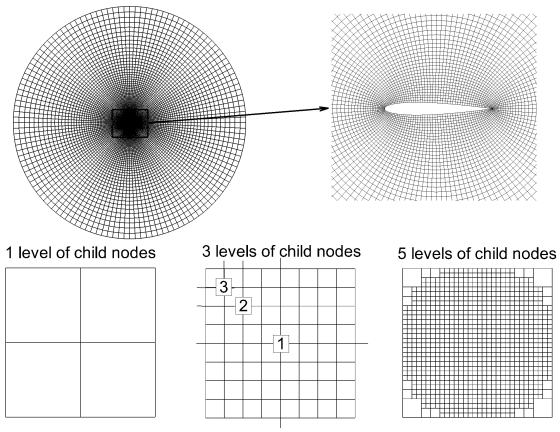


Figure 9: Illustration of the oct-tree structure created on a NACA0012 Euler grid with one, three and five maximum levels of child nodes.

The oct-tree structure allowed to significantly accelerate the process as it is presented in Figure 10. To further take advantage of the independence of the overlapping grids with each other, collar grids are often used to better assemble critical region. As it can be observed in Figure 7, the wing and fuselage solid components intersect each other. Generating a grid which is fitted to the wing and the fuselage would put many constraints on the grid, especially for a structured grid. The collar grid approach alleviates these constraints by using additional grids that are especially generated to define the intersection regions. Zooms on the tail in Figure 7 shows that several *collar grids* are used to define the tail-body intersection and the trailing edge and tip of the tail.

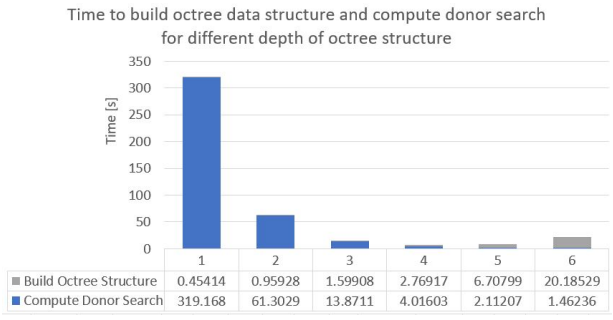


Figure 10: Time (in seconds) to build oct-tree data structure and compute donor search for different depths of oct-tree structure.

4.1 Application

One of the advantages of the approach is to easily add or remove elements from a configuration. In the overset grid presented in Figure 7, the tail can be modified by simply re-preprocessing the overset grids using different parameters. Good results are obtained on this overset grid. The flow conditions are a Reynolds number of 5 million, a Mach number of 0.85 and the angle of attack is automatically adapted to have a lift coefficient of 0.5. The simulation is carried out with the Block Gauss-Seidel scheme, the Spalart-Allmaras turbulence model is used and the convective fluxes are discretized with a first order Roe scheme.

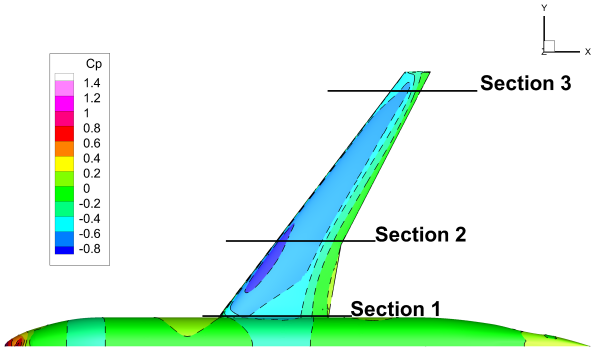


Figure 11: Pressure distribution on the NASA Common Research Model transonic Wing-Body configuration.

Figure 11 shows the pressure coefficient distribution on the aircraft surface and the figure 12 shows the extraction of the pressure coefficient at three spanwise locations. The pressure distributions are smooth and continue throughout the surface which is composed of several surface grids. Surface segments from different grids and collar grids compose the overall solid surface. The surface used to compute the pressure distribution is therefore also from different grids and collar grids. This shows a proper implementation and effectiveness of the different methods in the overset preprocessor. Figure 13 shows the convergences that were monitored for this simulation. The solution converges to machine accuracy and the spikes are caused by the algorithm to change the angle of attack in order to get the prescribed lift coefficient.

