JOURNAL OF SPACECRAFT AND ROCKETS
Vol. 46, No. 5, September–October 2009

Improved Method for Estimation of Spacecraft Free-Molecular Aerodynamic Properties

John D. Fuller* and Robert H. Tolson†
North Carolina State University, Raleigh, North Carolina 27695

DOI: 10.2514/1.43205

A numerical procedure for calculating free-molecular aerodynamic properties of spacecraft is revisited and redeveloped using improved discretization and shading techniques. New complex geometric shapes are included, allowing the user to model a high-fidelity spacecraft model divisible into elemental panels of approximately equal area to conserve resolution. The forces on exposed elements are summed across the entire geometry. New analytic methods for determining exposed panels have been developed. Combined with the use of array algebra, the improvements considerably enhance processing speeds. The fidelity of the data is tested by application to the geometry of the Mars Reconnaissance Orbiter. The resulting data are compared with direct simulation Monte Carlo predictions. Additional case studies serve to inspect implications of geometry design versus output accuracy and processing speed.

Nomenclature

A_e	=	elemental area
C_{Fx}	=	force coefficient in the x direction
C_{Fy}	=	force coefficient in the y direction
C_{Fz}	=	force coefficient in the z direction
C_{Mx}	=	moment coefficient about the x axis
C_{My}	=	moment coefficient about the y axis
C_{Mz}	=	moment coefficient about the z axis
$F_{n,e}$	=	normal elemental force
$F_{t,e}$	=	tangential elemental force
k	=	Boltzmann constant
m	=	atmospheric mean molecular mass
S	=	molecular speed ratio
T	=	freestream temperature
T_w	=	average spacecraft surface temperature
U	=	flow velocity
α	=	aerodynamic pitch angle
β	=	aerodynamic yaw angle
ε_n, ε_t	=	normal and tangential components of unit flow vector to element
θ	=	flow incidence angle
ρ	=	atmospheric density
σ_n, σ_t	=	normal and tangential momentum accommodation coefficients

Introduction

THE aerodynamic properties of spacecraft operating in hyperthermal, free-molecular flow have long been estimated by the calculation of momentum transfer based on gas-surface interaction theory [1–3]. This method has been found accurate when dealing with spacecraft operating at altitudes at which the mean free path between molecular collisions is larger than the spacecraft reference length. Dividing a spacecraft geometry using a discrete element panel method is the most convenient way to apply these calculations, because force contributions are functions of flow characteristics, incidence angle, and exposure. Evaluating exposure of the geometry to the flow suggests the use of the panel method as well. The procedure presented here is a new application of this theory with improved shading and discretization that also takes advantage of modern computer programming language in MATLAB®. This procedure shares the force calculation theory of previous methods, adds additional components, provides a complete redevelopment of all components, and includes new element generation and shading processes. Spacecraft designs may be input in the form of geometric shapes such as plates, prisms, cylinders, cones, frustums, and spherical dishes. These geometric shapes are referred to as components. Each component is discretized into several panels, called elements, of the user-defined resolution. This newly developed discretization process maintains a relatively constant elemental area across the geometry as opposed to routines that can result in elements differing significantly from the user-defined resolution. Smaller elements can lead to increased processing requirements with no added fidelity.

A common problem with estimating aerodynamic forces by means of discrete elements is determining elemental flow exposure, more commonly called shading. Shading analysis requires the most processing time of the method and is important when dealing with complex spacecraft geometries in which components are shielded by instruments, solar panels, and dish antennas. In this approach, elements that are shielded from the flow do not contribute aerodynamic forces. The ability to ascertain which elements are not exposed to the hyperthermal flow is essential to this analysis and can hamper processing speed if not handled appropriately. Analytic shading processes are developed using array manipulation, in which several elements can be analyzed with respect to a single shading component at a time. These analytic processes constitute a new design of the original procedure.

The implementation of this procedure into a MATLAB program, entitled FreeMat, sought to allow users to easily create a spacecraft geometry to complex specifications while maintaining minimal processing time for varying orientations. Using the array manipulation capabilities of MATLAB in combination with the new shading processes enabled a significant reduction in processing speeds. Specific components of satellites may be easily defined with the included component types. The program generates a set of aerodynamic coefficients given a range of yaw and pitch angles. This set of data may then be interpolated for trajectory calculations. For satellites that tumble or present a large variety of orientations to the flow, an alternate approach would be to calculate the aerocoefficients during the trajectory analysis. The program is applied to the Mars

Presented at the 2009 AIAA Aerospace Sciences Meeting, Orlando, FL, 5–8 January 2009; received 13 January 2009; accepted for publication 20 April 2009.

*Graduate Researcher, Department of Mechanical and Aerospace Engineering. Member AIAA

†Professor, Department of Mechanical and Aerospace Engineering. Associate Fellow AIAA.

Reconnaissance Orbiter (MRO) spacecraft and compared with results from direct simulation Monte Carlo (DSMC) analyses. DSMC is a widely used aerodynamic analysis tool that can be used to calculate both free-molecular and transition flow forces on spacecraft. A comparison with the DSMC results is used to validate the fidelity of the procedure. Additional case studies are performed to provide an understanding of the processing time required and data fidelity with respect to the desired resolution and model complexity.

Force and Moment Calculation

The equations [1] by which forces on the elements are calculated are the same as in the original procedure developed by Fredo and Kaplan [4] and Fredo [5], wherein a separate coordinate frame was introduced at each element centroid. The FreeMat procedure eliminates the transformation to the element coordinate frame, instead calculating the forces with the normal and tangential components of the gas flow at each element in the aerodynamic frame. The normal and tangential forces on a surface element are [1]

$$F_{n,e} = \frac{\rho A_e U^2}{2S^2}\left(\left[\frac{2-\sigma_n}{\sqrt{\pi}} S\varepsilon_n + \frac{\sigma_n}{2}\sqrt{\frac{T_w}{T}}\right]\exp[-(S\varepsilon_n)^2] + \left\{(2-\sigma_n)\left[\frac{1}{2}+(S\varepsilon_n)^2\right] + \frac{\sigma_n}{2} S\varepsilon_n \sqrt{\frac{T_w \pi}{T}}\right\}[1+\operatorname{erf}(S\varepsilon_n)]\right) \tag{1}$$

$$F_{t,e} = \frac{\rho A_e U^2 \sigma_t}{2S^2\sqrt{\pi}}\{S\varepsilon_t \exp[-(S\varepsilon_n)^2] + S^2\varepsilon_n\varepsilon_t\sqrt{\pi}[1+\operatorname{erf}(S\varepsilon_n)]\} \tag{2}$$

where ε_n represents the unit velocity component in the normal direction with respect to a surface element. The normal and tangential velocity components are obtained from the cosine and sine of the incident flow angle θ, respectively. The molecular speed ratio S is the ratio of the freestream flow velocity U to the mean thermal velocity at freestream temperature T:

$$S = \frac{U}{\sqrt{2kT/m}} \tag{3}$$

The tangential force lies along a vector tangent to the element surface, in the plane made by the element normal vector and flow velocity vector. The moment contribution from the element can be obtained by resolving the cross product between the element centroid location with respect to the spacecraft center of mass and the total force on the element. The overall force and moments on the spacecraft are then found by integrating the contributions from the exposed elements over the entire surface. Force and moment coefficients in each body axis are obtained from the final output.

Geometric Modeling

The geometric components used to represent the spacecraft are decomposed into panels based on a user-input maximum elemental area. These components and their elemental decompositions are shown in Fig. 1. Additional capabilities as compared with previous works include the inclusion of cone frustums, spherical dishes, and scalar triangular plates/prisms. Every element has a specific active side on which to calculate aerodynamic pressures. Single plates have only one active side, and three-dimensional components are defined such that the outside surface is active. The inactive sides of elements cannot be exposed to the flow during analysis. The consequences of these constraints include that all cylinders, cones, or dishes must be "capped" such that their inner surfaces are not exposed. Because inner surfaces of these components are not active, leaving one uncapped results in a "hole" in the spacecraft profile for which no force calculations are performed. Considering the applicable designs of spacecraft for analysis, this should not be an unreasonable requirement.

