

Enhancements to TetrUSS for NASA Constellation Program

Mohagna J. Pandya^{*}, Neal T. Frink[†], Khaled S. Abdol-Hamid[‡], Jamshid A. Samareh[‡]
NASA Langley Research Center, Hampton, Virginia 23681, USA

Edward B. Parlette[§]
ViGYAN, Hampton, Virginia 23666, USA

James R. Taft[¶]
Sienna Software, Inc, Reno, NV 89502, USA

The NASA Constellation program is utilizing Computational Fluid Dynamics (CFD) predictions for generating aerodynamic databases and design loads for the Ares I, Ares I-X, and Ares V launch vehicles and for aerodynamic databases for the Orion crew exploration vehicle and its launch abort system configuration. This effort presents several challenges to applied aerodynamicists due to complex geometries and flow physics, as well as from the juxtaposition of short schedule program requirements with high fidelity CFD simulations. NASA TetrUSS codes (GridTool/VGRID/USM3D) have been making extensive contributions in this effort. This paper will provide an overview of several enhancements made to the various elements of TetrUSS suite of codes. Representative TetrUSS solutions for selected Constellation program elements will be shown. Best practices guidelines and scripting developed for generating TetrUSS solutions in a production environment will also be described.

I. Introduction

NASA's Constellation program¹ was initiated to address the needs of U.S. access to space in the post Space Shuttle era with the mission to the moon, Mars and beyond. Unlike the Space Shuttle, where both crew and cargo are launched simultaneously, the proposed architecture for the Constellation program includes two separate launch vehicles: the Ares I for the crew and the Ares V for the cargo. The Ares I crew launch vehicle (CLV) is a two-stage rocket that incorporates a modified five-segment solid rocket booster from the Shuttle program while the upper stage is powered by the J-2X engine from the Apollo program. The major components of the vehicle as well as the proposed nominal flight trajectory are shown conceptually² in figure 1. The major components of Ares-I CLV consist of the Upper Stage (US) and the First Stage (FS) Solid Rocket Booster (SRB) that includes the frustum and the aft-skirt. Ares-I is the launch vehicle for the Orion Crew Exploration Vehicle (CEV) and its Launch Abort System (LAS) as also illustrated in figure 1.

The aerodynamic databases used for developing the Ares launch vehicles and the Orion CEV are being produced from a combination of wind tunnel testing and computational fluid dynamics (CFD) simulations. Since these vehicle geometries have such simple shapes, the original assumption was that CFD could serve as the primary source of aerodynamic data with wind tunnel testing in an anchoring role. However, the initial CFD analysis on these simple shapes quickly exposed complex flow fields primarily in association with blunt bases, protuberance effects, and close proximity jet interactions. The complex flow fields were also found to be challenging to model with CFD. Hence, the historical role of the wind tunnel as the primary source of aerodynamic data was restored, despite its own set of limitations and uncertainties. Even with CFD in a supporting role, thousands of solutions have been generated yielding an impressive amount of computational data for the Ares and Orion projects.

^{*} Research Aerospace Engineer, Configuration Aerodynamics Branch, Mail Stop 499, AIAA Senior Member

[†] Senior Aerospace Engineer, Configuration Aerodynamics Branch, Mail Stop 499, AIAA Associate Fellow

[‡] Senior Aerospace Engineer, Vehicle Analysis Branch, Mail Stop 451, AIAA Associate Fellow

[§] Research Engineer, 30 Research Drive

[¶] Senior Analyst, jtaft@siennasoftware.net

Within the subsonic/transonic/supersonic speed range, the computational aerodynamic database for Constellation was primarily generated from the NASA TetrUSS³/USM3D^{4,5}, and OVERFLOW⁶ flow solvers. Since code-to-code uncertainty was a key concern, some additional support was also provided from the NASA FUN3D⁷ code. The NASA Cart3D inviscid code⁸ was also used heavily for screening broad ranges of parameter space within the Orion project. For the Ares flight vehicles, the USM3D code served as the primary computational data source with supporting role from the OVERFLOW code. For the Orion CEV and LAS, the OVERFLOW code provided the majority of computational data with USM3D serving in a support role for quantifying code-to-code uncertainty. In both cases, the CFD teams worked seamlessly across the NASA centers to deliver the needed data on time, while collaborating often to solve the unforeseen challenges that surfaced.

Historically, TetrUSS has been a research tool. However, Ares and Orion were clearly development projects that demanded production-level efficiencies for data generation. Hence, many challenges were encountered while quickly transforming a research-oriented tool into a production-oriented tool in the midst of a fast-paced database generation campaign. Some of the challenges that were encountered during the application of TetrUSS tools in the Ares and Orion projects were:

Early challenges:

- Required grid sizes were much larger than USM3D could accommodate in memory,
- Solver inefficiencies for large grids prohibited production-level throughput with USM3D,
- How to quantify solution uncertainty,
- Running and managing a large number of cases,
- Post-processing and managing a massive amount of data.

Computational challenges for Ares Project launch vehicles:

- Need for 10-15 solutions per day on grids with 50 to 100+ million tetrahedral cells,
- Efficient generation of tetrahedral grids suitable for flight Reynolds number on slender launch vehicle with large disparity of geometric scale from small protuberances,
- Numerical stiffness associated with modeling flows at flight Reynolds number,
- Extracting line loads for structural design,
- Extracting and organizing aerodynamic loads on numerous individual protuberances for structural design.

Computational challenges for Orion Project LAS:

- Large uncertainty in axial force due to strong sensitivities to grid resolution and turbulence models,
- Inadequate turbulence models and gridding strategies for predicting strong jet interaction effects from abort motors,
- Need for autonomous solution-adaptive grid capability for resolving complex jet-plume flows,
- Throughput requests on order of 400 LAS jet interaction solutions per month.

This paper presents an overview of how most of these challenges were successfully resolved. The paper is organized with Section II presenting a broad overview of the TetrUSS system. Section III includes a discussion of key enhancements to the grid generation capability that enabled the rapid generation of high-quality tetrahedral grids. A description of the solution-adaptive grid approach is also included. Section IV describes enhancements to the flow solver that overcame show stoppers for large applications such as memory bottlenecks, slow run times, crippling robustness issues for high-Mach number flows and startup of jet flows, inadequate jet boundary condition capability, and deficient turbulence models for strong jet/aero-interaction flows. The section V describes the processes devised for running and post-processing large numbers of cases. Finally, Section VI highlights some representative applications for the Ares I, IX, and V launch vehicles and the Orion Launch Abort System (LAS).

II. Overview of TetrUSS Tool Set

The NASA Tetrahedral Unstructured Software System (TetrUSS) was developed during 1990s to provide a rapid aerodynamic analysis and design capability to applied aerodynamicists. The system is comprised of loosely integrated, user-friendly software that enables the application of advanced Euler and Navier-Stokes tetrahedral finite volume technology to complex aerodynamic problems. A schematic view of the TetrUSS system is presented in figure 2. The system consists of component software for setting up geometric surface definitions (GridTool), generating tetrahedral grids (VGRID), computing Euler and Navier-Stokes flow solutions (USM3D), and extracting meaningful information from analysis of results (SimpleView). The system also allows for imposing design or aeroelastic shape changes by interfacing with other codes, such as CDISC⁹. An overview of the capabilities of

TetrUSS system in early 2000 is presented in reference 3. The salient features of the VGRID grid generator, USM3D flow solver, and various visualization as well as data analysis tools and utilities are presented below.

A. Grid Generation

VGRID is a tetrahedral grid generator based on the Advancing Front Method¹⁰ (AFM) for generation of the ‘inviscid’ field cells and the Advancing Layers Method¹¹ (ALM) for generation of thin-layered ‘viscous’ cells. Both techniques are based on a marching process in which tetrahedral cells grow on an initial front (triangular surface mesh) and gradually accumulate in the field around the subject geometry. Unlike the conventional AFM, which introduces cells in the field around in a totally unstructured manner, the ALM generates layers of thin tetrahedral cells in a more orderly fashion (one layer at a time) while maintaining the flexibility of AFM. Once the advancing front process is completed in VGRID, an additional post-processing step is required using POSTGRID to close any open pockets and to improve grid quality. VGRID input files are generated by an interactive geometry manipulation program, GridTool¹². It can import surface definitions from IGES files (NURBS surfaces and NURBS curves) and from PLOT3D files (point definitions) and manipulate them to define necessary geometric (*surface patches*) and grid-spacing (*sources*) parameters. It uses OpenGL for 3D graphics. The graphical interface is based on the Fast Light Toolkit. GridTool is available for Mac and Linux systems.

B. Flow Solver

The USM3D^{4,5} code is a parallelized tetrahedral cell-centered, finite volume Navier-Stokes flow solver. The term cell centered means that the finite volume flow equations are solved at the centroid of each tetrahedral cell. Inviscid flux quantities are computed across each tetrahedral cell face using various upwind schemes. Spatial discretization is accomplished by a novel reconstruction process, based on an analytical formulation for computing solution gradients within tetrahedral cells. The solution is advanced in time by a second-order Newton time step scheme¹³, or to a steady-state condition by an implicit backward-Euler scheme. Several turbulence models are available: the Spalart-Allmaras (SA) one-equation model⁵, the two-equation k - ϵ turbulence model, the Menter Shear Stress Transport (SST) two-equation model, and the nonlinear Algebraic Reynolds Stress Models (ARSM) of Girimaji and Shih/Zhu/Lumley¹⁴. Detached Eddy Simulation (DES) has been implemented in all of the turbulence models. A capability to trip the flow at specified locations on aerodynamic surfaces has been implemented for the k - ϵ turbulence model, but fully turbulent flow is assumed for the results to follow. USM3D has capabilities for dynamic grid motion and overset grids¹⁵.

C. Post Processing

TetrUSS system has an interactive menu-driven post-processing program, SimpleView, which allows users to visualize unstructured grid and flow solution details. SimpleView uses OpenGL for 3D graphics and is available for Mac and Linux systems. The Mac version uses the native Cocoa user interface, while the Linux version uses a Motif-based interface. It can read grid files at any stage of the grid generation process and present either the entire grid generated or the surface mesh only. VGRID *front* files can also be read, either in conjunction with the grid or in a stand-alone mode. Several diagnostic tools are available inside SimpleView, particularly for the viscous layers of the grid. Users can verify the presence of viscous layers above any face on a viscous surface and obtain a count of the number of cells in the layers. Simpleview reads USM3D nodal solution file to display five conserved flow variables, as well as a number of derived variables, such as pressure coefficient or entropy. It can perform surface (line) and volume (planar) cuts. Surface data and planar cuts may be displayed as solid-shaded or contour lines. It can display oil flow and velocity vectors on the aerodynamic surfaces. It can also display velocity vectors and streamlines in the flow-field and find iso-surfaces.

Standalone codes have also been developed to transform TetrUSS grid and solution files for visualization and post-processing using some of the major commercial software, such as TecPlot[®], FieldView[®], and EnSight[®]. Other stand-alone post-processing tools have also been developed for the data analysis. For example, a utility to extract sectional load distributions is available that will be described later in more detail. A TetrUSS grid and solution cutter¹⁶ utility for generalized sub-domain extraction of over 50 user-specified flow functions has recently been developed for efficient post processing of solutions.

D. Utilities

Several useful utility codes are maintained through a common interface called ‘usgutil’. They are:

1. Mirroring of grid or VGRID input file,
2. Extracting cross-sectional geometry and flow field data,

3. Estimating grid spacing and stretching parameters used in VGRID for viscous grids, and
4. Scaling, translating, rotating of grid and swapping of coordinates.

Additional standalone utilities have also been developed to enhance TetrUSS productivity. For example, a utility is developed to interpolate solution from coarse grid to a finer adapted grid. Another utility is available to identify trouble spots in difficult-to-converge or diverging solutions that can aid in the localized improvements to the grid quality.

III. Enhancements to Grid Generation

Grid generation presents a number of challenges within a production environment. Geometries are continually evolving and require frequent generation of new grids. The demands of airframe design necessitate adequate grid resolution to yield high-fidelity loads on very small components, such as the various protuberances and appendages on the Ares launch vehicle. The Ares and Orion flows are inherently complex and can manifest nonlinear effects due to shocks, separated flows, strong jet interactions, fluid/structure interactions, and blunt-base wakes. Most of these flow features can only be efficiently grid resolved using solution-adaptive grid capability. The following subsections present some of the key enhancements to the grid generation capability that enabled the rapid generation of high-quality tetrahedral grids and flow solutions.

A. NURBS Patches

NURBS patches are a new versatile patch type incorporated in VGRID based on geometry routines from GridTool. NURBS patches can be defined with highly irregular boundary shapes and can be composed of an arbitrary number of sides. VGRID NURBS patches require an underlying NURBS surface definition. NURBS patches lead to a significant reduction in time spent for generating patches, because a NURBS surface definition can almost always be directly turned into one patch. The older parametric patches usually required extensive work to split a surface definition into 'good' patch shapes. Another benefit of NURBS patches is that grid points generated on NURBS patches are guaranteed to be on the original surface definition. Consequently, there is no need for time-consuming projection of the surface grid onto the underlying surface definition. Also, NURBS patches can accommodate multiple-loop patches, where a patch has one or more holes in the middle. This allows for an easy insertion, deletion or modification of small geometric features that may be embedded in a larger feature, such as a protuberance on a fuselage. These features can be patched and sourced in a separate file, and subsequently placed anywhere on the original configuration. The features may even be transformed and translated to different positions and orientations, and inserted in multiple locations.

B. Volume and Surface Sources

A new approach for the distribution of grid points on the surface and in the volume has been developed and incorporated in VGRID¹⁷. In addition to the point and line sources, the new approach uses surface and volume sources for automatic curvature-based grid sizing and convenient point distribution in the volume. A new exponential growth function produces smoother and more efficient grids and provides superior control over the distribution of grid points in the field. Anisotropic grid stretching is still available for all types of sources for grid economy. The new approach makes use of an efficient bounding-box auxiliary medium for storing grid parameters defined by surface sources instead of three-dimensional background grids. These enhancements make the new approach less memory intensive and more computationally efficient. The latest version of VGRID has been used to generate a grid consisting of one billion tetrahedral cells around a generic wing-body transport configuration on an intel-Mac desktop.

