International Journal of Aeronautical and Space Sciences
https://doi.org/10.1007/s42405-022-00558-0

IJASS International Journal of Aeronautical and Space Sciences

ORIGINAL PAPER

An Automated Aerodynamic Analysis System in Missile Based on Open-Source Software

Kang Kuk You[1] · Jung Hyun Ha[1] · Sang Chul Lee[1]

Received: 28 October 2021 / Revised: 19 August 2022 / Accepted: 16 November 2022

Abstract
An automated aerodynamic analysis system that predicts aerodynamic characteristics of the missile configuration at the early design stage is developed using open-source software and presented. This system consists of three modules for geometry modeling, mesh generation, and flow analysis. The missile configuration defined by Missile DATCOM is created using OpenCASCADE software. The unstructured tetrahedral mesh is generated by NETGEN software with minimal input parameters. In particular, the density-based coupled solver, TSLAeroFoam, is used to predict aerodynamic coefficients accurately in the compressible flow regime. The presented system is verified for three representative missile configurations, and the results show good agreements in aerodynamic coefficients with the experiment.

Keywords Automated aerodynamic analysis system · Open-source · OpenCASCADE · NETGEN · OpenFOAM

1 Introduction

Multidisciplinary design analysis and optimization (MDAO) is a technique used widely in the conceptual design stage. The missile design includes analysis of various disciplines such as aerodynamics, propulsion, trajectory, etc. MDAO with design parameters can allow designers to understand the overall characteristics of missiles but requires a large number of analysis cases. For that reason, low-fidelity solvers like Missile DATCOM [1] and Aeroprediction [2], which are semi-empirical codes and do not require CAD, are mainly adopted for MDAO. However, some studies showed that the low-fidelity solvers cannot predict aerodynamic performance accurately compared to the experiment results [3, 4]. Therefore, a computational fluid dynamics (CFD) can be an alternative option for aerodynamic analysis.

There were some challenges in applying CFD to MDAO. Unlike semi-empirical solvers, CFD requires water-tight geometry representation from the design parameters, and it is also necessary to create the volume mesh for flow analysis. These time-consuming processes are performed manually by the user and a large number of simulation cases for similar configurations should be conducted repeatedly in MDAO. Consequently, the automation of geometry representation and mesh generation is essential for work efficiency.

Several studies have been previously conducted to develop partially or fully automated processes. Marco et al. [5] developed an automated aircraft modeling API, named JPAD, which uses geometric parametrization, but the automated processes for mesh generation and flow analysis were not included. Tomac et al. [6] developed a module for generating the mesh automatically as a part of the CEASIOM framework. The module was developed using the open-source mesh generation software, but it had a low robustness in the creation process of prism layer for thin airfoil. Ordaz et al. [7] introduced an automated process of generating a tetrahedral mesh for sonic boom analysis of supersonic configuration, but a regionally restricted license is applied to the mesh generation software, AFLR3. Gu et al. [8] presented a fully automated analysis workflow that can be applied for overall aircraft design, but the open-source software was used only for the geometry generation while other processes used the commercial software such as Pointwise and ANSYS Fluent.

This study aims to develop a fully automated aerodynamic analysis system based on open-source software. Additionally, a special treatment is performed on the missile geometry to solve a problem related with the robustness for generating the mesh of high quality. For the flow analysis,

✉ Sang Chul Lee
sclee@nextfoam.co.kr

[1] Research Center, NEXTfoam Co., LTD., Seoul, Republic of Korea

Published online: 15 December 2022

TSLAeroFoam [9], a density-based coupled solver which was developed previously by our research group for compressible flow, is used. The verification of presented system is conducted for three different missile configurations.

This paper is organized as follows: Sect. 2 describes the modules in an automated aerodynamic analysis system. The mathematical expression for TSLAeroFoam is briefly introduced in Sect. 3. Section 4 shows the results of aerodynamic analyses for test cases, and the conclusion of this study is given in Sect. 5.

2 Automated Aerodynamic Analysis System

The overall automated system for the missile aerodynamic analysis depicted in Fig. 1. The system consists of three individual processes: geometry modeling, mesh generation, and flow analysis. For fully automated analysis without any user's intervention, the required input parameters for each module are defined in the XML (eXtensible Markup Language) file that can store hierarchical data.

First of all, for geometry modeling, the geometric parameters based on the definition method in Missile DATCOM are required, then OpenCASCADE (OCC) [10], an open-source geometry modeling kernel, uses those parameters to create geometry. OCC is a fully object-oriented C++ API and supports a full-scale BRep (Boundary Representation) format. OCC has a large number of modeling functions and can generate sophisticated geometry rapidly. In this study, pythonOCC [11], an OCC which is wrapped into python, is used for geometry generation, and the created geometry is exported as the STEP file format.

Secondly, SALOME [12] software, an open-source platform for pre-and post-processing of numerical simulation, is used to generate volume mesh for flow analysis. Among the modules of SALOME, the GEOM and SMESH modules are used for mesh generation. In the GEOM module, the geomerty-related procedures are carried out such as the creation of the computational domain with the boolean operation and the identifying process of the specific faces to set the mesh sizes. In the SMESH module, the processes of setting mesh sizes and generating the volume mesh are performed. The meshing algorithms for unstructured tetrahedral mesh are adopted from the NETGEN [13] software, which is embedded in SALOME, then the volume mesh is exported as the UNV file format.

Finally, the flow analysis is conducted using OpenFOAM [14], which is an open-source CFD software package. OpenFOAM has over 100 standard solvers that can be applied to a wide range of flow simulations and over 150 standard utilities for pre-processing, mesh generation and conversion, and post-processing. The pre-processing for flow analysis is conducted using the mesh created from the previous module and the input parameters.

2.1 Geometry Modeling

The geometric parameters for the body geometry are shown in Fig. 2. There are two different options to define the body in Missile DATCOM. The geometric parameters for each option are summarized in Table 1. Option 1 defines the body by dividing it into the nose, centerbody, and afterbody, and five nose shapes (conical, tangent ogive, power series, and Haack/Karman series) are available. Option 2 defines it using the longitudinal stations and corresponding width of each station. The cross-section of the body can be defined as circle or ellipse in both options. The three different types of nose are shown in Fig. 3, and all nose types are available for both options. The geometric equations for various nose shapes and types are presented in Ref. [15]. In this study, the sharp nose is modified into a blunted nose with a small radius to facilitate the mesh generation and flow analysis.