A unique characteristic of discretization that this procedure uses is the attempted conservation of elemental area across an entire component. Past implementations [4–7] of discrete paneling produced elements that were a fraction of the user-specified element area, such as those at the center of a circular plate or the tip of a cone. Other cases would allow elements to be too large, causing inaccurate shading results. Variably sized elements either mean the geometry is not adequately resolved at some points or overresolved with a subsequent increase in processing time. A visual comparison of these methods applied to relevant component types is shown in Fig. 2.

The performance gains from using the constant area approach can be estimated by calculating the number of elements generated for each component. Table 1 lists the discretization of these components with 1 m^2 of surface area, with a desired element area of 10 cm^2. The plates are one sided, and no elements are used in the base of the cone. For uniform spacing, the circular plate, sphere, and cone are discretized such that the largest elements are approximately of the desired element area. The ideal case is composed of 1000 elements, each 10 cm^2 in area. This new procedure ensures an efficient distribution of elements that is nearly that of the desired resolution.

Geometric components within the FreeMat program are separated into two categories: fundamental components and supercomponents. Fundamental components are basic shapes that are each considered for shading analysis; these include plates, cones, cylinders, dishes, and spheres. Supercomponents are shapes that are constructed of fundamental components, such as double-sided plates and prisms. Supercomponents are included for convenience in construction and are decomposed into their respective fundamental components for shading and force analysis in a preprocessor. Users may choose to define supercomponents or fundamental components in the body coordinate frame with an origin coordinate along with the Euler orientation angles and dimensions or with the definition of the

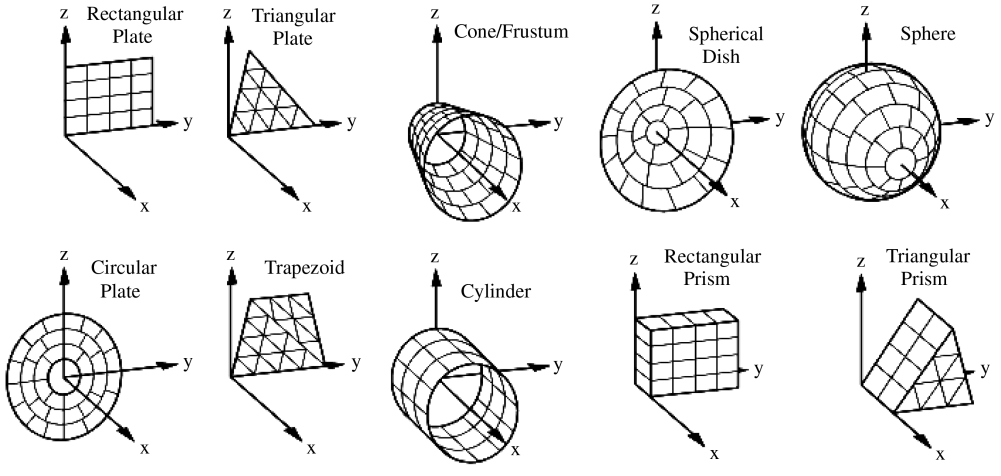

Fig. 1 Component coordinate systems and schema for subdivision into elements.

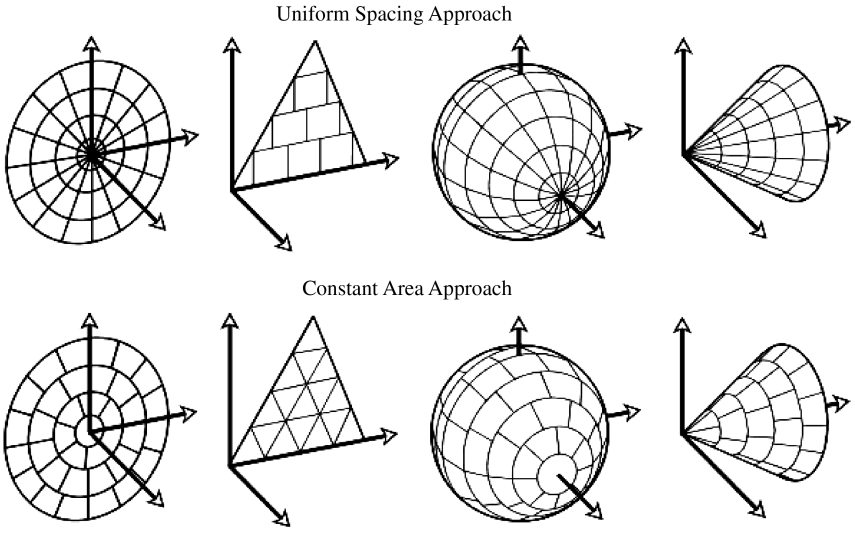

Fig. 2 Comparison of discretization techniques.

component vertices. The geometry input is structured within a text input file.

FreeMat includes three coordinate frames: the component reference frame, the body reference frame, and the aerodynamic frame. The component reference frame is defined differently for each type of component, with the axes oriented as shown in Fig. 1. The body reference frame is the coordinate system in which the geometry is designed. Determining the aerodynamic properties of an orientation of the craft requires the body frame to be rotated within the aerodynamic frame for which shading and force calculations are implemented. This rotation is accomplished with a yaw angle and pitch angle in sequence, defined as β and α, respectively. Figure 3 shows a rotation by a positive yaw and negative pitch. The aerodynamic frame is oriented such that the positive Z_A axis points into the flow. The unit velocity vector is the negative Z_A axis location within the body frame. The rotation between these coordinate frames is defined by

$$\begin{Bmatrix} X_A \\ Y_A \\ Z_A \end{Bmatrix} = \begin{bmatrix} \cos\beta & \sin\alpha\sin\beta & \cos\alpha\sin\beta \\ 0 & \cos\alpha & -\sin\alpha \\ -\sin\beta & \sin\alpha\cos\beta & \cos\alpha\cos\beta \end{bmatrix} \begin{Bmatrix} X_B \\ Y_B \\ Z_B \end{Bmatrix} \quad (4)$$

Shading Analysis

Determining which elements are shaded by other components is the most costly process of this procedure in terms of processing time. Aerodynamic pressure is assumed to be zero on elements shaded by an upwind component or for which the active side faces away from the flow. Because elements are grouped with respect to their parent components, shading relations can be drawn from the component locations without introducing the element centroid locations until necessary. To speed computation, a preprocessor is implemented to determine these relations without introduction of element locations.

A single shading comparison consists of considering one component lying between another component and the free-molecular flow. For a spacecraft model composed of N components, there are N^2–N possible shading cases, because for every two components there are two comparisons and no components may shade themselves. FreeMat records shading within a square, binary array. The rows represent shading components whereas the columns represent shaded components. The array is filled with ones and zeros according to possible shading relations, wherein 1 means shading is possible while 0 means shading is impossible. A 1 located in cell (5, 8) of the array means that component 5 may possibly shade component 8. Conversely, a 0 located in cell (5, 8) means that component 5 cannot shade component 8. This array is updated before final shading calculation to expedite processing.

Maximum Extent Projection

In the interest of minimizing shading calculations, several methods have been applied to detect irrelevant shading cases and discard them before the final shading analysis. Many shading instances can be disregarded based on the component geometry. Cases in which it is impossible for a particular component to shade another particular component will be referred to as irrelevant shading comparisons. For example, when analyzing the shading possibilities of the rectangular and/or triangular plates that compose a prism, it can be established that no two plates within the prism may shade one another. Leeward plate components do not experience molecular collisions, and they also cannot shade other components from the flow. In addition to these inherent relations, other irrelevant shading cases may be determined by the maximum extent projection technique.