Figure 3 demonstrates an example of a grid around Apollo Launch Abort System using the enhanced VGRID. This configuration features several geometric complexities that present formidable challenges to grid generation such as truss beams in very close proximity with tight angles at end junctures, and grid extending through the abort motor nozzles and into the plenum. The grid in figure 3 was created from only seven volume sources around the vehicle and two volume sources in base wake region. The resulting grid of 48,355,970 tetrahedral cells and 8,216,471 nodes was generated in 88 minutes of Central Processing Unit (CPU) time on a MacBook Pro laptop with a 2GHz Intel Core Duo processor and 2GB of RAM. The grid in the complex truss region is controlled by a single conical volume source that results in a high-quality thin-layered tetrahedral grid suitable for viscous flow simulations.

GridTool (used to prepare the input file for VGRID) has also been upgraded with several usability enhancements and includes support for the latest VGRID (version 4.1) features. In addition, it has been ported to several compute platforms such as Mac and Linux.

C. Improved Gridding Practices

Grid spacing around cylindrical features of a slender configuration can be prescribed with either a ring of line sources or a single cylindrical volume source. Although a ring of line sources can add flexibility by allowing for localized changes in grid spacing, they also introduce undulations in the mesh as these sources deviate from an exact circle. For the small cylindrical features on the Constellation vehicles, such as extruding rings around the launch vehicle or steps around the launch abort vehicle tower, it was determined that cylindrical volume sources produce markedly better quality grids. The numerical resolution of such features requires a high longitudinal concentration of cells with relaxed concentration in the circumferential direction. It was found that a circumferential stretching of 10:1 was generally acceptable, although the stretching was lowered to 5:1 in some regions.

Nozzle jet flow regions were resolved with approximately 50 cells across the plume. Best results were achieved with uniform mesh size within the nozzle, which were generated with no stretching using a truncated conical volume source (a variation of the cylindrical source). Outside of the nozzle, the plumes were resolved by solution adaptive gridding, which offers more optimal grid point distribution.

D. Solution Adaptive Gridding

Solution adaptive gridding was accomplished by the CraftTech CRISP CFD[®] code¹⁸. CRISP CFD[®] is a stand-alone tool for mesh modification and quality improvement of three-dimensional mixed-element unstructured meshes. Meshes comprised of tetrahedral, prismatic, and hexahedral regions may be readily modified to generate more accurate flow solutions through local refinement and coarsening. The adaption is driven by either an estimate of the solution error, or gradients of flow features. For tetrahedral cells as used in the present work, local refinement is achieved by means of a constrained Delaunay refinement algorithm combined with a circumcenter point placement strategy. Any inconsistency between the circumradius of a tetrahedron and some desired point spacing triggers the point insertion procedure. This iterative cell refinement is repeated until the cell circumradii are consistent with the prescribed point spacing. Coarsening of the tetrahedral region is also permitted through an edge collapse procedure. In regions where the grid is distorted or where solution errors are negligible, edges may be selected for removal. All cells incident to the deleted edge are removed from the mesh, the adjacent cells are redefined, and the two nodes of the edge are collapsed to a single vertex. Capability exists to modify the viscous layers of tetrahedral grids and underlying surface mesh. However, our initial experience has led us to only adapt in the inviscid region of the grid at this time.

IV. Enhancements to Flow Solver

The transition of USM3D from a research-oriented code to a production-oriented capability within a short time frame presented a challenge. The applications for the Constellation program required high throughput on very large grids. The immediate need for USM3D was to overcome several barriers to large-scale applications such as memory bottlenecks, slow run times, and crippling robustness issues for high-Mach number flows. Furthermore, improvements were required in the jet modeling capability within USM3D to address inadequate jet boundary conditions, difficulties with starting jet flows, and deficiencies of turbulence models for strong jet/aero-interaction flows. These deficiencies prevented any meaningful use of the tool within this project environment.

The following enhancements were made to the flow solver to resolve these shortcomings. While largely driven by the urgent needs of the Constellation program, these enhancements will clearly benefit the broader aeronautics community.

A. Overcoming Barriers

1. *Memory management for large-scale applications*

USM3D offers a large degree of flexibility in selecting the number of processors for running a solution from a global grid and restart file. Earlier versions of the flow solver made extensive use of the collective communication paradigm and supported FORTRAN unformatted grid files with large records. However, these features carried a large memory footprint within the code that impeded scalability of memory. The original implementation of parallel-processing inside USM3D was ideally suited for small to medium size grids (less than 20 million cells) running on shared memory machines, such as NASA Advanced Supercomputing (NAS) Columbia Altix. Unfortunately, this implementation was very inefficient on the NAS Columbia machine for grids over 50 million cells, due to excessive swapping of memory buffers.

The Constellation program presented a growing need to resolve intricate geometric details on a large vehicle as seen in figure 4. Consequently, grids grew from approximately 20 million tetrahedral cells to over 120 million tetrahedral cells. Attempts to run USM3D on such large grids at NAS were further hampered in 2008 by the advent

of distributed-memory machines with smaller memory per core (e.g., the Pleiades supercomputer with Xeon E5472 Harpertown dual quad-core processors). To address this difficulty, several improvements were made to USM3D to reduce its runtime memory. For example, the enhanced flow solver could handle binary grid files and allowed for external pre-processing of grids for a parallel run. Additionally, extensive modifications were made to the flow solver data structures and inter-processor communication strategies to restore memory scalability.

As a demonstration of this enhanced capability, a USM3D Navier-Stokes solution was generated on a civil transport configuration for the 4th AIAA Drag Prediction Workshop (DPW IV) using a grid of 212 million tetrahedral cells and the Spalart Allmaras turbulence model. This solution required a wall time of 4.62 seconds per iteration, using 700 grid partitions and 712 1-GB cores of NAS Pleiades Harpertown nodes. To date, the largest case to be successfully computed on the NAS Pleiades machine with the enhanced USM3D code is on a 623 million-cell mesh around the DPW IV geometry.

2. Runtime Speedup

The increased grid size also resulted initially in a significant increase in solution turnaround time, attributable to higher CPU and inter-processor communication times. In order to overcome this difficulty, USM3D's internal grid pre-processing strategy was revised for a multi-processor run. Additionally, cache characteristics of the flow solver were improved. As a result of these enhancements, the speed up of the latest USM3D is five to six times the 2006 version of USM3D.

The upgraded USM3D has greatly increased its efficiency and productivity for the Constellation program. For example on an Ares I A106 task, the code was used to compute 800 high-quality Navier-Stokes solutions using a 90 million-cell grid. These solutions were generated at a rate of 20 solutions per day using 6,000 CPUs on the NAS Pleiades cluster.

3. Robustness for High-Mach Number Flows

The Apollo Launch Abort System was selected in the precursor studies for developing simulation guidelines for the Orion project LAS database. The intricate details in the Apollo LAS including truss geometry (see fig. 3) coupled with high Mach flows adversely impacted robustness of USM3D solutions and brought the LAS simulation guideline study to a halt.

To address these difficulties, three new inviscid flux schemes, namely, HLLC, AUSM+, and LDFSS were implemented in the flow solver. Additionally, several enhancements were made to the implicit solution procedure. These improvements made it possible to readily obtain converged Apollo Launch Abort System solutions up to Mach 6 using SA and SST turbulence models, as illustrated in figure 5.

B. Modeling of Jet Flows

The Ares series of launch vehicles utilize large numbers of control jets during ascent and stage separations to maintain stable flight. Similarly the Orion LAS utilizes powerful Abort Motor (AM) jets and smaller Attitude Control Motor (ACM) jets during a launch abort that dominate the aerodynamic characteristics of the vehicle. All of these jets operate at very high velocities and temperatures in close proximity to aerodynamic surfaces over a wide range of flight Mach numbers, from subsonic to supersonic. Within such flows, large gradients in pressure and temperature have a profound effect on the development of turbulence for transonic and supersonic jets. For high-speed, high-temperature jet flows, standard turbulence models lack the ability to predict the experimentally observed mixing rate of a shear layer, which affects plume spreading and results in large predictive errors for the aerodynamics of jet/surface interactions. Prior to the Constellation program, USM3D had limited capability for prescribing multiple jets and was deficient in the turbulence models for jet flows. Hence, the following improvements were made.

1. Generalized supersonic boundary conditions

The baseline USM3D flow solver allowed modeling of up to four nozzles using a cumbersome input procedure. In order to increase the productivity of USM3D, a generalized supersonic inflow boundary condition was implemented that not only allowed for an unlimited number of nozzle jets but also obviated the need for painstaking specification of nozzle orientations. These boundary conditions are prescribed on selected patch surfaces for each jet using a simple utility code that generates an additional input file for the flow solver. The approach to defining the jet boundary conditions for the applications to Ares launch vehicles and Orion LAS configurations is described in Appendix A.

2. Initialization of jet flows

Jet plume computations for Ares and Orion flight vehicles involved very high nozzle pressure ratios that presented severe startup problems with USM3D. In the earlier USM3D code, cells could only be initialized to freestream conditions at startup. When a jet boundary surface supporting a very high density, pressure, and

temperature falls adjacent to volume cells initialized to freestream conditions, highly unstable and volatile non-physical numerical instabilities are produced. This was resolved by initializing a subset of cells inside the nozzle or plenum near the jet boundary surface with the approximate jet conditions. A simple scheme was contrived for initializing volume cells using the existing minimum distance function devised for the turbulence models. During preprocessing of the grid, cells within a desired distance from selected surface patches inside the nozzle or plenum are flagged for initialization.

The robustness of this technique is demonstrated in figure 6 on the Apollo Launch Abort System. With a challenging nozzle pressure ratio of 4,200 in a freestream Mach number of 4, the solution readily started and subsequently converged into a strong jet/surface interaction flow.

3. Extended 2-equation turbulence models

The need for better physical modeling of jet flows in the USM3D turbulence models became apparent early during the Orion and Ares campaigns. Several approaches to address this deficiency have been advanced in the literature to modify the turbulence transport equations in a variety of ways. Some of the most proven and simple modifications to two-equation turbulence models have been selected and implemented in the USM3D flow solver¹⁹. Table 1 shows the range of turbulence models in USM3D. As denoted by cross-hatching in table 1, the modifications include the addition of correction terms for compressibility (Comp) and pressure dilatation terms (PD) in the turbulence transport equations for high-speed flows, and the addition of a simple modification to the Boussinesq's closure model coefficient for high-temperature (Temp) jets. Two compressibility correction formulations are available, one from Sarkar and another from Wilcox. An extensive guideline study, yet to be published, was performed on the Orion LAS with Abort Motors and Attitude Control Motors that led to the selection of the SST with Comp+PD corrections as the preferred model for the jet-interaction computations.

**Table 1. Available turbulence models in USM3D ver. 20100611.
(New enhancements¹⁹ denoted by cross-hatching)**

Turbulence Model	Standard	Corrections			DES	Hybrid RANS/ LES
		Comp	Comp+PD	Temp		
SA						
SST						
k-ε linear						
k-ε ARSM (Girimaji)						
k-ε ARSM (SZL)						
k-ω (Wilcox)	2006 and 1988 models					

V. Enhancements for Large-Scale Application

A. Script-based run processes

The effective use of TetrUSS in a production role required more than just improving VGRID and USM3D capabilities. Such tasks necessitate the rapid generation of dozens of grids and hundreds of flow solutions, which inherently present challenges in case management, data management, data analysis and post-processing, and quality control. These are process-related functions that were addressed using various scripting procedures. Scripts were developed to execute and manage processes for Ares and Orion, such as directory management, job submittals, best practices, post processing, data extraction, data plotting, and archiving. Examples will be included in the section on sample applications.

B. Data analysis

The Ares and Orion projects had some specific data analysis requirements that necessitated new processes and some modest development of capability. Process scripts were setup to 1) extract sectional load distributions, 2) subtract surface pressure values to create ΔC_p contours plots, 3) a cross-sectional cutter utility for creating C_p line

plots, and 4) enhanced component force and moment integration capability to quantify loads on small features of the Ares rocket.

1. Sectional Load Distributions

The Ares structural and aeroelastic analyses require the aerodynamic loads to be defined at the launch vehicle centerline. A process was developed to transfer aerodynamic loads from a general three-dimensional unstructured surface mesh to the launch vehicle centerline. The process is based on the discrete data transfer technique developed by Samareh²⁰. This technique guarantees conservation of aerodynamic forces and moments to within machine accuracy. The algorithm is described in appendix B.

2. ΔC_p surface contour plots

Jet interference effects were best discerned using ΔC_p surface contour plots. This involved extracting the difference in contours of surface pressure coefficient, C_p , between jets on and jets off conditions. A simple ΔC_p surface contour process was developed using the TecPlot[®] analysis tool. The result is a color contour map displaying aerodynamic influence from the jets. Examples of this capability will be included with the sample applications in the next Section.

3. Cross-sectional cutter for C_p line plots

Capabilities within TecPlot[®] were exploited to automatically extract surface line plots of C_p . Process-control scripts were developed to create multiple comparison plots with experimental data. Examples of this capability will be demonstrated on the Ares and Orion vehicles in the next Section.

4. Enhanced component force and moment integration

An effort was initiated during the course of Ares I ADAC-2A (A101) computational analysis to establish a procedure to isolate the contribution from each protuberance toward vehicle's total integrated aerodynamic forces and moments. A sample result from this analysis is presented in figure 7 that displays axial force contribution from either an individual or a group of protuberances as a fraction of vehicle's total axial force at various Mach numbers. The computed results in figure 7 readily indicate that among all protuberances, the Booster Deceleration Motors (BDMs), stiffeners/kick rings, and LAS nozzles principally contribute toward the total axial force. It can also be seen that the axial force contribution from BDMs peaks at Mach 0.9 and decreases thereafter with increasing speed. Similar analysis has also been performed for all other computed coefficients. Such a data analysis has aided the design of vehicle protuberances from an aerodynamic perspective.