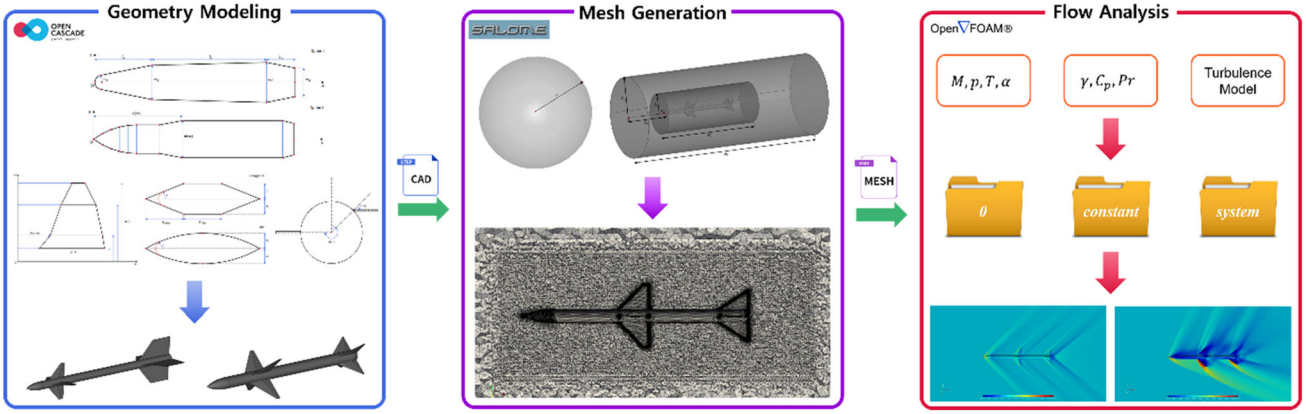


Fig. 1 Automated aerodynamic analysis system

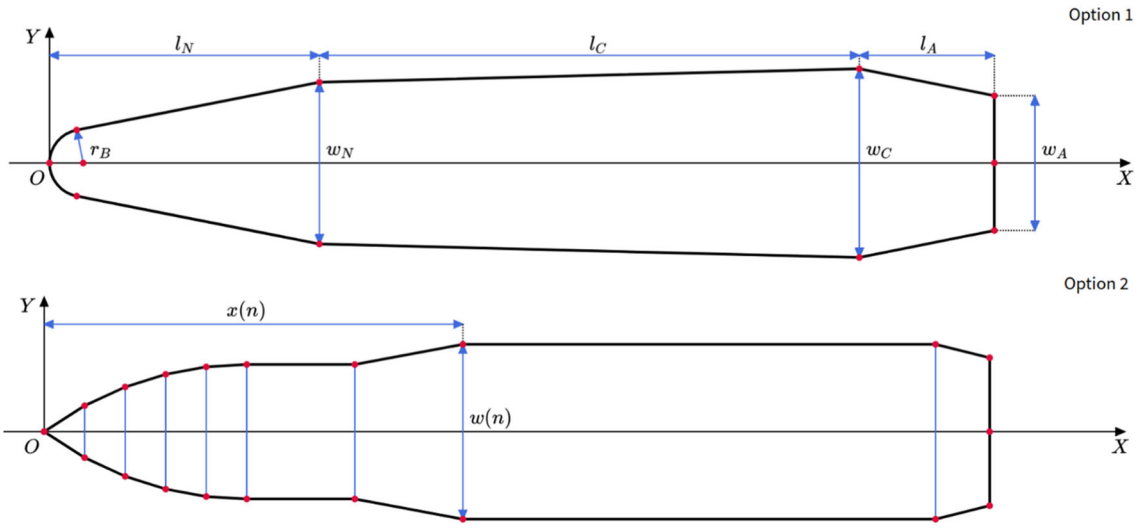


Fig. 2 Body geometric parameters

Table 1 Summary of body geometric parameters

Option	Parameters	
Option1	Nose length	l_N
	Nose width	w_N
	Blunted/truncated nose radius	r_B
	Centerbody length	l_C
	Centerbody width	w_C
	Afterbody length	l_A
	Afterbody width	w_A
	Ellipticity	E
Option2	n-th longitudinal coordinates	$x(n)$
	n-th station width	$w(n)$
	Blunted/truncated nose radius	r_B
	Ellipticity	E

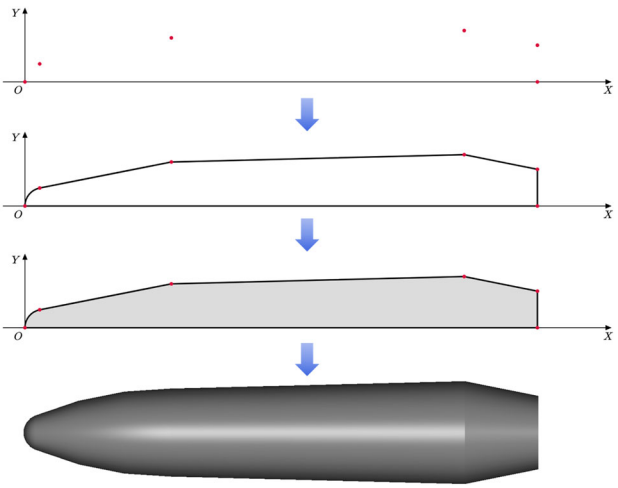


Fig. 4 Process of generating body solid

The process of generating the body solid is described in Fig. 4. The coordinates of the points representing the outlines are calculated with the given parameters, and the outline edges are created using those points as the vertices. After that, a face is created using a set of closed wires that consist of edges, and the body solid is generated by rotating the face to the x-axis. Additionally, the scale operation for ellipticity can be performed when the cross-section is an ellipse.

The process of generating finset geometry consists of defining planform, selecting airfoil for each station and attaching method to the body. Table 2 summarizes the param-

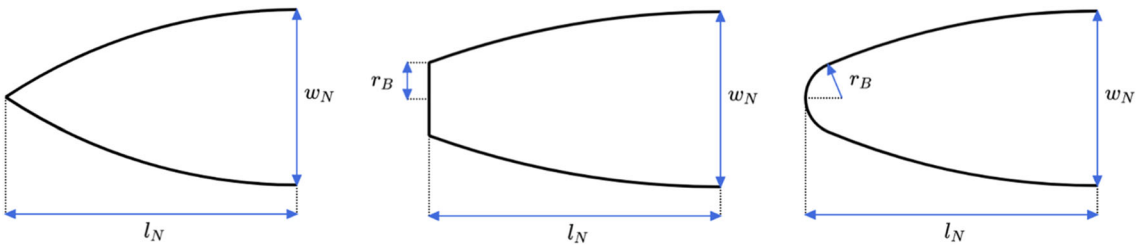


Fig. 3 Nose options

Table 2 Summary of Finset geometric parameters

Category	Parameters	
Planform	n-th station chord length	$c(n)$
	n-th station semi-span location	$s(n)$
	n-th station distance from nose to leading edge	$X_{\mathrm{LE}}(n)$
Airfoil	Thickness-to-chord ratio of upper surface	z_{u}
	Thickness-to-chord ratio of lower surface	z_{l}
	Fraction of chord from leading edge to maximum thickness section	$L_{\max}$
	Fraction of chord of constant thickness section	L_{flat}
	Leading edge radius	r
Attaching Method	Roll angle of each panel	$\phi(n)$
	Dihedral angle of each panel	$\gamma(n)$

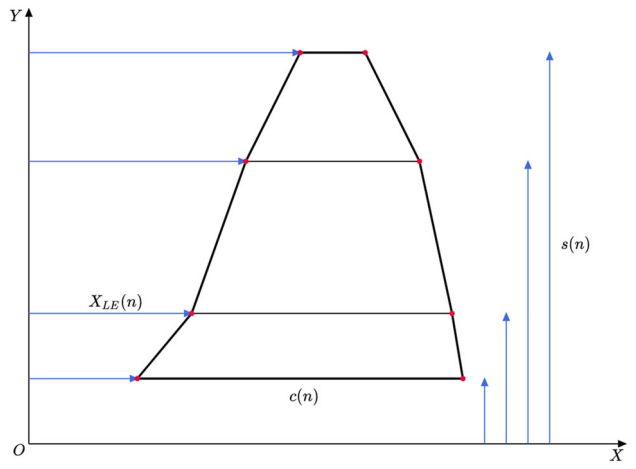


Fig. 5 Planform geometric parameters

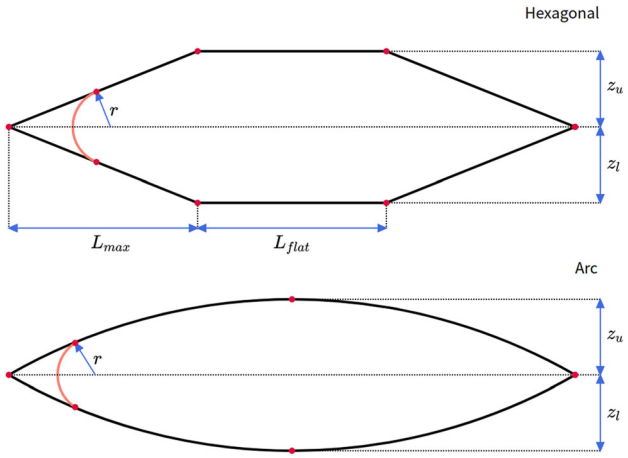


Fig. 6 Airfoil geometric parameters

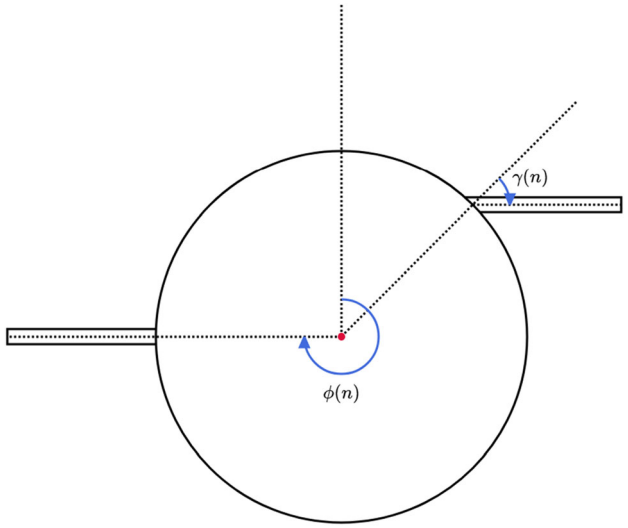


Fig. 7 Geometric parameters for attaching method

eters that define the finset geometry, and the parameters of the planform are described in Fig. 5. The planform is defined by multiple spanwise stations that consist of chord length, semi-span length, and the distance from the nose to the leading edge of root chord at each station.