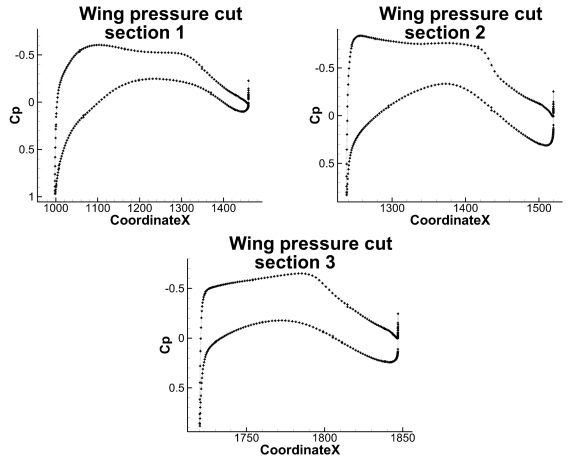


Figure 12: Pressure distribution for three different cut sections along the wing of the NASA Common Research Model transonic Wing-Body configuration.

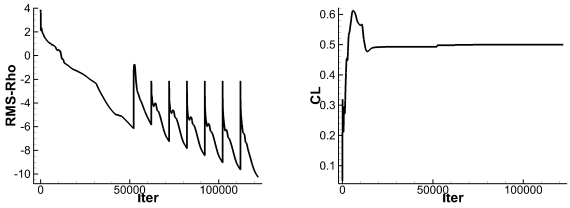


Figure 13: Convergence on the NASA Common Research Model transonic Wing-Body configuration using the full overset approach. On the left is the convergence of density and on the right the lift coefficient.

5 CONCLUSION

In this paper, the recent developments made in the CHApel Multi-Physics Simulation software, namely the addition of a stability solver for flow solutions and the treatment of overset grids, are presented. Both methods are verified for industrial level applications namely the prediction of the transonic buffet on an aircraft and the computation of the steady flow around an aircraft meshed with an overset grid.

ACKNOWLEDGEMENT

This work is supported by a NSERC/CRIAQ/Bombardier Industrial Research Chair. Computations were performed on Compute Canada/Calcul Québec Beluga and Narval clusters.

REFERENCES

[1] Espen Åkervik, Luca Brandt, Dan S. Henningson, Jérôme Hoepffner, Olaf Marxen, and Philipp Schlatter. 2006. Steady Solutions of the Navier-Stokes Equations by Selective Frequency Damping. *Physics of Fluids* 18, 6 (2006), 068102. https://doi.org/10.1063/1.2211705

[2] O. Baş, A. R. Cete, S. Mengi, I. H. Tuncer, and U. Kaynak. 2017. A Novel Alternating Cell Directions Implicit Method for the Solution of Incompressible Navier Stokes Equations on Unstructured Grids. *Journal of Applied Fluid Mechanics* 10, 6 (2017), 1561–1570. https://doi.org/10.29252/jafm.73.245.27654

[3] K. N. Christodoulou and L. E. Scriven. 1988. Finding Leading Modes of a Viscous Free Surface Flow: An Asymmetric Generalized Eigenproblem. *Journal of Scientific Computing* 3, 4 (1988), 355–406. https://doi.org/10.1007/BF01065178

[4] Noah Kim and William Chan. 2011. Automation of Hole-Cutting for Overset Grids Using the X-rays Approach. AIAA Paper 2011-3052. https://doi.org/10.2514/6.2011-3052

[5] Dimitri Mavriplis. 2005. Grid Resolution Study of a Drag Prediction Workshop Configuration Using the NSU3D Unstructured Mesh Solver. AIAA Paper 2005-4729. https://doi.org/10.2514/6.2005-4729

[6] Robert Meakin. 2001. Object X-rays for cutting holes in composite overset structured grids. AIAA Paper 2001-2537. https://doi.org/10.2514/6.2001-2537

[7] F. R. Menter. 1994. Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications. *AIAA Journal* 32, 8 (1994), 1598–1605. https://doi.org/10.2514/3.12149

[8] Joe Morrison. 2021. 4th AIAA CFD Drag Prediction Workshop. https://dpw.larc.nasa.gov/DPW4/overset_Boeing/.

