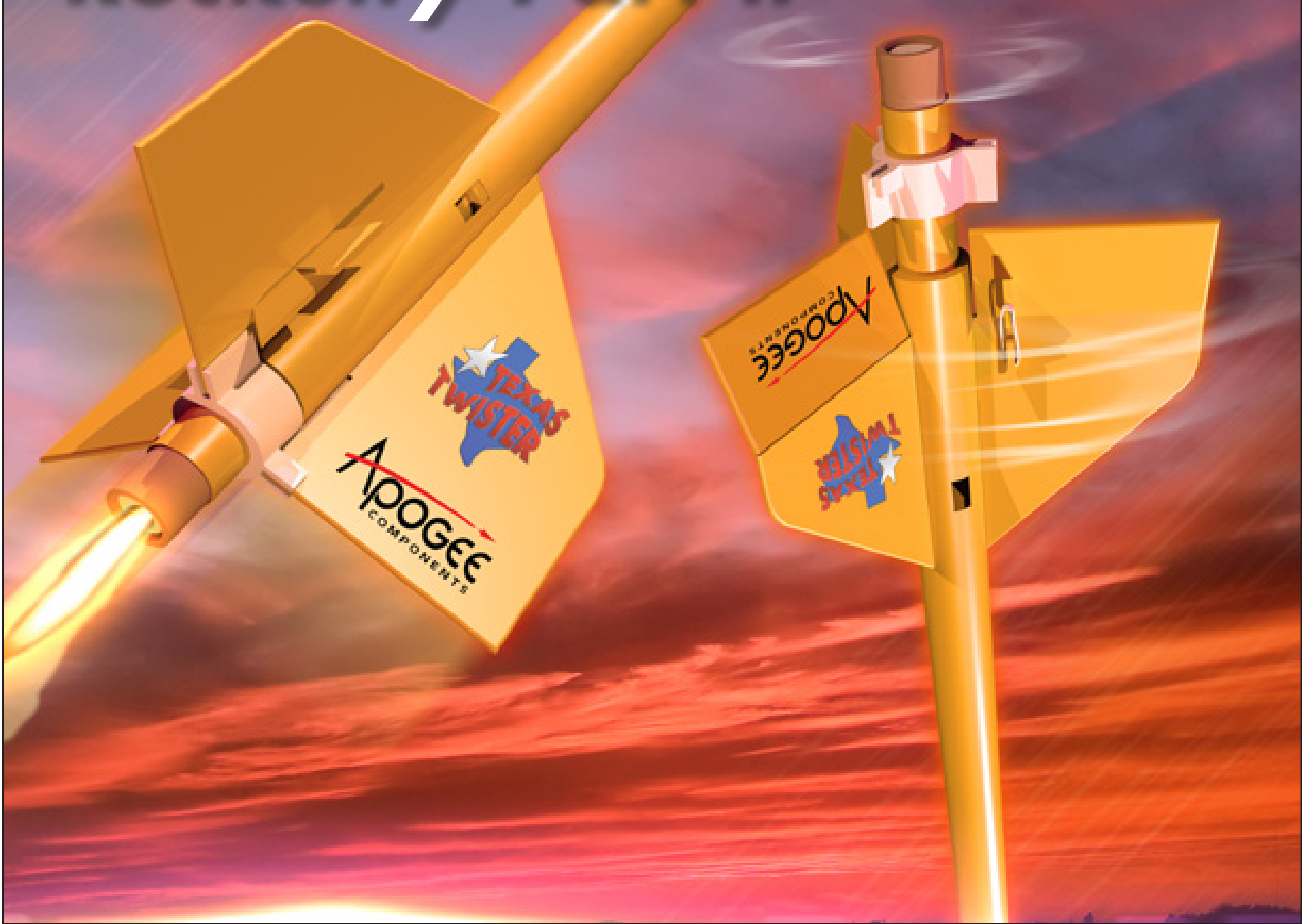
Issue 642 / December 31st, 2024

PEAK OF FLIGHT NEWSLETTER

Apogee Components, Inc. / ApogeeRockets.com / Colorado Springs, CO

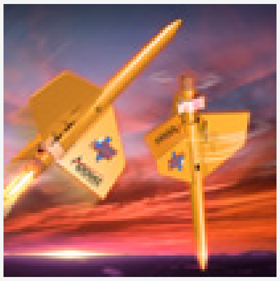
Computational Fluid Dynamics for Model Rocketry Part II



PEAK OF FLIGHT NEWSLETTER

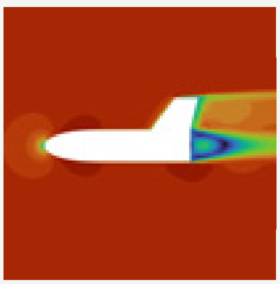
Issue 642 / December 31st, 2024

COVER PHOTO



Apogee Texas Twister Model Rocket

FEATURED ARTICLES



Computational Fluid Dynamics for Model Rocketry Part II

by Ken Karbon

A Continuation from Part I on Computational Fluid Dynamics (CFD) in Model Rocketry

Apogee Components, Inc.
4960 Northpark Dr.
Colorado Springs, CO 80918
1-719-535-9335
www.ApogeeRockets.com

About this Newsletter

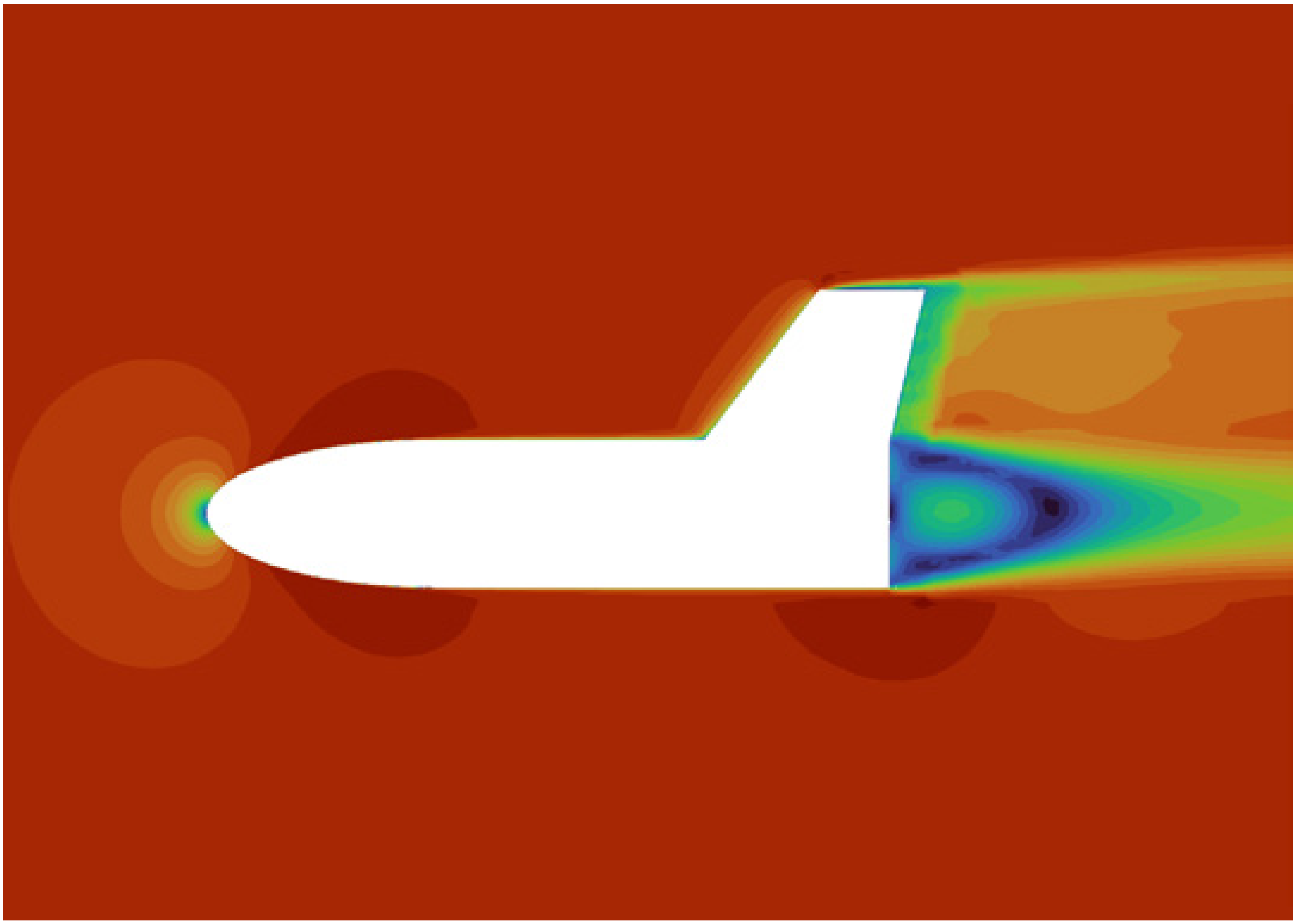
You can subscribe to recieve this e-zine FREE at the Apogee Components website: www.ApogeeComponents.com, or by clicking the link here **Newsletter Sign-Up**

Editor-in-Chief: Tim Van Milligan
Managing Editor: Michelle Mason
Content Editor: Martin Jay McKee
Layout Design: Ogden Sikel

Apogee Rockets Quick Draw Launch, Pueblo, CO

Would you like to see your launch photo featured in the *Peak-of-Flight* newsletter? Submit your photo at apogeerockets.com.