VI. Sample Applications

Since 2006, an extensive amount of vetted computational aerodynamic and loads data have been delivered to the respective projects for the Ares I, I-X, and V launch vehicles and the Orion CEV and LAS configurations. Over 7,000 USM3D solutions have been generated on the Ares vehicles, and well over 1,000 solutions produced on the Orion configurations. The following describes some typical applications of the TetrUSS toolset to the Ares and Orion vehicles.

A. Ares I, I-X, and V Launch Vehicles

1. Ares I Ascent Aero

The Ares I Design Analysis Cycle (ADAC) outer mold line (OML) definition has evolved since 2006. During this evolution, aerodynamic databases have been developed for at least five major Ares I configurations with wind tunnel testing and CFD. CFD has played an important role in providing data for vehicle aerodynamics during ascent and first-stage-separation phases of a flight. CFD has also provided information on aeroelastic characteristics during the ascent phase. CFD solutions have been obtained using three flow solvers, namely, USM3D, OVERFLOW, and FUN3D. In the first two years of this project, extensive CFD computations were made to establish confidence in the ability of CFD to accurately simulate configurations similar to Ares I. This was accomplished via code-to-code and code-to-wind-tunnel comparisons. USM3D was selected as the primary flow solver for the aerodynamic assessment of Ares I during the ascent phase of flight. OVERFLOW was used to analyze aerodynamics of first-stage separation, whereas FUN3D was used to analyze aeroelastic characteristics of the vehicle. USM3D results have played a multi-faceted role in the Ares I ascent aerodynamic studies as described below.

Due to inherent limitation of wind tunnel measurements associated with a slender geometry, comprehensive measurements of surface pressures is very difficult. Therefore, USM3D CFD results were anchored at selected locations and conditions with experiment and then used to provide more detailed vehicle surface pressures and sectional line loads. These data have been used by various groups for venting and debris analysis and load analysis for the Orion capsule and individual launch vehicle protuberances. Since early 2007, majority of USM3D results were completed prior to any wind tunnel testing. These predicted results aided wind tunnel model design,

protuberance fabrication, test matrix development, and instrumentation in the region of major flow variations. The USM3D predictions were validated against subsequently available wind tunnel test measurements²¹⁻²³. As shown in the following, and in reference 24, good correlation between USM3D predictions and wind tunnel data further added confidence in the validity of the CFD results. Consequently, USM3D results were utilized in a number of trade studies to assess aerodynamic effects due to geometrical modifications and repositioning of the existing protuberances and addition of new ones to improve vehicle performance.

The Guidance, Navigation, and Control (GN&C) aerodynamic database was primarily developed from the wind tunnel data on the 1-percent model due to its ability to generate a large quantity of data in a timely manner. The USM3D force and moment results were utilized to provide flight Reynolds number increments to the wind-tunnel derived database as described in reference 25.

USM3D was also used to generate the aerodynamic database for jet interactions from the Roll Control System (RoCS) on the Ares I full protuberance configuration. All USM3D computations were based upon ideal gas modeling of the flow. The baseline computational grid for this study was constructed following the best practices for the Ares I ascent power-off condition. Approximately 2 million grid cells surrounding each of the 12 RoCS thrusters were added to provide adequate grid resolution to support plume development and interactions with the free-stream flow. Reference 26 provides more details about this study.

During the entire Ares I project, USM3D solutions for the ascent phase of the vehicle were based on tetrahedral grids that ranged from 10 million cells for the clean configuration at wind tunnel conditions to 90 million cells for full protuberance RoCS configuration at flight conditions. A converged solution typically needed about 20,000 iterations and 5,000 CPU hours on NAS Pleiades Harpertown cluster. Nearly 1,000 USM3D solutions were generated within two months for the latest Ares I configuration (ADAC-3 A106) to satisfy all project requirements for CFD data. Extensive use of scripts was made to set up (1) input stream for various freestream Mach numbers (2) grid rolling necessary for generating solutions at various roll angles (3) run directories for various solutions (4) user alerts for solution status, and (5) data extraction and graphical reduction of results. Use of scripts resulted in hands-off production of data in support of project requirements. Sample validation results from these studies are presented below.

Figure 8 demonstrates convergence history for the Ares I A106 USM3D solution obtained at Mach 1.6, an angle of attack of 8°, a roll angle of 0°, and at wind tunnel Reynolds number. All the six components of forces and moments around the vehicle geometry were converged to within a certain percentage of their average values over the preceding 2,000 iterations. A value of 0.1% was used for the axial as well as normal forces and pitching moment whereas a value of 0.5% was used for the side force and rolling as well as yawing moments. As seen from the figure, lateral forces and moments took longer to achieve prescribed convergence criterion. The results shown in the figure 8 have been automatically generated using scripts.

The Ares I ADAC-2B (A103) configuration has been tested at three wind tunnel facilities²¹⁻²³ covering low subsonic to high supersonic flow conditions and low to high Reynolds number. A sample comparison of a USM3D solution with wind tunnel measurements for this configuration at Mach 1.6, an angle of attack of 7°, and low Reynolds number wind tunnel conditions is presented in figure 9. Comparison of computed and measured surface pressures along the top centerline and around one of the largest protuberances (LH2) on the vehicle is presented in the figure. USM3D validation results for the same configuration using transonic high Reynolds number wind tunnel test²³ data can be found in reference 24. Assessment of USM3D results at flight Reynolds number is presented later.

Comparison of the sectional normal load distributions over the Ares I A103 clean configuration as obtained from USM3D and wind tunnel data at various Mach numbers is presented in figure 10. As is evident from the figure, USM3D predicted the sensitivity of sectional loads to the changes in free stream Mach number. Sectional load extraction utility described earlier was used to derive sectional load variations along the stream wise direction.

2. Ares I-X Flight Test Vehicle Configuration

The Ares I-X is the first Flight Test Vehicle (FTV) for Ares I. For the Ares I-X FTV, USM3D flow solver was used to conduct all pre-flight simulations that were incorporated into the forces and moment databases. The merged liftoff, ascent and high angle-of-attack aero and line loads databases provide the basis for the GN&C algorithms and were used to calculate the trajectory and the footprint for the recovery of the first stage as well as for the loss of control or abort cases.

The baseline configuration for Ares I-X FTV was Ares I ADAC-2A (A101) full protuberance configuration for which a wind tunnel derived database existed. USM3D solutions were used to account for the changes in configuration between A101 and Ares I-X FTV and to generate an Ares I-X FTV database at wind tunnel conditions. Additional USM3D solutions were generated at Ares I-X FTV flight conditions that provided Reynolds number increments between wind tunnel and flight. The latter increments were included in the dispersion bounds

that were provided to the GN&C team. The RoCS Jet Interaction (JI) flight database was developed from hundreds of USM3D solutions.

The generation of Ares I-X FTV aerodynamic database, and comparison of pre-flight computations and flight data are detailed in reference 27. A sample result from this effort is presented in figure 11 that shows longitudinal surface pressure distributions near the front of the vehicle at four azimuth locations. The USM3D predictions were obtained using two different turbulence models, namely, SA and SST. The computed pressures from two models are nearly identical. The results show a very good correlation between USM3D predictions and flight measurements.

Over the entire Ares I-X project, more than 5,000 USM3D solutions have been generated based on grids that consisted of 70-140 million tetrahedral cells. All of these solutions have been generated on NAS Pleiades and Altix supercomputers. A typical Ares I-X solution was started and run with spatially first-order differencing for a prescribed number of iterations, then switched to globally second-order differencing for the remainder of the computation. With most numerical flow solvers, strong shocks and sharp corners can often induce locally non-physical phenomenon such as spurious entropy, negative pressures, or excessive pressure levels. For the Ares I-X computations, cells with pressure dropping below a minimum threshold of 0.0014 of freestream pressure or exceeding a maximum threshold of 20-percent above freestream total pressure were locally switched to first-order differencing to cushion any potential instability.

3. Ares V Cargo Launch Vehicle

The Ares V launch system is in an early stage of planning and design with attendant revisions to vehicle configuration. Due to limited availability of wind tunnel data, CFD solvers are being utilized to generate initial aerodynamic databases for the Ares V project. In this effort, OVERFLOW is the primary flow solver whereas USM3D is lending a supporting role providing approximately 40% of data. Reference 28 provides details of database development and comparisons of CFD and wind tunnel results.

USM3D is also being used to establish best practices for investigating Plume Induced Flow Separation (PIFS) characteristics along the future Ares V heavy lift vehicle. In this precursor study²⁹, USM3D is used to compute PIFS along the Saturn V heavy lift launch vehicle with F-1 rockets firing. Flight test PIFS data at four supersonic Mach numbers are available for this configuration. The USM3D solutions have been computed at the corresponding supersonic freestream speeds, with angle of attack of 0°, sideslip of 0°, and flight Reynolds number. In this study, sensitivity of the computed results to grid resolution, turbulence models, and engine exit boundary condition specification approaches is evaluated. As an illustration, the Mach contours along the Y=0 plane intersecting the Saturn V vehicle is shown in figure 12a at the Mach = 6.5 condition²⁹. PIFS can be seen up the side of the vehicle from the nozzles by the blue contours of low Mach number flow. The Mach contours along the Y=0 plane intersecting the Ares V vehicle are shown in figure 12b for the Mach 6.5 condition.

B. Orion CEV Launch Abort System

The aerodynamic database development of the Orion Launch Abort System (LAS) is led out of NASA Johnson Spaceflight Center (JSC) under the CEV Aerosciences Project (CAP) that provides multi-center coordination for wind tunnel testing (WTT) and computational support. TetrUSS/USM3D is contributing to CAP by investigating various sensitivities, such as WTT sting interference effects, WTT-to-Flight Reynolds number scaling with Abort Motors (AM), AM thrust offset, and Abort-Motor/Attitude-Control-Motor (AM/ACM) jet interactions. Each of these tasks required a delivery of hundreds of flow solutions with stringent data requirements and very tight schedules. Hence, the TetrUSS enhancements discussed earlier were critical for delivering a quality product. The following will describe a typical task within the CAP environment.

A study was conducted to quantify WTT-to-Flight Reynolds number scaling for the LAS 60-AA WTT configuration with powered AM jets (fig. 13). This task required the delivery of 112 USM3D simulations of the Orion LAS with AM jets firing, in configurations representing both free-flight and the 60-AA WTT. Data covered a Mach range from 0.7 to 2.5 and angles of attack from 0° to -16° in 4° increments to match the available wind tunnel data. The objective was to produce data for evaluating USM3D flow simulations of WTT AM jet increments and absolute magnitudes through comparison with wind tunnel data and similar computations performed using Cart3D and OVERFLOW. In addition, the analysis of WTT and Flight conditions provided WTT-to-Flight increments that were used for database corrections and uncertainty evaluations. As in all of the tasks, the computational data had to be post-processed and delivered under specific requirements for archiving. This was accomplished using numerous scripting processes.

Four baseline grids were generated for this task. Typical surface triangulations are shown in figure 14. Two grids were required for the vehicle with WTT AM nozzle inner mold lines (IML), and two for the IML of the Flight AM nozzle. Based on prior guideline studies, a finer baseline grid of approximately 25 million cells was required for the

transonic Mach conditions of 0.9 and 1.05, and a coarser grid of approximately 20 million cells was needed for all other Mach numbers. A crosscut of the volume cells of the WTT coarse grid is shown in figure 15. Note that the cells tend to be smaller close to the geometry and coarsen in the field away from the surface, which is characteristic of the Advancing Front Method. This grid does not provide adequate resolution to model the AM jet plumes and vehicle base flows. Hence, the CRISP CFD[®] grid adaption code was used to refine the grid around errors of density, velocity, and pressure as illustrated in figure 16 for Mach 0.9 with the AM jet firing with a thrust coefficient, $CT=4$. Note the significant clustering of field cells in the AM jet plume and vehicle wake regions. For this case, the 25 million cell baseline grid was adapted resulting in an increase to 34 million cells.

Prior to the Orion LAS AM and ACM jet tasks, an extensive guideline study was performed to develop best practices and estimate solution uncertainty. A wide range of strategies for creating the baseline grids, adapting the grid, and turbulence models were explored. A representative analysis of the effect of turbulence model is presented in figure 17 for the LAS 60-AA configuration with $CT=4$ and one adaptive grid strategy. This plot compiles a weighted statistical composite of how much each solution deviated from the experimental data according to:

$$\Delta = \sum_{i=1}^5 w_i |C_{X,USM3D,i} / C_{X,exp,i} - 1| / \sum_{i=1}^5 w_i$$

where

$$C_X = C_N, C_A, C_L, C_D, C_M \text{ and } w_i = 1, 1, 1, 1, 2$$

The pitching moment C_M was weighted twice as heavily as the other coefficients because it carried more importance and uncertainty in the aero database development. From figure 17, the SST Sarkar Comp + PD was selected as the turbulence model of choice having the least statistical deviation from experiment. Note also the large deviations at Mach 0.9. Wind tunnel Schlieren imagery suggests that this deviation was most likely caused by strong wall-interference effects in the test data at transonic conditions.