There are four types of airfoil in Missile DATCOM: hexagonal, circular arc, NACA, and the shape as defined by x, y coordinates. The parameters of each airfoil are shown in Fig. 6, and the diamond-shaped airfoil is a special case of the hexagonal airfoil which sets L_{FLAT} to zero. The sharp edges for hexagonal and arc airfoils are modified to have a small radius on the leading and trailing edges to facilitate mesh generation and flow analysis.

The parameters for the attaching method to the body are shown in Fig. 7. The roll angle ($\varnothing$) is measured clockwise from the top center line, and the positive dihedral angle (γ) increases the roll angle. The roll angle and dihedral angle can be set individually for each panel.

The process of creating the finset solid is described in Fig. 8. First of all, the airfoil wire at each spanwise station is created with the given parameters. After that, the planform solid is constructed by lofting the airfoil wires, and the roll and dihedral angles are applied by rotating each panel.

As described above, the missile geometry is generated using fuse operation of body and finset solids which were created from vertex to solid, so it has the water-tight characteristic which is applicable to mesh generation and flow analysis.

XML input files applied for generating body and finset solids are shown Figs. 9 and 10, respectively, and the

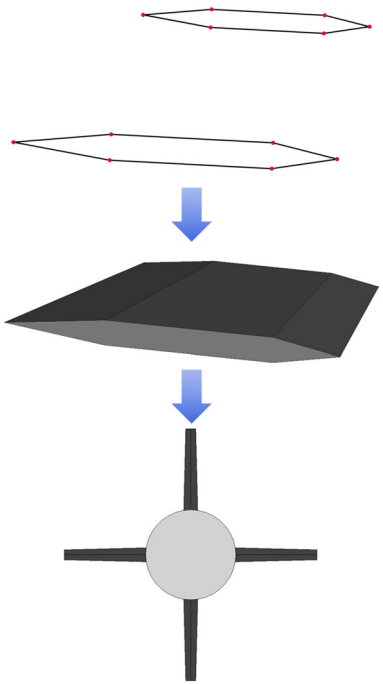


Fig. 8 Process of generating finset solid

```
<Item Name="Body" Type="Body1">
    <Item Name="Nose" Type="NOSE">
        <X0>0.000000</X0>
        <TNOSE>OGIVE</TNOSE>
        <LNOSE>0.068600</LNOSE>
        <DNOSE>0.030480</DNOSE>
        <BNOSE>0.001524</BNOSE>
        <TRUNC>.FALSE.</TRUNC>
    </Item>
    <Item Name="Centr" Type="CENTR">
        <LCENTR>0.468400</LCENTR>
        <DCENTR>0.030480</DCENTR>
    </Item>
    <Item Name="Aft" Type="AFT">
        <TAFT>CONE</TAFT>
        <LAFT>0.011600</LAFT>
        <DAFT>0.027430</DAFT>
    </Item>
</Item>
```

Fig. 9 XML example for geometry modeling (Body)

parameter of list (i.e., array) type is indicated by 'param' tag. Figure 11 illustrates missile solids created automatically with different parameters. The example solids show that the present module for geometry modeling can create various configurations according to the different input parameters while the same algorithm is executed. A series of tasks to

```
<Item Type="FINSET" Name="Finset1">
    <SSPAN>
        <param>0.015240</param>
        <param>0.074940</param>
    </SSPAN>
    <CHORD>
        <param>0.074200</param>
        <param>0.014500</param>
    </CHORD>
    <SWEEP>
        <param>45.000000</param>
    </SWEEP>
    <XLE>0.150000</XLE>
    <SECTYP>HEX</SECTYP>
    <ZUPPER>
        <param>0.029990</param>
        <param>0.029990</param>
    </ZUPPER>
    <ZLOWER>
        <param>0.029990</param>
        <param>0.029990</param>
    </ZLOWER>
    <LMAXU>
        <param>0.568700</param>
        <param>0.568700</param>
    </LMAXU>
```

Fig. 10 XML example for geometry modeling (Finset)

generate the whole missile geometry takes only a few seconds (1–2 s).

2.2 Mesh Generation

An automated mesh generation begins from the CAD geometry created in the previous process. The input parameters are required minimally, and they are about the computational domain, the mesh size, and the prism layer. The parameters of the computational domain are shown in Fig. 12 and summarized in Table 3. The computational domain consists of the spherical far-field region and two cylindrical refinement regions. The refinement regions are used to accurately capture the shock wave and the wake flow generated around the missile. The sizes of the computational domain and refinement regions are adjusted by given parameters.

The parameters related to the mesh size are composed of the maximum size, minimum size, and growth rate. The mesh size can be set individually for missile surfaces, each refinement region, and far-field region. Also, to predict the

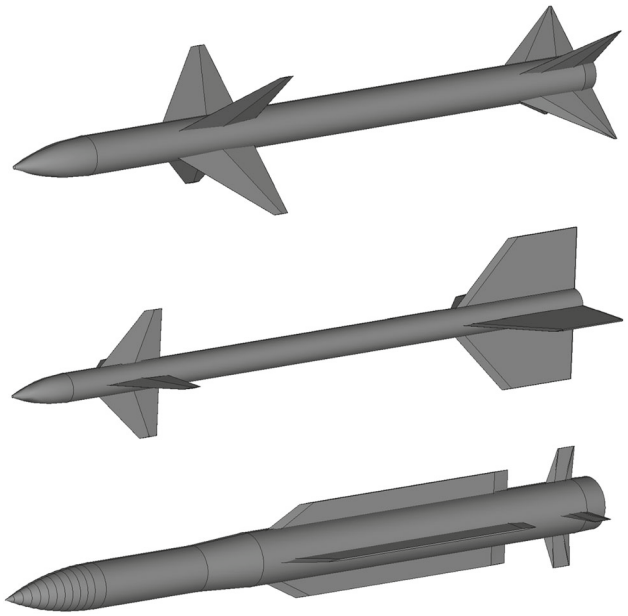
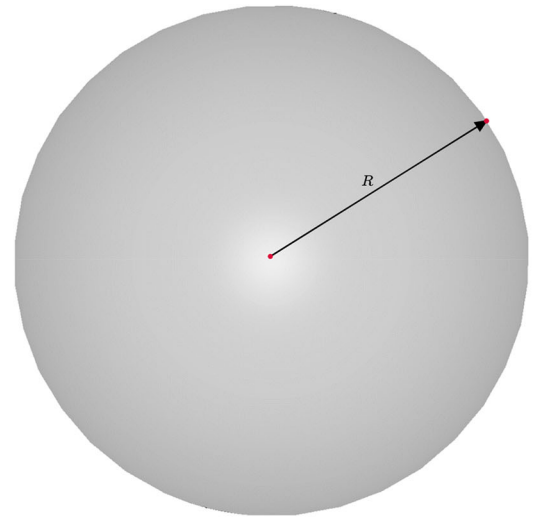
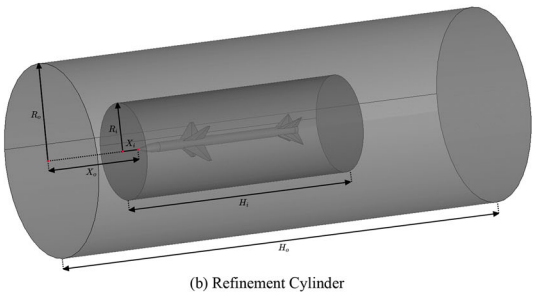


Fig. 11 Various missile configurations



(a) Computational Domain



(b) Refinement Cylinder

Fig. 12 Computational domain parameters

Table 3 Summary of computational domain parameters

Parameters	
Far-field region radius	R
Inner refinement region radius	R_i
Inner refinement region height	H_i
Distance from nose for inner refinement region	X_i
Outer refinement region radius	R_o
Outer refinement region height	H_o
Distance from nose for outer refinement region	X_o

```
<Item Name="Surfmesh" Type="SURFMESH">
    <Max>0.004500</Max>
    <Min>0.000250</Min>
    <growth>0.200000</growth>
</Item>
<Item Name="Volmesh" Type="VOLMESH">
    <Item Name="Prism" Type="PRISM">
        <nlayers>20</nlayers>
        <stretch>1.200000</stretch>
        <yplus>1.000000</yplus>
    </Item>
    <Item Name="Fine1" Type="FINE1">
        <X>-0.050000</X>
        <R>0.150000</R>
        <H>1.150000</H>
        <Max>0.004000</Max>
        <growth>0.200000</growth>
    </Item>
```

Fig. 13 XML example for mesh generation

aerodynamic coefficients accurately, the local refinement is applied to the nose and the base face of the body. The cell size for face refinement is automatically calculated using the diameter of the missile from geometric parameters. The parameters for the generation of the prism layer consist of the number of layers, the stretch factor, and the desired y^+. The total thickness of the prism layer is calculated using freestream conditions from the flow analysis parameters.