PEAK OF FLIGHT NEWSLETTER
Computational Fluid Dynamics for Model Rocketry Part II
Issue 642 / December 31st, 2024
www.ApogeeRockets.com



Computational Fluid Dynamics for Model Rocketry — Part II

by Ken Karbon

Introduction

In Part I of this article, I gave some background on Computational Fluid Dynamics (CFD) and how it has become more feasible for hobby rocketry. I also introduced the typical workflow for building and simulating CFD models (**Figure 1**), including the steps of geometry, selecting the appropriate physics, and defining boundary conditions. Now, I will continue with the remaining steps, starting with the setup for "meshing."

PEAK OF FLIGHT NEWSLETTER
Issue 642 / December 31st, 2024
www.ApogeeRockets.com

Computational Fluid Dynamics for Model Rocketry Part II

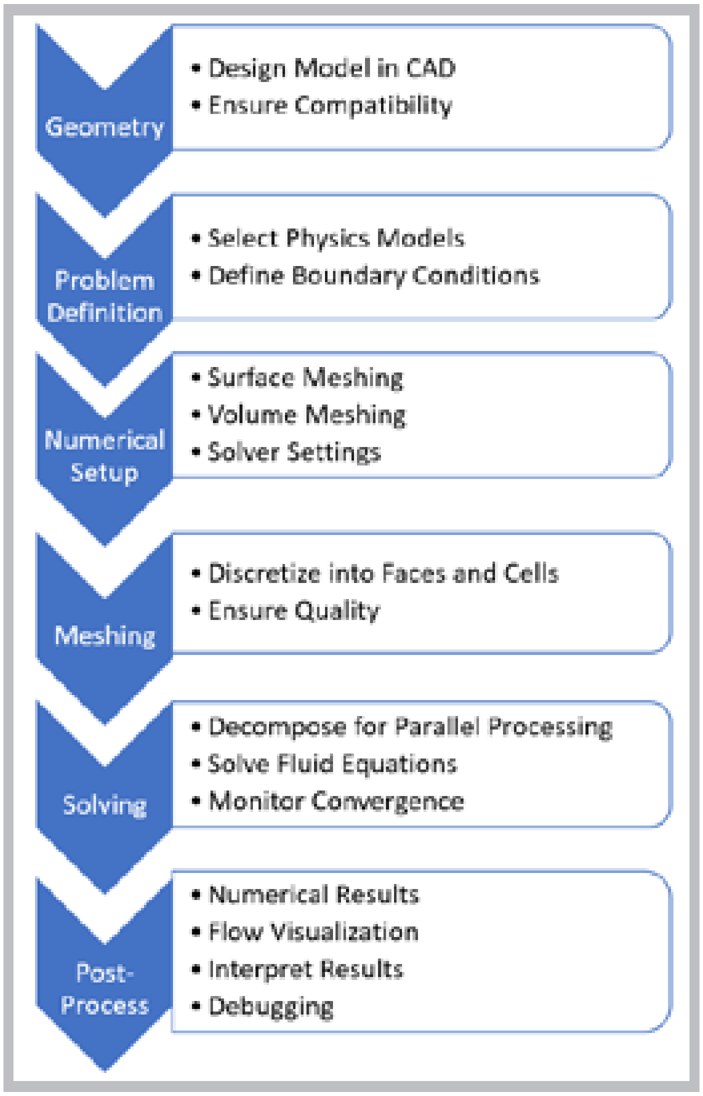


Figure 1. CFD Workflow Diagram

Meshing Overview

The meshing process consists of discretizing the rocket CAD surfaces into faces, and the surrounding fluid volume into cells. The entire assembly enclosed in the computational domain defined previously is the mesh. See **Figure 2**. Generating a mesh involves solving partial differential equations, just like those governing fluid flow or heat transfer problems. It is a subset of Computer Aided Engineering (CAE), and CAE software are developed solely for meshing. Even though this process is highly automated, the user must have a good eye for the rocket geometry and envisioning the 3D space around it. Nearly all CFD failures and questionable results are due to a poor mesh.

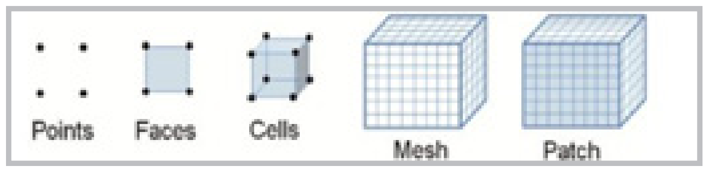


Figure 2. Mesh Entities

There are several numerical schemes to solve the Navier-Stokes equations across the mesh. The Finite Volume Method (FVM) is the most popular. Each cell can be considered a control volume, and conservation of mass, momentum, and energy is preserved for each one in the mesh. Cells can be any shape, including hexahedrons (cubes), tetrahedrons, pyramids, prisms, or polyhedral. Hexahedrons are considered the most accurate type for a given number of cells.

What is the best mesh size? There is no correct answer and no single approach to meshing. CFD engineers rely on experience, tribal knowledge within their organization, computer limitations, or correlation to physical tests. There are some systematic approaches, such as repeatedly reducing the mesh size until the solution stops changing (within a tolerance). At this point, the model is deemed "grid independent."

Hex-dominant, cartesian, meshing algorithms that are work with OpenFOAM will first start with a coarse background or “base” mesh, then begin to progressively refine it around the rocket surface and additional regions of interest. The base mesh is considered to be “Level 0” and is usually the element edge length that is used on the boundaries of the computational domain. Also, Subsequent refinement levels are defined as a percentage of the base mesh size or as a geometric progression, with each Nth level reducing in size by a factor of 2N. **Figure 3** shows an example of this halving mesh.

Level	Edge Length, mm
0	100.00
1	50.00
2	25.00
3	12.50
4	6.25
5	3.13
6	1.56
7	0.78
N	$100/2^N$

