33RD CONGRESS
OF THE INTERNATIONAL COUNCIL
OF THE AERONAUTICAL SCIENCES
STOCKHOLM, SWEDEN, 4-9 SEPTEMBER, 2022

ICAS 2022 SWEDEN

COMPARING POTENTIAL FLOW SOLVERS FOR AERODYNAMIC CHARACTERISTICS ESTIMATION OF THE T-FLEX UAV

Fanglin Yu[1], Julius Bartasevicius[1] & Mirko Hornung[1]

[1] Institute of Aircraft Design, Technical University of Munich, Boltzmannstrasse 15, 85748 Garching
Phone: +49.89.289.15988, Fax: +49.89.289.15982

Abstract

The FliPASED project aims to design an actively-controlled wing with an aero-servo-structural optimization toolchain to achieve drag reduction, and test the new design on the T-FLEX demonstrator, which is a 7m wingspan unmanned aircraft with a V-Tail. Accurate and fast predictions of aerodynamic characteristics is vital in this preliminary design optimization process. This paper discusses the applicability of several low order aerodynamic tools to such aircraft while comparing them with a higher order CFD tool. The lifting-line based XFLR5 and PAWAT, the vortex-lattice based AVL, Tornado, PyTornado and VSPAERO, the 3D panel based FlightStream, and the commercial CFD tool STAR-CCM+ were used. The results include a convergence study, computational performance of each tool and comparison of calculated force and moment coefficients. The user experience of these tools are shared.

Keywords: Aerodynamics, UAV, T-FLEX, FLiPASED, FLEXOP

1. Introduction

As the climate impact of air traffic-related emissions gets more concerned, increasing the efficiency of aircraft and reducing the emission is of high interest. The EU-funded project Flight Phase Adaptive Aero-Servo-Elastic Aircraft Design Methods (FliPASED) aims to improve aerodynamic performance and cut down fuel consumption of aircraft by a concept for active wing shape adaptation [1]. Such concept can reduce the drag by maintaining optimal lift distribution during various flight conditions through actively deflecting control surfaces. Based on the concept, an active-controlled wing design can be achieved by a multidisciplinary design optimization (MDO) toolchain which incorporates aerodynamics, structural design, aeroelastic simulation and control design. The wing design will be tested on the T-FLEX demonstrator, which is a UAV developed in the previous EU project Flutter Free FLight Envelope eXpansion for ecOnomic Performance Improvement (FLEXOP) [2].
The aerodynamic solver is vital in the wing design process for accurate and fast estimation of the aerodynamic characteristics of the wing, especially drag. Computational fluid dynamics (CFD) tools can yield results with high fidelity, but are too computational expensive to be applied in such case, even with today's computation power. Therefore only low-order aerodynamic tools can make the design optimization computational feasible. There are plenty of low-order aerodynamics tools based on different theories, e.g. Lifting-line theory (LLT), Vortex lattice method (VLM) and panel method. An overview of some available tools and their theory can be found in the review from Technical University of Delft [3].
To find out the most suitable tool to be used in the active-controlled wing design, this study is conducted. The low order aerodynamic tools, Athena Vortex Lattice (AVL) [4], XFLR5 [5], PyTornado [6], Tornado, VSPAERO, PAWAT, FlightStream are compared regarding aspects such as prediction accuracy of aerodynamic characteristic, computational performance and applicability to the T-FLEX UAV . In the comparison, the data from the CFD tool STAR-CCM+ serve as the reference.

1.1 Description of the Demonstrator

The T-FLEX demonstrator is a 65 kg take-off weight, 7m wingspan unmanned aircraft with a swept wing and a V-Tail and is powered by a jet engine (Figure 1). A custom airfoil (internal name *try6*) in two versions- 10 percent thickness for the root and 8 percent for the wingtip- has been developed for the wing. A symmetrical 7.5 percent thickness airfoil is used for the tail surfaces.

Figure 1 – T-FLEX Subscale flight demonstrator during landing phase.

The geometry of the aircraft is summarized in table 1.

Table 1 – Geometry of T-FLEX UAV

Wing span, m:	7.0	Tail projected span, m:	1.27
Wing area, m^2:	2.478	Tail area, m^2:	0.39
Wing aspect ratio:	19.77	Tail aspect ratio:	4.2
Wing incidence, $\deg$:	-0.5	Tail incidence, $\deg$:	-4.33
Wing 0.25c sweep, $\deg$:	19.14	Tail 0.25c sweep, $\deg$:	19.83
Wing taper ratio:	0.5	Tail taper ratio:	0.52
Wing twist, $\deg$:	-2	Tail dihedral, $\deg$:	35
Number of wing control surfaces:	8	Number of tail control surfaces:	4
Fuselage length, m:	3.42		
Fuselage maximum height, m:	0.315		
Fuselage maximum width m:	0.3		

2. Aerodynamic Modelling

The increase in computational power has enabled complex numerical methods to be used for solving aerodynamic problems. Starting with Prandtl's Lifting Line approach up to the three-dimensional Direct Numerical Simulations, all methods can be solved on high-performance personal computers these days. However, out of the full range of methods some are more popular than others due to a good balance between their precision, complexity, reliability and computational cost. This group of methods are the new implementations of the Lifting Line Theory, Vortex or Doublet Lattice Method and 3D Panel Method. The mathematics of these methods will not be explained here, as this is not the goal of this study. However, descriptions of these methods are provided with references for an interested reader.

2.1 AVL

AVL is a program for performing aerodynamic analysis of rigid aircraft of arbitrary configurations [4]. It uses the VLM method to model the lifting surfaces. One advantage of AVL is the implementation of the slender body theory for fuselage modelling. Because of an intrinsic limitation of VLM, AVL is only suitable for inviscid calculation at small angles of attack and sideslip. Beside the global aerodynamic coefficient, flight stability characteristics can be acquired from eigenmode analysis in AVL.

2.2 Tornado

Tornado is a Vortex Lattice Method for linear aerodynamic wing design applications in conceptual aircraft design or in aeronautical education [7]. The method is built in MATLAB [8] and is based on the description as provided by Moran [9].
The VLM implementation in Tornado ignores the thickness effects of the airfoil, but includes the camber. In Tornado, modelling of control surfaces is possible. Experimental functions that generate a Trefftz-plane analysis can be used. Different options for mesh creation (linear panel distribution, cosine panel distribution, etc.) are available. A graphical interface is available which can plot coefficients of interest, display geometries and mesh.
Initially, Tornado was designed only to include linear aerodynamics. However, the code has been updated to include viscous effects as well [10].
If required, stability derivatives can be calculated using central-difference approximation around the trim condition. It is also possible to calculate trimmed polars.