Figures 13 and 14 show the example of XML input files used for mesh generation and the automatically generated volume mesh in the core refinement region, respectively. The local face refinements applied on the blunted nose and base face and the prism layers created on the junction between the body and finset and the tip face of each finset panel are shown clearly.

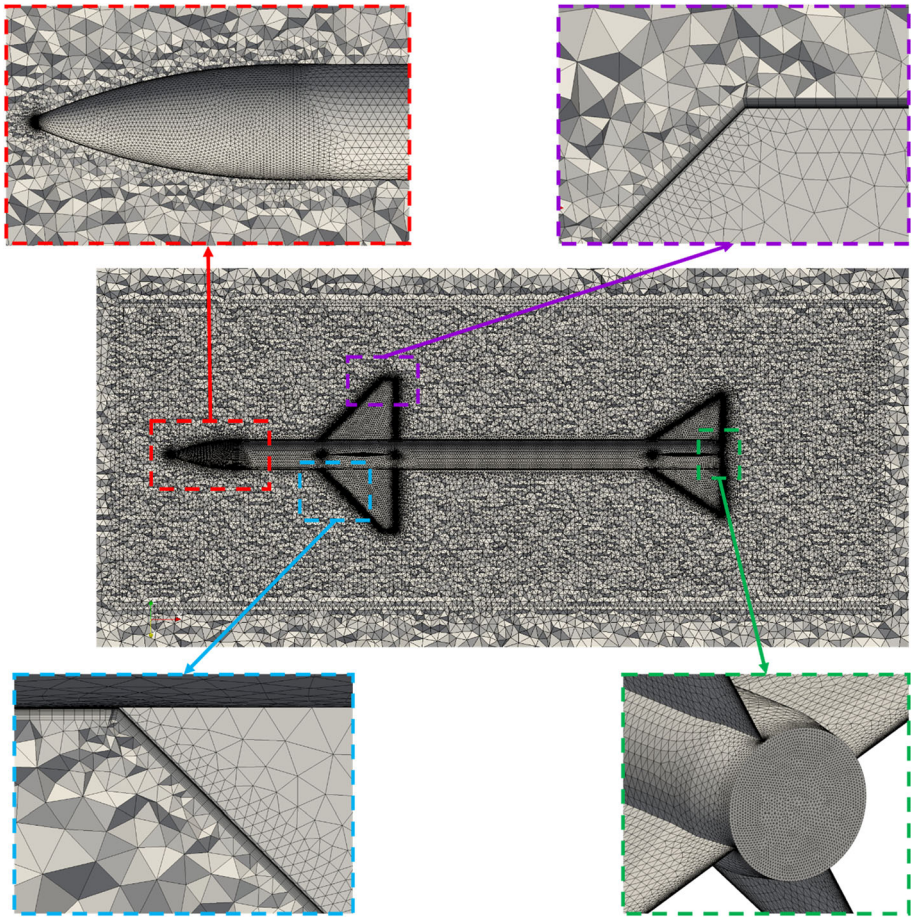


Fig. 14 Automatically generated volume mesh

2.3 Flow Analysis

OpenFOAM does not provide GUI, unlike commercial CFD programs, and uses text-based user interface. The simulation case directory consists of three subdirectories: 0, constant, and system. In the 0 directory, the information related to initial and boundary conditions is stored. A separate file should be written for each field variable. In the constant directory, the information that does not change during the analysis (e.g. the volume mesh, physical properties, turbulence model, and other constants) is stored. Finally, the system directory contains the information related to the simulation control. The parameters for conducting simulation such as the start/end time, CFL number, discretization schemes, and linear solvers are written in this directory.

Because there are a large number of options that can be selected for setting the case, the options for automated case setting should be limited. For instance, $k-\omega$ SST model [16] is fixed for the turbulence model in this study, and TSLAeroFoam which was previously developed by our research group as a density-based coupled solver for compressible flow is fixed also for the numerical solver. The governing equations and algorithm of this solver are briefly introduced in the next chapter. Consequently, a few parameters summarized in Table 4 can be selected as changeable variables for flow analysis. Figure 15 shows an example XML file for flow analysis.

Table 4 Summary of changeable variables for flow analysis

Category	Parameters	
Flow conditions	Mach number	M
	Static pressure (Pa)	p
	Static temperature (K)	T
Gas constant	Specific heat ratio	γ
	Heat capacity at constant pressure	C_p
	Prandtl number	Pr

The automated case setting process starts with creating a 'Default' directory. In the analyses for the same geometry, constant and system directories contain identical data regardless of the change of flow condition. Therefore, constant and system directories are created within the 'Default' directory. Then, the volume mesh generated in the previous

```
<Items>
    <Item Name="TURB" Type="TURB">
        <model>KOmegaSST</model>
        <Intensity>0.01</Intensity>
        <Beta>0.5</Beta>
        <Prt>0.9</Prt>
        <muOption>Sutherland</muOption>
    </Item>
    <Item Name="Solution" Type="SOLUTION">
        <startTime>Start</startTime>
        <endTime>5000</endTime>
        <WriteInterval>100</WriteInterval>
        <maxCo>1</maxCo>
        <nprocs>16</nprocs>
    </Item>
</Items>
```

Fig. 15 XML example for flow analysis

process is converted to OpenFOAM mesh using *ideasUnvToFoam* utility. Since all boundary types are converted as 'patch' type by default, the boundaries for the missile surface must be changed to 'wall' type after the mesh is converted. When the simulation directories are created for each flow condition, constant and system directories are copied from the 'Default' directory. After that, 0 directory which reflects the corresponding flow condition is created.

3 Numerical Solver

The conservation of mass, momentum, and energy equations of compressible flow are as follows

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \left(\rho \vec{U} \right) = 0, \tag{1}$$

$$\frac{\partial \left(\rho \vec{U} \right)}{\partial t} + \nabla \cdot \left(\rho \vec{U} \otimes \vec{U} \right) + \nabla p = \nabla \cdot \overline{\tau_f}, \tag{2}$$

$$\frac{\partial (\rho E)}{\partial t} + \nabla \cdot \left(\rho H \vec{U} \right) = \nabla \cdot \left(\overline{\tau_f} \cdot \vec{U} \right) + \nabla \cdot (k \nabla T), \tag{3}$$

where ρ is the density, $\vec{U}$ is the velocity vector, p is the pressure, $\overline{\tau_f}$ is the stress tensor, E is the energy, H is the enthalpy, k is the thermal conductivity, and T is the temperature. The integral form of the system of equations is as follows

$$\int_V \frac{\partial \vec{W}}{\partial t} dV + \int_S \left(\vec{F_c} - \vec{F_v} \right) \mathrm{d}S = 0, \tag{4}$$

where $\vec{W} = \left[\rho, \rho \vec{U}, \rho E \right]$ is conservative variable vector, $\vec{F_c}$ is convective flux vector, and $\vec{F_v}$ is viscous flux vector.

The previous equation is discretized in space as

$$V_i \frac{\partial \vec{W}_i}{\partial t} + \sum_{j \in N(i)} \left(\vec{F}_{c,ij} - \vec{F}_{v,ij} \right) S_{ij} = 0, \tag{5}$$

where V_i is volume of the cell, $\vec{W}_i$ is average value at each cell center. $N(i)$ is the set of neighbor cells, S_{ij} is the face component oriented from cell i to j.

For temporal discretization in Eq. (5), the first order backward Euler is applied as follows,

$$\frac{V_i}{\Delta t_i} \left(\vec{W}_i^{\,n+1} - \vec{W}_i^{\,n} \right) + \sum_{j \in N(i)} \left(\vec{F}_{c,ij}^{\,n+1} - \vec{F}_{v,ij}^{\,n+1} \right) S_{ij} = 0, \tag{6}$$

and both convective and viscous fluxes are linearized using Taylor's series expansion as follows

$$\frac{V_i}{\Delta t_i} \Delta \vec{W}_i^{\,n} + \sum_{j \in N(i)} \left(A_{c,ij} - A_{v,ij} \right) \Delta \vec{W}_{ij}^{\,n} S_{ij} = -\mathrm{Res}_i^{\,n}, \tag{7}$$

where $\Delta \vec{W}_i^{\,n} = \vec{W}_i^{\,n+1} - \vec{W}_i^{\,n}$, $A_c = \partial \vec{F_c} / \partial \vec{W}$ is convective flux Jacobian, $A_v = \partial \vec{F_v} / \partial \vec{W}$ is viscous flux Jacobian, and $\mathrm{Res}_i^{\,n} = \sum_{j \in N(i)} \left(\vec{F}^{\,n}_{c,ij} - \vec{F}^{\,n}_{v,ij} \right) S_{ij}$ is residual term. Since a large amount of memory is required to store these Jacobian, the lower–upper symmetric Gauss–Seidel (LU-SGS) algorithm [17] is used.