Figure 3. Mesh Levels

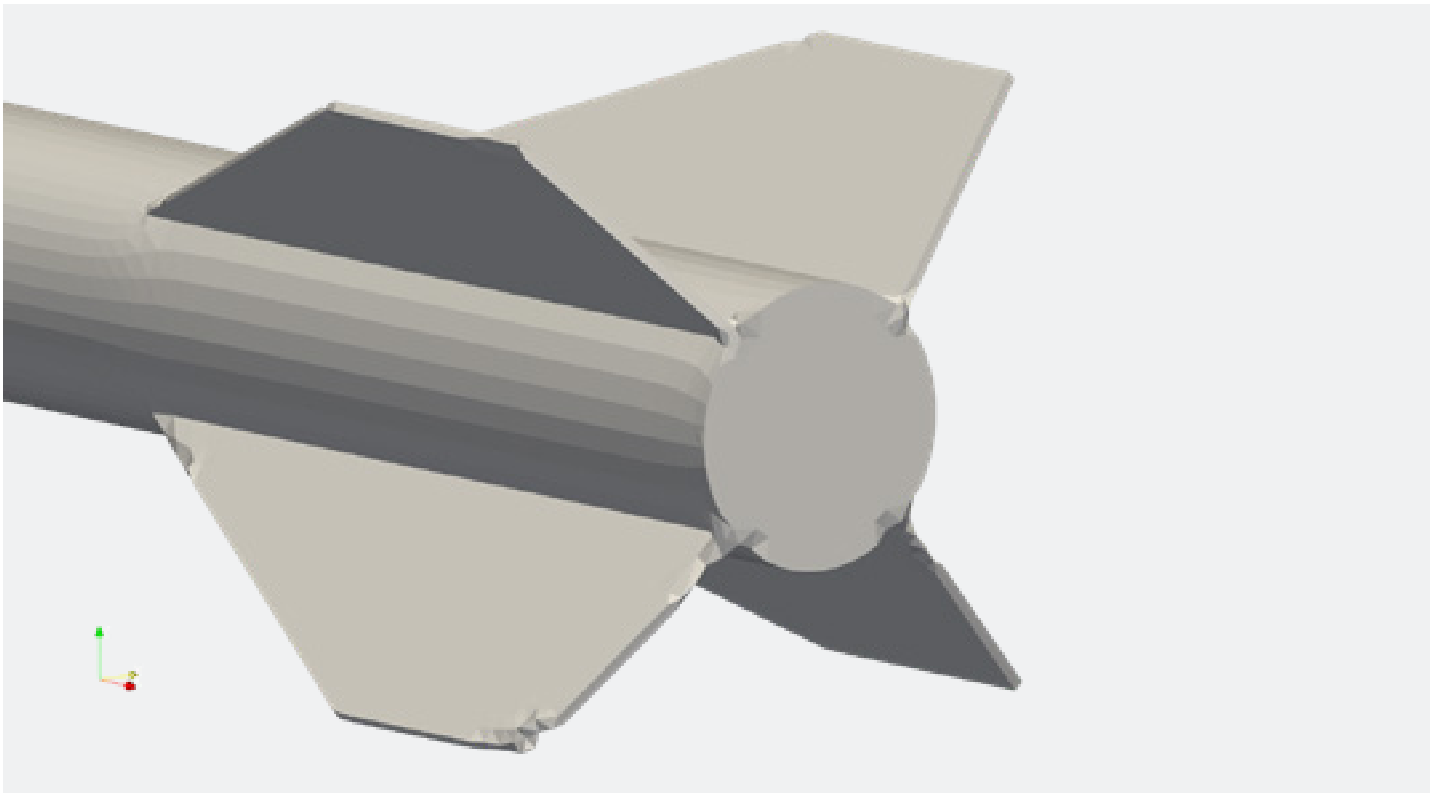


Figure 4. Poor Surface Mesh

Surface Meshing Setup

The first aspect of meshing to consider is the discretization of the rocket surfaces with faces. The main objective is to keep the faces small enough to capture the rocket curvature and edges. The rocket body tube and nose cone should “look” round and smooth. Fin edges should look “straight” and corners “sharp.” I also like the fins to be more than one cell thick.

At first glance, the rocket mesh may look fine if zoomed too far away. A good way to visualize the rocket mesh quality is to zoom in to a shaded view with the face edges turned off. **Figure 4** shows a very poor resolution of the body tube and fins.

I have been working on CFD simulations of typical 4FNC rocket models that have a diameter up to 75 mm. After trial and error, I found that face edge lengths of 2.0 mm to 3.0 mm on the rocket surface seem sufficient. I also like to increase the refinement to another level (1.0 mm to 1.5 mm) on fin edges and the trailing edge of the body tube. This ensures that these lines remain sharp, especially when rotating the model at angles of attack. **Figure 5** shows a surface mesh that is pleasing to my eye and simulates very well. The hexagonal volume cells produce near perfect squares on the rocket faces.

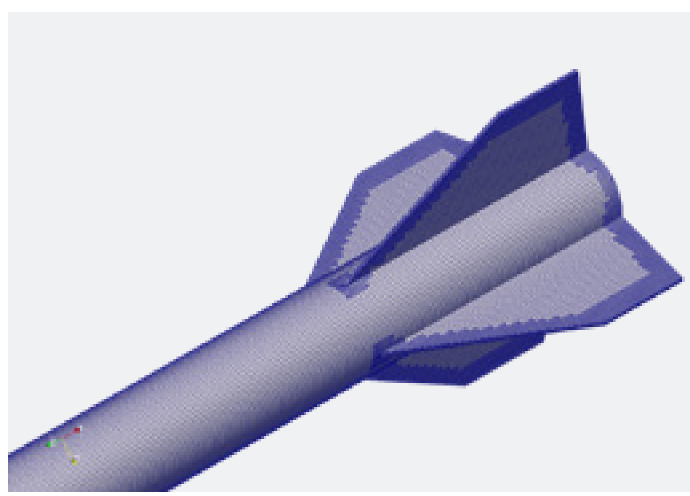


Figure 5. Good Surface Mesh with Edge Refinements

Another area to watch out for is the nose cone tip. A very sharp tip may not mesh correctly and cause a numerical error in stagnation pressure. **Figure 6** shows the nose tip radius resolved nicely with local mesh refinement.

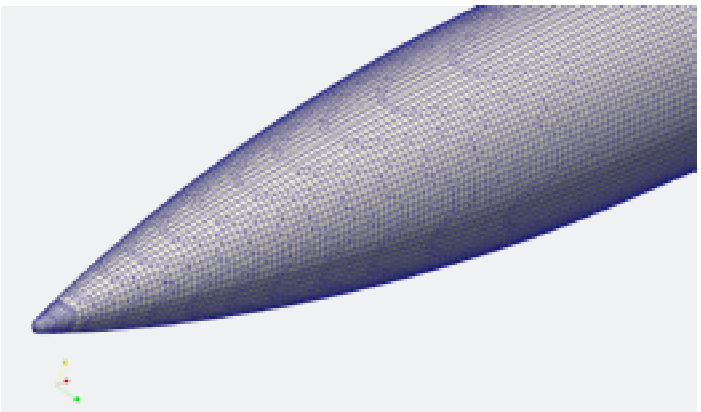


Figure 6. Nose Cone Mesh Resolution

Volume Meshing Setup

The goal of volume meshing is for us to refine the mesh in the computational domain, from the rocket surface all the way out to the boundaries. This mesh should capture important regions of the air flow, such as the boundary layer and wake regions. A smooth progression from fine to coarse reduces numerical jumps and helps keep the total cell count manageable.

A specialized mesh is needed to model the boundary layer, which is the thin layer of air in contact with the rocket surface. This mesh is highly refined and structured to accurately solve for the velocity gradient and shear stress (skin friction) that contribute to aerodynamic performance, like drag. Boundary layer modeling is extremely complicated and involves turbulence, the Reynolds Number, and characteristic lengths of the rocket geometry. The dimensionless quantity y+ ("y plus") is a CFD value used to size boundary layers correctly. Online y+ calculators are available.

For the incompressible flow simulation of model rockets using RANS and k-omega turbulence, a y+ value of 30 is a good starting assumption. A calculator will then suggest about 0.2 mm as the first layer height. I then choose 4-6 cells to span the height of the boundary layer mesh and transition smoothly to the outer bulk mesh. The boundary layer cells are typically prisms oriented normal to the rocket surface and aligned to the flow direction. **Fig. 7** shows a close-up cross section of the boundary layer mesh.