2.3 PyTornado

PyTornado is an aerodynamic tool for conceptual aircraft design. Short computation times make it possible to easily obtain estimates of aerodynamic loads and to benchmark different concepts [6]. Although a similar name as Tornado, PyTornado has been implemented from scratch within the European research project AGILE. A Vortex Lattice Method is implemented in this code. It has a user interface, pre- and post-processing in Python and a calculation core routine in C++ [11], which guarantees a user friendly interface and computational efficiency. It can be used as a standalone aerodynamic solver or can be integrated into a MDO toolchain. The deformation feature, which is under development, could be potentially used for aeroelastic analysis.

2.4 XFLR5

XFLR5 is a software tool designed specifically with model sailplanes in mind [5][12]. Therefore, it focuses on wings operating at low Reynolds numbers. The tool uses XFoil [13] (XFoil v6.99 since XFLR5 v6.55) to calculate the 2D aerodynamics of an airfoil. Non-linear Lifting Line Theory (based on the NACA technical note 1269 [14]), Vortex Lattice Method with quadrilateral rings (as recommended by Katz and Plotkin [15]) or 3D Panel Method (based on Maskew [16]) can be used for 3D wing and tail analysis. Body analysis is not recommended by the author [12].
Unlike the usual VLM solvers, the VLM method implemented in XFLR5 provides a viscous drag correction. In such case, lift-related characteristics (lift distribution, induced drag) are kept inviscid and after local lift distribution is calculated, viscous drag correction using 2D airfoil polars is applied. The lift distribution is not changed. This method is also used during this study. However, the author of the software raises awareness that such correction is not scientifically sound, as using 2D polars ignores any spanwise effects [12].

2.5 VSPAERO

VSPAERO [17] is the aerodynamic analysis tool integrated within the conceptual aircraft design package OpenVSP [18]. The tool has two methods available - the Vortex Lattice Method with a simple stall prediction methodology (not used in this study) and a 3D Panel method [19]. Propellers can be included in the simulation. The tool also incorporates the possibility to calculate the parasite drag using the component build-up method. In the current study, only the VLM method is used.

2.6 PAWAT

The Preliminary Design Tool for Propeller-Wing Aerodynamics (PAWAT) is an aerodynamic tool for the conceptual design of aircraft [20]. The calculation of the steady state lifting surface aerodynamics in PAWAT is based on a modified three-dimensional nonlinear lifting line theory with a fixed wake model employing nonlinear airfoil data to model nonlinear and viscous effects to a certain extent [20]. PAWAT is also capable of modelling propellers and it allows investigations of the interaction effects between wing and propeller.
The method is built in MATLAB [8]. The description of the lifting line method used is described by Phillips and Snyder [21].

2.7 FlightStream

FlightStream is a novel surface vorticity solver capable of using structured or unstructured surface meshes. As a vorticity-based solver, the code can be expected to be substantially more robust and stable compared to pressure-based potential-flow solvers and less sensitive to surface perturbations, and it also allows the use of coarser meshes with an acceptable level of fidelity [22].
To account for viscous effect, integral boundary layer was implemented in FlightStream and was coupled with inviscid solver via displacement of the inviscid boundary equal to the displacement thicknesses of the local boundary layers. More features like prediction of flow separation and stall characteristics are also enabled by this implementation.

2.8 STAR-CCM+

Simcenter STAR-CCM+ is a multiphysics computational fluid dynamics (CFD) software. In this study, it is used to provide the reference data for comparison.
For these simulations, the geometry was prepared within the CAD software SolidWorks. The T-FLEX geometry was imported and the bullet-shaped domain was created (15 spans in radius and 45 spans in length, Figure 2). The trailing edges of the lifting surfaces were trimmed to total thickness of 0.2 percent of the reference chord length of the lifting surface. The geometry was then transferred to STAR-CCM+. In-order to reduce the computational effort, symmetry condition was used.
A polyhedral mesher was used. For turbulent simulations, a prism layer mesher was applied. Custom mesh controls were set on the different boundaries (domain, fuselage, wing and tail) as well as on the trailing and leading edges. Wake controls (2 spans in length and 15° in spread angle) were used on the fuselage, wing and tail (Figure 3).
Steady inviscid and steady fully turbulent simulations were done. In the latter, Reynolds-averaged Navier-Stokes (RANS) equations were solved. Spalart-Allmaras turbulence model was used together with all y+ treatment. Near wall thickness of the prism layer was therefore adjusted for y+ = 1 condition. Total thickness of the prism layer was adapted to fit the complete boundary layer at AoA = 2° (Figure 4).

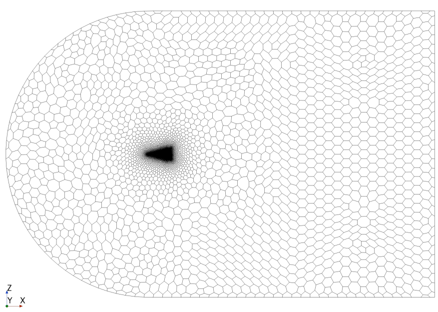


Figure 2 – The domain used in STAR-CCM+ simulations.

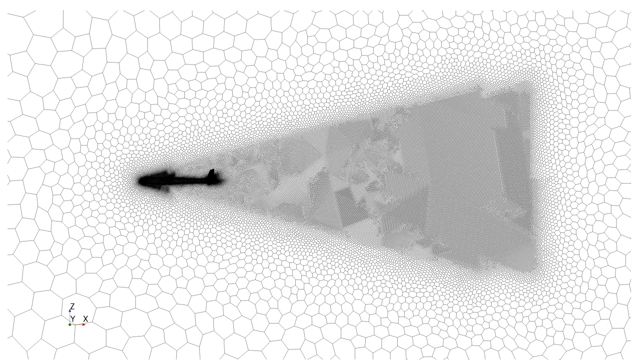


Figure 3 – The wake control applied on the geometry.