The best practices developed from the guideline study were captured in script-based run processes designed to execute and manage all runs. These scripts provided complete hands-off execution of preprocessing, job submittal, adapting the grid, terminating solution at convergence, post-processing and plotting results, and archiving the solutions. The following script-controlled process was developed for LAS jet flow tasks:

1. Run 1st-order USM3D flow solution on baseline grid for 3,000 iterations using SST Sarkar Comp + PD turbulence model.
2. Perform grid adaptation using CraftTech CRISP CFD[®] code.
 - a. Adapt to errors of density, velocity, and pressure (ρ, u, v, w, p)
 - b. Constrain minimum inviscid spacing at 1.0 inches (full scale)
 - c. Interpolate baseline solution onto adapted grid
3. Restart interpolated solution on adapted grid with 1st-order scheme for 1,500 iterations.
4. Restart 1st-order solution with 2nd-order scheme.
5. Automatically stop 2nd-order solution when the following convergence criteria are met.
 - a. Variation of C_L, C_D, C_M, C_N , and C_A less than 1-percent over last 1,000 iterations
 - b. Average of last 1,000 iterations within 1-percent of average of preceding 1,000 iterations
 - c. Absolute difference in the minimum and maximum values of the coefficients over last 1,000 iterations less than 0.004
 - d. Residual errors decrease by 3 orders of magnitude and remain constant or keep decreasing

Extensive analysis and convergence plots were generated during post-processing. Figure 18 shows a typical crosscut of Mach contours with AM jets on and off, along with the corresponding surface C_p contours. Also shown is a $\Delta C_p = C_{p,AM_ON} - C_{p,AM_OFF}$ surface contour plot which illuminates the jet interactions. Similarly, C_p line plots were created and compared with WTT data as represented in figure 19.

The adapted grids in this study ranged from 26- to 30-million cells, with the exception of the grids for transonic Mach 0.9 and 1.05 that ranged from 32- to 35-million cells. The solutions were run on the NAS Columbia Altix computer typically with 128 processors. Resource usage ranged from 1,000- to 3,800-CPU hours per solution, with an average of 2,100-CPU hours per solution for the 112 cases.

VII. Concluding Remarks

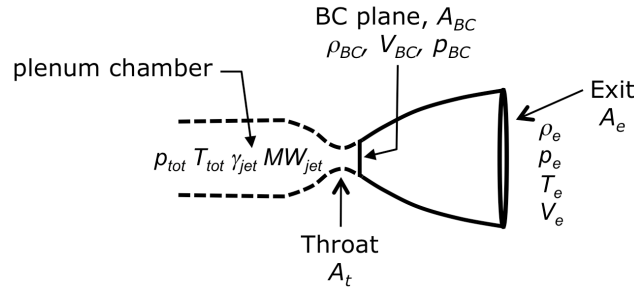
The NASA Constellation program is relying heavily on Computational Fluid Dynamic (CFD) tools that complement ground-based wind tunnel testing by aiding the design and aerodynamic database development for the Ares I Crew Launch Vehicle (CLV), Orion Crew Exploration Vehicle (CEV) and Launch Abort System (LAS). The TetrUSS system serves as the lead CFD tool within the Ares project, providing over 7,000 solutions to develop the computational aerodynamic database and loads for the vehicles up through supersonic speeds. The USM3D solver is also lending critical support to the aerodynamic development of the CEV and its LAS configuration with over 1000 solutions, but only after the implementation of considerable enhancements.

The fast-paced application-driven environment has presented many challenges and opportunities to implement required capabilities within the TetrUSS system. Key enhancements added to the VGRID grid generator and its application practices enabled the routine generation of numerous viscous tetrahedral grids between 30 and 100+ million cells on the LAS and Ares flight vehicles. A commercial tool for autonomous adaption of the grid to the flow solution was seamlessly folded into processes for computing strong jet interaction flows on the LAS. Memory bottlenecks that limited USM3D application to grids in the 20 to 30 million-cell ranges were removed from the flow solver to extend its range to grids up to 623 million cells. Improvements were made to coding practices that led to code execution at least 5- to 6-times faster than a 2006 version. Robustness of computing the strong jet interactions of the LAS was enabled by the implementation of new flux schemes, better linearization, and new initialization functions. The accuracy of the jet interaction flows was improved by the implementation of compressibility and temperature corrections within several two-equation turbulence models. Scripting processes were developed to execute and manage thousands of flow solutions. Data analysis capability was developed for extracting section load distributions, ΔC_p surface contour plots, and organizing extensive component loads.

Over the past five years, this tool enhancement effort has largely been successful because of the close day-to-day collaboration between in-house code experts and the project application experts. The impact of the enhanced TetrUSS system on the Constellation program has been substantial. Although largely driven by the urgent needs of the Constellation program, these enhancements will clearly benefit the broader aeronautics community.

Appendix A – Method for Prescribing Jet Boundary Conditions

The methodology for prescribing the Ares and Orion LAS jet boundary conditions in ideal-gas flow solvers presented below was provided via Excel spreadsheet by Mr. James Greathouse at NASA Johnson Spaceflight Center. The jet boundary conditions (BC) are defined on a plane within the nozzle slightly down stream of the throat as illustrated in the sketch below. A general assumption of one-dimensional isentropic flow is applied to determine the flow conditions at the nozzle exit plane, and then transfer them to the BC plane.



Schematic of nozzle jet boundary condition prescription.

The relations for one-dimensional isentropic flow through a nozzle are

$$\frac{A}{A^*} = \frac{1}{M} \left[\frac{2}{\gamma+1} \left(1 + \frac{\gamma-1}{2} M^2 \right) \right]^{\frac{\gamma+1}{2(\gamma-1)}} \quad (A1)$$

$$\frac{T_{tot}}{T} = 1 + \frac{\gamma-1}{2} M^2 \quad (A2)$$

$$\frac{p_{tot}}{p} = \left(\frac{T_{tot}}{T} \right)^{\frac{\gamma}{\gamma-1}} \quad (A3)$$

$$\frac{\rho_{tot}}{\rho} = \left(\frac{T_{tot}}{T} \right)^{\frac{1}{\gamma-1}} \quad (A4)$$

The following process is used to determine the jet boundary conditions on the BC plane.

1. Prescribe initial conditions and parameters:

a. Freestream conditions

$$M_\infty, p_\infty, T_\infty, \rho_\infty, q_\infty, a_\infty = \sqrt{\gamma_{air} p_\infty / \rho_\infty}, S_{ref}, \gamma_{air}, \text{ and } R_{air}$$

b. Jet plenum gas properties

$$\gamma_{jet}, R_{jet}, \text{ and } T_{tot,jet}$$

c. Geometric areas for nozzle exit, throat, and BC plane

$$A_e, A_t, \text{ and } A_{BC}$$

2. Expand plenum chamber conditions to nozzle exit plane using jet gas properties

a. Using the prescribed value of γ_{jet} , adjust nozzle exit Mach number, $M_{e,jet}$, in equation A1 to match the choked area ratio with the geometric area ratio,

$$(A_e/A^*)_{jet} = A_e/A_t$$

⇒ This defines $M_{e,jet}$

b. Compute isentropic relations at nozzle exit by substituting γ_{jet} and $M_{e,jet}$ into equations A2-A4

$$(p_{tot}/p_e)_{jet} \quad (T_{tot}/T_e)_{jet} \quad (\rho_{tot}/\rho_e)_{jet}$$

3. Calculate nozzle exit plane conditions assuming jet gas properties

a. Define a target nozzle thrust coefficient

$$C_{T,target} = target_thrust / q_\infty S_{ref}$$

b. Adjust plenum chamber pressure, p_{tot} in thrust equation

$$C_T = [p_{tot} / (p_{tot} / p_e)_{jet} \cdot A_e (1 + \gamma_{jet} M_{e,jet}^2 \cdot \cos(25^\circ))] / (q_\infty S_{ref})$$

until C_T matches $C_{T,target}$. The $\cos(25^\circ)$ imposes the C_T of each offset abort nozzle (see fig. 16) in the body axial direction.

⇒ This determines p_{tot} for the jet conditions

c. Complete the nozzle exit conditions from known quantities

$$p_{e,jet} = p_{tot} / (p_{tot} / p_e)_{jet}$$

$$T_{e,jet} = T_{tot} / (T_{tot} / T_e)_{jet}$$

$$\rho_{e,jet} = p_{e,jet} / (R_{jet} T_{e,jet})$$

$$V_{e,jet} = M_{e,jet} * \sqrt{\gamma_{jet} R_{jet} T_{e,jet}}$$

4. Switch from *jet* gas to *air* and recalculate chamber conditions. The general assumption is to maintain the thrust. Since thrust is a function of exit-plane pressure and jet momentum, the values nozzle exit pressure, density, and velocity will remain unchanged. The primary impact of switching gas from *jet* to *air* is to recompute the nozzle exit temperature. This directly impacts the speed of sound and exit Mach number, which in turn alters the values of the isentropic relations in equations A2-A4 and the total conditions.

- a. Adjust nozzle exit jet temperature and Mach number for air

$$T_{e,air} = p_{e,jet} / (R_{air} \rho_{e,jet})$$

$$M_{e,air} = V_{e,jet} / \sqrt{\gamma_{air} R_{air} T_{e,air}}$$

- b. Assign remaining jet properties to nozzle exit as air

$$\rho_{e,air} = \rho_{e,jet}$$

$$p_{e,air} = p_{e,jet}$$

$$V_{e,air} = V_{e,jet}$$

- c. Perform sanity check of thrust by comparing new $C_{T,air}$ with $C_{T,target}$

$$C_{T,air} = [p_{e,air} \cdot A_e (1 + \gamma_{air} M_{e,air}^2 \cdot \cos(25^\circ))] / (q_\infty S_{ref})$$

- d. Compute new isentropic relations at nozzle exit using equations A2-A4 with air properties γ_{air} and exit Mach number, $M_{e,air}$

$$(p_{tot}/p_e)_{air} \quad (T_{tot}/T_e)_{air} \quad (\rho_{tot}/\rho_e)_{air}$$

- e. Compute for total jet conditions with assumed air

$$p_{tot,air} = p_{e,air} \cdot (p_{tot}/p_e)_{air}$$

$$T_{tot,air} = T_{e,air} \cdot (T_{tot}/T_e)_{air}$$

$$\rho_{tot,air} = \rho_{e,air} / (\rho_{tot}/\rho_e)_{air}$$

5. Calculate A/A^* at the BC plane just down stream of the throat

- a. Use equation A1 to compute new $(A_e/A^*)_{air}$ using γ_{air} and $M_{e,air}$

- b. Compute new choked throat area A^* is

$$A^* = A_e / (A_e/A^*)_{air}$$

6. Propagate the exit plane conditions to the BC plane

- a. Using γ_{air} , adjust Mach number on the BC plane, $M_{BC,air}$, in equation A1 until its $(A/A^*)_{BC}$ matches A_{BC}/A^*

\Rightarrow This defines $M_{BC,air}$

- b. Substitute γ_{air} and $M_{BC,air}$ into isentropic relations equations A2-A4 to compute BC-plane ratios

$$(p_{tot}/p_{BC})_{air} \quad (T_{tot}/T_{BC})_{air} \quad (\rho_{tot}/\rho_{BC})_{air}$$

- c. Compute the BC-plane properties

$$p_{BC} = p_{tot,air} / (p_{tot}/p_{BC})_{air}$$

$$T_{BC} = T_{tot,air} / (T_{tot}/T_{BC})_{air}$$

$$\rho_{BC} = \rho_{tot,air} / (\rho_{tot}/\rho_{BC})_{air}$$

$$V_{BC} = M_{BC} \cdot \sqrt{\gamma_{air} R_{air} T_{BC}}$$

- d. Compute USM3D non-dimensional BC values using freestream conditions:

$$\rho_{usm3d} = \rho_{BC} / \rho_\infty$$

$$V_{usm3d} = V_{BC} / a_\infty$$

$$p_{usm3d} = p_{BC} / (\gamma_{air} p_\infty)$$

7. Apply jet boundary conditions

- a. Directly apply ρ_{usm3d} and p_{usm3d} across BC plane

- b. Smoothly fan the velocity vectors from the center of the BC plane until parallel to the nozzle walls. Maintain nozzle axis component of V_{usm3d} constant across plane.
- c. Initialize the tetrahedral volume cells within the nozzle to approximate the conditions on the BC plane to facilitate startup.

Appendix B – Launch Vehicle Line Load

The Ares structural and aeroelastic analyses use line load coefficients that are defined as the axial derivatives of force coefficients (e.g., dC_N/dx) on a set of user-defined points along the vehicle centerline. The launch vehicle surface is defined by a set of triangles, and CFD results are provided as pressure coefficients and skin friction coefficients at each node on the launch vehicle surface. The algorithm described here maintains conservation of forces and moments: the integration of line load will reproduce the CFD aerodynamic force and moment coefficients within machine accuracy.

The line load coefficients are required at each user-defined point (i), which is surrounded by a control volume (Δx_i) defined as:

$$\Delta x_1 = \frac{x_1 - x_2}{2}, \Delta x_i = \frac{x_{i-1} - x_{i+1}}{2}, \Delta x_N = \frac{x_{N-1} - x_N}{2},$$

$$s_1 = \frac{3x_1 + x_2}{4}, s_i = \frac{x_{i-1} + 2x_i + x_{i+1}}{4}, s_N = \frac{x_{N-1} + 3x_N}{4}$$

where s_i represents the center of control volume surrounding node i (see fig. B1). The integrated aerodynamic force and moment coefficients (ΔC) are calculated for each control volume (Δx_i). This is a computationally intensive process that requires cutting surface mesh at discrete set of planes to calculate loads for triangles within the control volume and triangles partially cut by the control volume boundaries. The aerodynamic line load coefficients are then calculated as:

$$\left(\frac{dC}{dx} \right)_i^{\text{CFD}} = \left(\frac{\Delta C}{\Delta x} \right)_i^{\text{CFD}}$$

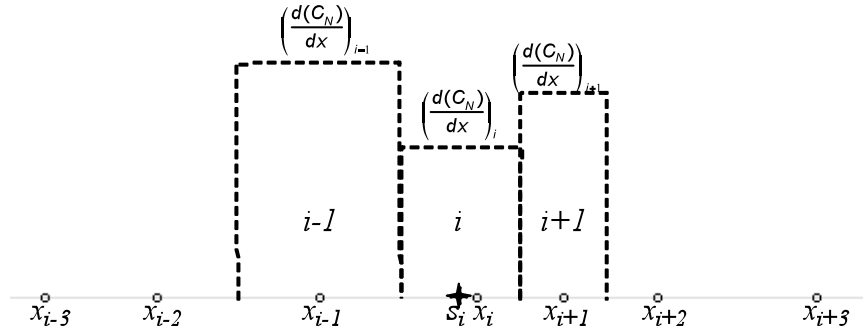


Figure B1. User-defined point distribution and corresponding control

Figure B2 shows a sample result for an Ares-I configuration. The blue line shows vehicle centerline with diamond symbols for the locations of user-defined points. Black line shows the outline of launch vehicle shape. The red line shows the variation of line load for normal force coefficient, C_N .