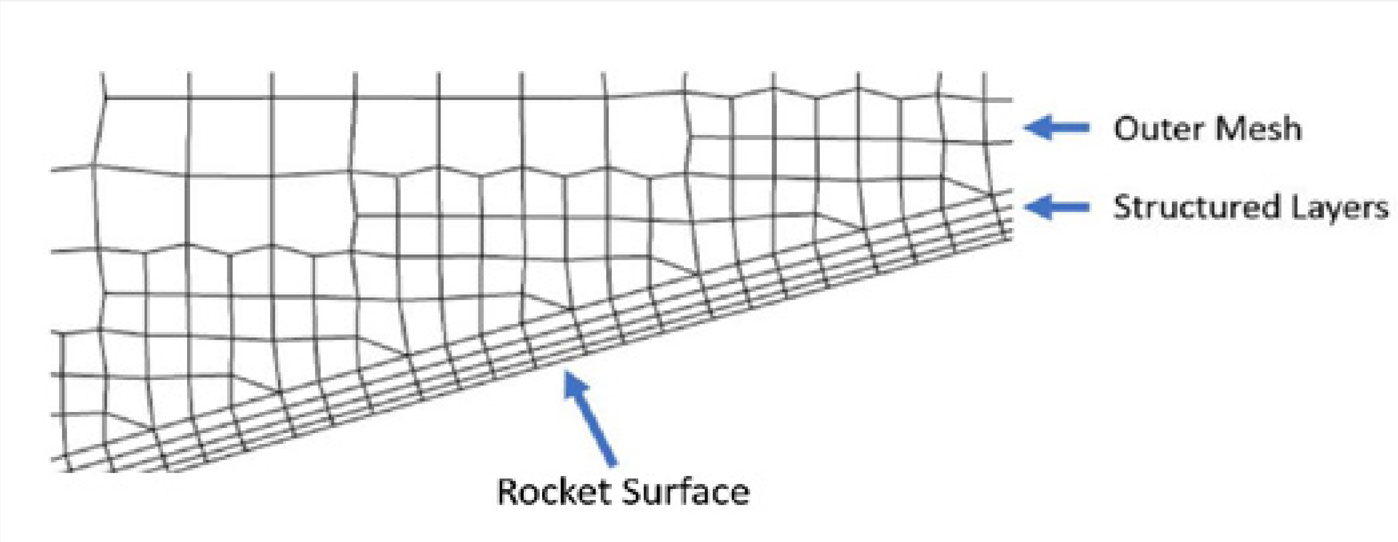


Figure 7. Boundary Layers

The mesh then transitions to the outer base mesh size in a series of steps using distance specifications or user-defined volume refinement boxes/cylinders. The wake regions behind the rocket should be refined to model vortex shedding and recirculating flow. I strive for about one rocket length of wake refinement on a streamlined shape. A bluff shape, like a spool rocket, would require a longer and wider wake refinement volume.

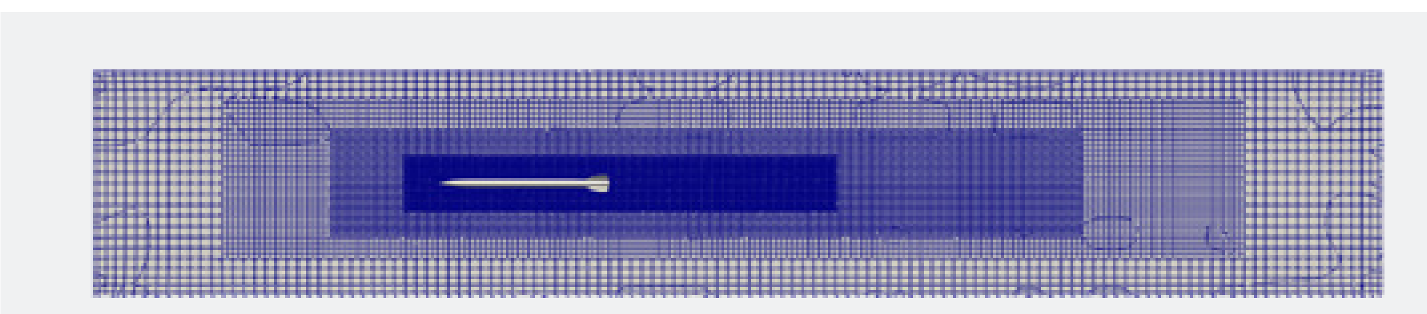
Figure 8 show a completed mesh domain for a high-power rocket about 1500 mm long and 75 mm in diameter. The rocket is biased with less mesh in the front and more mesh to the rear of the rocket. This is for efficiency, as the flow behind the rocket is more complicated and takes longer to settle out. The base mesh length is 100 mm, and three volume refinement boxes are specified along with a wrap around the rocket (too small to see in this picture).

The Meshing Process

After setup, the meshing process is kicked off on the computer to discretize the model. This is done in the background. Using the algorithms in CfdOF, I can generate a complete mesh in a few minutes on my desktop computer. The size specifications I mentioned previously will result in about 2 million total cells for a low-power rocket and about 5 million cells for a larger, high-power rocket. These are arbitrary goals, but after a series of trial and error, I am happy with the resulting computational time of the CFD solver and quality of the results. I did checks of sensitivity on a few models by altering the mesh resolution rather aggressively around the rocket surface. I saw up 2% difference in results, such as CP location.

There are mesh checks that occur during the process, like cell skewness, aspect ratio, and negative volume. These checks are

Figure 8. Cross Section of Meshed Domain



reported in the log files. If cells fall out of bounds, the mesher will attempt to correct them automatically and continue. The final mesh should ideally have no failed cells, but many times "close enough" will still work, as the modern CFD solvers are robust and can handle a few non-ideal cells.

Along with cell checks, you should also visually inspect the resulting mesh, just as I showed in the previous figures. Ensure that the rocket looks correct, and the mesh regions are resolved as intended. CfdOF has a mesh visualization, but even better is a post-processing software. ParaView is the popular open-source software for CAE visualization.

Solver Setup

The fluid medium needs to be defined. Air at 15 C is a standard condition for aircraft. For isothermal, incompressible flows, only the air density and dynamic viscosity are needed. Molar mass, specific heat, and Sutherland temperatures are used for high Mach Number, compressible flow.

Air properties on the boundaries are specified, just like velocity, pressure, temperature, and turbulence. I like to use Mach 0.3 for incompressible, subsonic, simulations. I simulated up to Mach 4.0 for doing my supersonic CFD simulations.

"Initializing" the model means to give a starting value for all the flow variables in the mesh so the solver can begin iteratively computing. These values can be 0 or taken from the boundaries. The most common approach is to perform a potential flow (inviscid, irrotational, and incompressible) calculation. This calculation takes mere seconds to complete.

Solving the Navier-Stokes conservation equations is tricky, especially the coupling of velocity and pressure. Many numerical techniques with clever acronyms were invented over the decades. How these algorithms work is beyond the scope of this article, but I will mention a few of them by name in case they appear as options in the CFD software.

The SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithm and its variants are popular for solutions of incompressible, steady-state, RANS problems. For transient, incompressible, solutions, there is PISO (Pressure Implicit with Splitting of Operators). PIMPLE (PISO-SIMPLE) is a hybrid of the two. Compressible problems usually need a "density-based" solver for transonic, supersonic, and hypersonic simulations. CfdOF includes the HISA (High-Order Implicit Shock Capturing Algorithm) method which is good for shock wave sims of rockets & aircraft.

CFD is an iterative technique, so the user must specify how to step through the calculations and when to stop them. For transient runs, the time step (Δt) and end time must be given. The Δt also determines the stability of the numerics, so it must be chosen wisely for both accuracy and to avoid blowing up the simulation. As mentioned previously, time-accurate CFD can be very costly in terms of calculation time.

Steady-state RANS calculations are easier on the compute wallet and easier to monitor the progression. Calculation steps are measured in terms of iterations, not time. An iteration is one loop through the steps of the SIMPLE algorithm, for example. The user specifies the maximum number of iterations to perform.