One has to emphasize that most of the tools simulated the wing and tail. Fuselage was included only in simulations of STAR-CCM+ and FlightStream. A study was done with STAR-CCM+ to investigate

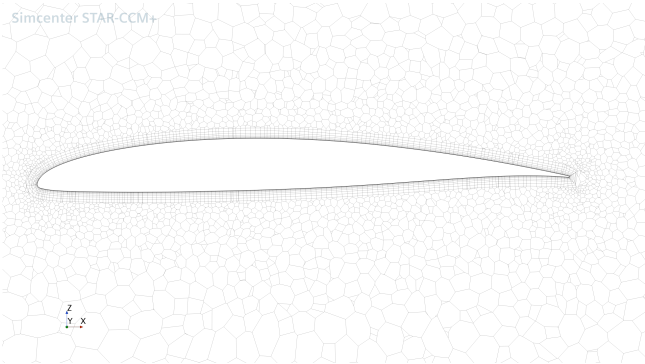


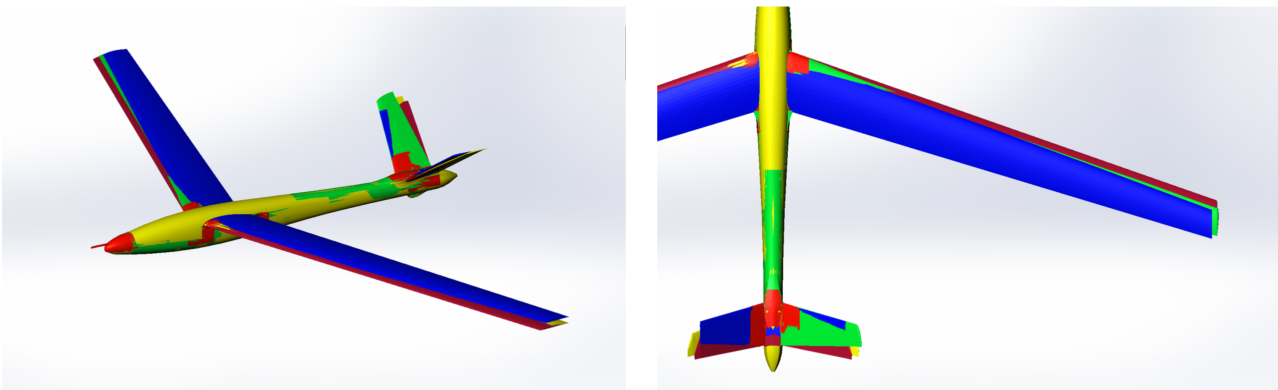
Figure 4 – Prism layer mesh around the wing cross section.

the influence of the fuselage on the spanwise lift distribution. A small influence was noted at low angles of attack. However, at high angles of attack the fuselage does change the flow at the wing root.

3. Practical Recommendations

3.1 Geometry modelling

Special care must be taken when defining the aircraft geometry within different aerodynamic or structural modelling tools. As such tools are usually used during the preliminary design stage, where no CAD models are available, the aircraft geometry must be defined via geometry parameters. From the tools tested during this study, only FlightStream supports direct import of CAD models.
In the current case, a collection of parameters describing the overall geometry were available from early stages of the project. These were initially used for all the aerodynamic modelling software. However, it became clear soon, that the parameters not only differed from the actual build geometry (due to modelling and manufacturing errors), but also the different software tools interpreted these parameters differently.
As a result the geometry modelled in each of the tools was exported in a .stl format. Then the actual build geometry was 3D scanned and all the geometries were compared within a CAD software. The initial missalignment of the geometries within different tools are shown in the Figure 5.



(a) Isotropic view. (b) Top view.

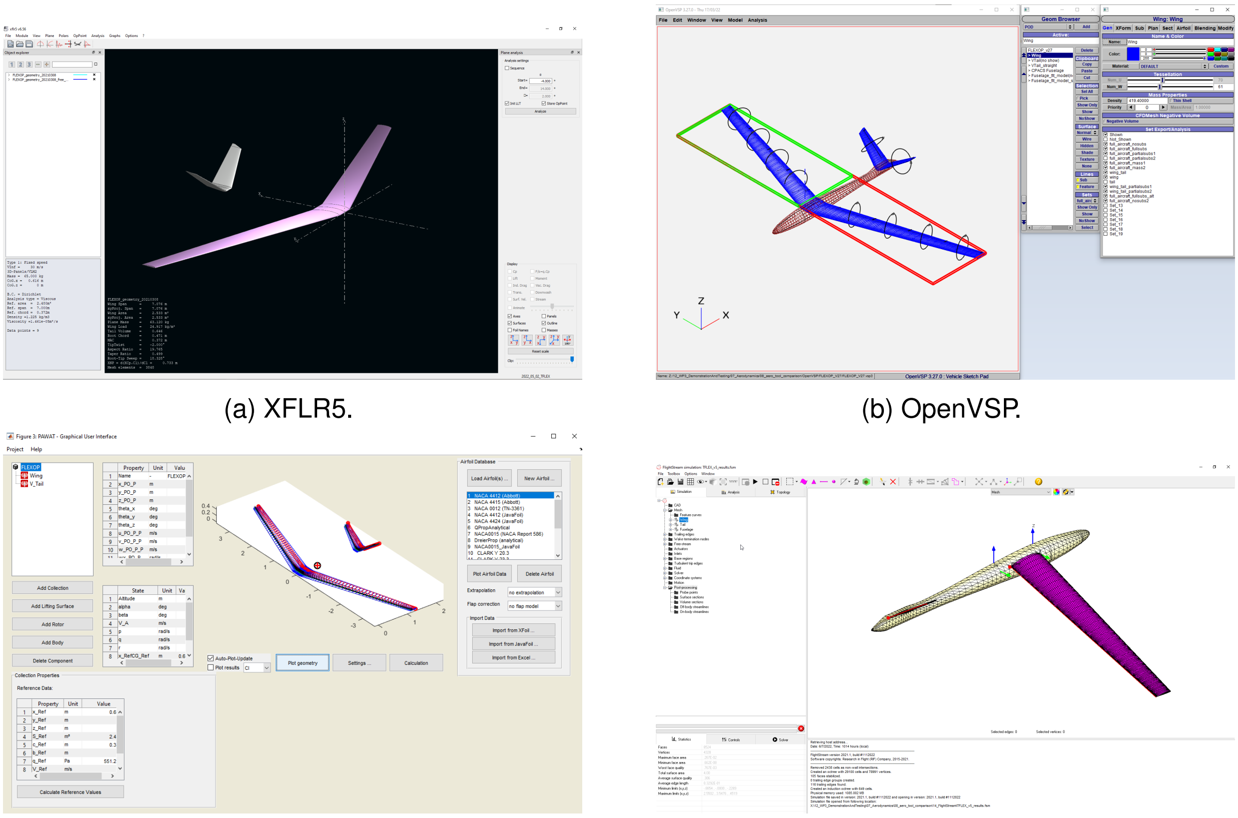
Figure 5 – The visual comparison of the geometries modelled in different tools. Bright red - 3D scanned geometry; green - the CAD geometry after aligning it with the scan; yellow - geometry as initially defined in the project; blue - XFLR5; dark red - VSPAERO.