First of all, the Jacobian matrices are simplified using Steger–Warming's flux vector splitting method [18] for convective flux and thin shear layer approximation (TSL) for viscous flux as below

$$\frac{V_i}{\Delta t_i} \Delta \vec{W}_i^{\,n} + \sum_{j \in N(i)} \left(A_{c,i}^{\,+} - A_{v,i}^{\,*} \right) \Delta \vec{W}_i^{\,n} S_{ij} + \sum_{j \in N(i)} \left(A_{c,j}^{\,+} - A_{v,j}^{\,*} \right) \Delta \vec{W}_j^{\,n} S_{ij} = -\mathrm{Res}_i^{\,n}. \tag{8}$$

Secondly, the diagonal terms are split into lower(owner) and upper(neighbor) triangular terms.

$$\frac{V_i}{\Delta t_i} \Delta \vec{W}_i^{\,n} + \sum_{j \in N(i)} \left(A_{c,i}^{\,+} - A_{v,i}^{\,*} \right) \Delta \vec{W}_i^{\,n} S_{ij} + \sum_{j \in L(i)} \left(A_{c,j}^{\,-} - A_{v,j}^{\,*} \right) \Delta \vec{W}_j^{\,n} S_{ij} + \sum_{j \in U(i)} \left(A_{c,j}^{\,-} - A_{v,j}^{\,*} \right) \Delta \vec{W}_j^{\,n} S_{ij} = -\mathrm{Res}_i^{\,n}. \tag{9}$$

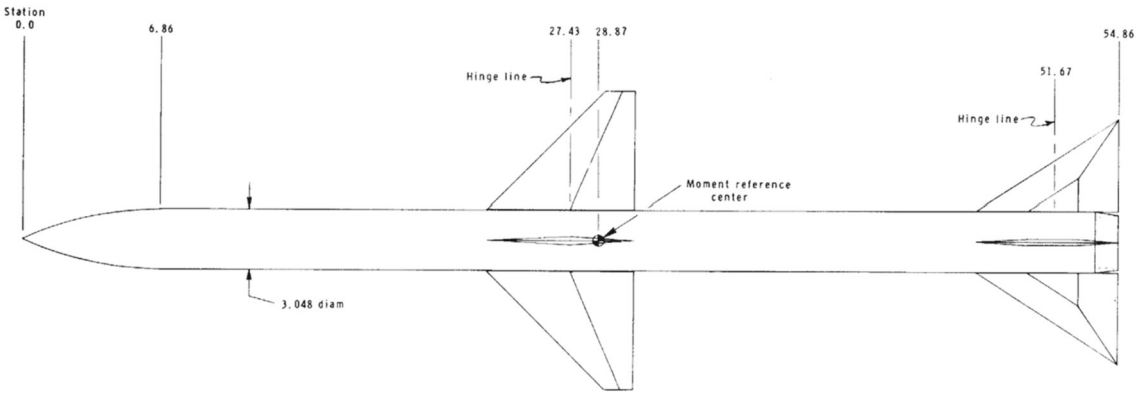


Fig. 16 NASA TP 1078 (Sparrow) [19]

The previous equation is represented as block matrix system

$$(D + L + U)\Delta W^n = -R^n. \tag{10}$$

Finally, factorization approximation is applied to previous equation and represented using two sweep symmetric Gauss–Seidel method as follows

$$\text{Forward} : (D + L)\Delta W^* = -R^n, \tag{11}$$

$$\text{Backward} : (D + U)\Delta W^n = D\Delta W^*. \tag{12}$$

4 Test Cases

Three representative missile configurations which have wind tunnel data are chosen as the validation cases for the automated system. The model geometry, flow condition, and analysis result for each missile configuration are presented in the following sections.

4.1 Sparrow Missile

The first missile configuration, which is called the Sparrow [19], is depicted in Fig. 16. The Sparrow consists of an ogive-cylindrical body, cruciform wings, and in-line tails. A trapezoidal planform with diamond airfoil is used for the wings and a delta planform with the hexagonal airfoil is used for the tails. In this study, the configuration with a roll orientation angle of 45° is used for analysis. The selected flight conditions are reported in Table 5.

Figure 17. shows the automatically generated surface mesh and volume mesh in the core refinement region. The spherical far-field region is created with a radius of 20 missile lengths. The average y^+ is set to be 1.0, and 20 prism layers are created on the missile surface. The total number of surface mesh is 348,516 and the volume mesh is approximately 23.3 million.

Table 5 Flight conditions for Sparrow

Flow conditions	Value
Mach number	1.5
Static pressure (Pa)	18114.8
Static temperature (K)	233.793
Angle of attack (°)	0, 5, 11, 17, 23, 27, 32

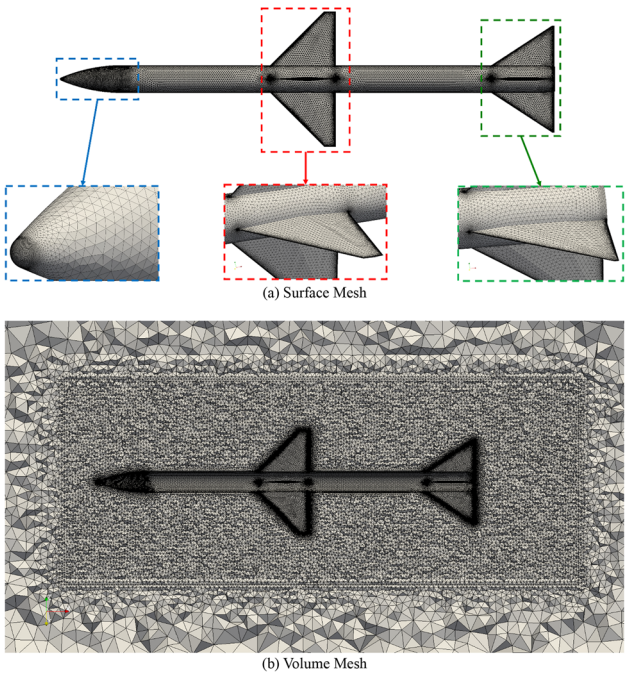
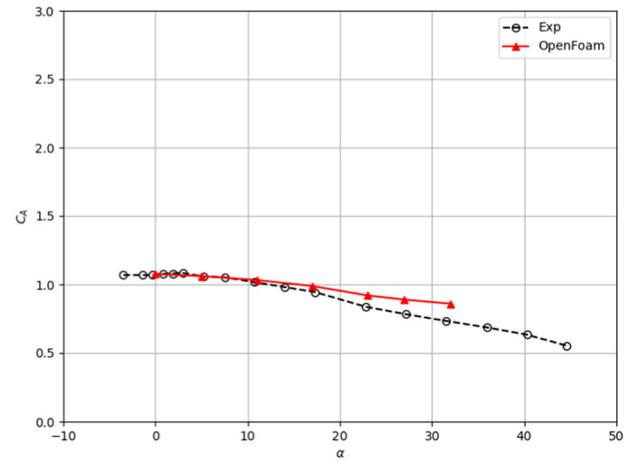
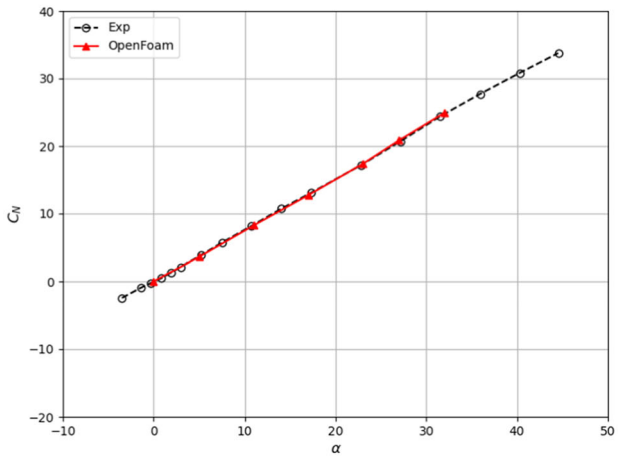