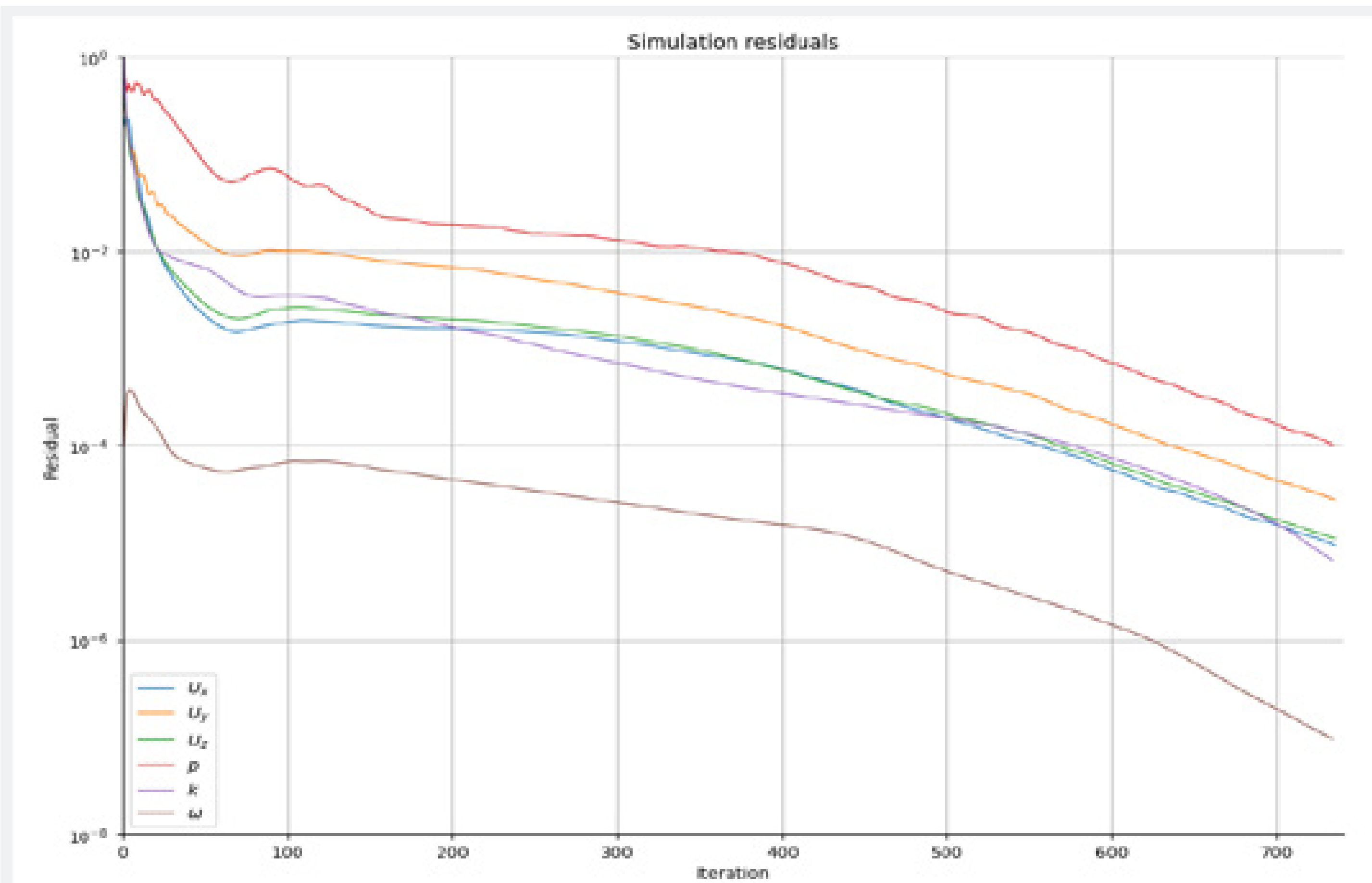
The Solving Process

CFD codes allow for parallel processing, meaning the model can be broken or “decomposed” into smaller pieces and distributed across multiple cores or threads of the computer. Multiple cores will run the calculations faster than one core. The speed up is not perfectly 1:1 linear, since there is overhead associated with passing information across the interfaces. My machine has four cores which I specify in the solver settings.

As the computations start to march through time or iterations, information is written into log files and then plotted, including the “residuals.” Residuals are a measure of how much the solution fails to meet the conservation of mass, momentum, and energy. As the solution improves, the residuals get smaller. The simulation will stop once a user-specified threshold is met, which is a means of establishing convergence. Convergence implies that flow variables stop changing appreciably from one step to the next.

Residuals are specified by orders of magnitude and plotted on a log scale. **Figure 9** shows a nice reduction in residuals from a steady-state solution. I specified 1000 maximum iterations and a residual tolerance of 10-4. All variables reached this threshold at 730 iterations. The pressure variable (p) usually takes the longest time to converge.

Figure 9. Residuals



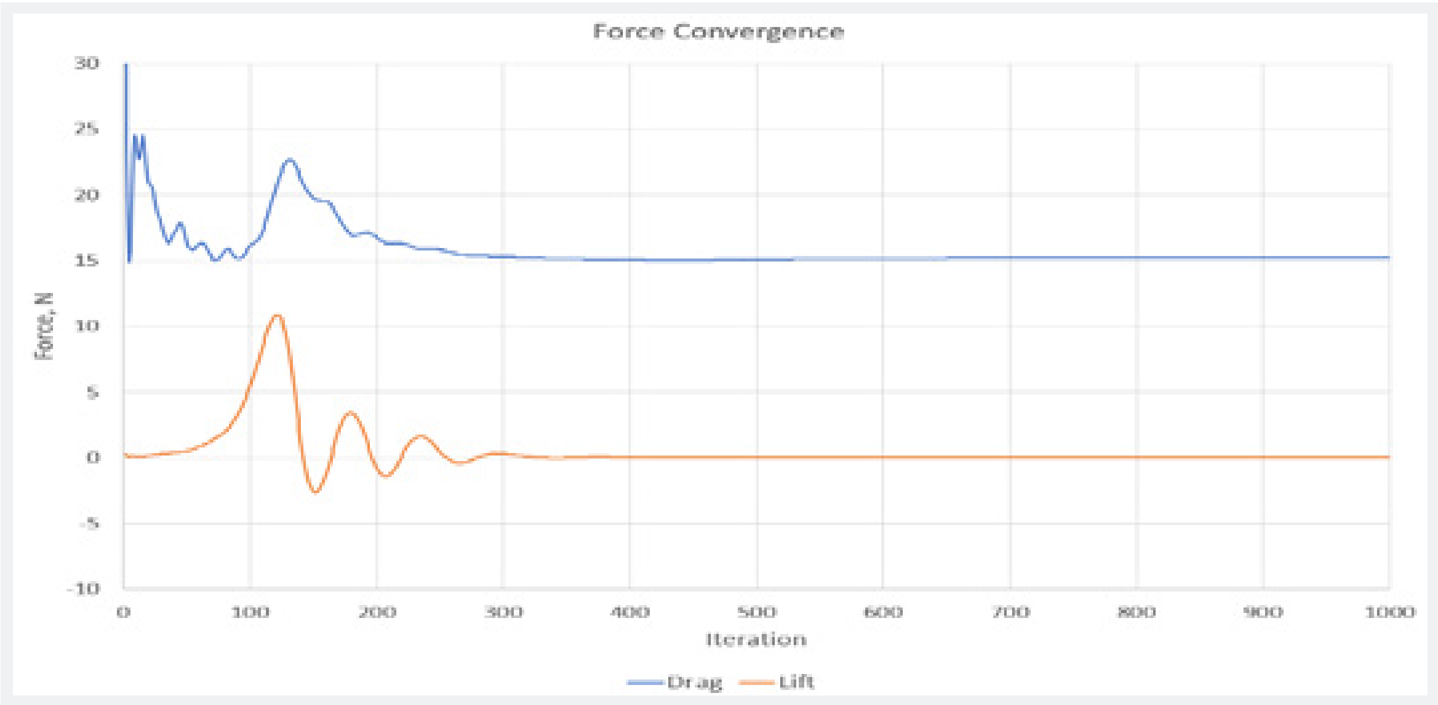


Figure 10. Force History

Residuals are rather meaningless when it comes to transient simulations or even RANS simulations with some periodic behavior. The solution will oscillate, and residuals will never reduce beyond a couple orders of magnitude. A better measure of convergence is something practical and reflective of the problem. This could be a point to monitor pressure or a plane to measure flow rate. Since most of our rocket aerodynamics involve drag and lift, a force monitor is usually best.

The user defines the drag and lift directions in the model coordinate system. I use X and Z, respectively, with the origin on the tip of the nose cone. These forces are computed by the solver by integrating the pressure and wall shear stress on each mesh face of the rocket. The force results are written to file at each time step or iteration.

Figure 10 is a plot of drag and lift history on my 75 mm rocket. The initial 400 iterations or so is the startup of the simulation where the forces bounce around as the solver works to satisfy the Navier-Stokes equations. After 400 iterations, the forces are stable, and the solution is considered converged. Since I ran the calculations a lot longer than needed with no noticeable change in forces, I am confident that this is a good result. The model is symmetric and at zero degrees angle of attack, so the lift is zero.

The very last drag result at iteration 1000 should not be taken as the "answer." Sometimes the force may oscillate around a mean value. This is absolutely the case in transient sims and can occur in steady sims of very bluff rockets or if an asymmetry is present (like yaw angle or the camera shroud I showed in Part I). Always take an average of the force over the last section of the simulation time that includes a couple periods of oscillation. For a response like in **Figure 11**, I would average from 600 to 1000 iterations.

The mesh and typical steady-state solver settings I discussed previously, along with my computer hardware, yields a reasonable, converged, aerodynamic result in 3 to 6 hours depending on the size of the rocket. I am happy with this performance and think it is a good blend of accuracy and efficiency for home computing.