Past applications [4–6] have used a type of shadow projection technique in which the profile of the spacecraft seen by the flow is projected onto a series of vertical columns. Shading relations were then established by determining which components shared a particular column, narrowing the subset of shading components to be checked for each particular element. FreeMat, conversely, uses the extents of each component in all three axes of the aerodynamic frame to determine shading possibilities. This procedure, named maximum extent projection, minimizes the number of shading comparisons without requiring the generation of element centroids. The maximum extent projection process is referred to as shade 1.

The spacecraft is projected in the X_A–Y_A plane, representing the geometry as seen by the incoming flow. By circumscribing bounding

Table 1 Element totals of discretization techniques

Discretization		Circular plate	Equilateral triangular plate	Sphere	Cone ($R = h$)
Uniform spacing approach	Element count	2034	1091	1653	2090
	Mean A_e, cm^2	4.9164	9.1659	6.0496	4.7847
Constant area approach	Element count	1009	1024	1020	1014
	Mean A_e, cm^2	9.9108	9.7656	9.8039	9.8619

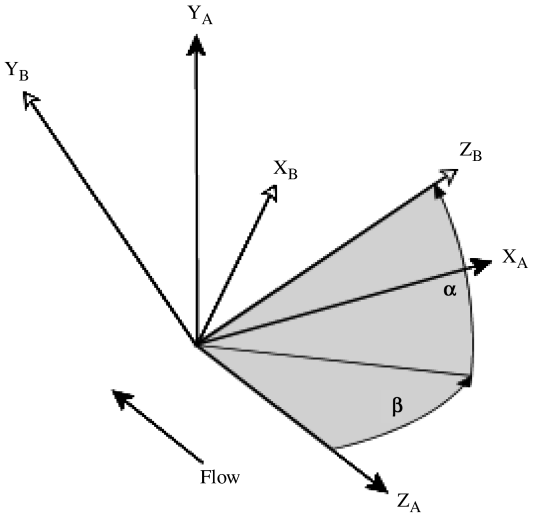

Fig. 3 Rotation between body and aerodynamic frames.

rectangles with sides parallel to the X_A and Y_A axes about each component, irrelevant shading comparisons can be determined from components for which the limits do not overlap in the plane. Finding the bounding rectangle of each component can be performed analytically based on the type of profile projected onto the X_A–Y_A plane. Once all extents have been determined, shading possibilities between any two components are determined. Furthermore, in the cases of components that share bounding rectangles, the minimum and maximum extents in the Z_A axis can be compared to find cases for which one component lies completely behind or in front of another component.

This process can be demonstrated by a simple arrangement of three cubes A, B, and C in the aerodynamic frame, as shown in Fig. 4. In comparison 1, the minimum and maximum X_A and Y_A limits of the cubes' projections are compared. It is seen that cubes A and B may possibly shade one another because the projection of cube A overlaps the projection of cube B. Cube C cannot shade or be shaded by either because its y_{min} is greater than the y_{max} of both cubes A and B. Comparison 2 is a further assessment of possible shading candidates to determine if one is in front of another. As shown, cube B is in front of cube A because the Z_A boundaries do not overlap. Instead of testing all six possible cases of shading, the shade 1 process of FreeMat has narrowed the search to only include analysis for the case of cube B shading cube A. The comparison of maximum extents is improved from past versions of this procedure [6] by comparing the limits in all three dimensions. The process requires no inclusion of the particular elements of a component and relies on the geometric limits of the fundamental components alone.

Examples of upwind-view profiles of the main fundamental components are shown in Fig. 5, along with the corresponding maximum bounds. Rectangular and triangular plates are the simplest components for which to find the boundaries. The user input defines the vertices of the plate, and these vertices define the bounding rectangle that circumscribes the plate. The vertices are compared, and the limits are derived from those coordinates. To determine the boundaries of a circular plate with respect to the flow, visualize a disk suspended in the aerodynamic frame. With respect to the flow, this disk looks like an ellipse with a semimajor axis equal to the original circular plate and a semiminor axis that is the original radius times the cosine of the incidence of the plate to the flow. In other

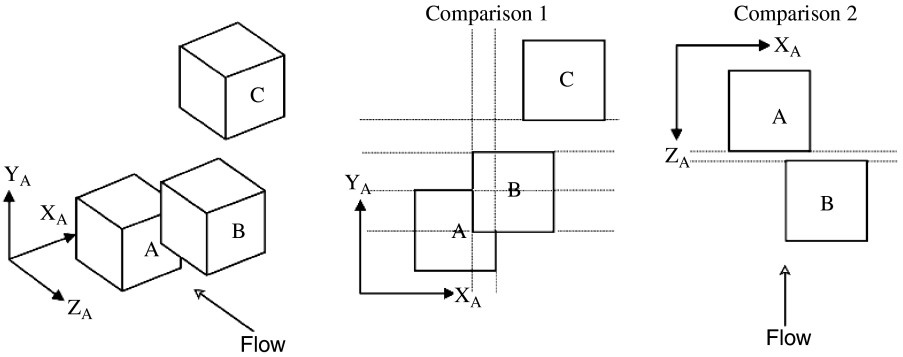

Fig. 4 Maximum extent projection technique.

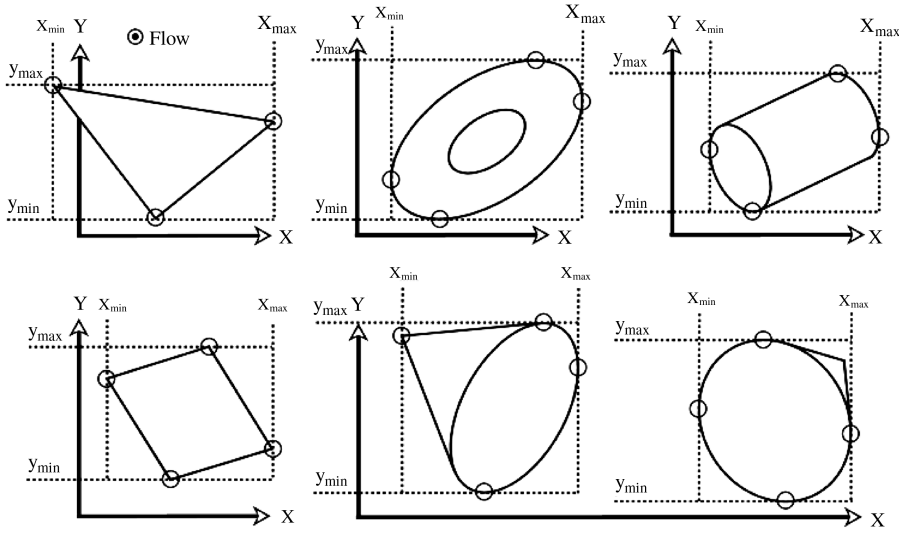

Fig. 5 Maximum extent projection boundaries by component type.

words, the equation of the ellipse in the X_A–Y_A plane is fully described by the plate's radius, centroid location, and incidence to the flow. The boundaries of the ellipse may then be determined with analytic geometry.

The circular plate maximum extent method is also used for other components with elliptical profiles, such as cylinders and cones. A cylinder or cone frustum profile consists of two ellipses in the X_A–Y_A plane. Limits for these ellipses are found as though both were separate circular plates. If a cylinder/frustum has an axis of symmetry normal to the flow, the profile becomes a rotated rectangle/trapezoid with vertices that constitute the possible minimum and maximum extents of the component. The cone profile is slightly different in that it can be constructed with one ellipse and a single point representing the vertex. The bounding rectangle of the cone profile can only be attributed to the vertex or to the maximum reaches of the ellipse. The limits of the ellipse can be found using the ellipse limit method and then compared with the vertex coordinates, giving the overall minimum and maximum boundaries. Figure 5 shows two different cases of such a cone profile, for which the vertex does and does not play a role in the boundaries.

