Geometry Manipulation by Automatic Parameterization (GMAP)

Casey T. Raymondson[1], Myles L. Baker, Ph.D.[2], Steven P. Doyle[3], and Scott M. Young[4]
M4 Engineering, Inc., Long Beach, CA, 90807

and

Daniel Tejtel[5]
Air Force Research Laboratory, Dayton, OH, 45433

The GMAP Mesh application is a program for parameterizing and morphing engineering analysis models using a simple abstraction called Morphing Tools. Morphing Tools allow complex geometry changes to be boiled down to a few simple design variables, allowing analysis models to be modified with ease in a parametric trade study or optimization study. Using GMAP Mesh allows one to set up multidisciplinary optimization systems that are completely automated once they have been started, providing time savings over systems that require a "man in the loop."

I. Background

OVER many years of working in multidisciplinary design optimization at M4 Engineering, we have consistently had trouble developing parametric geometry models that are useful for generating high-fidelity analysis models.[1] The difficulties range from problems importing/exporting geometry from CAD systems, to unpredictable topology of the resulting models (e.g., small geometry changes result in features appearing or vanishing, often breaking downstream analyses). Many of these difficulties arise because the parametric geometry models are fundamentally based on a "regeneration" approach, in which the geometry is reconstructed when parameters are updated.

A. The Morph Toolset

In order to address these difficulties, we developed the GMAP (Geometry Manipulation by Automatic Parameterization) morphing system based on a "modification" approach, in which the original geometry model is used to define the topology, and this topology is maintained as the coordinates of the various components are modified. While the "modification" approach cannot address all possible types of design variables (for example, the number of fins cannot be easily varied with this approach), it addresses the majority of anticipated design variables with a very high degree of reliability.

The basic idea behind the GMAP (Geometry Manipulation by Automatic Parameterization) morphing system is a unique mapping of one 3-D space to another. In its most general form, this can be expressed as a simple function, shown below in Eq. (1), which is applied to the coordinates of each point in a geometry or analysis model:

---

[1] Software Development Manager, Member AIAA.
[2] President, Associate Fellow AIAA.
[3] Staff Engineer.
[4] Lead Engineer.
[5] Aerospace Engineer, Senior Member AIAA.

$$\vec{r}_{new} = f(\vec{r}_{old}, X) \quad (1)$$

Where $\vec{r}_{old}$ is a vector representing the coordinates of the point in the original "baseline" model, $\vec{r}_{new}$ is a vector representing the corresponding point in the updated model, and X is a vector containing the geometric design variables. While this approach is clearly adequate for defining a wide variety of geometric parameterizations, the difficulty lies in how to define the function (f), without a large amount of abstract, difficult, and error-prone programming.

B. Morphing Volumes

GMAP takes the complete space domain and breaks it down into a set of "morphing volumes". These volumes are represented as Bezier volumes.[2] The coordinates of the control points defining these morphing volumes then can be used as low-level geometric design variables. As the position of a morphing volume node is changed, the geometry inside that morphing volume is "warped" based on the new shape. This technique is known in the computer graphics field as free-form deformation (FFD).[34]