After this was done, it was ensured that the geometries modelled in the different software tools were identical in the end.
This problem can be particularly troublesome for the tools, which do not document in detail the interpretation of the geometry parameters. Also, the use of V-Tails introduced additional geometrical complexity. In such cases, a clear understanding of the definitions like the incidence of the tail

(incidence around the aircraft y-axis or around the quarter-chord line of the V-Tail surface?) or span (projected on the horizontal plane or along the surface?) is a must.

3.2 User interface

The user interface is an important criterion for evaluating software tools. The intuitive and user-friendly interface can lower the barrier to entry for new users. This section briefly summarises the user experience of the different tools in this regard.
Out of the 7 tools that are presented here, only AVL, Tornado and PyTornado do not have a graphical user interface. For these tools, menu options are chosen from the terminal and command line inputs must be done. Only in the results processing step (plotting) graphical interaction is allowed.
The remaining four softwares all have a complete graphical user interface (Figure 6). All of them are self-explanatory and comparably easy to use.
The results, generated by the solvers, can be accessed interactively. It should be noted that while all the generated data (each simulation data point) is accessible in XFLR5 via the graphs menu, data access with the other three solvers can be somewhat more complicated, if specific graphs are required. For example, in XFLR5 all previous simulations can be accessed for the same project all the time, but this is not possible in the viewer of VSPAERO, where only the results from a single simulation can be viewed at once. Similarly, PAWAT also only displays results of a single data point.



(a) XFLR5.

(b) OpenVSP.

(c) PAWAT.

(d) FlightStream.

Figure 6 – Graphical user interfaces of the compared tools.

The results from the different tools can be exported in various format - tool specific and generic. But all the investigated tools support export in text format, which enables users to conduct postprocessing in any preferred external processing routine. The content of exported data varies among tools but all the tools can export basic aerodynamic information easily.

3.3 Automation integration possibilities

As an intrinsic requirement from the MDO toolchain, the aerodynamic solver has to be executed automatically in batch mode without GUI.

AVL and PyTornado only provide a terminal interface which is not intuitive to use compared to other tools, but it makes it easily scripted and thus integrated into a toolchain. As Matlab can be ran in a no GUI mode, Matlab based TORNADO and PAWAT could also be good options for a MDO toolchain. VSPAERO can also be run from the command line. XFLR5 supports scripting in the GUI mode to run a batch analysis.

4. Results

In this section, results from all 7 tools are compared against the inviscid and viscous CFD data. Lift, drag, moment and lift distribution results are discussed separately.

4.1 Mesh convergence and computational performance

To make sure that the computational results are independent of the mesh, convergence studies have been performed. The number of panels have been increased in chordwise and spanwise direction separately until the difference in resulting lift coefficient was less than 1 percent of the absolute value. The comparison for all the VLM-based tools can be found in Figure 7. Solver times for each mesh study was also noted, the comparison can be seen in Figure 8.

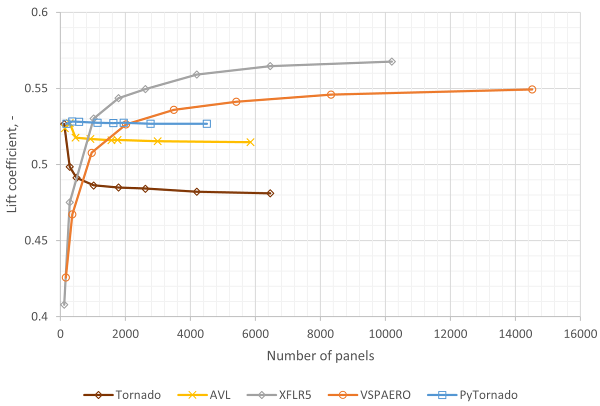


Figure 7 – Mesh convergence study for the VLM-based tools.

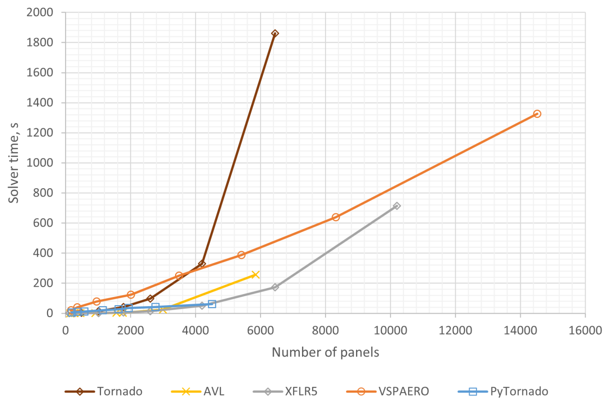


Figure 8 – Computational time comparison for the VLM-based tools.

While XFLR5 and VSPAERO require the highest number panels for convergence, it was Tornado that took the longest to compute. On the contrary, PyTornado required the lowest number of panels for convergence while also being the most time efficient tool.

PAWAT, being a lifting line method tool, requires two orders of magnitude fewer panels to reach convergence criteria and therefore is not displayed in the plots. It takes up to 3 seconds to perform calculation with any mesh.

For FlightStream, the mesh is generated following the official guideline. The mesh used in this study has 29100 cells and 78991 vertices, which is much more than the panels used in the VLM-based method. The calculation time of the simulation with viscous coupling is around 1 minute.

4.2 Mesh convergence for STAR-CCM+

STAR-CCM+ being a CFD tool requires significantly more resources for computations. Therefore, these simulations were performed on the Leibniz-Rechenzentrum Linux Cluster CoolMUC-2 [23]. Two nodes with 28 cores each were used.