The maximum boundaries of spheres are trivial to determine because the profile of a sphere, in any orientation, is a circle. These boundaries represent a cube of side length twice the sphere radius in the aerodynamic frame. The spherical dish, however, is the most complex component type that is included in FreeMat geometry. It can take on the shape of a half-sphere or that of a shallow dish similar to a circular plate. Though the rim of the dish may be treated as an ellipse, the possibilities of the profile shape are varied and finding the maximum extents is challenging. A series of coordinates along the crescent edge projected into the X_A–Y_A plane are generated at 3 deg increments. These coordinate locations as well as the minimum and maximum extents of the dish rim are sorted for the limits of the bounding rectangle. Figure 6 shows this method, which finds the maxima of the dish within 0.03% of the radius of the dish.

Elemental Exposure Determination

Once all irrelevant shading comparisons have been discarded, the remaining cases are analyzed to determine which elements are shaded and thus not included in pressure calculations. This procedure uses the array manipulation capabilities of MATLAB to enhance processing speed. Each shading scenario is considered between two fundamental components that share bounding limits as determined in the maximum extent projection technique. The shaded fundamental component is decomposed into its respective elements in the component frame and defined in the aerodynamic frame, whereas only the geometric boundaries of the shading fundamental component in the aerodynamic frame are required in the analysis. This portion of the procedure represents a complete redesign of shading analytics as compared with previous methods.

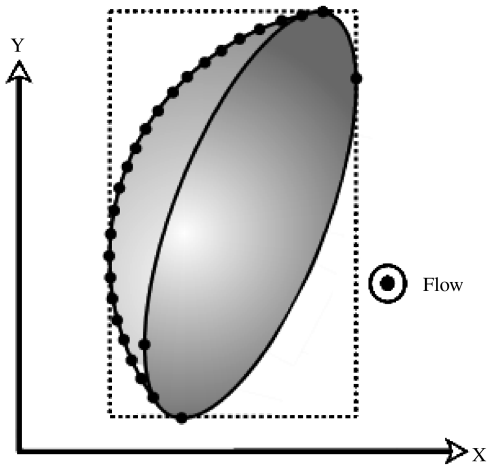

Fig. 6 Spherical dish bounding rectangle.

Each shading fundamental component will project a profile onto the X_A–Y_A plane based on the type of component and orientation of the spacecraft with respect to the flow. The geometry of the shaded fundamental component is irrelevant because the elemental centroid locations in the aerodynamic frame are defined. Sorting through these centroids to determine which lie within the boundaries of the projected shading profile as well as behind a defined plane through the shading component will establish a subset of shaded elements. Array processing allows the comparison to be conducted much quicker than older FORTRAN languages in which other free-molecular tools have been written, which had to compare each element to the shading component sequentially. The method presented could also be applied using any language that supports efficient array arithmetic and logical operations.

The following subsections will describe each shading case categorized by the shading component type. A more thorough description can be found in [8]. This process is named shade 2 within FreeMat. All shading calculations take place in the aerodynamic frame, focusing on the X_A–Y_A plane projection. Each algorithm consists of a function that requires the input test group of possibly shaded elements for analysis as well as the geometric data of the shading component. The output is a binary list corresponding to the elements, for which 0 represents a shaded element and 1 represents an active element. After all shading cases have been considered for that component, the list of active elements is used for aerodynamic force calculations.

This process is the most costly portion of programming in terms of processing speed, and so many considerations to increase efficiency are applied. The use of array processing is maximized to eliminate "for" loops. Specific processes that save time include the elimination of leeward plates from shading analysis, because a plate that faces away from the flow experiences no aerodynamic pressure nor can it shade any other fundamental component. The user may also specify a minimum area multiplier that will eliminate components for which the surface area does not meet that multiple of the elemental area from the shading analysis. This multiplier can reduce processing times for cases with large elemental areas that do not accurately define smaller components. Implications of this method will be further discussed in the performance section.

Triangular, rectangular, circular, and trapezoidal plates are fundamental components for which the profiles are polygons. All shading cases dealing with a polygon shading component will be dealt with using the logic that for every n-sided polygon there are n equations of lines as well as one planar surface that describe shading. By projecting the polygon into the X_A–Y_A plane and determining which elements lie within these lines *and* behind this planar surface, a shaded coordinate set may be determined. Consider a rectangular plate shading a plane of elements, as shown in Fig. 7.

The shaded points share a common trait with the centroid in that they obey the same relationships to the plate boundaries. The centroid lies above the lower and leftmost lines and below the upper and rightmost lines. This idea forms the basis of theory behind polygon shading. By establishing the location of any polygon centroid, the points that lie within the boundaries in the X_A–Y_A plane may be determined. A line between each pair of adjacent vertices is calculated, and the points lying on the other side from the

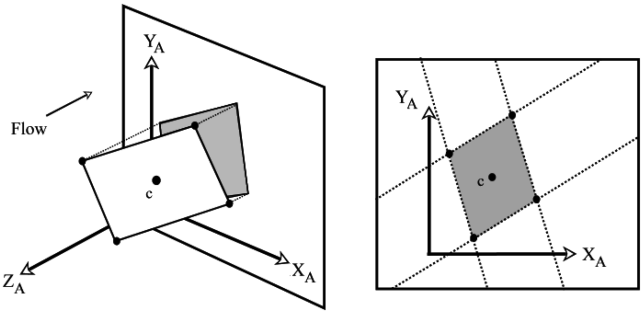

Fig. 7 Rectangular plate shading scenario.

polygon centroid will be omitted. Once all lines have been considered, the remaining elements for which the centroid Z_A values fall downwind of the plane made by the plate surface will be shaded. This method will work for any n-sided polygon plate that acts as a shader, allowing users to include their own type of plates as they deem necessary.

The shadow of a circular plate in the X_A–Y_A plane will be an ellipse that is a function of the plate radius, center coordinate, and incidence to the flow. Arithmetic determines which elements in consideration are inside the elliptical shadow as well as behind the plane made by the plate. Additionally, if the circular plate has a hole in the center, then the points lying inside the ellipse projected in the X_A–Y_A plane by the hole are not shaded. Shading for a sphere is also determined using this process, because a sphere will always project a circular profile, equivalent to a circular plate at zero incidence.

There are several scenarios of shading that are encountered with cylinders, cones, and cone frustums; however, there are some similarities that allow them to be grouped together. Any X_A–Y_A projection of the aforementioned shading components will consist of, at most, two ellipses connected by a pair of lines representing the sides of the geometry. Example profiles of these components are shown in Fig. 8. A cylinder will consist of two parallel sides that meet the ellipses at their major axes, whereas a cone frustum will have nonparallel sides meeting the ellipses at points of tangency. A cone will have two intersecting sidelines that meet the ellipse base at points of tangency. Each case will use the method for single ellipses introduced earlier, with special considerations for each shading component type.

A cylinder can be fully defined by a radius and the centroids of the end plates. Any projection of this cylinder onto the X_A–Y_A plane can be described by two ellipses and a rectangle for which the vertices reside at the outermost extents of the ellipses. Thus, any element of the shaded plane for which the coordinates lie within the bounds of these three regions are possible shaded elements. Elements within this group that also fall behind the planes made by the two end plates and the plane made by the vertices of the rectangular region are shaded by the cylinder. This procedure is accomplished by the ellipse shading process for each end plate and then the polygon shading process for the rectangular region.