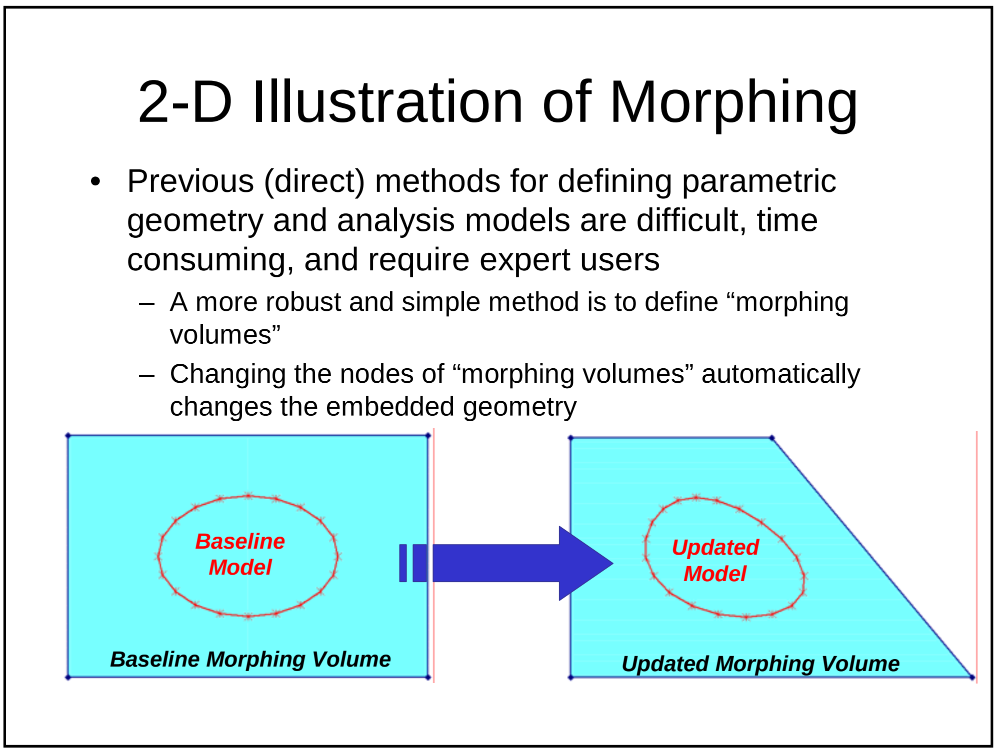

Figure 1. 2-D illustration of morphing approach taken in GMAP.

II. The GMAP Mesh Application

GMAP Mesh is an innovative piece of software currently undergoing development as part of a Phase II SBIR effort with the Air Force Research Laboratory.[5] It allows the user to visually morph an analysis model through a simple graphical user interface (GUI). The GUI was implemented in a combination of Python and C++, leveraging the wxPython6 library for GUI elements and the Visualization Toolkit[7] for 3D graphics. GMAP Mesh is effective for parametrically modifying a model's shape, such as by lengthening an aircraft's nose or increasing its wingspan, without having to modify the CAD geometry and remesh the analysis model. GMAP Mesh offers this parametric morphing capability in a realtime environment that provides instant visual results to the user.