For the mesh independence studies, the number of cells were gradually increased from 0.9 to 22.0 million and global lift, drag and moment coefficients at AoA = 2° were compared (Figure 9 and Table 2). In the end, a grid with 12.3 million cells was chosen for the study.

The prism layers were removed for inviscid simulations while keeping the rest of the mesh parameters the same. This resulted in a mesh size of 5.9 million cells.

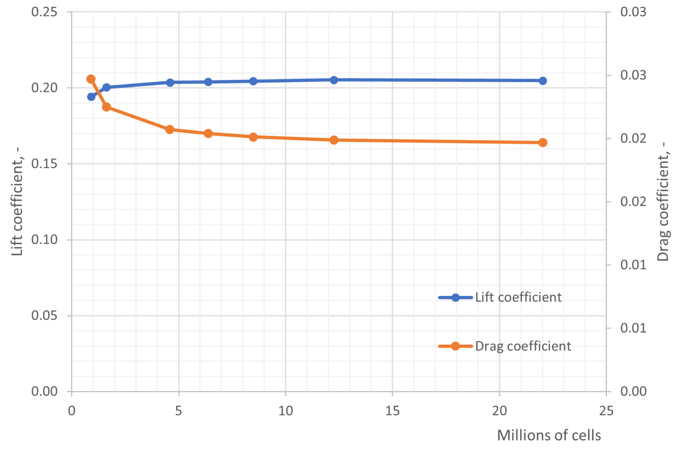


Figure 9 – Investigation of the mesh independence performed with the STAR-CCM+ software.

Table 2 – Comparison of lift, drag and moment coefficients during the mesh independence study.

N_{cells}	C_D	ΔC_D, %	C_L	ΔC_L, %	C_m	ΔC_m, %	t_{solver}, s
0.9M	0.0247		0.1942		0.1400		139
1.6M	0.0225	-9.8	0.2004	3.1	0.1388	-0.9	237
4.6M	0.0207	-8.6	0.2037	1.6	0.1346	-3.1	705
6.4M	0.0204	-1.6	0.2040	0.1	0.1332	-1.0	958
8.5M	0.0201	-1.3	0.2045	0.3	0.1330	-0.2	1403
12.3M	0.0199	-1.3	0.2053	0.4	0.1324	-0.4	2008
22.0M	0.0197	-1.0	0.2049	-0.2	0.1319	-0.4	4284

4.3 Global aerodynamic coefficients

4.3.1 Lift

The lift coefficient data is plotted with respect to the angle of attack in Figure 10 as well as in Figure 11 for the linear part of the slope. The lift curve slope coefficients C_{L_α} and zero angle lift coefficients C_{L_0} are shown in Table 3.

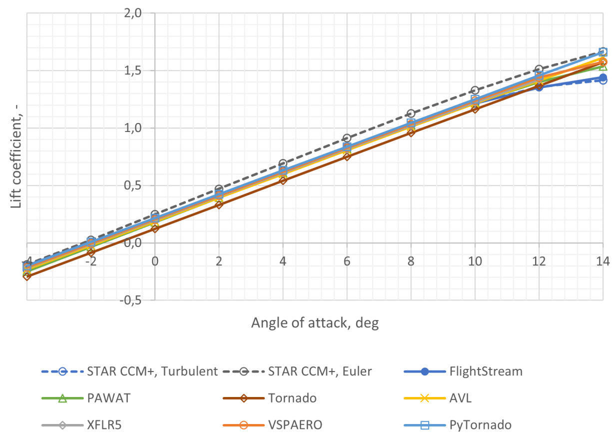


Figure 10 – Lift coefficient C_L with respect to the angle of attack α.

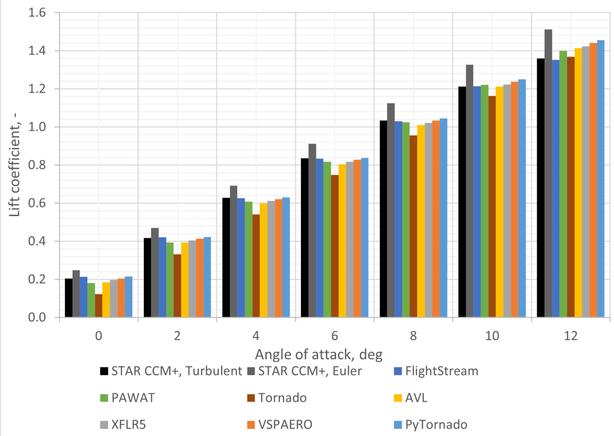


Figure 11 – Lift coefficient C_L with respect to the angle of attack α.

Significant reduction in lift is apparent when comparing the turbulent simulations to Euler simulations. This is expected, as the viscous boundary layer on the top surface of the wing reduces the effective camber line, therefore reducing the aerodynamic angle of attack. Interestingly, most of the tools show better alignment with the turbulent simulations than with the inviscid ones, even though only PAWAT and FlightStream take viscosity into account when calculating lift.

Table 3 – Comparison of lift curve slope C_{L_α}, zero angle lift coefficient C_{L_0}, minimum drag coefficient $C_{D_{min}}$, pitching moment curve slope C_{m_α} and zero angle pitching moment coefficient C_{m_0} for different aerodynamic modelling tools.

	Turbulent	**Euler**	**FlightStream**	**PAWAT**	**Tornado**	**AVL**	**XFLR5**	**VSPAERO**	**PyTornado**
C_{L_α}	0.106	0.111	0.103	0.107	0.105	0.104	0.104	0.104	0.104
C_{L_0}	0.206	0.248	0.214	0.180	0.122	0.185	0.198	0.205	0.215
$C_{D_{min}}$	0.020	0.005	0.015	0.016	0.001	0.002	0.015	0.012	0.002
C_{m_α}	-0.027	-0.028	-0.030	-0.047	-0.050	-0.032	-0.032	-0.026	-0.030
C_{m_0}	0.132	0.141	0.117	0.103	0.193	0.214	0.147	0.128	0.159

When only the linear part of the lift curve is concerned, the calculated curve slope agreed with each other. The zero angle of attack lift shows deviations among tools. Tornado differs most from the other tools. Taking into account that all the tools are meant for preliminary design phase, the differences between them could be categorised as being insignificant.

The nonlinear part of the curve is predicted by both PAWAT and FlightStream. Even at high angle of attack, the lift curve from FlightStream matches quite well with CFD turbulent result. However, as no CFD simulations above 14° were done, the $C_{L_{max}}$ could not be estimated.