Cone frustum shading is similar to cylinders in that the projection is composed of two connected ellipses. The projection of a cone has a single vertex coordinate in the place of one ellipse. The processes used are the same as for cylinders, using the ellipse and polygon shading methods to determine the points that lie behind the corresponding regions of the projection. The differences arise when finding the coordinates of the vertices that bound the trapezoidal/triangular region. Because the sides are not parallel, they do not intersect the ellipses at the major axes but instead at points of tangency.

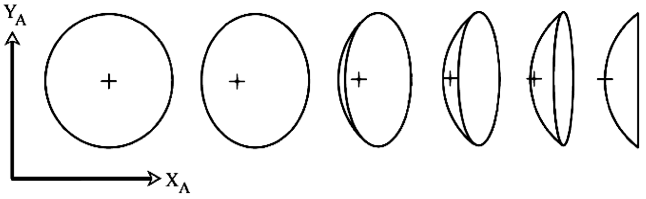

Fig. 9 Rotating spherical dish X_A–Y_A projection.

The spherical dish is the most complex geometric shape within the procedure, both to generate and process for shading. The difficulty of projecting the spherical dish into the X_A–Y_A plane can hamper the shading analysis. To visualize these projections, imagine a case in which the dish fully opposes the flow and is slowly rotated to where the rim is tangential to the flow, as shown in Fig. 9. If the dish were to continue rotating, it would project the same profiles progressing from right to left as the dish rim normal becomes parallel with the flow again.

It is sufficient to assume that any spherical dish projection can be described by a Z_A rotation of an interim case from this scenario. With this in mind, the general case of shading would dictate that any elements behind the ellipse projection as well as behind the crescent region should be shaded. As elliptical shading has already been determined, the difficulty enters when trying to quantify the crescent region. The crescent region is bound by a circular arc of a radius matching that of the sphere from which the dish was cast. The arc intersects the ellipse at points of tangency on the ellipse depending on the incidence to the flow. Cases in which the crescent is visible may be determined analytically using dish geometry and the angle of incidence of the dish rim to the flow.

The points of tangency can be determined and, by comparing the location of the sphere origin to the line drawn through these coordinates, the elements shaded by this region can be determined, as shown in Fig. 10. In this example, the center of an imaginary sphere tangent to the dish surface is beneath the line between the two coordinates of intersection. Thus, the elements for which the centroid coordinates in the aerodynamic frame are above this line as well as contained within the circular arc are shaded. Additionally, any elements contained within the rim ellipse and behind the plane it creates are shaded, fully defining the shading boundaries of the spherical dish.

Mars Reconnaissance Orbiter Application

The Mars Reconnaissance Orbiter is a satellite equipped with a high-resolution camera to study the surface of Mars in greater detail than ever accomplished. Launched in 2005, it used aerobraking upon arrival at Mars to achieve the required science orbit. The geometry of

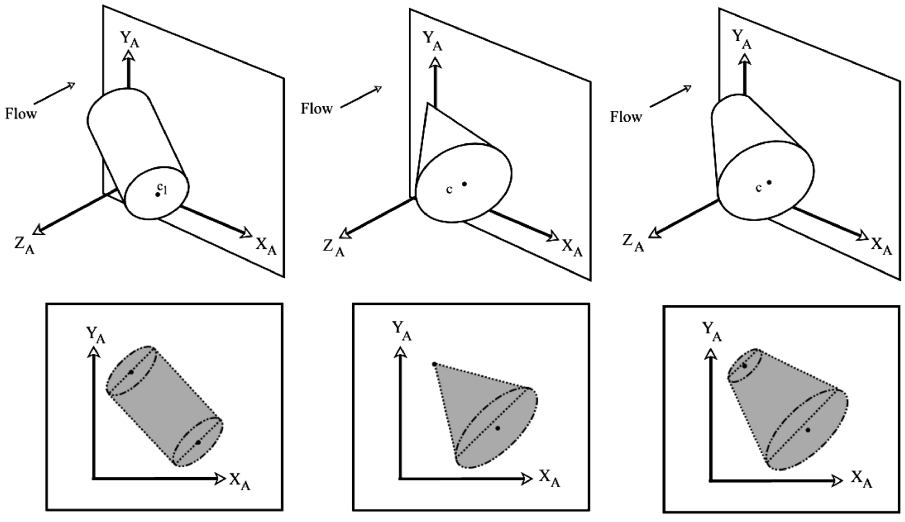

Fig. 8 Cylinder, cone, and frustum shading scenarios.

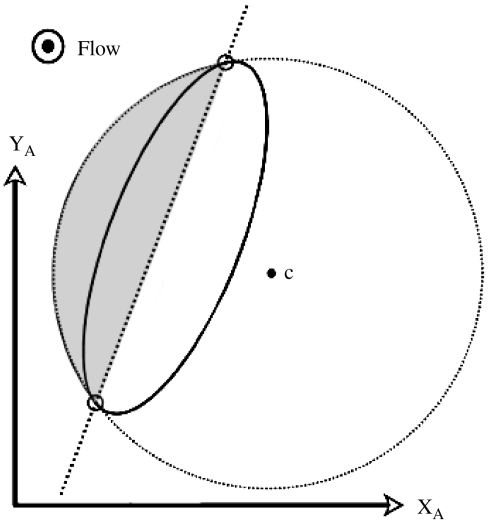

Fig. 10 Crescent shading region.

the MRO spacecraft was input into FreeMat as a collection of geometric figures representing the solar array, dish, main bus, and instrument decks. The model was designed to match the MRO configuration during the aerobraking phase. The geometry submitted for analysis was a collection of 220 fundamental components that required approximately 30 h to design. The versatility of the input format for FreeMat allows the creation of a highly detailed geometry. The body axes are centered at the theoretical aerobraking center of mass such that the Z_B axis extends out the main thruster, the X_B axis extends along the port solar array, and the Y_B axis extends to zenith along the high-gain antenna. Figure 11 shows a comparison of the science deck geometry and the elemental discretization from FreeMat. The MRO geometry was analyzed with FreeMat on a range of pitch angles $-40 \text{ deg} \leq \alpha \leq 40 \text{ deg}$ at no yaw with full momentum accommodation, with a freestream velocity of 4000 m/s and density of $3e\text{-}8 \text{ kg/m}^3$. The runs required between 2 and 4 s on an Intel Centrino® dual core machine to process at each orientation with a target element area of $dA = 8 \text{ cm}^2$.

Evaluating the fidelity of the output from the improved procedure in FreeMat required comparison with proven methods. Direct simulation Monte Carlo [2] is a numerical method for modeling rarefied gas flows and their interaction on spacecraft surfaces. A given geometry is input and subjected to a simulated hypersonic molecular flow. Intermolecular collisions and molecule-surface collisions are calculated using probabilistic, phenomenological models. The DSMC method assumes that the molecular movement and collision phases can be decoupled over time periods smaller than the mean collision time [2]. Conducting DSMC runs at free-molecular conditions represents a comparable test for output from the procedure introduced. The data output from the DSMC code for the Mars Reconnaissance Orbiter (Δ) at comparable flow conditions is compared with output from FreeMat in Fig. 12. The DSMC results are provided in [11]. The DSMC model consists of a triangular plate mesh of varying resolution [9–11] with mean and median element areas of approximately 12 and 1 cm^2, respectively. The FreeMat model was run with a mean element area of approximately 8 cm^2. The reference area and length used are 36.771 m^2 and 13.530 m, respectively. A single orientation analysis for DSMC on a comparable machine requires approximately 30 s, and FreeMat requires between 2 and 4 s. The force coefficients differ from the DSMC by no more than 1% for the data points given. The moment coefficient differs by less than 5% at points not including those near zero.