[9] Joe Morrison. 2021. 7th AIAA CFD Drag Prediction Workshop. https://aiaa-dpw.larc.nasa.gov/.

[10] Hélène Papillon Laroche, Simon Bourgault-Côté, Matthieu Parenteau, and Éric Laurendeau. 2021. Development of an Aircraft Aero-Icing Suite Using Chapel Programming Language. In *Chapel Implementers and Users Workshop*.

[11] Matthieu Parenteau, Simon Bourgault-Cote, Frédéric Plante, Engin Kayraklioglu, and Eric Laurendeau. 2021. Development of Parallel CFD Applications with the Chapel Programming Language. AIAA Paper 2021-0749. https://doi.org/10.2514/6.2021-0749

[12] Stuart E. Rogers, Norman E. Suhs, and William E. Dietz. 2003. PEGASUS 5: An Automated Preprocessor for Overset-Grid Computational Fluid Dynamics. *AIAA Journal* 41, 6 (2003), 1037–1045. https://doi.org/10.2514/2.2070

[13] Youcef Saad and Martin H. Schultz. 1986. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.* 7, 3 (1986), 856–869. https://doi.org/10.1137/0907058

[14] Fulvio Sartor, Clément Mettot, and Denis Sipp. 2015. Stability, Receptivity, and Sensitivity Analyses of Buffeting Transonic Flow over a Profile. *AIAA Journal* 53 (2015), 1980–1993. https://doi.org/10.2514/1.j053588

[15] Denis Sipp, Olivier Marquet, Philippe Meliga, and Alexandre Barbagallo. 2010. Dynamics and Control of Global Instabilities in Open-Flows: A Linearized Approach. *Applied Mechanics Reviews* 63, 3 (2010), 030801. https://doi.org/10.1115/1.4001478

[16] Philippe R. Spalart and Steve R. Allmaras. 1992. A One-Equation Turbulence Model for Aerodynamic Flows. AIAA Paper 1992-0439. https://doi.org/10.2514/6.1992-439

[17] Kunihiko Taira, Steven L. Brunton, Scott T. M. Dawson, Clarence W. Rowley, Tim Colonius, Beverley J. McKeon, Oliver T. Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S. Ukeiley. 2017. Modal Analysis of Fluid Flows: An Overview. *AIAA Journal* 55, 12 (2017), 4013–4041. https://doi.org/10.2514/1.J056060

[18] Vassilios Theofilis. 2003. Advances in Global Linear Instability Analysis of Nonparallel and Three-Dimensional Flows. *Progress in Aerospace Sciences* 39, 4 (2003), 249–315. https://doi.org/10.1016/S0376-0421(02)00030-1

[19] Vassilios Theofilis. 2011. Global Linear Instability. *Annual Review of Fluid Mechanics* 43, 1 (2011), 319–352. https://doi.org/10.1146/annurev-fluid-122109-160705

[20] Sebastian Timme. 2020. Global Instability of Wing Shock-Buffet Onset. *Journal of Fluid Mechanics* 885 (2020), A37. https://doi.org/10.1017/jfm.2019.1001

[21] H. K. Versteeg and W. Malalasekera. 2007. *An introduction to computational fluid dynamics: the finite volume method* (2nd ed ed.). Pearson Education Ltd, Harlow, England ; New York. OCLC: ocm76821177.

[22] K. Yamaguchi, T. Kunii, K. Fujimura, and H. Toriya. 1984. Octree-Related Data Structures and Algorithms. *IEEE Computer Graphics and Applications* 4, 1 (1984), 53–59. https://doi.org/10.1109/mcg.1984.275901