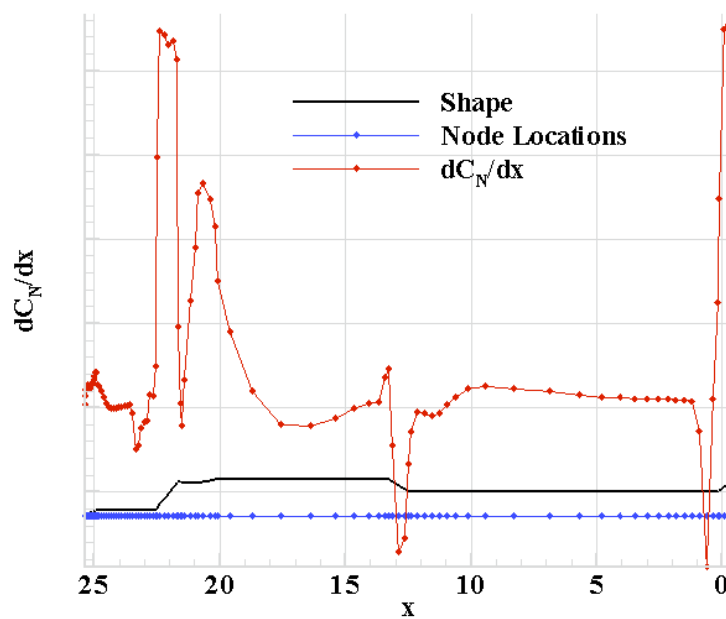


Figure B2. Sample line load coefficients

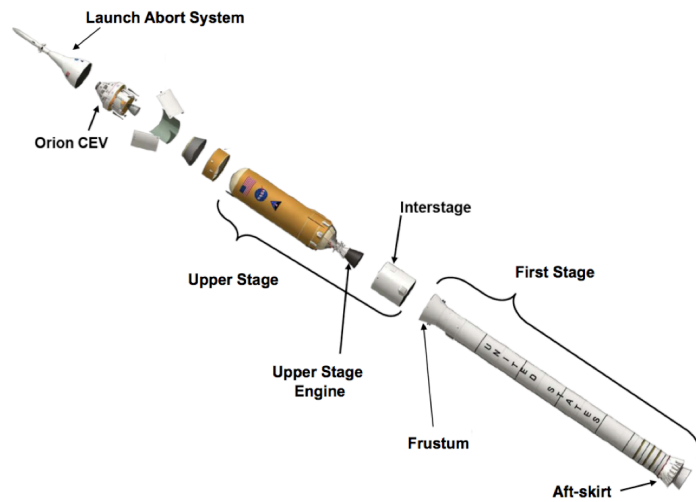
Acknowledgments

The authors would like to acknowledge following individuals at NASA LaRC for their contributions to the work reported herein. The authors wish to sincerely recognize several enhancements made by the late Shahyar Pirzadeh to VGRID that were crucial in the application of TetrUSS to the Constellation program. The authors would also like to thank Paul Pao for developing a solution cutter utility and Steven Massey for developing solution interpolation utility and scripts to generate and process multiple solutions. The authors wish to thank Steve Bauer, William Compton, Karen Deere, Eric Ding, Alaa Elmilgui, Farhad Ghaffari, Johannes Kerimo, Steve Krist, and Susan Mcmillin for their feedback leading toward TetrUSS improvements. The first author would like to acknowledge his former employer, Analytical Services and Materials, Inc., Hampton, VA, for providing support during the execution of this study.

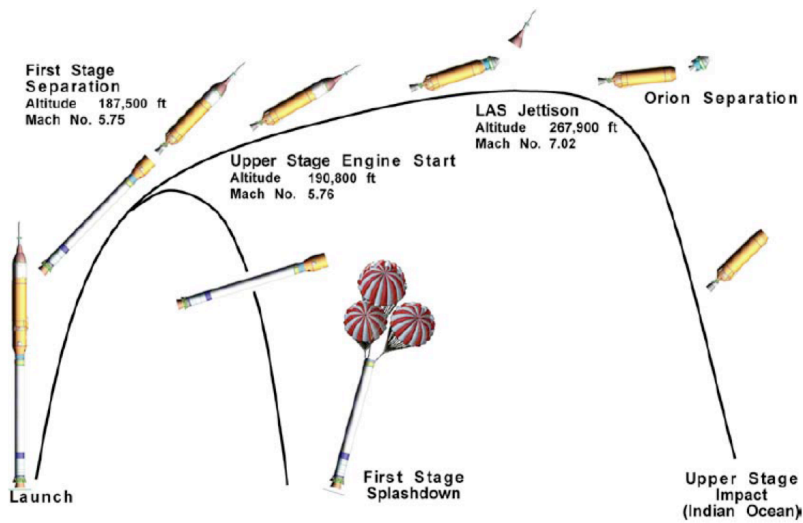
References

- ¹NASA, "NASA's Exploration Systems Architecture Study," NASA-TM-2005-214062, November 2005.
- ²Huebner, L. D., Smith, R. M., Campbell, J. R., and Taylor, T. L., "Ares-IX Flight Vehicle Similitude to the Ares I Crew Launch Vehicle," IAC-08-D2.6.7, 59th International Astronautical Congress, September 29-October 3 2008, Glasgow United Kingdom.
- ³Frink, N. T., Pirzadeh, S. Z., Parikh, P. C., Pandya, M. J., and Bhat, M. K., "The NASA tetrahedral unstructured software system (TetrUSS)," *The Aeronautical Journal*, Vol. 104, No. 1040, 2000, pp. 491-499.
- ⁴Frink, N. T., "Upwind scheme for solving the Euler equations on unstructured tetrahedral meshes," *AIAA Journal*, Vol. 30, No. 1, 1992, pp. 70-77.
- ⁵Frink, N. T., "Tetrahedral unstructured Navier-Stokes method for turbulent flows," *AIAA Journal*, Vol. 36, No. 11, 1998, pp. 1975-1982.
- ⁶Nichols, R. H., Tramel, R. W., and Buning, P. G., "Solver and Turbulence Model Upgrades to OVERFLOW 2 for Unsteady and High-Speed Applications," AIAA Paper 2006-2824.
- ⁷Alexandrov, N. M., Atkins H. L., Bibb K. L., Biedron R. T., Carpenter, M. H., Gnoffo P. A., Hammond D. P., Jones W. T., Kleb W. L., Lee-Rausch E. M., Nielsen E. J., Park M. A., Raman, V. V., Roberts T. W., Thomas J. L., Vatsa V. N., Viken S. A., White J. A., Wood W. A., "Team Software Development for Aerothermodynamic and Aerodynamic Analysis and Design", NASA TM-2003-212421, November 2003.
- ⁸Aftosmis, M. J., Berger, M. J., and Adomavicius, G., "A Parallel Multilevel Method for Adaptively Refined Cartesian Grids with Embedded Boundaries," AIAA Paper 2000-0808, January 2000.
- ⁹Campbell, R. L., "Efficient Viscous Design of Realistic Aircraft Configuration (Invited)," AIAA Paper 1998-2539.
- ¹⁰Lohner, R. and Parikh, P., "Three-dimensional grid generation by the advancing front method," *International Journal for Numerical Methods in Fluids*, Vol. 8, Issue 10, 1988, pp. 1135-1149.

- ¹¹Pirzadeh, S. Z., "Three-dimensional unstructured viscous grids by the advancing layer method," *AIAA Journal*, Vol. 33, No. 1, 1996, pp. 43-49.
- ¹²Samareh, J., "GridTool: A Surface Modeling and Grid Generation Tool," Proceedings of the Workshop on Surface Modeling, Grid Generation, and Related Issues in CFD Solutions, NASA CP-3291, 9-11 May, 1995.
- ¹³Pandya, M.J., Frink, N.T., Abdol-Hamid, K.S., and Chung, J.J., "Recent enhancements to USM3D unstructured flow solver for unsteady flows," AIAA Paper 2004-5201, August 16-19 2004.
- ¹⁴Abdol-Hamid, K. S., Frink, N. T., Deere, K. A., and Pandya, M. J., "Propulsion Simulations Using Advanced Turbulence Models with the Unstructured-Grid CFD Tool, TetrUSS," AIAA Paper 2004-0714, January 2004.
- ¹⁵Pandya, M.J., Frink, N.T., and Noack, R.W., "Overset-grid moving body capability in the USM3D unstructured flow solver," AIAA Paper 2005-5118.
- ¹⁶Pao, S. P., "USMC – A TetrUSS Grid and Solution Cutter," User's Guide, Configuration Aerodynamics Branch, NASA Langley Research Center, February 2009.
- ¹⁷Pirzadeh, S. Z., "Advanced unstructured grid generation for complex aerodynamic applications," *AIAA Journal*, Vol. 48, No. 5, 2010, pp. 904-915.
- ¹⁸Cavallo, P., Grismer, M., "A Parallel Adaptation Package For Three-Dimensional Mixed-Element Unstructured Meshes", *Journal of Aerospace Computing, Information, and Communication* 2005, 1542-9423 Vol. 2, No. 11, (433-451) doi: 10.2514/1.16220.
- ¹⁹Pandya, M. J., Abdol-Hamid, K. S., and Frink, N. T., "Enhancement of USM3D unstructured flow solver for high-speed high-temperature shear flows," AIAA Paper 2009-1329.
- ²⁰Samareh, J. A., "Discrete Data Transfer Technique for Fluid-Structure Interaction," AIAA Paper 2007-4309.
- ²¹Pinier, J. T., Niskey, C. J., Hanke, J. L., and Tomek, W. G., "Ares I Aerodynamic Testing at the Boeing Polysonic Wind Tunnel," AIAA Paper 2011-0998.
- ²²Erickson, G. E., and Wilcox, F. J., "Ares I Aerodynamic Testing at the NASA Langley Unitary Plan Wind Tunnel," AIAA Paper 2011-0999.
- ²³Jacobs, P. F., Chan, D. T., Capone, F. J., and Hanke, J. L., "0.01-Scale CLV ADAC-2B (A103) Pressure Model Testing in the NASA Langley Research Center National Transonic Facility from Mach 0.5 to Mach 1.08", ARES-AD-TA-0015, August 2008.
- ²⁴Abdol-Hamid, K. S., Ghaffari, F., and Parlette, E. B., "Overview of Ares-I CFD Ascent Aerodynamic Data Development and Analysis Based on USM3D," AIAA Paper 2011-0015.
- ²⁵Pamadi, B. N., Pei, J., Covell, P. F., Favaregh, N. M., and Gumbert, C. R., "Aerodynamic Analyses and Database Development For Lift-Off, Transition and First Stage Ascent of the Ares I A106 Vehicle," AIAA Paper 2011-0012.
- ²⁶Deere, K. A., Pao, S. P., and Abdol-Hamid, K. S., "Computational Analysis of Ares I Roll Control System Jet Interactions on Rolling Moment," AIAA Paper 2011-0172.
- ²⁷Bauer, S. X., Krist, S. E., and Compton, W. B. "Generation of the Ares I-X Flight Test Vehicle Aerodynamic Data Book and Comparison To Flight", AIAA Paper 2011-0011.
- ²⁸Kiris, C. C., Housman, J., Gusman, M., Schauerhamer, D., Deere, K. A., Elmilgui, A. A., Abdol-Hamid, K. S., Parlette, E., Andrews, M., Blevins, J., "Best Practices for Aero-Database CFD Simulations of Ares V Ascent," AIAA Paper 2011-0016.
- ²⁹Deere, K. A., Elmilgui, A. A., and Abdol-Hamid, K. S., "USM3D Simulations of Saturn V Plume Induced Flow Separation", AIAA Paper 2011-1055.



a)- Major components.



b)- Nominal flight trajectory (from ref. 2).

Figure 1. Major components and nominal flight trajectory of Ares-I and Orion flight vehicles.

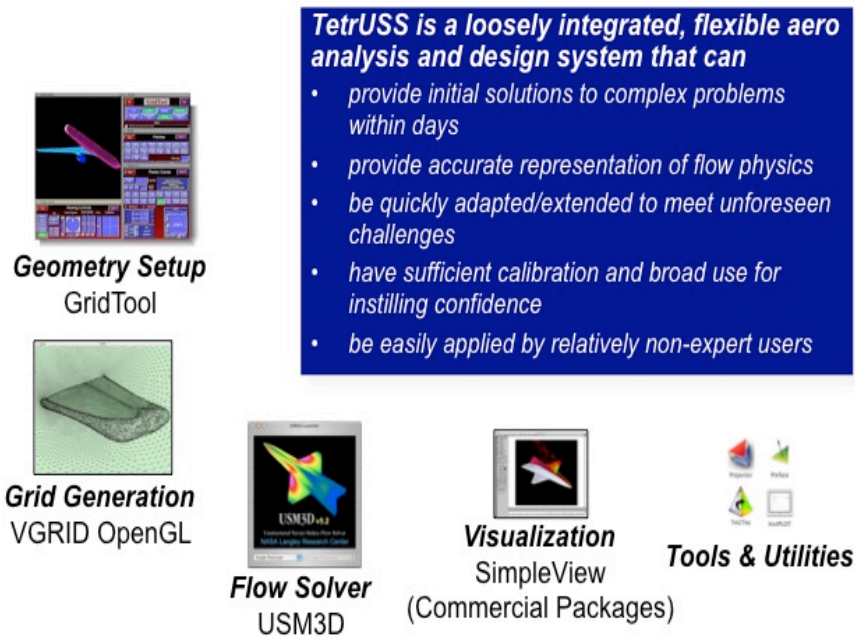


Figure 2. The Tetrahedral Unstructured Software System (TetrUSS).