Performance Process Time Case Study

To gain insight into the factors behind analysis time in this procedure, recall the structure of the theory on which it operates. The method operates on the breakdown of component geometry into several elements. Each coordinate is reviewed for shading characteristics at each orientation, and then the final flow calculations are applied to the active elements. The biggest time consumer in this process is the shading review of each element, or the shade 2 process. Thus, the largest contributing factor in processing time is the number of elements for analysis. This number is directly tied to the user-input desired elemental area, or resolution. Because higher resolutions (smaller elemental areas) will give more accurate results, a compromise between analysis time and data accuracy must be made.

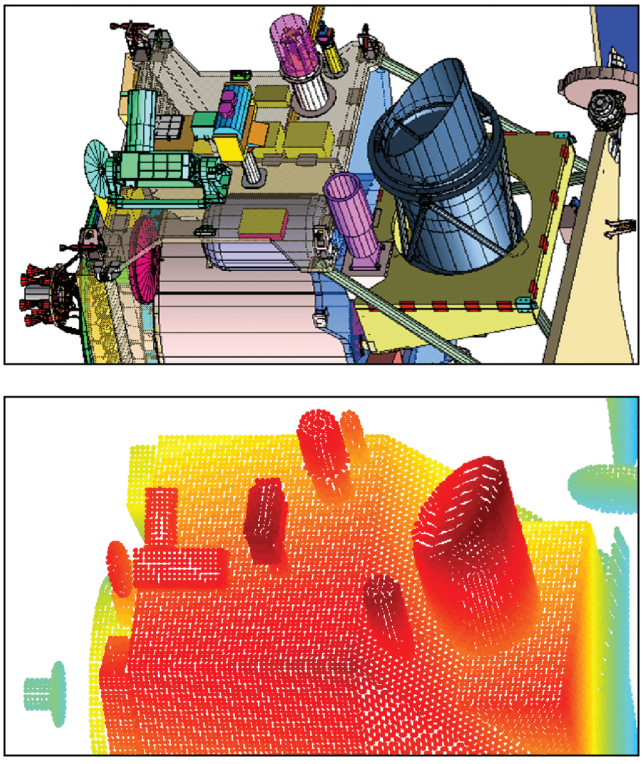

Fig. 11 Comparison of MRO CAD geometry to FreeMat geometry.

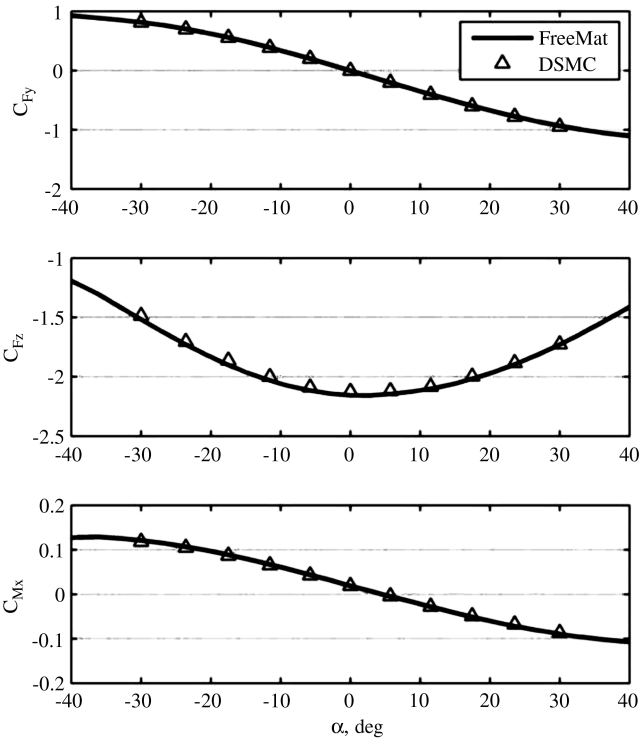

Fig. 12 Aerodynamic coefficients for MRO $\beta = 0$ deg.

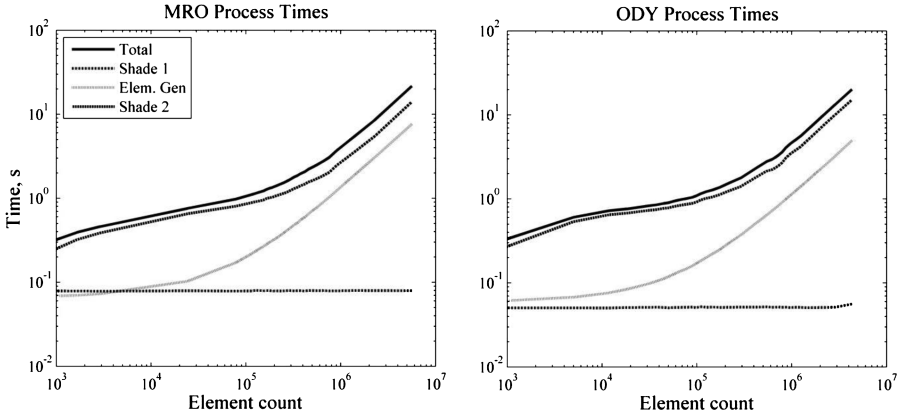

Fig. 13 MRO and ODY analysis process timing.

This section will focus on the implications of analysis time versus resolution.

A case study was performed involving two spacecraft geometries. These models correspond to the Odyssey (ODY) and MRO spacecraft. The geometries were analyzed using FreeMat with varying resolutions for a set of orientations with a minimum area multiplier of 3. The analysis was conducted on an Intel Centrino® dual core machine. Each orientation and resolution was run five times and the mean process times were taken. The log plots in Figs. 13 and 14 show the resulting data. The total time is shown as well as the time for each process. The maximum extent shading process and element exposure shading process are labeled as shades 1 and 2, respectively.

The trends in the data confirm that total analysis time is inversely related to resolution, with an exponential increase in run time as element count increases. Note that the shade 1 process remains independent of element count N, because the process does not take any element locations into consideration. Shade 1 reduces the shade 2 shading comparisons conducted from the worst case of 48,180 to approximately 1000 for MRO and from 19,460 to approximately 1200 for ODY. This requires little processing time due to the analytic method used in shade 1 and its independence from element coordinates. Figure 14 confirms that the MRO model has approximately 4 times the surface area of the ODY model, for which similar element counts correspond to a larger elemental area for the MRO model. The reference areas used for ODY and MRO are 10.500 and 36.771 m^2, respectively.

Process time is dependent mainly on two factors: fundamental component count and element count. The amount of fundamental components in each model stays constant for this analysis at high element counts, when ODY and MRO have 140 and 220 fundamental components, respectively. From the timing plot, it is evident that for the element generation and shade 2 processes at higher resolutions, time is proportional to element count, or $T \propto N$, where N is the number of elements. This is applicable in the region where the slopes of the curves asymptotically approach 1. At low element counts, the element generation and shade 2 process times are driven by both element count and component count. Without the use of a minimum area multiplier, the slope of these curves would approach zero when component count is the only factor behind process time. The data shown includes a minimum area multiplier of 3, which means components for which the surface areas do not exceed 3 times the element area chosen are eliminated from the analysis. This is confirmed by the drop in the curve at high element areas where component count begins to decrease. Care must be taken when using minimum area multipliers, as they effectively remove components from the geometry as though they are not a part of the spacecraft. Proper design considerations are also necessary when using the multiplier, such that the geometry has an active surface beneath removable components that still contributes aerodynamic pressure.

The element generation process becomes a significant portion of the total process time at higher element counts due to hindrance from for loops within the centroid generation processes. The gap between the MRO and ODY curves for the element generation process is due to the higher component count of MRO. Shade 2 takes longer for ODY than MRO despite its lower component count, presumably due to the nature of its design. At the orientations used, the geometry of ODY is more condensed when viewed from upstream, resulting in more shading comparisons. It is possible to reduce the process time used by shade 2 and, thus, the overall time by parsing the shading comparisons to multiple processors on a capable machine. Because shading comparisons are independent of one another, conducting the iterations using dual quad processors could reduce the time for shade 2 by a factor of 8. These results give insight into geometric design implications as well as elemental area chosen, because both are the primary factors that affect process time.