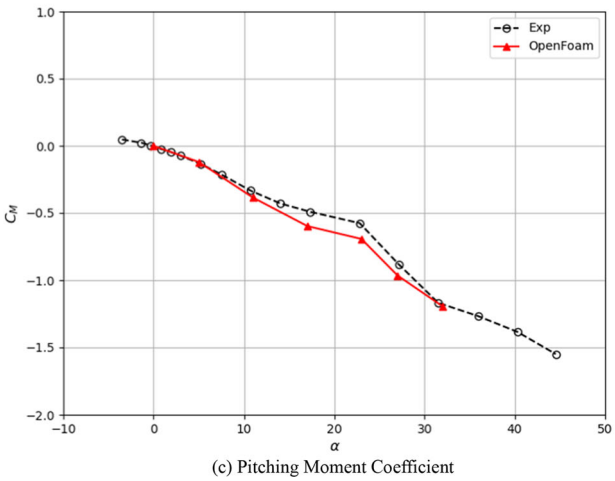
Fig. 17 Automatically generated mesh for Sparrow



(a) Axial Force Coefficient



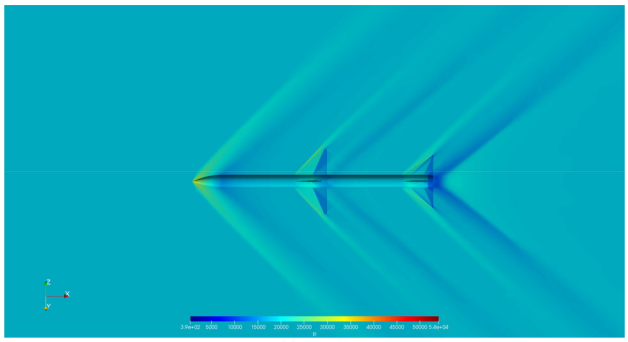
(b) Normal Force Coefficient



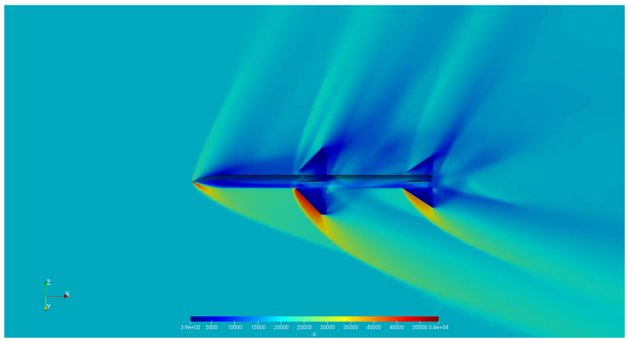
(c) Pitching Moment Coefficient

Fig. 18 Longitudinal aerodynamic coefficients for Sparrow

Figure 18. shows the results of longitudinal aerodynamic coefficients versus the angle of attack. The axial force coefficient (C_A) predicted by TSLAeroFoam is in good agreement with experimental data at low angles of attack, but at high angles of attack, C_A values are little bit over-predicted but match with the overall trend line well. The normal force coefficient (C_N) matches very well with the experiment at all angles of attack, and the pitching moment coefficient (C_M) matches with the overall trend line compared to the wind tunnel test.



(a) $\alpha = 0°$



(b) $\alpha = 32°$

Fig. 19 Pressure contours for Sparrow

Figure 19. shows the static pressure contours for two angles of attack. The shock wave and wake flow are clearly captured for both angles of attack with the help of the refinement region. The pressure distribution and shock wave are symmetric at the zero-degree angle of attack. At a high angle of attack, the shock wave is generated asymmetrically due to flow direction, and higher pressure distribution occurs on the lower surface of the missile.

4.2 NASA Tandem Control Missile

NASA Tandem Control Missile (TCM) [20] is selected as the second test case. The missile configuration is shown in Fig. 20. The model has a tangent ogive nose and cylindrical body, cruciform canards, and tail fins. The canards have a trapezoidal planform with diamond airfoil, and the tail fins have a trapezoidal planform with the modified double-wedge airfoil. The flow analysis is performed on the configuration with b_t/b_c of 1.25 in Ref. [20]. The flight conditions used for the flow analyses are reported in Table 6.

Figure 21 shows the automatically generated surface mesh and volume mesh in the core refinement region. Like the first

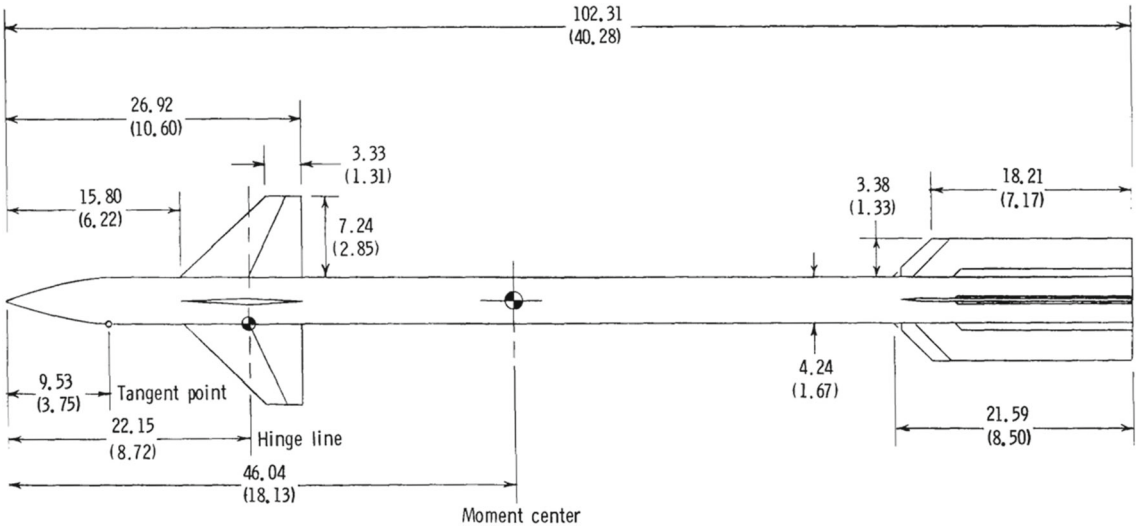


Fig. 20 NASA TP 2157 (TCM) [20]

Table 6 Flight conditions for NASA TCM

Flow conditions	Value
Mach number	3.5
Static pressure (Pa)	1696.55
Static temperature (K)	94.2029
Angle of attack (°)	−4, 0, 4, 10, 14, 18

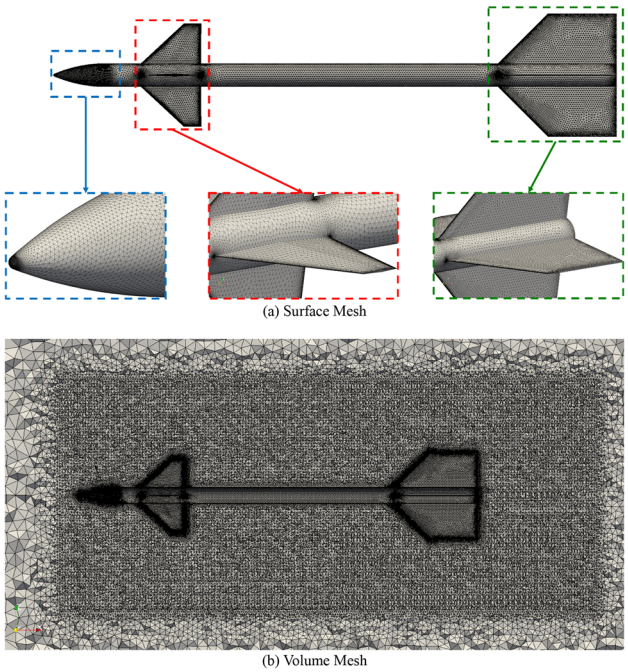


Fig. 21 Automatically generated mesh for NASA TCM

configuration, the spherical far-field region has a radius of 20 missile lengths. The average y^+ is set to be 1.0, and 20 prism layers are created on the missile surface. The overall surface mesh count is 353,508 and the volume mesh count is approximately 25.5 million.

The longitudinal aerodynamic coefficients of NASA TCM are presented in Fig. 22. C_A values from the experiment show some oscillation on the data. TSLAeroFoam little bit over-predicts the axial force compared to the test but matches well in the overall trend line. For normal force and pitching moments, it is in very good agreement at all angles of attack.