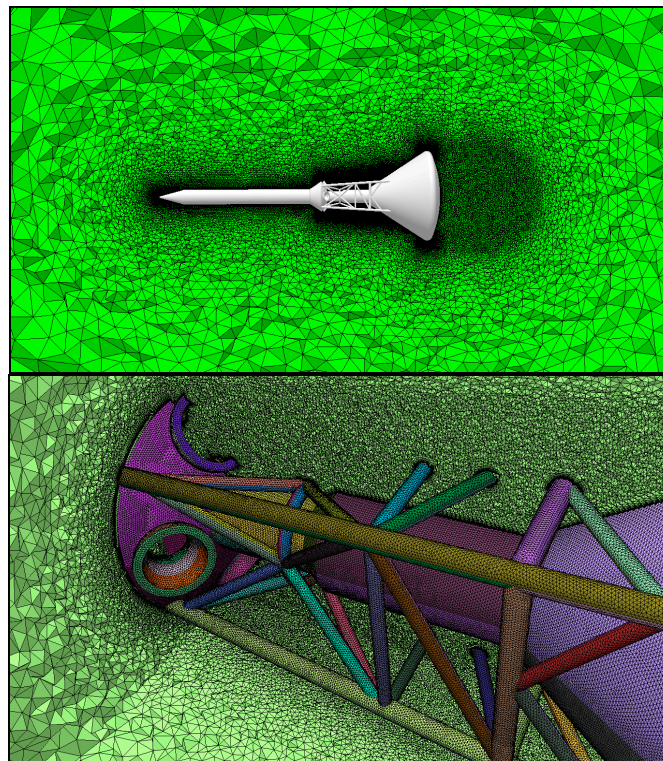


Figure 3. Baseline grid for the Apollo Launch Abort System (LAS) with Abort Motor (48.3 million cells).

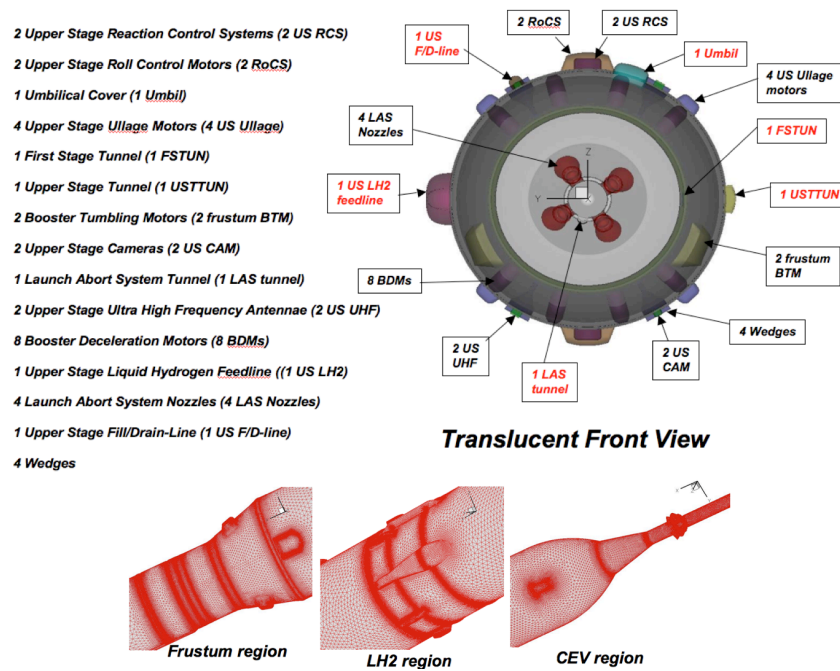


Figure 4. Front view of ARES I ADAC-2B (A103) configuration with protuberances and a view of computational grid around selected region.

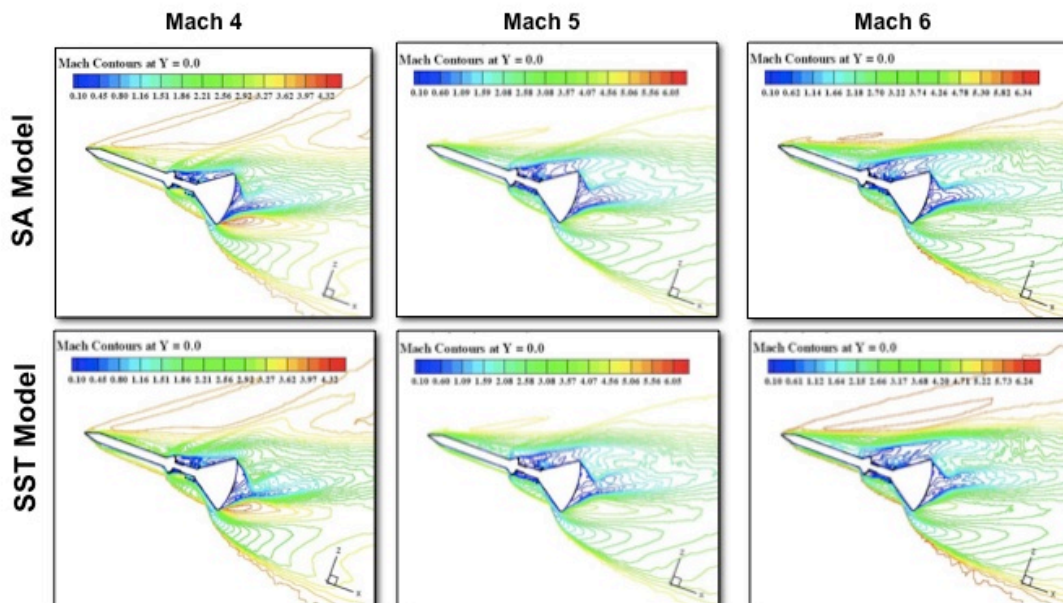


Figure 5. USM3D solutions for Apollo Launch Abort System using SA and SST turbulence models at free stream Mach 4, 5, and 6. Solutions obtained without plumes.

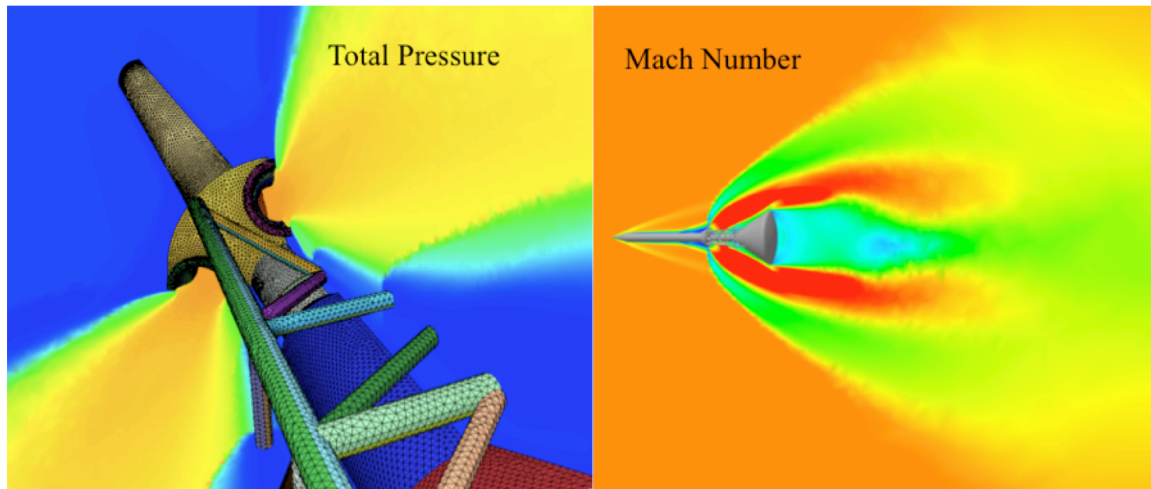


Figure 6. USM3D solution for Apollo Launch Abort System using SST turbulence model for free stream Mach 4, Nozzle pressure ratio 4,200.

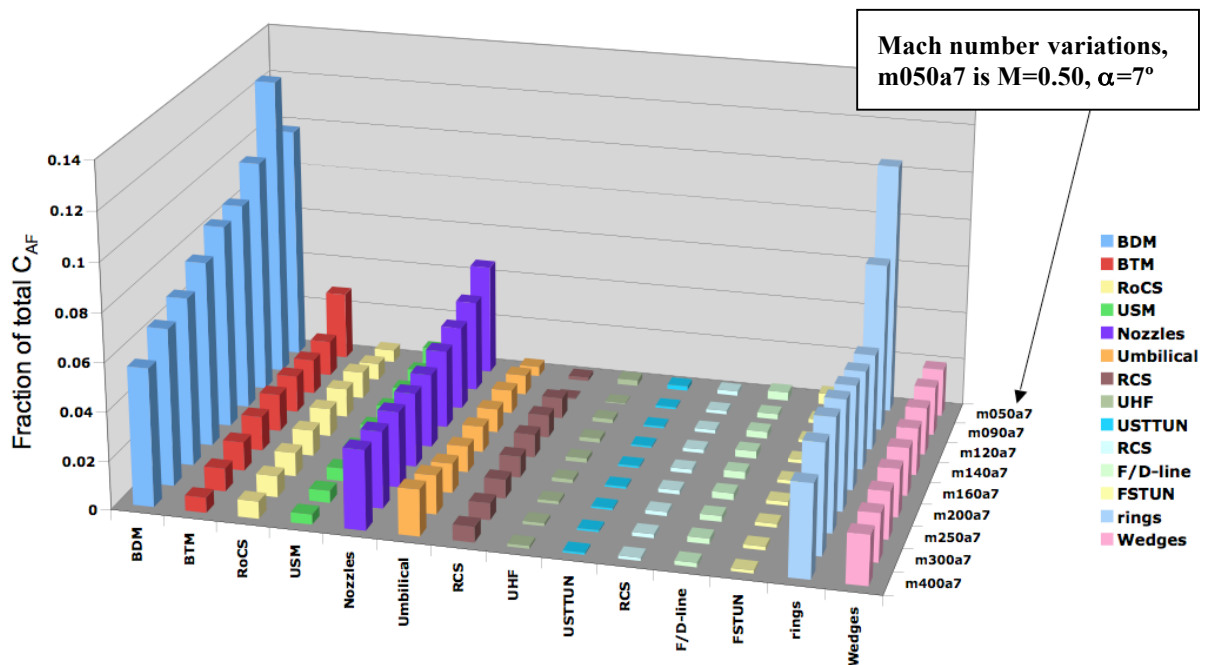


Figure 7. Individual/group protuberance(s) contribution to the computed C_{AF} for the ADAC-2A (A101) C1 configuration, $\alpha=7^\circ$, WT Reynolds number.

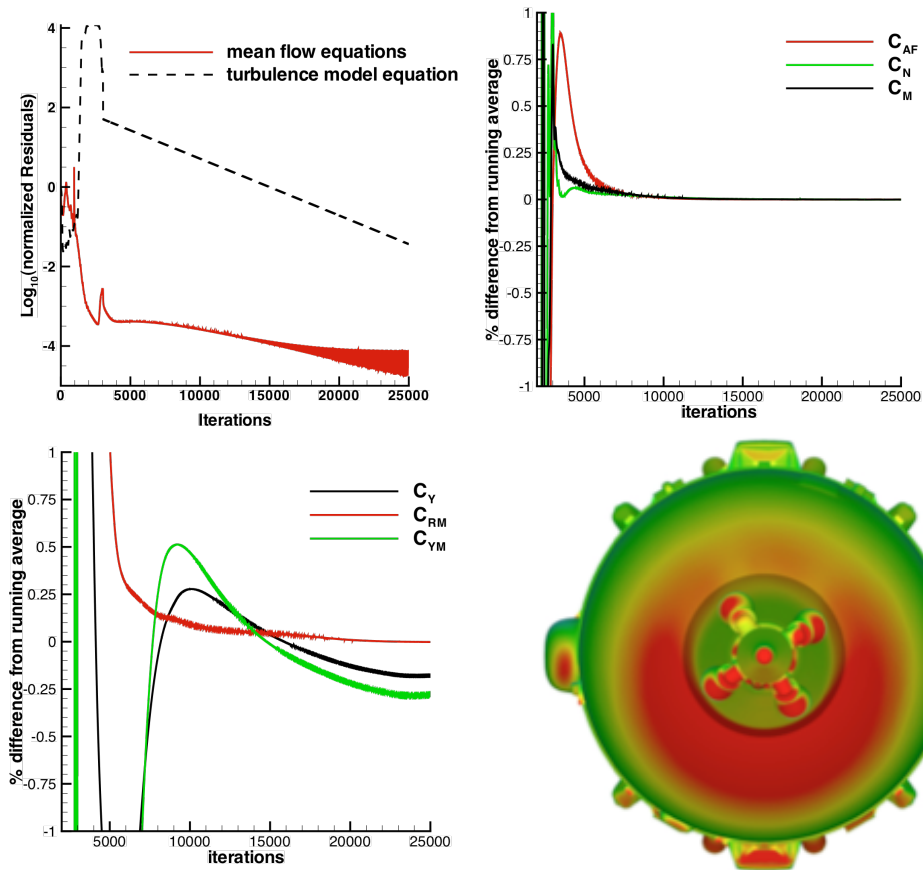


Figure 8. Typical solution convergence for the Ares I ADAC-3 (A106) configuration, $M=1.6$, $\alpha=8^\circ$, $\Phi=0^\circ$, WT Re.