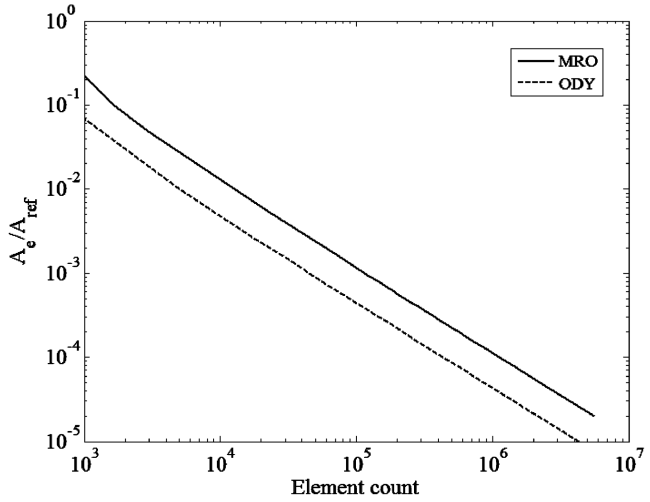

Fig. 14 Relation of element count and element area.

Data Fidelity Case Study

The fidelity of the data output by this procedure is tied directly to the resolution the user inputs in the form of elemental area. As the elemental area becomes large, the shading analysis becomes more inaccurate, however, less costly computationally. Consider a large plane of elements behind a shading triangular plate. The elements are described by their centroid locations and are thus shaded by the relation of the triangular projection and these coordinates. Figure 15 shows this scenario with an increasing element size. As resolution increases, the shaded area approaches that of the actual projection area of the triangular plate. With this in mind, the question arises of how much deviation in the output will result from changing the resolution.

Another case study was performed on the Mars Reconnaissance Orbiter model in which the output force and moment coefficients were recorded from running FreeMat at various elemental areas at a fixed orientation. The elemental areas ranged from 1 cm^2 to 1 m^2,

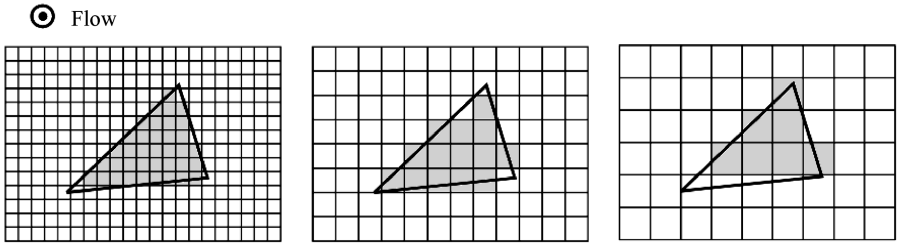

Fig. 15 Triangular plate shading with decreasing resolution.

logarithmically spaced at an untrimmed orientation. The axes used are the same as those used in the DSMC comparison. The force and moment coefficient percent differences with respect to the value at the highest resolution are plotted in Fig. 16, showing larger deviation as element area increases. The force coefficients vary by no more than 3% and the moment coefficients by no more than 6%. Note that, when the elemental area becomes large, some components will still limit the element sizes to maintain accuracy. Such examples of this would be the surface of a cone or cylinder, for which the minimum amount of elements in each circumferential ring is maintained at 6. This prevents the creation of a single large element to represent the entire curvature of the component. Safeguards such as this will prevent extreme deviations due to large elemental areas. The plots indicate that an elemental area of approximately 10 cm^2 may be used, resulting in little deviation from higher resolution values.

Model Complexity Case Study

The analyses conducted in the previous sections have dealt with highly complex geometric models of MRO and ODY. High component counts increase processing time due to more element generation routines and more shading comparisons between components. Some applications will not require such complex models, specifically those that deal with small deviations from trim orientation. Another case study was conducted to compare the aerodynamic coefficients from the complex models of MRO and ODY with the coefficients from the simpler models. The complex models were the same as used earlier, with 220 fundamental components in MRO and 140 in ODY. The simple models were designed with the goal of including only structures for which the upwind surface area exceeded about 5% of the reference area using the minimum components possible. For both models this included the solar array, main bus, high-gain antenna, and camera assembly. The simpler models of MRO and ODY each contained 21 fundamental components. Figure 17 shows relevant coefficient outputs for the models.

The simpler models give output that is comparable to the complex models at orientations close to the trim aerobraking orientation, near zero β and α. An analysis of orientations further from trim confirms that the simpler models begin deviating by larger margins, especially in moment coefficients. Orientations at which both β and α are varied also show larger deviations from the complex model output. It should be noted that the reference points about which the data are taken must be corrected for the simpler models, as their centers of pressure differ from those of the complex models. The theoretical center of mass of the MRO simple model was referenced approximately 2 cm from the center of mass of the complex model to match moment coefficient trends. The ODY simple model's center of mass was referenced approximately 7 cm from that of the complex model. The aerodynamic coefficients not pictured showed negligible deviation as well. The mean run times for the complex and simple MRO models were 1.854 and 0.636 s, respectively. The mean run times for the complex and simple ODY models were 1.390 and 0.294 s, respectively. This analysis confirms that users may input simpler models to improve process times with little loss in data accuracy for

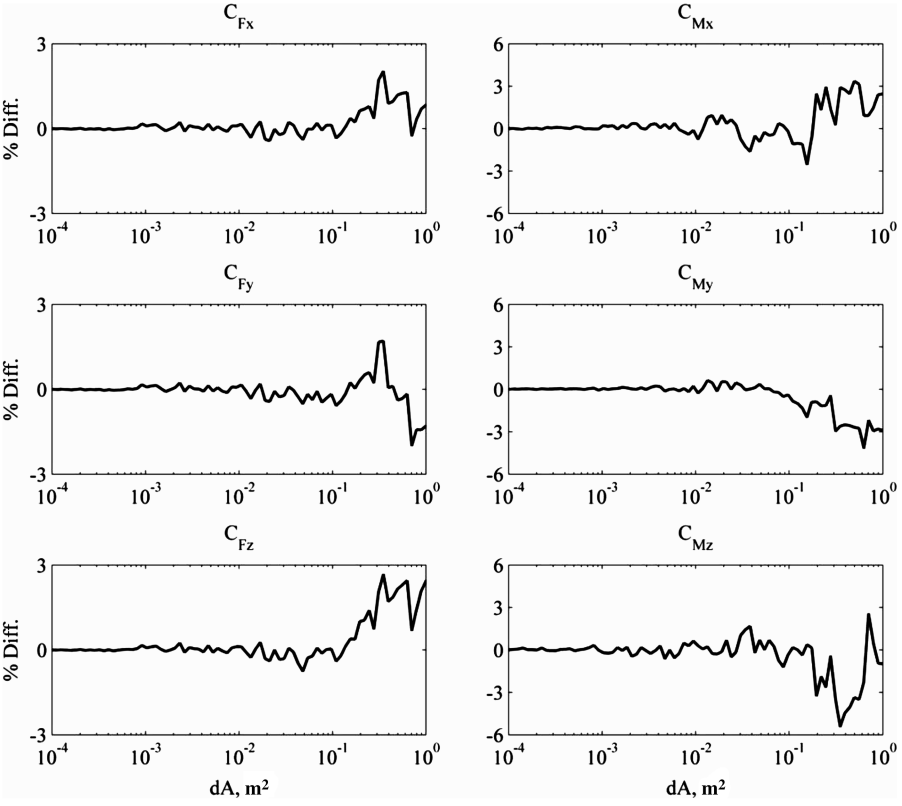

Fig. 16 Aerodynamic coefficient deviation with varied resolution.