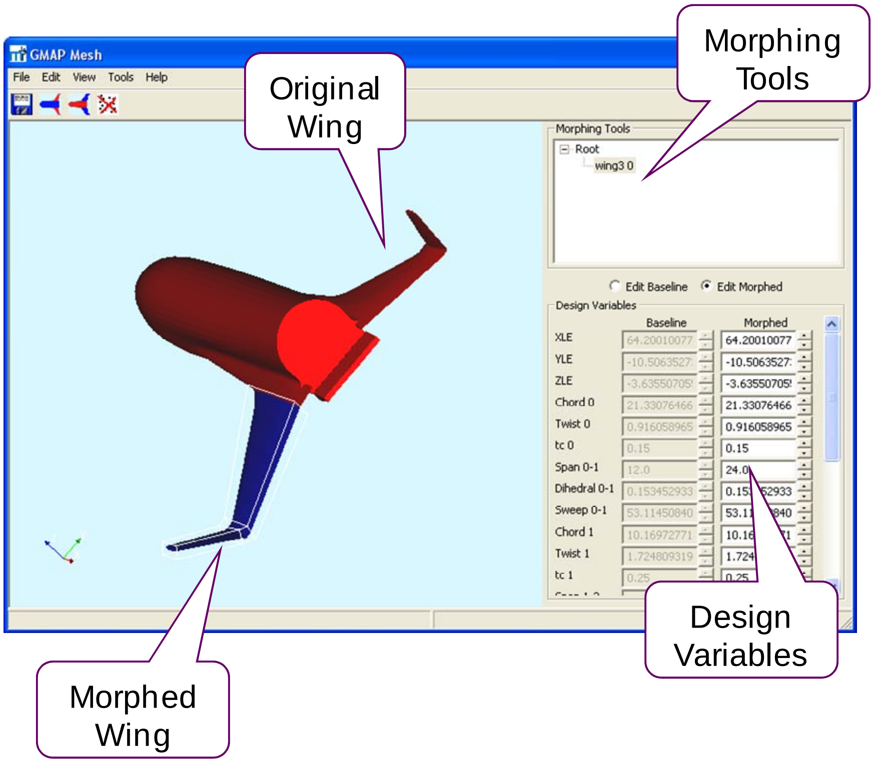

Figure 2. The GMAP Mesh GUI application.

If baseline analysis models for more than one discipline are present, such as a finite element model and a CFD model, the same morphing parameters can be applied to each model. Since GMAP Mesh only moves the points within the analysis model, the morphed model for each discipline will still be consistent with one another after morphing. GMAP Mesh supports a variety of CAE file formats, including STL, S/HABP, PANAIR, Plot3D, NASTRAN, ATAC, and Cart3D.

C. Morphing Tools

For all but the most trivial of morphing tasks, we have found that another level of abstraction above morphing volumes can be useful. GMAP Mesh improves on the concept of FFD by grouping related morphing volumes together into "Morphing Tools." A Morphing Tool is a high-level collection of morphing volumes that is designed to work on a specific part of a model, such as a wing or tail, or to implement a special constraint. Tools reduce the burden on the engineer of having to manipulate dozens of vertex coordinates by hand. A tool will typically expose a handful of high-level design variables, such as wing span, chord length, thickness, etc. As the engineer modifies these high-level parameters, GMAP Mesh automatically resizes and moves the morphing volumes appropriately, morphing the model in the process.

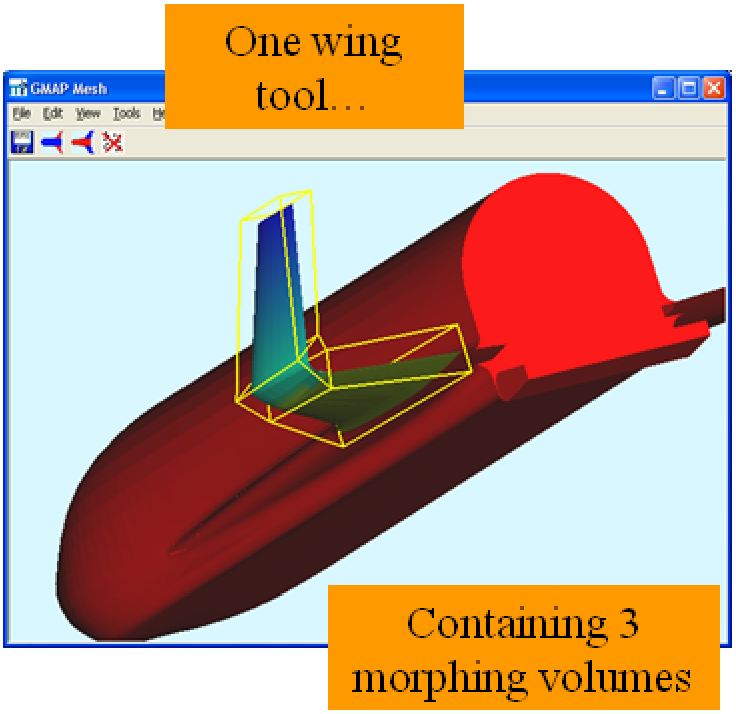

Figure 3. A wing tool contains one morphing volume for each wing segment.

This additional level of abstraction has several advantages over moving the control points of the Bezier volumes directly. It reduces clutter and information overload in the GMAP Mesh GUI, allowing a user to more quickly and easily try out different designs. Expressing the tools in terms of high-level design variables rather than control point locations makes the system far more accessible to novice users. Finally, the generally small number of design variables that morphing tools expose makes them significantly more usable in a parametric optimization study. Performing geometry optimization using Morphing Tools results in a much lower order design space than if the optimization had to modify the X, Y, and Z locations of each control point.