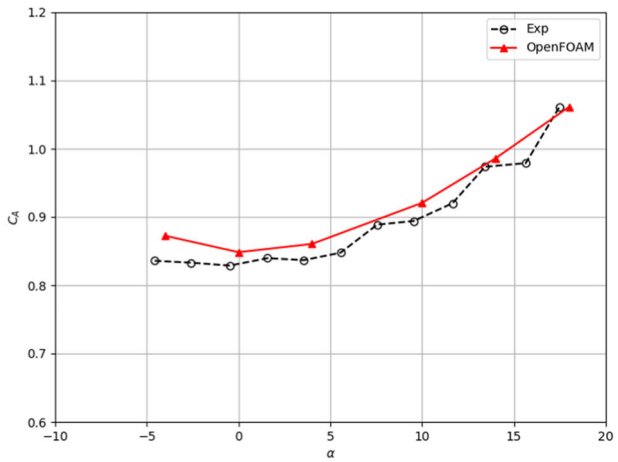
Figure 23 shows static pressure contours for two angles of attack. The shock wave and wake flow are accurately captured again for both angles of attack. The flow field is predicted symmetrically for the zero-degree angle of attack. At a high angle of attack, the shock wave is generated asymmetrically due to flow direction. Also, the large pressure distribution occurs on the lower surface of the missile.

4.3 NASA TM 2005 – Configuration 3

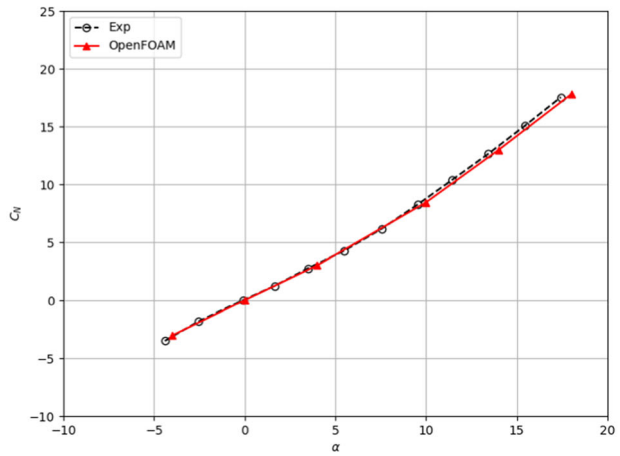
The missile configuration 3 of Ref. [21] is chosen as the last test case. As shown in Fig. 24, Configuration 3 is composed of the body with small nose and tail fin of hexagonal airfoil. The selected flight conditions are summarized in Table 7.

Figure 25 shows automatically generated surface mesh and volume mesh in the core refinement region. The information of far-field region and prism layer is same as other test cases. The overall surface mesh count is 334,382 and the volume mesh count is approximately 22.8 million.

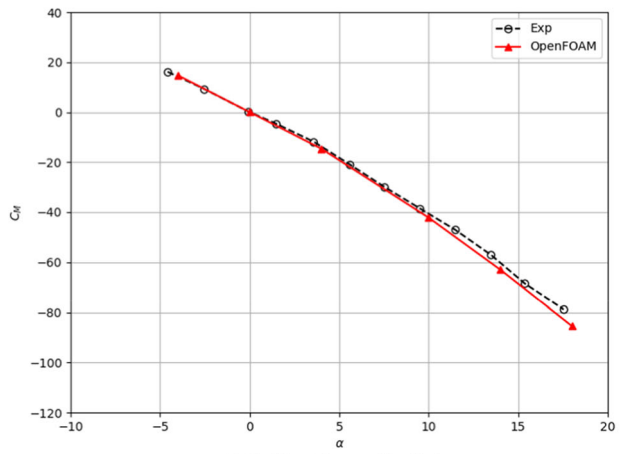
The results of longitudinal aerodynamic coefficients versus the angles of attack are shown in Fig. 26. As mentioned in



(a) Axial Force Coefficient

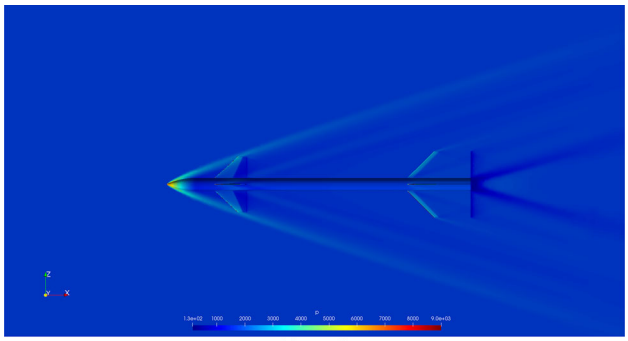


(b) Normal Force Coefficient

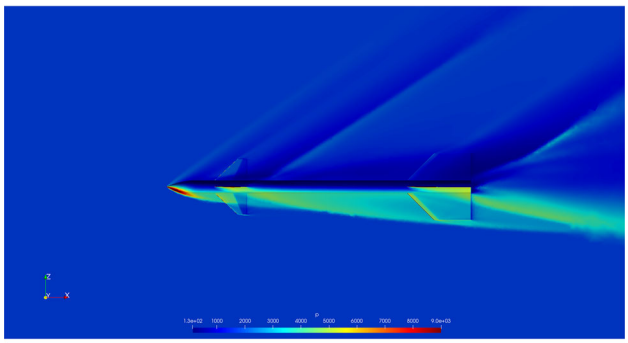


(c) Pitching Moment Coefficient

Fig. 22 Longitudinal aerodynamic coefficients for NASA TCM



(a) $\alpha = 0°$



(b) $\alpha = 18°$

Fig. 23 Pressure contours for NASA TCM

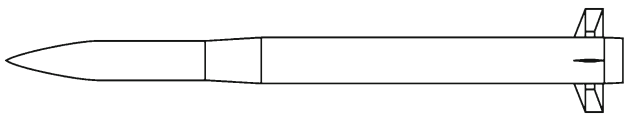
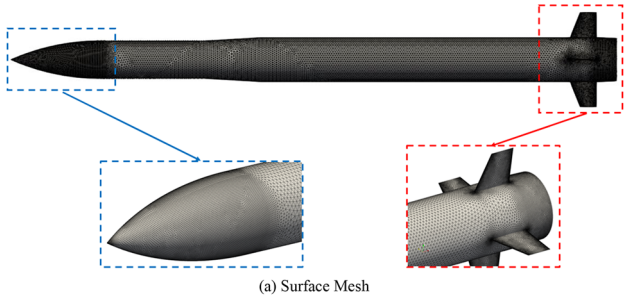


Fig. 24 NASA TP 2005—configuration 3 [21]

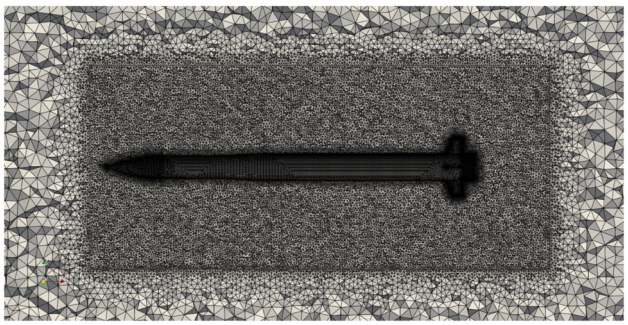
Table 7 Flight conditions for NASA TM 2005—configuration 3

Flow conditions	Value
Mach number	2.36
Reynolds number (per foot)	1.5×10^6
Angle of attack (°)	−1, 0, 2, 4, 6, 17

Ref. [4], there seems to be an error in the position of the center of moment provided in Ref. [21], so the pitching moment is calculated by moving the center of moment 1.75 inches forward compared to the original value. The flow solver overpredicts the axial force for all angles of attack, but predicts the normal force and pitching moment very accurately.



(a) Surface Mesh



(b) Volume Mesh

Fig. 25 Automatically generated mesh for NASA TM 2005—configuration 3

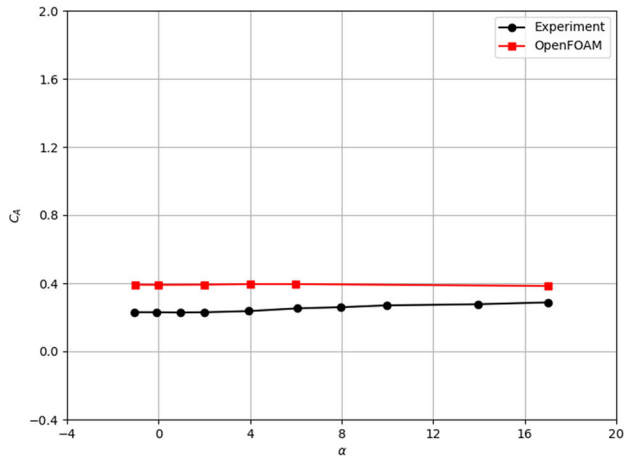
Figure 27 shows static pressure contours for two angles of attack. The shock wave and wake flow are clearly captured like other test cases. At a high angle of attack, the large pressure distribution occurs on the lower surface of the geometry.