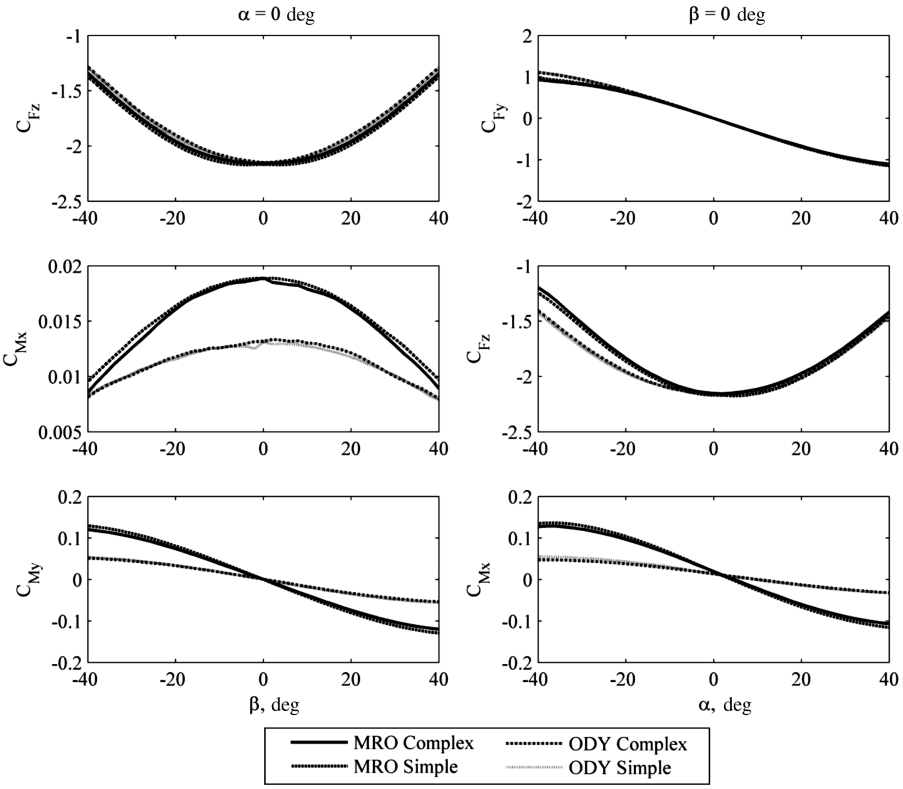

Fig. 17 FreeMat output comparison for models of differing complexity.

orientations near trim conditions. Complex models prove more robust when analyzing the geometry at a wide range of orientations.

Applications

The output may be used for orbit perturbation analysis, aerobraking case studies, and wind error compensation studies. The current program structure allows for the calculation of an aerodynamic database at a range of orientations that may be interpolated within trajectory analyses. Users may edit this procedure to provide constant calculation of aerodynamic coefficients for tumbling or variable geometry spacecraft within their own analysis. The structure of geometry input and force determination also allows customization of component properties, such as the specification of unique accommodation characteristics of one surface as compared with the rest of the spacecraft. The user may create custom geometric components that are composed of fundamental components with ease. Custom supercomponents may be created without determining new shading methods. Custom fundamental components can also be created; however, the inclusion of corresponding shading methods is required.

This procedure may also be edited for analysis of solar pressure. Photons that strike the surfaces of the geometry obey the same assumptions tied to free-molecular flow force calculation with the application of a new momentum transfer equation. Using the same shading techniques, the force and moment calculation method may be revised for photon momentum transfer based on provided absorption ratios rather than accommodation coefficients. Future work dealing with solar pressure will be conducted.

Conclusions

The goal behind developing this procedure and subsequent FreeMat programming was to significantly improve the output fidelity and processing requirements of the free-molecular force calculation procedure. The improvements to the method include a complete redesign of the shading analytics as well as improved discretization processes. These enhancements coupled with the array processing capabilities of MATLAB served to provide high-fidelity output with low computational costs. Consequently, the use of this method is attractive to spacecraft designers, enabling a rapid turnaround time for analyses than most free-molecular analysis tools currently available. The program was written so that other users could tailor the procedure for their own use. The spacecraft design may be as complex or as simple as the designer requires, providing force and moment coefficients at relatively high fidelity. The reduced processing time when compared with previous applications allows the inclusion into real-time trajectory analyses if desired. Compared with programs that require several iterative loops requiring calculation at each element, the procedure's use of array manipulation has increased its processing efficiency drastically.

The output of the procedure has been evaluated with several case studies to study the impact of model complexity and resolution on the fidelity of output and processing time. Results show that the programming is versatile to the user, in that it can provide high accuracy even in cases of lowered resolutions or simplified geometric models. The studies show that model complexities and resolution are directly correlated to processing requirements. Varying model complexities most significantly affected output coefficients at non-trim orientations for which shading results have a higher impact on force and moment calculation. At high resolution and complex geometric input, the output is comparable to that of direct simulation Monte Carlo with differences of less than 5% for the aerodynamic coefficients. Overall, this method provides convenience, customizability, and accuracy to an otherwise tedious analysis.

Acknowledgment

This research was supported by NASA Langley Research Center under cooperative agreement NCC-1-02043.

References

[1] Schaaf, S. A., and Chambre, P. L., "Flow of Rarefied Gases," *Fundamentals of Gas Dynamics*, edited by, H. W. Emmons, Princeton Univ. Press, Princeton, NJ, 1958, pp. 687–708.

[2] Bird, G. A., *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*, Oxford Univ. Press, Oxford, 1994.

[3] Knechtel, E. D., and Pitts, W. C., "Normal and Tangential Momentum

Accommodation for Earth Satellite Conditions," *Astronautica Acta*, Vol. 18, No. 3, 1973, pp. 171–184.
[4] Fredo, R. M., and Kaplan, M. H., "Procedure for Obtaining Aerodynamic Properties of Spacecraft," *Journal of Spacecraft and Rockets*, Vol. 18, No. 4, July–Aug. 1981, pp. 367–373. doi:10.2514/3.28061
[5] Fredo, R. M., "A Numerical Procedure for Calculating the Aerodynamic Coefficients for Complex Spacecraft Configurations in Free-Molecular Flow," M.S. Thesis, Pennsylvania State Univ., University Park, PA, Aug. 1980.
[6] Chipman, D., "Freemac FORTRAN 90," George Washington Univ., Technical Report, Aug. 1996.
[7] Shane, Russell, W., "Aerothermodynamics of the Mars Global Surveyor Spacecraft," M.S. Thesis, George Washington Univ., Washington, DC, Dec. 1997.
[8] Fuller, J. D., and Tolson, R. H., "Program for the Estimation of Spacecraft Aerodynamic Properties," AIAA Paper 2009-728, Jan. 2009.
[9] Wilmoth, R. G., LeBeau, G. J., and Carlson, A. B., "DSMC Grid Methodologies for Computing Low-Density, Hypersonic Flows About Reusable Launch Vehicle," AIAA Paper 96-1812, June 1996.
[10] Nance, R. P., Wilmoth, R. G., and Hassan, H. A., "Comparison of Grid-Definition Schemes for Monte Carlo Simulations," *Journal of Thermophysics and Heat Transfer*, Vol. 11, No. 2, April–June 1997.
[11] Liechty, D. S., "Aeroheating Analysis for the Mars Reconnaissance Orbiter with Comparison to Flight Data," *Journal of Spacecraft and Rockets*, Vol. 44, No. 6, Nov.–Dec. 2007, pp. 1226–1231. doi:10.2514/1.25843

I. Boyd
Associate Editor