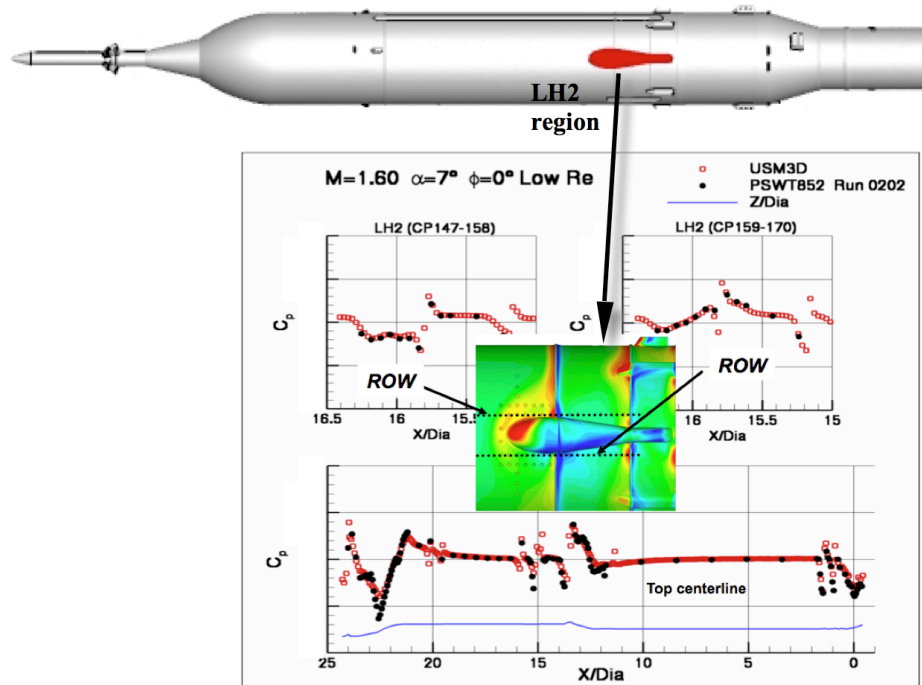


Figure 9. Computed and measured surface C_p on the Ares I ADAC-2B (A103) configuration, $M=1.6$, $\alpha=7^\circ$, WT Re.

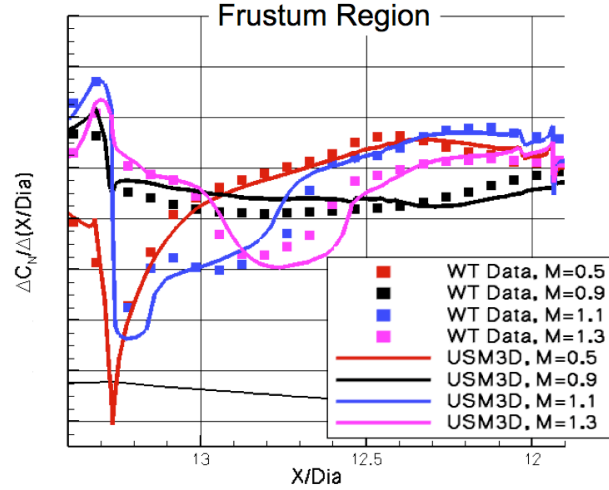


Figure 10. Computed and measured line loads over the frustum for the clean ADAC-2B (A103), $\alpha=7^\circ$, WT Re.

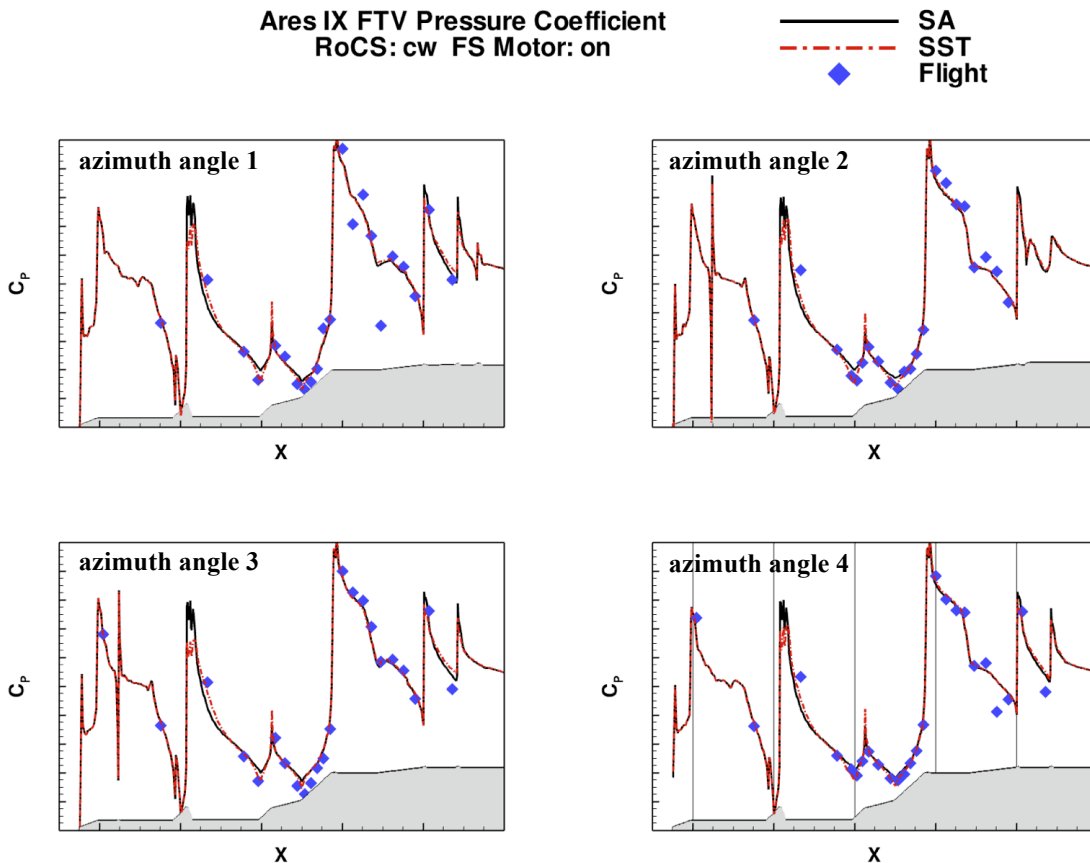


Figure 11. Comparison of USM3D surface pressures with the flight test data for the Ares I-X Flight Test Vehicle. Computations using SA and SST turbulence models (from ref. 27).

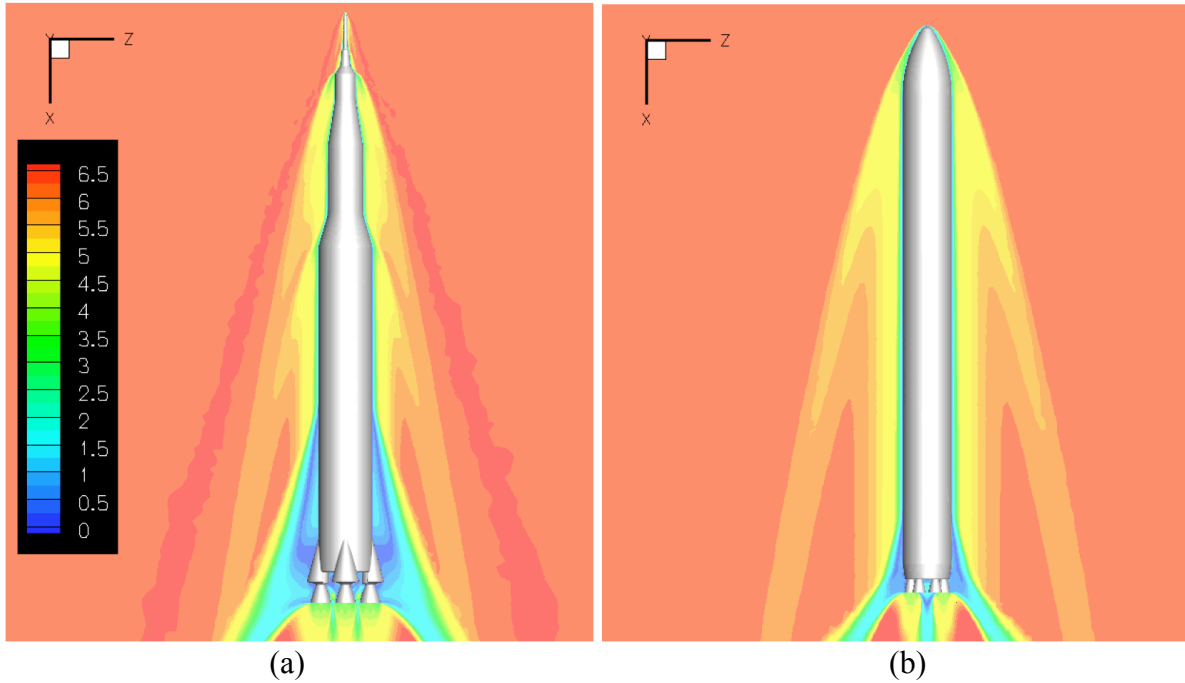


Figure 12. Symmetry plane Mach contours for two heavy lift launch vehicle configurations at $M=6.5$, $\alpha=0^\circ$, $\beta=0^\circ$, flight Reynolds number. (a) Saturn V rocket (b) Ares V rocket (from ref. 29).

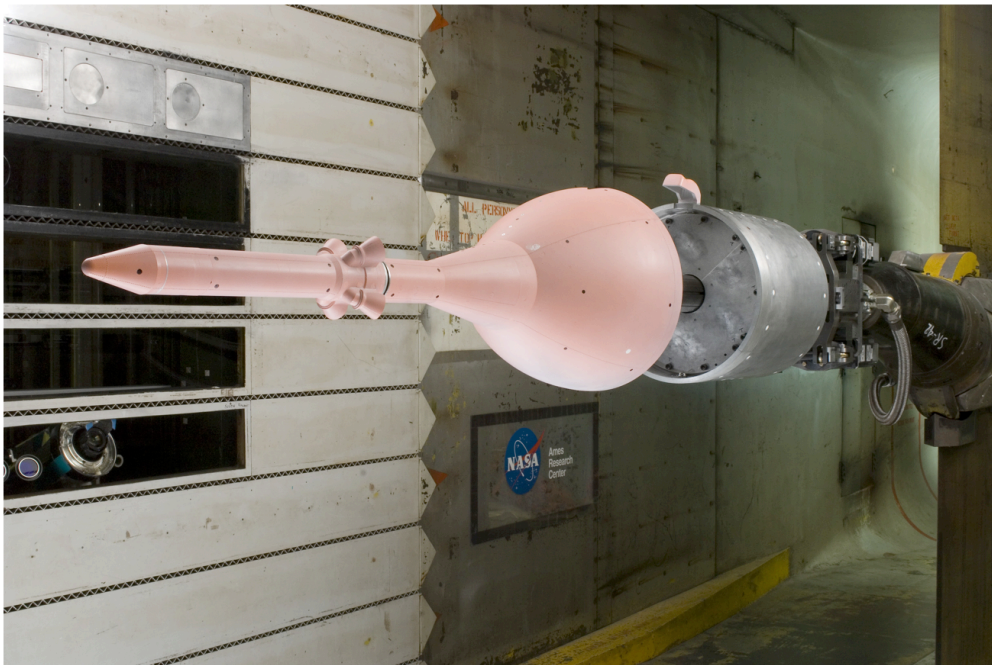


Figure 13. The Project Orion LAS 60-AA wind tunnel model mounted in the NASA Ames Research Center 11-foot tunnel.

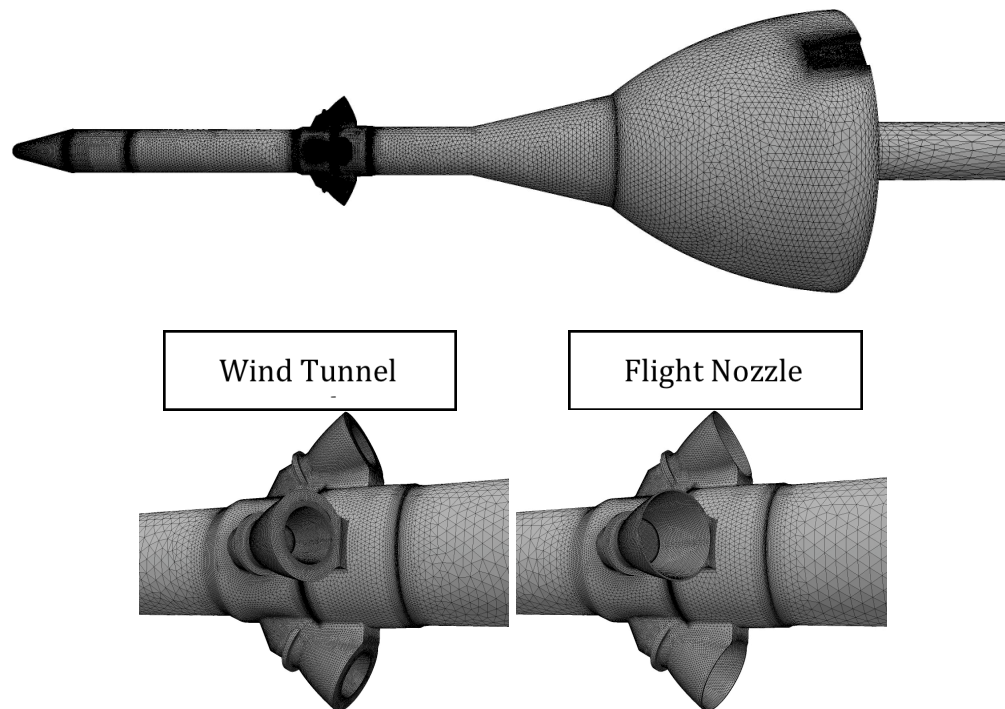


Figure 14. Coarse grid surface triangulation for LAS 60-AA configuration.