We are creating an ever-growing library of Morphing Tools for common tasks such as wing planform deformation, control surface deflection, and nose shaping. These tools are easy to adapt to a new vehicle by changing the baseline design variables they use as input. These customizations can be made directly in the GMAP Mesh GUI, providing instant visual feedback that verifies the correctness of the values chosen. Once a user has adapted a Morphing Tool for use with a specific vehicle, the Morphing Tool can be saved for future use. Then, when performing a parametric trade study or optimization study, the same Morphing Tool and baseline analysis model(s) can be used again and again, with only the "morphed" design variables changing on each iteration.

If the existing library of Morphing Tools is not sufficient to solve a user's problem, they can create their own that is customized exactly to their needs. Morphing Tools use a simple XML-based description that defines the topological layout of the tool's Bezier volumes, lists which design variables are exposed for modification, and defines the relationship between the Bezier control points and the design variables. Control point locations can be defined using simple equations. If the user desires a particularly "smart" tool that uses more complex calculations, additional processing can be embedded in the tool using the Python scripting language. While creating new tools does have a bit of a learning curve, a tool to perform a certain task only needs to be developed once. Once a tool has been created, it can easily be used by a novice to morph the model.

D. Compatibility Between Tools

Another issue that comes up frequently when performing morphing is the interface between different sections of a model. When increasing the root chord length of a wing, for example, the nodes on the fuselage surrounding the wing must be moved as well in order to prevent any elements from having a very skewed shape, or even turning inside out. Likewise, a tool designed to deflect a flap on a wing must still work properly regardless of whether the wing as a whole is being morphed by another tool. Our solution is to let the user organize all the currently loaded tools into a hierarchical structure with parent tools and child tools. When a parent tool morphs, it modifies not only the points in the model owned by that tool, but also the nodes of the child tool.

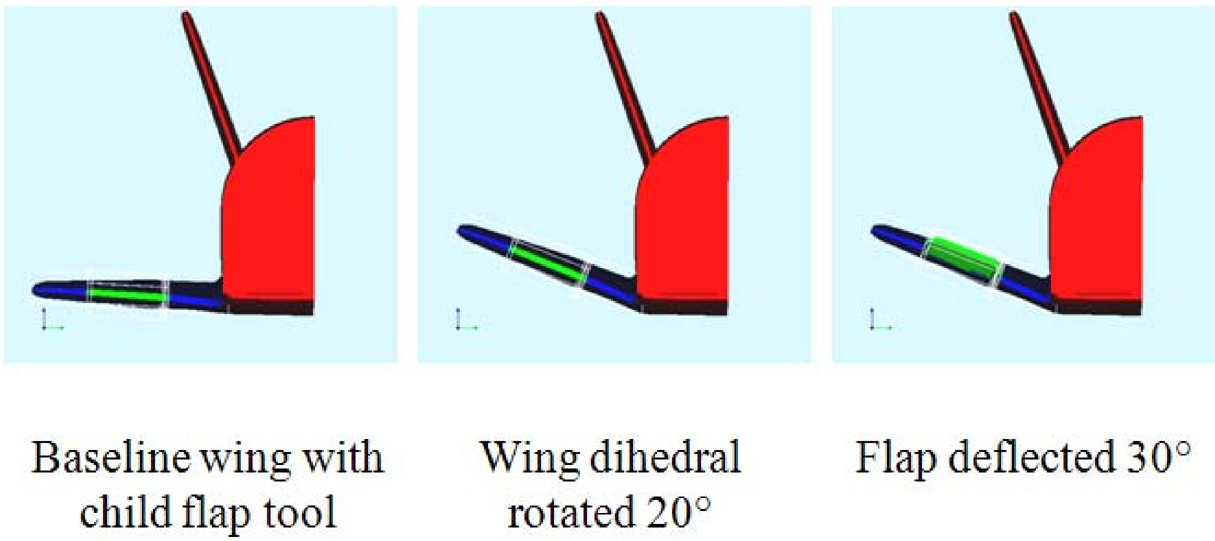

Figure 4. A flap deflection tool is attached to a wing dihedral rotation tool. When the dihedral is changed, the flap tool automatically rotates along with the rest of the wing.