Results

Numerical data of forces and moments is available to calculate CD, CL, CNa, and CP. CFD also can visualize the airflow in many ways to help understand the aero performance. Post-processing is really an art, and you need to plan for what you want to see. In business, CFD is often mocked as "Colors for Directors," but it is a good way to get your point across to the bosses (or totally confuse them, if done poorly.)

Most commercial CFD codes have post-processing software included in their suite of tools. However, there are powerful, stand-alone products that specialize in visualizing scientific data. ParaView and EnSight are two good ones for OpenFOAM, but they come with steep learning curves. Like the convergence analysis above, visual inspection of the air flow is a way to assess the goodness of the simulation. The following examples will cover a few basics.

Figure 11. Oscillating Force History

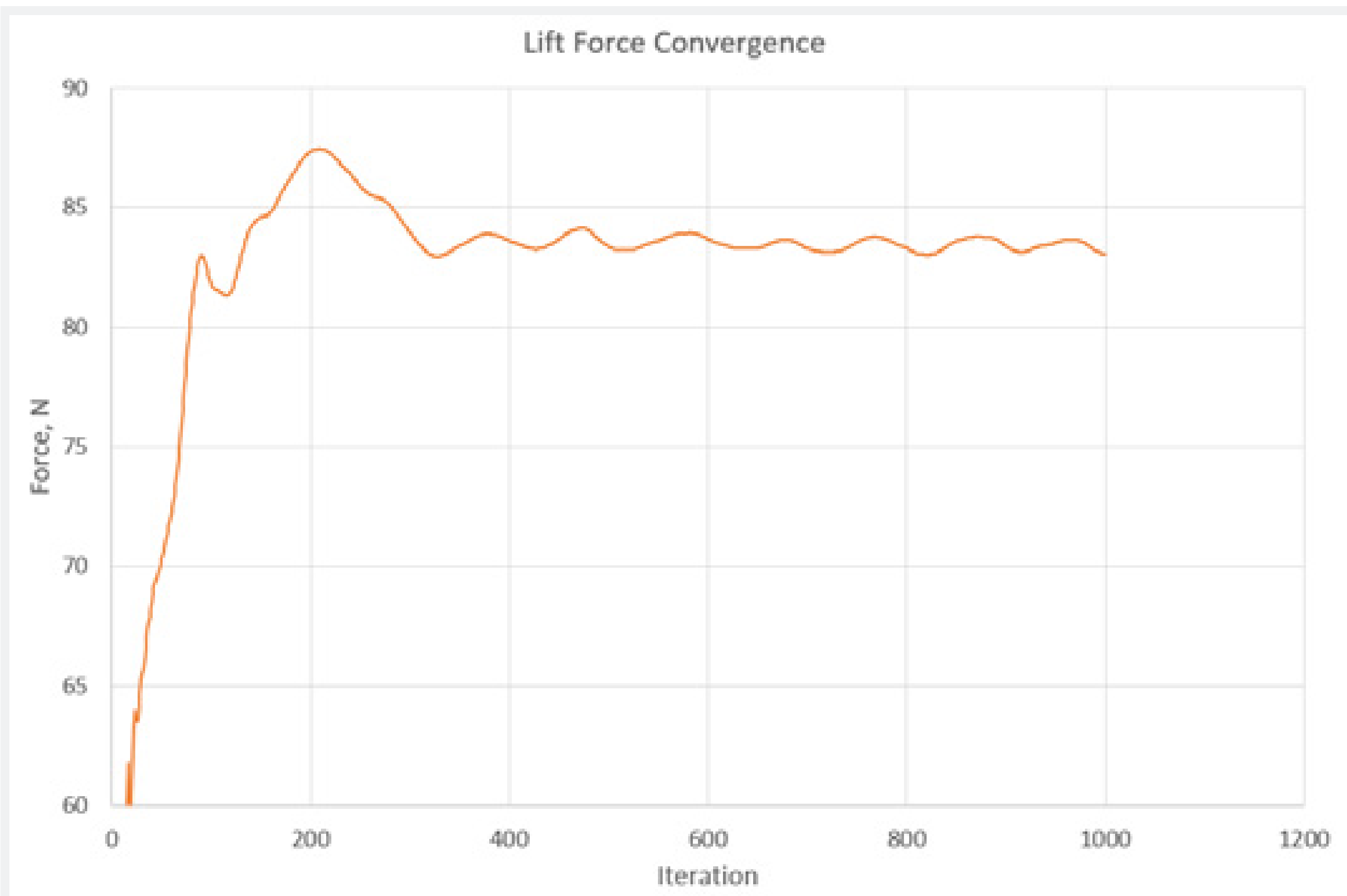


Figure 12 shows us the rocket colored by static pressure normalized as Pressure Coefficient:

Cp can range from +1 to -3 or more. At zero angle of attack, the blunt tip of the nose cone should be the stagnation point with Cp equal to 1.0 or nearly so (maybe 1.05 due to slight numerical error). As the flow accelerates over the curved nose cone, the pressure become negative. The body tube is mostly neutral pressure. Positive high pressure is evident again on the leading edge of the fins. Strong low pressure is seen behind the fin leading edges as there is some minor flow separation off the square profile. Moderate low pressure is on the rocket base. All these observations make sense from the aerodynamics point of view.

Figure 13 is a 2D slice through the fluid domain along the rocket centerline and colored by velocity magnitude. The velocity is zero on the nose at the stagnation point. The speed is greater than the free stream (110 m/s vs. 100 m/s) where the air flows over the curved nosecone. 10% to 20% velocity acceleration is typical. Low velocity is in the wake behind the body tube and fin.

Figure 14 shows 2D velocity vector streamlines which indicate speed and direction in the slice. The flow remains attached to the nose and body and separates off the fin and base. In the base wake are two, symmetric, counterrotating, vortices. The wake quickly closes about one caliber behind the rocket. This is typical of a steady-state solution. Note that this simulation represents a coasting rocket without the motor exhausting into the wake.

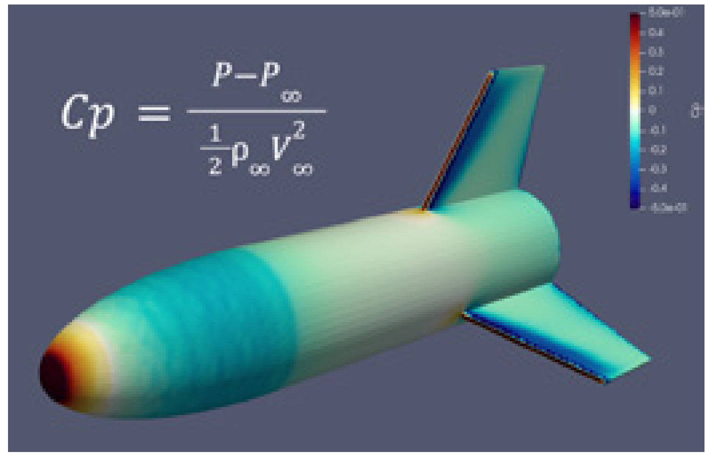


Figure 12. Pressure Coefficient

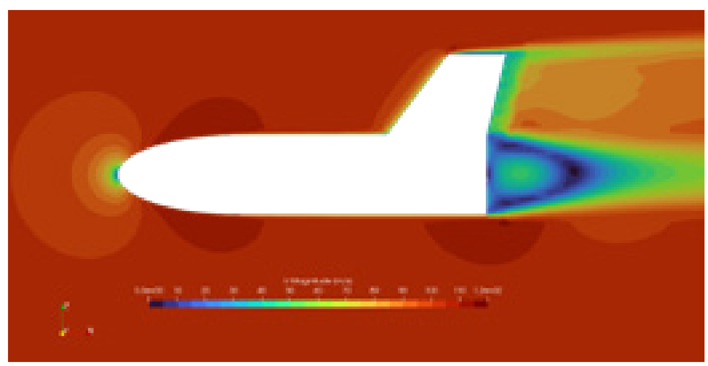


Figure 13. Velocity Slice

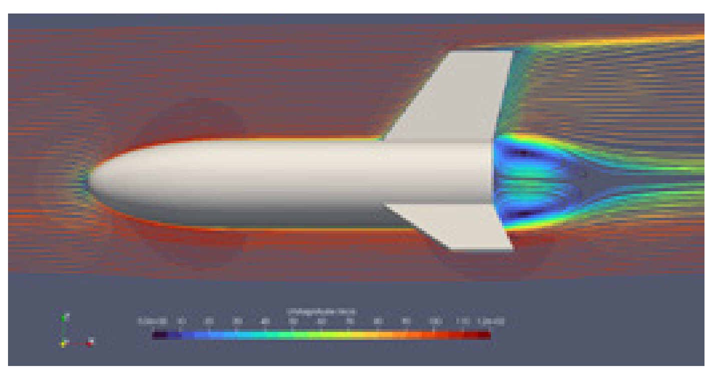


Figure 14. Streamlines on the Slice

Figure 15 is a 3D view of velocity streamlines seeded behind a camera shroud. They show a local wake behind the shroud and how that disturbance propagates downstream.