5 Conclusions

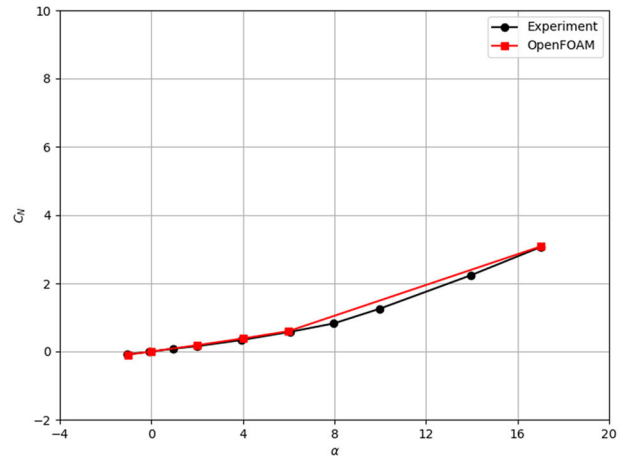
This paper presents the fully automated aerodynamic analysis system to predict aerodynamic characteristics for arbitrary missile configuration in the early design stage. The proposed system includes the modules of geometry modeling, mesh generation, and flow analysis. The required parameters for each module are defined in the XML file format, and all modules are developed using the open-source software such as OpenCASCADE, SALOME, and OpenFOAM.

The automated system is verified with three representative missile configurations. The flow analysis results show that the predicted aerodynamic characteristics are in good agreement with the wind tunnel tests.

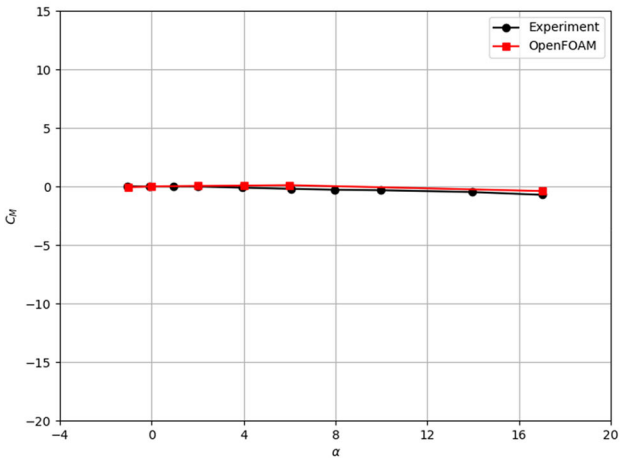
In the following studies, an automated analysis system for predicting the control performance of the finset will be developed.



(a) Axial Force Coefficient

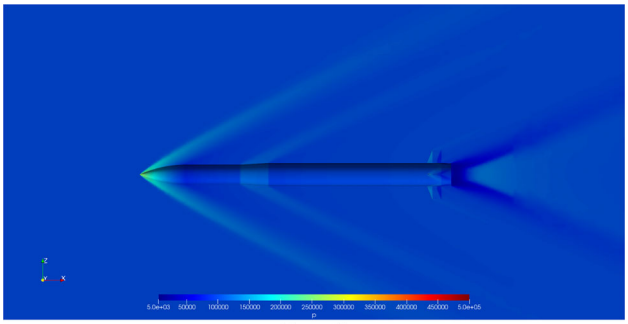


(b) Normal Force Coefficient

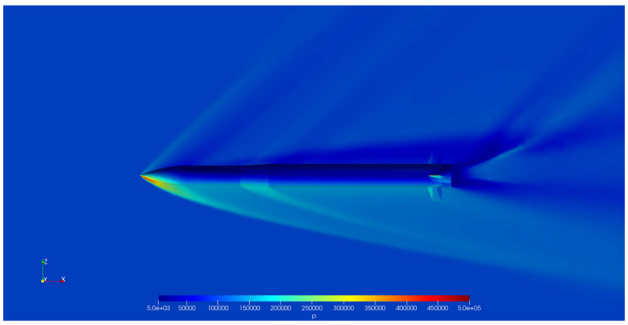


(c) Pitching Moment Coefficient

Fig. 26 Longitudinal aerodynamic coefficients for NASA TM 2005—configuration 3



(a) $\alpha = 0°$



(b) $\alpha = 17°$

Fig. 27 Pressure contours for NASA TM 2005—configuration 3

Acknowledgements This work was supported by Grant (UE191079CD) from Agency for Defense Development (ADD) and Defense Acquisition Program Administration (DAPA).

References

1. Vukelich SR, Jenkins JE (1984) Missile DATCOM: aerodynamic prediction of conventional missiles using component build-up techniques. In: 22nd AIAA aerospace sciences meeting. https://doi.org/10.2514/6.1984-388
2. Moore FG, McInville RM, Hymer TC (1999) Application of the 1998 version of the aeroprediction code. J Spacecr Rocket 36(5):633–645. https://doi.org/10.2514/2.3495
3. Atik H, Erdem B, Ilgaz M et al (2008) Prediction capabilities and comparison of panel, semi-empiric and CFD codes for missile aerodynamic analyses. In: 26th AIAA applied aerodynamics conference. https://doi.org/10.2514/6.2008-6224
4. Doyle JB, Paul J, DeSpirito J et al (2020) Results of the missile and projectile aeroprediction discussion group case study. In: AIAA SciTech 2020 forum. https://doi.org/10.2514/6.2020-1992
5. Marco AD, Stasio MD, Vecchia PD et al (2020) Automatic modeling of aircraft external geometries for preliminary design workflows. Aerosp Sci Technol 98:105667. https://doi.org/10.1016/j.ast.2019.105667
6. Tomac M, Eller D (2011) From geometry to CFD grids—an automated approach for conceptual design. Prog Aerosp Sci 47(8):589–596. https://doi.org/10.1016/j.paerosci.2011.08.005
7. Ordaz I, Li W, Campbell RL (2014) Automated tetrahedral mesh generation for CFD analysis of aircraft in conceptual design. In: 52nd AIAA aerospace sciences meeting. https://doi.org/10.2514/6.2014-0118
8. Gu X, Ciampa PD, Nagel B (2018) An automated CFD analysis workflow in overall aircraft design applications. CEAS Aeronaut J 9:3–13. https://doi.org/10.1007/s13272-017-0264-1
9. Gill JH, Kim BY, Kim JH et al (2013) Development and validation of a density-based implicit solver using LU-SGS algorithm. In: 8th international OpenFOAM workshop
10. Open CASCADE. https://www.opencascade.com. Accessed 27 Aug 2021
11. pythonOCC. http://www.pythonocc.org. Accessed 27 Aug 2021
12. SALOME. https://www.salome-platform.org. Accessed 27 Aug 2021
13. Schöberl J (1997) NETGEN An advancing front 2D/3D-mesh generator based on abstract rules. Comput Vis Sci 1:41–52. https://doi.org/10.1007/s007910050004
14. OpenFOAM. https://openfoam.org. Accessed 27 Aug 2021
15. Underwood ML, Rosema CC, Wilks BL et al (2007) Recent improvements to Missile DATCOM. In: 25th AIAA applied aerodynamics conference. https://doi.org/10.2514/6.2007-3936
16. Menter FR, Kuntz M, Langtry R (2003) Ten years of industrial experience with the SST turbulence model. In: 4th international symposium on turbulence, heat and mass transfer
17. Yoon S, Jameson A (1988) Lower-upper symmetric-Gauss–Seidel method for the Euler and Navier–Stokes equations. AIAA J 26(9):1025–1026. https://doi.org/10.2514/3.10007
18. Steger JL, Warming RF (1981) Flux vector splitting of the inviscid gas dynamic equations with application to finite-difference methods. J Comput Phys 40(2):263–293. https://doi.org/10.1016/0021-9991(81)90210-2
19. Monta WJ (1977) Supersonic aerodynamic characteristics of a Sparrow III type missile model with wing controls and comparison with existing tail-control results. NASA TP 1078
20. Blair AB, Allen JM, Hernandez G (1983) Effect of tail-fin span on stability and control characteristics of a canard-controlled missile at supersonic Mach numbers. NASA TP 2157
21. Allen JM (2005) Aerodynamics of an axisymmetric missile concept having cruciform strakes and in-line tail fins from Mach 0.60 to 4.63. NASA TM 2005-213541