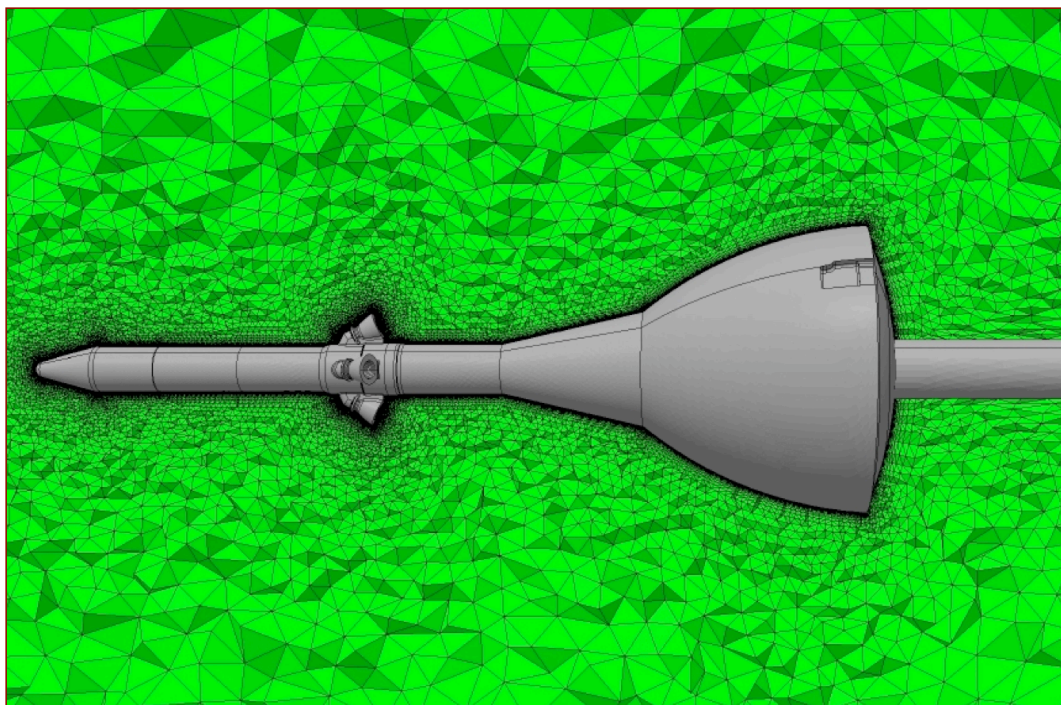


Figure 15. Cross cut of baseline coarse grid (20 million cells) for the LAS 60-AA WTT configuration.

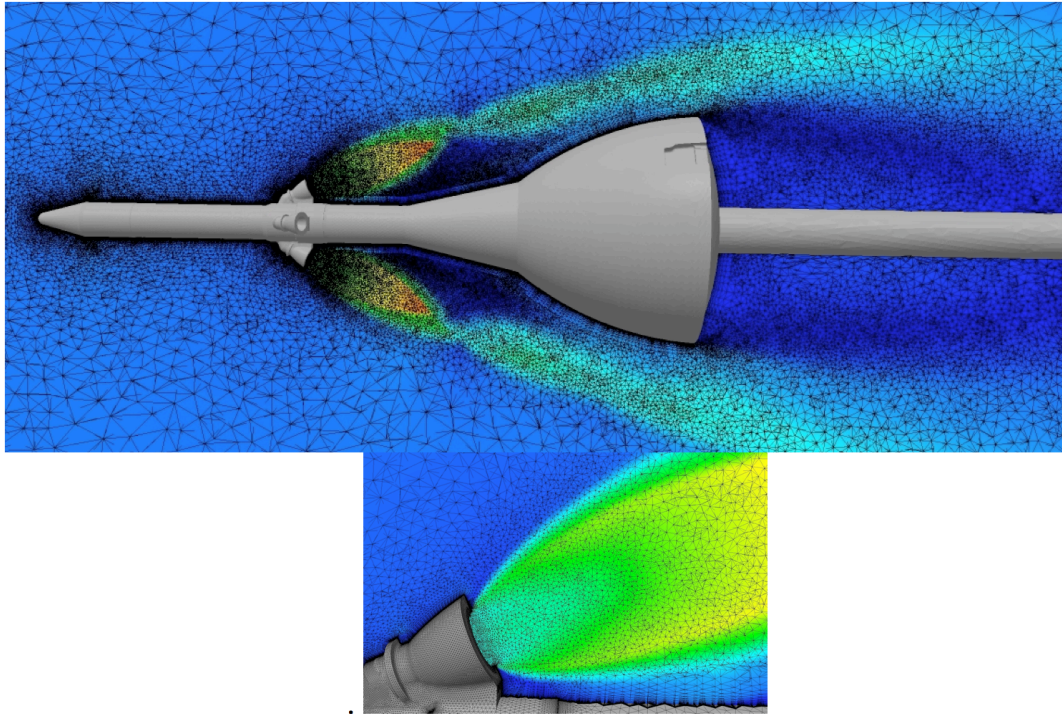


Figure 16. Solution adapted grid from guideline study on LAS 60-AA for $M_\infty=0.9$, $\alpha=-8^\circ$, $AM_CT=4$, SST Sarkar Comp + PD turbulence model (adapted grid has 34 million cells, baseline grid has 25 million cells).

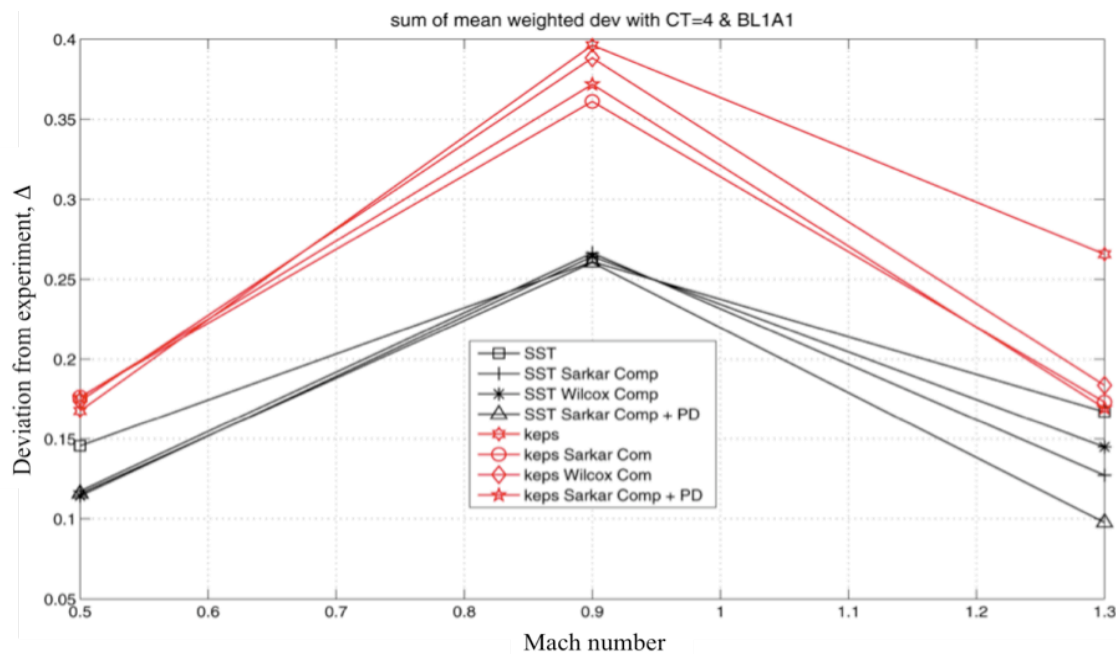


Figure 17. Statistical assessment of USM3D turbulence model variations from experimental data for LAS 60-AA.

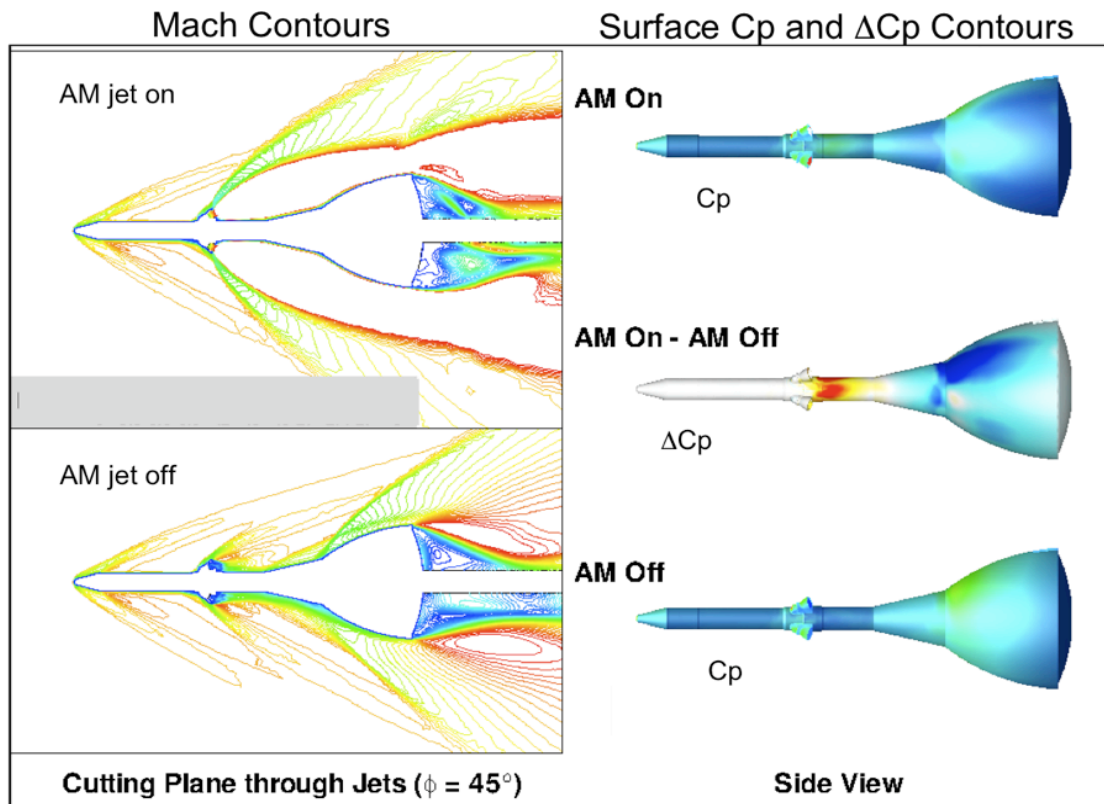


Figure 18. Typical Mach crosscut and surface C_p , ΔC_p contour plots for the LAS 60-AA configuration. $M_\infty=2.5$, $\alpha=-8^\circ$, $Re_D=2.5 \times 10^6$, $AM_CT=3.689$.

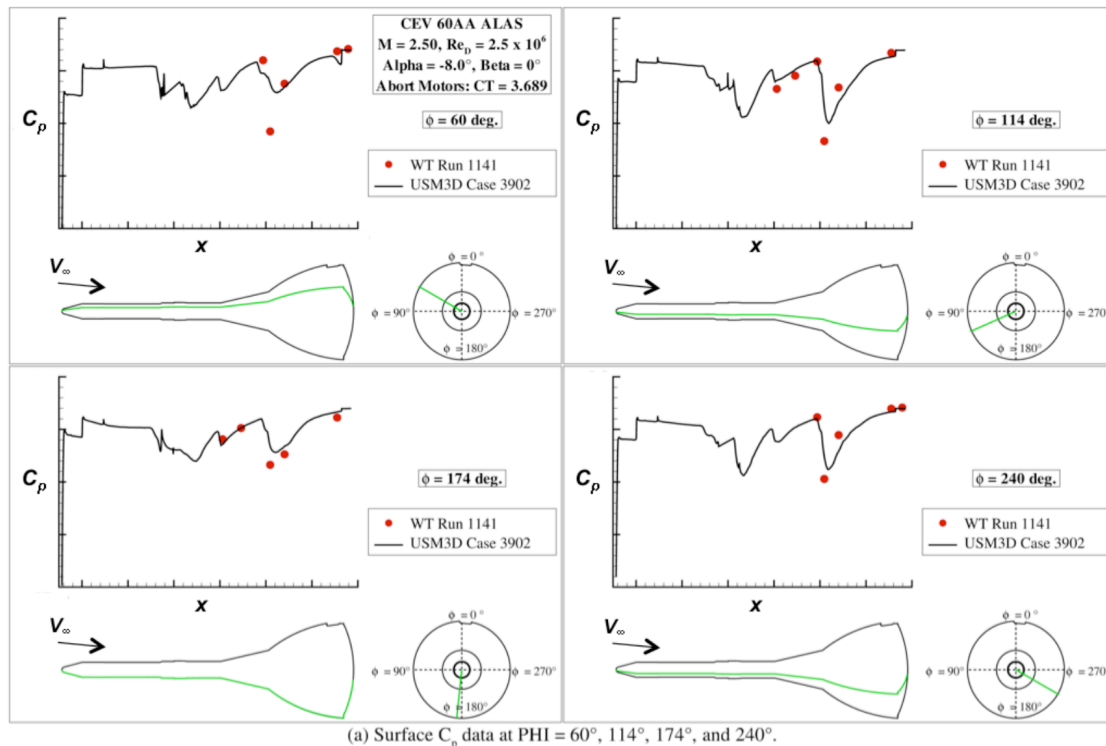


Figure 19. Typical surface C_p line plots for the LAS 60-AA configuration. $M_\infty=2.5$, $\alpha=-8^\circ$, $Re_D=2.5 \times 10^6$, $AM_CT=3.689$.