Figure 16 is the Army-Navy Finner model, a benchmark case for CFD and wind tunnel correlations, flying at Mach 1.5. The velocity slice is colored by Mach Number and shows shock waves. This was simmed with the HISA algorithm for supersonic flights.

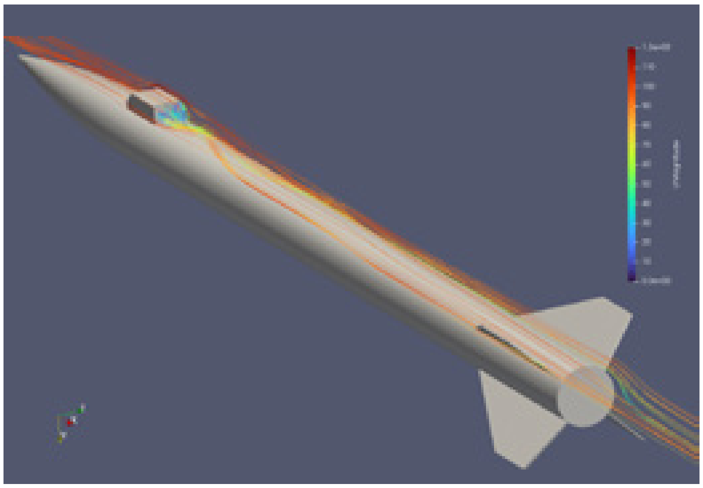


Figure 15. 3D Streamlines

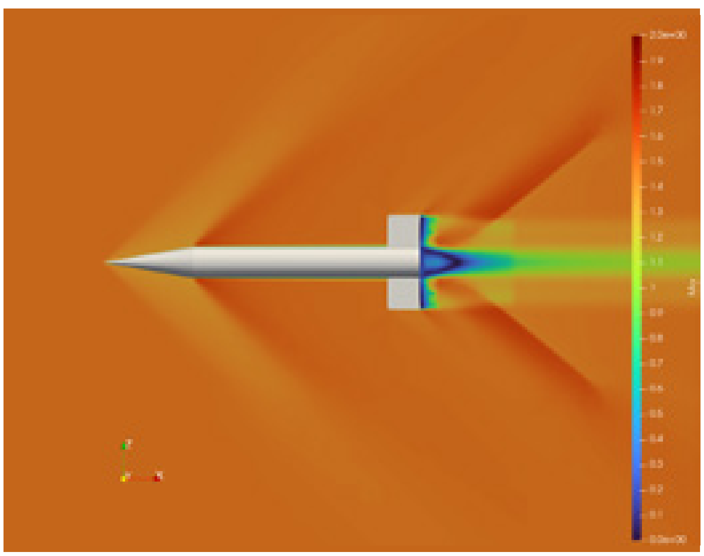


Figure 16. Slice Colored by Mach Number

Accuracy

So, how good are these fancy-schmancy CFD results? Like most simulations, CFD is most powerful when comparing multiple designs against each other to determine the best performer. However, absolute values can be measured against other computer sims, wind tunnel simulations, or free flight tests. Here are a couple cases I performed.

A simplified, subsonic, Aerobee 350 was a validation case for Jim Barrowman's thesis and R&D report on the theoretical center of pressure. **Figure 17** shows the CP location predicted by Barrowman and other simulations. The CFD result falls in nicely with the other models. CFD is somewhat overkill for this basic rocket shape, but for geometries that violate the Barrowman equations, such as complicated fins and asymmetric components, it would be worth the effort to obtain an accurate CP.

Figure 18 shows the drag coefficient of the Army-Navy Finner

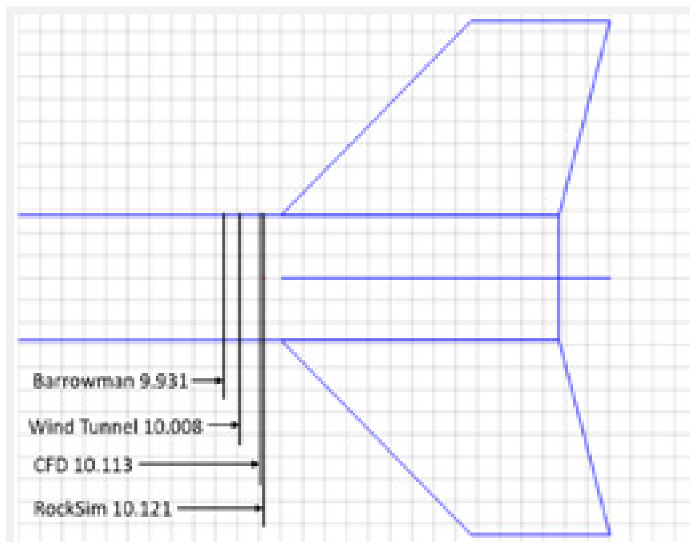


Figure 17. Aerobee 350 CP Location Predictions (m)