The spanwise normalized lift distribution for $\alpha = 2^\circ$ is plotted in Figure 12. As only the shape of the distribution is of importance here, the local lift coefficients are normalized with respect to the maximum local lift coefficient for the same tool.

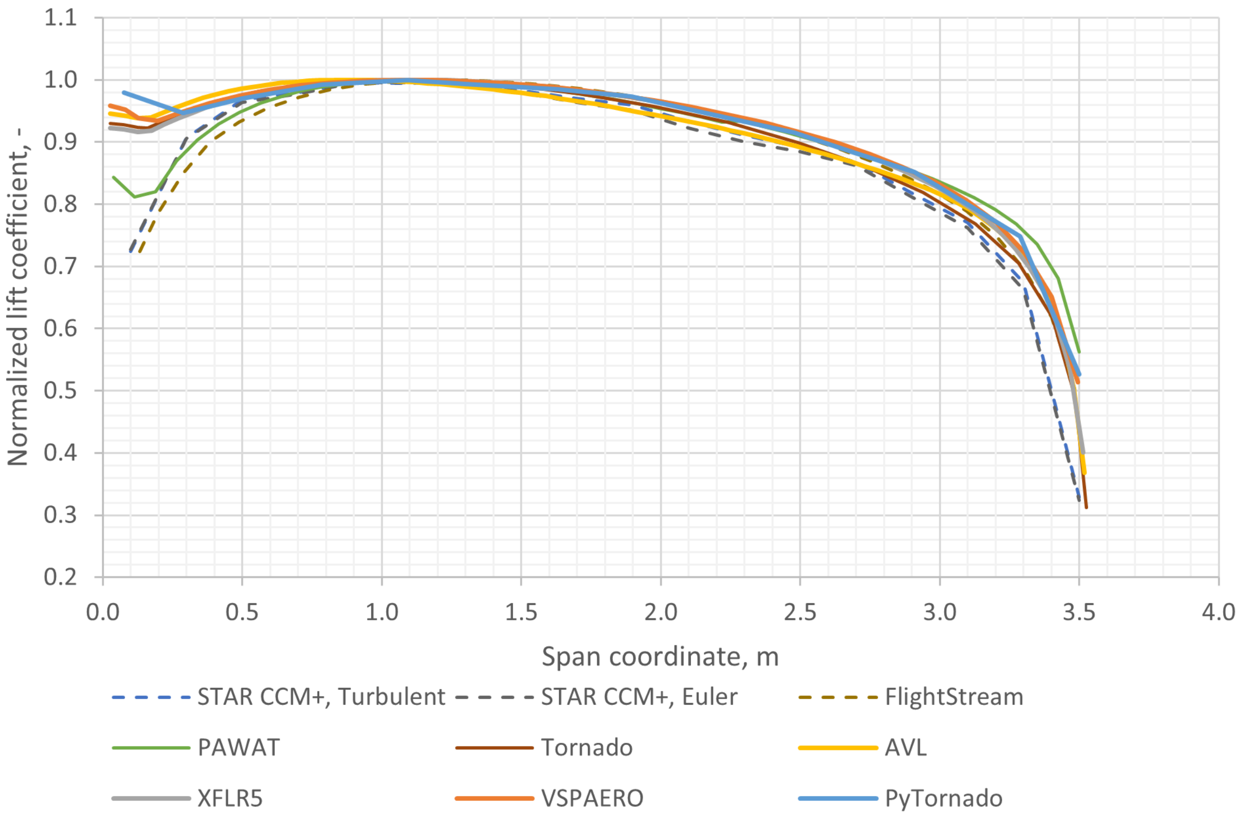


Figure 12 – Spanwise normalized lift distribution for $\alpha = 2^\circ$. The local lift coefficients are normalized with respect to the maximum local lift coefficient of the same tool.

The normalized lift distributions between the turbulent and Euler simulations are almost identical. The estimated maximum local lift location is similar for all the tools. The overall shape is very similar with some discrepancies at the root and tip areas. The differences between the STAR-CCM+ results and the other tool results at the wingtip might be due to the poor discretization when extracting the lift distribution from STAR-CCM+.

4.3.2 Drag

Figure 13 shows the inviscid drag polar. While all the VLM tools and the panel-based method Flight-Stream agree mainly, the differences compared with the STAR-CCM+ Euler simulation are noticeable even at low lift coefficient.

One has to note that the inviscid drag extracted from STAR-CCM+ here is the pressure drag component acting on the aircraft. Strictly speaking, this is not equal to the induced drag by definition. The separation of induced and profile drag from CFD is not straight-forward, and if Euler simulations are used the induced drag due to viscous effects are then ignored. Nowadays there exist some methods to extract these two drag components from CFD [24], but they were not implemented at the time of writing this article.

The total drag coefficient shown in Figure 14 includes both viscous and inviscid drag. Significant differences can be seen in between the tools that correct for viscous drag (STAR-CCM+ (turbulent), FlightStream, PAWAT, XFLR5, VSPAERO) and the ones that do not (Tornado, AVL, PyTornado).

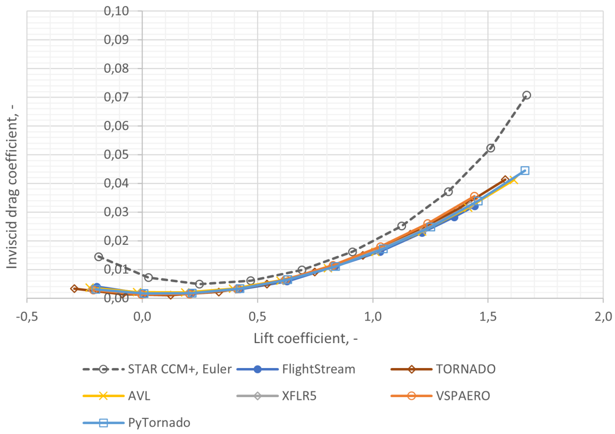


Figure 13 – Inviscid drag coefficient C_{D_i} with respect to the angle of attack α.