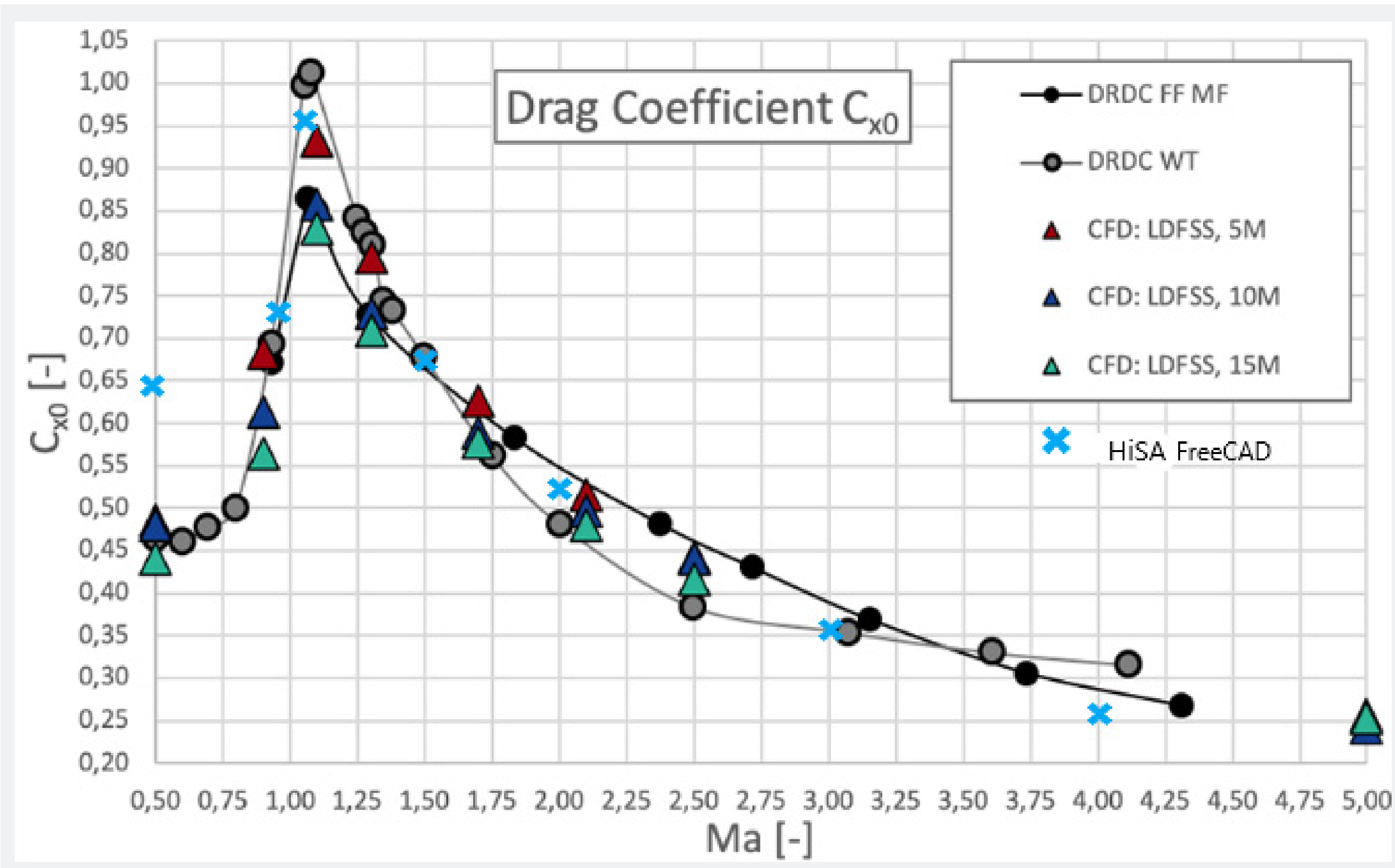


Figure 18. Army-Navy Finner Drag Predictions

over a range of Mach Numbers. I used a 2.4M cell HISA model and plotted the results along with other CFD models, wind tunnel, and free flight tests. Other than the subsonic condition at Mach 0.5, my CFD result agrees with rest of the supersonic data, even with a relatively low-cost simulation.

Conclusion

New developments in CAD and CAE software have opened the door to hobbyists looking to go to the next level with simulations. Computational Fluid Dynamics can help answer model rocket aerodynamic questions more accurately than ever before. There are many scenarios where CFD is needed for drag and stability problems that cannot be simulated correctly any other way. I am working on a couple applications to share in the future.

About the Author

Ken Karbon is a rocketeer from Michigan. He is a retired engineer from the auto industry where he specialized in CFD and aerodynamics.

PEAK OF FLIGHT NEWSLETTER

Article Submission Guidelines

Issue 642 / December 31st, 2024

www.ApogeeRockets.com

SUBMITTING ARTICLES TO APOGEE

We are always looking for quality articles to publish in the *Peak-of-Flight* newsletter. Please submit the "idea" first before you write your article. It will need to be approved first.

When you have an idea for an article you'd like to submit, please use our contact form at **https://www.apogeerockets.com/Contact**. After review, we will be able to tell you if your article idea will be appropriate for our publication.

Always include your name, address, and contact information with all submissions. Including best contact information allows us to conduct correspondence faster. If you have questions about the current disposition of a submission, contact the editor via email or phone.

CONTENT WE ARE LOOKING FOR

We prefer articles that have at least one photo or diagram for every 500 words of text. Total article length should be between 2000-4000 words and no shorter than 1750 words. Articles of a "how-to" nature are preferred (though other types of articles will be considered) and can be on any rocketry topic: design, construction, manufacture, decoration, contest organization, etc. Both model rocket and high-power rocket articles are accepted.

CONTENT WE ARE NOT LOOKING FOR

We don't publish articles like "launch reports." They are nice to read, but if you don't learn anything new from them, then they can get boring pretty quick... Example: "Bob flew a blue rocket on a H120 motor for his certification flight." As mentioned above, we're looking for articles that have an educational component to them, which is why we like "how-to" articles.

You can see what articles and topics we've published before at: **https://www.apogeerockets.com/Peak-of-Flight?pof_list=archives&m=education**. You might use this list to give you an idea or two for your topic.

Here are some of the common articles that we reject all the time, because we've published on these topics before:

- How to get a L1, L2, or L3 Cert
- Building cheap rockets and equipment (pads & controllers)
- How to 3D print parts, or a Rocket Kit
- How to Build a cheap Rocket Kit
- Getting Back Into Rocketry After a Long Hiatus

ARTICLE & IMAGES SUBMISSION

Articles may be submitted by emailing them to the editor. Article text can be provided in any standard word processor format, or as plain-text. Graphics should be sent in either a vector format (Adobe Illustrator, SVG, etc.) or a raster format (such as jpg or png) with a width of at least 600 pixels for single column images or 1200 pixels for two-column images. It is preferable for images to be simple enough to be readable in a two-column layout, but special layouts can be used.

Send the images separately via email as well as show where they go by placing them in the word processor document.

ACCEPTANCE

Submitted articles will be evaluated against a rubric (available here on our website). All articles will be evaluated and the results will be sent to the author. In the evaluation process, our goal is to ensure the quality of the content in *Peak-of-Flight*, but we want to publish your article! Resubmission of articles that do not meet the required standard are heavily encouraged.

ORIGINALITY

All articles submitted to Peak-of-Flight must not run in another publication before inclusion in the *POF* newsletter, but it may be based on another work such as a prior article, R&D report, etc. After we have published and paid for an article, you are free to submit them to other publications.

RATES

Apogee Components offers **$300** for a quality-written article over 2,000 words in length. Payment is pro-rated for shorter articles.

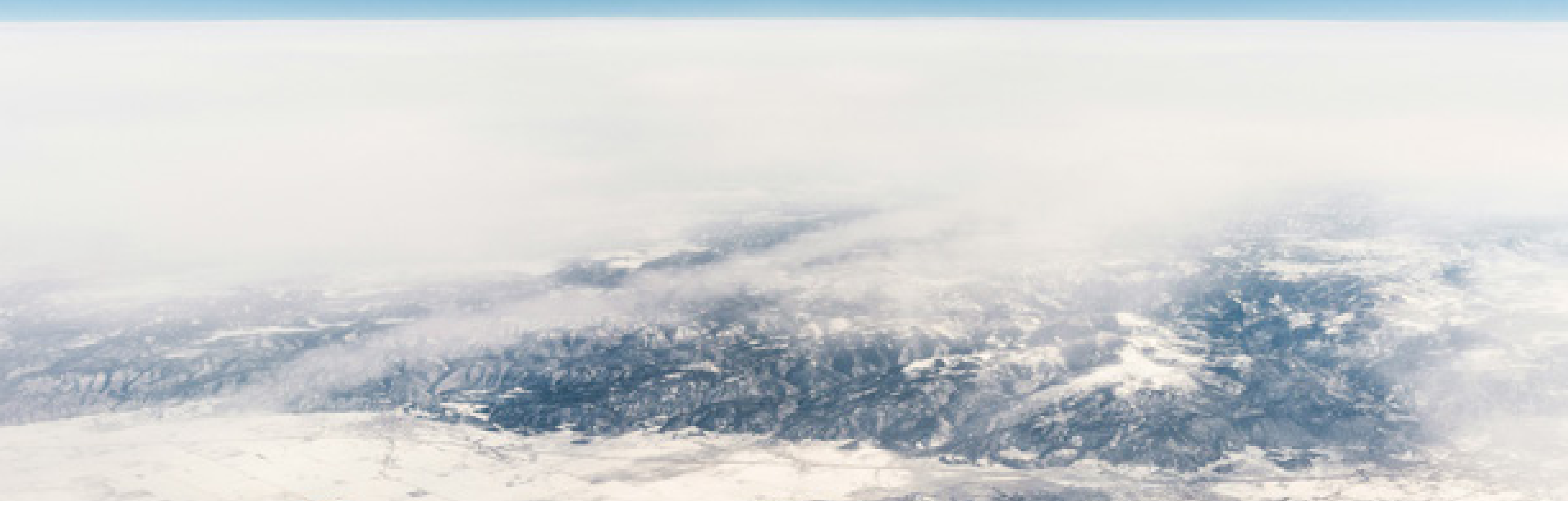
WHERE WILL IT APPEAR?

These articles will mainly be published in our free newsletter, *Peak-of-Flight*. Occasionally some of the higher-quality articles could potentially appear in one of Tim Van Milligan's books that he publishes from time to time.


FREQUENT FLYER
PROGRAM
Become a V.I.R. member!
Earn useful GIFTS & points towards DISCOUNTS!
RECOVERY CREW
VEHICLE TECHNICIAN
SCIENCE OFFICER
PAYLOAD SPECIALIST
FLIGHT ENGINEER
MISSION SPECIALIST
PILOT
COMMANDER

Apogee Components, Inc.
4960 Northpark Dr.
Colorado Springs, CO 80918
719-535-9335

Your Success Is Our Mission!



www.ApogeeRockets.com