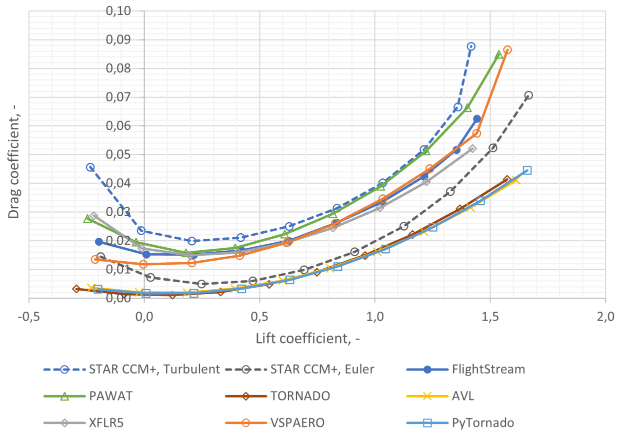


Figure 14 – Total drag coefficient C_D with respect to the angle of attack α.

Different methods were used to correct the viscous drag in different software tools. Variation of viscous drag is clearly visible in the Figure 14.

Both PAWAT and XFLR5 correct viscous drag based on 2D airfoil polar data. For XFLR5, 2d viscous drag is interpolated from local wing lift coefficient. The interactive boundary layer, which is a coupling method between potential flow and viscous flow on surfaces, is not implemented in the VLM available in XFLR5 [5]. The consequence of underestimation of viscous drag is confirmed in the Figure 14.

In PAWAT, equations are established for wing segments based on the aerodynamic force derived from three-dimensional vortex lifting law and the aerodynamic force derived from nonlinear airfoil characteristics of the segment and the segment area [20]. An iterative procedure is needed to solve the equations. Total drag coefficient from PAWAT matches quite well with CFD data.

In FlightStream, the integral boundary layer is coupled with the inviscid surface solver to account for viscous drag. Even though, the total drag seems to be underestimated.

4.3.3 Pitching moment

The pitching moment coefficient with respect to angle of attack is shown in Figure 15.

As already shown in Table 3, there is a good (differences up to 10 percent) agreement of the pitching moment slopes of the linear part in between the STAR-CCM+, FlightStream, VSPAERO and PyTornado. The zero angle pitching moment coefficient is predicted well by both FlightStream and

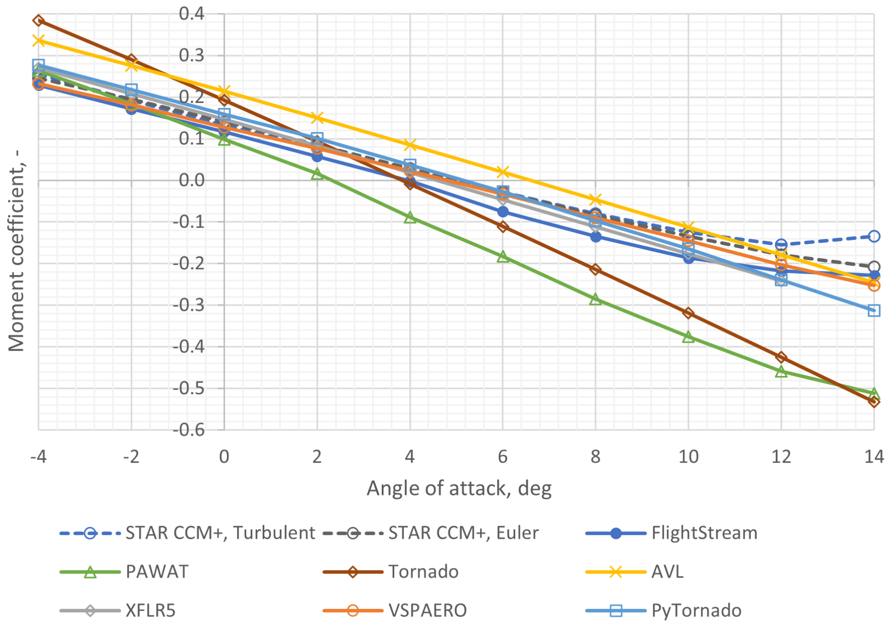


Figure 15 – Pitching coefficient C_m with respect to the angle of attack α .

VSPAERO, with the latter having even smaller deviation from the STAR-CCM+ results. This is unexpected, since the fuselage, which should have an influence on the pitching moment, is not modelled in the OpenVSP.
A pitch-up trend can be noted at high angles of attack from the STAR-CCM+ simulations. This is due to the early stall at the tip section of the wing (Figure 16). Only FlightStream captures the effect of the pitch-up, even though not as pronounced as with the STAR-CCM+.

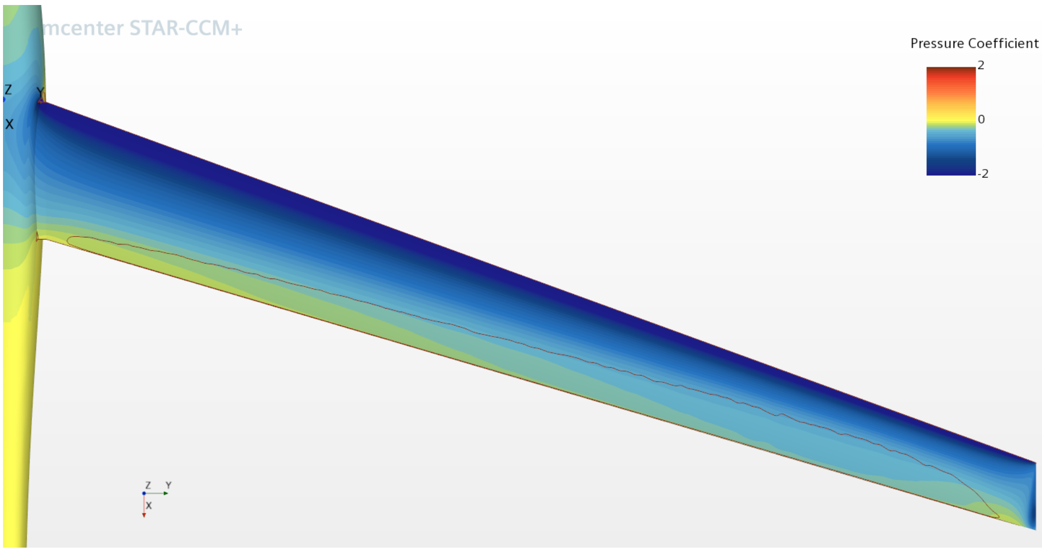


Figure 16 – Pressure coefficient distribution at $\alpha = 14°$. The line marks the location where the wall shear stress changes direction.

It should be noted that while PAWAT provided good calculations for lift and drag, the results for pitching moment are not satisfactory.

5. Conclusions

The selection of aerodynamic tools is highly dependent on the purpose and the aircraft configuration. Based on the user experience and simulation results of T-FLEX UAV, following recommendations are given.

- If a fast polar calculation is of main interest, VSPAERO(OpenVSP), XFLR5 and PAWAT could be good options. All of them have an intuitive user interface for modelling, calculation and postprocessing. The simulations run relatively quickly.
- If the boundary layer and nonlinear effects are of concern, FlightStream provides the best results due to the coupled integral boundary layer solver.
- Pitching moment coefficient curves, as calculated by VSPAERO (linear part) and FlightStream (nonlinear part as well), match very well with the STAR-CCM+ results.
- If an aerodynamic tool is needed for a MDO task, AVL, PyTornado and VSPAERO(OpenVSP) could be good choices. The input file for these tools can be easily automatically prepared. The execution of these tools can be invoked automatically and fast.

The aforementioned recommendations should be applied with caution to similar configurations as T-FLEX which is an UAV equipped with a swept, high aspect ratio wing and a V-tail.

6. Acknowledgements

The work presented has been conducted within the framework of project FliPASED (grant agreement No. 815058) funded from the European Union's Horizon 2020 research and innovation program.

7. Contact Author Email Address

To contact the author, please use the following email address: fanglin.yu@tum.de

8. Copyright Statement

References

[1] FLIPASED Consortium. *FLIGHT PHASE ADAPTIVE AERO-SERVO-ELASTIC AIRCRAFT DESIGN METHODS | FLiPASED Project | H2020 | CORDIS | European Commission*. URL: https://cordis.europa.eu/project/id/815058 (visited on 05/13/2021).

[2] FLEXOP Consortium. *FLEXOP Project Homepage*. URL: https://flexop.eu/news (visited on 10/24/2018).

[3] B Peerlings. *A review of aerodynamic flow models, solution methods and solvers-and their applicability to aircraft conceptual design Literature study report*. 2018.

[4] Mark Drela. *AVL*. URL: http://web.mit.edu/drela/Public/web/avl/ (visited on 05/05/2021).

[5] Andre Deperrois. *XFLR5*. URL: http://www.xflr5.com/ (visited on 05/05/2021).

[6] Alessandro Gastaldi and Aaron Dettmann. *PyTornado*. URL: https://pytornado.readthedocs.io/en/latest/index.html (visited on 05/05/2021).

[7] Tomas Melin. "A Vortex Lattice MATLAB Implementation for Linear Aerodynamic Wing Applications". Royal Institute of Technology (KTH), Dec. 2000. URL: https://www.researchgate.net/publication/238671899.

[8] *MATLAB - MathWorks - MATLAB & Simulink*. URL: https://uk.mathworks.com/products/matlab.html (visited on 05/16/2021).

[9] J. Moran. *An Introduction to Theoretical and Computational Aerodynamics*. Dover Books on Aeronautical Engineering. Dover Publications, 2003. ISBN: 9780486428796. URL: https://books.google.de/books?id=4eVP3yWZ1LgC.

[10] A. Da Ronch et al. “Benchmarking ceasiom software to predict flight control and flying qualities of the B-747”. In: *27th Congress of the International Council of the Aeronautical Sciences 2010, ICAS 2010* 4.September (2010), pp. 2906–2912.

[11] Tomas Melin. *Implementation Of A Vortex Lattice Method In A Heterogeneous Programming Language Environment*. 2018. URL: https://www.researchgate.net/publication/327733453.

[12] André Deperrois. *Guidelines for XFLR5: Analysis of foils and wings operating at low Reynolds numbers*. Tech. rep. February. 2013, pp. 1–72. URL: http://scholar.google.com/scholar?hl=en{\&}btnG=Search{\&}q=intitle:Analysis+of+foils+and+wings+operating+at+low+Reynolds+numbers{\#}0.

[13] Mark Drela. *XFOIL*. URL: http://web.mit.edu/drela/Public/web/xfoil/ (visited on 05/05/2021).

[14] James C Sivells and Robert H Neelly. “Method for calculating wing characteristics by lifting-line theory using nonlinear section lift data”. In: *National Advisory Commitee for Aeronautics* 865 (1947), pp. 75–93.

[15] Joseph Katz and Allen Plotkin. *Low-Speed Aerodynamics*. 2nd. New York: Cambridge University Press, 2001. ISBN: 978-0-521-66219-2.

[16] Brian Maskew. “Program VSAERO theory document”. In: *Nasa Cr-4023* (1987). URL: http://scholar.google.com/scholar?hl=en{\&}btnG=Search{\&}q=intitle:Program+VSAERO+Theory+Document{\#}0.

[17] David Kinney. *Using VSPAERO*. URL: http://openvsp.org/wiki/doku.php?id=vspaerotutorial (visited on 06/03/2022).

[18] OpenVSP. *Software Package, Version 3.22.0*. 2021. URL: http://openvsp.org/ (visited on 05/05/2021).

[19] David Kinney. *VSPAERO... What’s New?* 2021.

[20] H.-J. Steiner. *Preliminary Design Tool for Propeller-Wing Aerodynamics Part II: Theory*. Bauhaus Luftfahrt., 2010.

[21] W F Phillips and D O Snyder. “Modern Adaptation of Prandtl’s Classic Lifting-Line Theory”. In: *JOURNAL OF AIRCRAFT* 37.4 (2000). DOI: 10.2514/2.2649. URL: http://arc.aiaa.org.

[22] Erik D. Olson and Cindy W. Albertson. “Aircraft high-lift aerodynamic analysis using a surface-vorticity solver”. In: vol. 0. American Institute of Aeronautics and Astronautics Inc, AIAA, 2016. ISBN: 9781624103933. DOI: 10.2514/6.2016-0779.

[23] *CoolMUC-2 - Leibniz-Rechenzentrum (LRZ) Dokumentation*. URL: https://doku.lrz.de/display/PUBLIC/CoolMUC-2 (visited on 05/31/2022).

[24] George Loubimov and Michael Kinzel. “A novel approach to calculating induced drag from computational fluid dynamics”. In: *AIP Advances* 11.7 (2021). ISSN: 21583226. DOI: 10.1063/5.0051080.