E. Automation

With GMAP, once the baseline geometry has been provided in a CAE format, and the Morphing Tools have been applied to the model with the appropriate baseline design parameters, a trade study can be run for hundreds or thousands of design variations without any need for user interaction. Since the CAD geometry is not touched and no remeshing takes place, and GMAP works directly with high-level design variables to determine the new geometry, such a trade study can be run 100% automation. Figure 5 below illustrates typical engineering trade study workflows with and without GMAP.

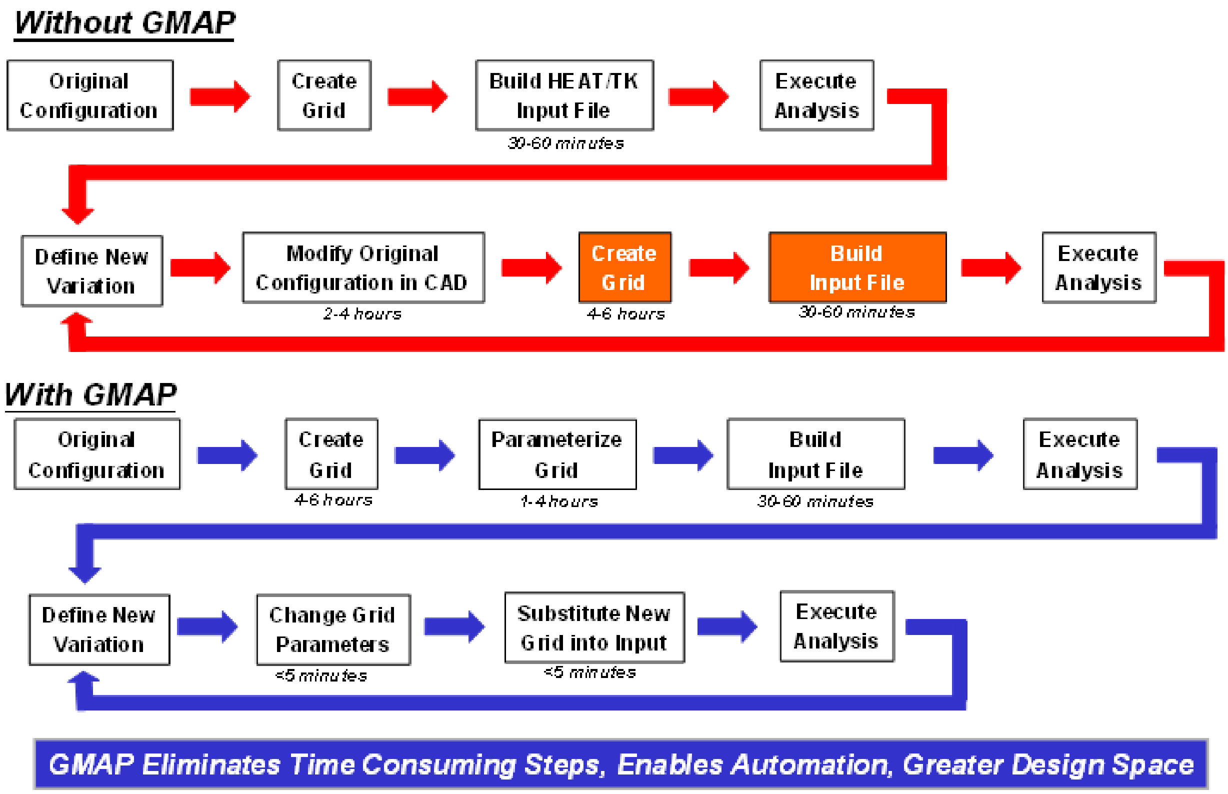

Figure 5: Workflows with and without GMAP.

III. Application: Nose Constant Area Tool

The following application shows an example of a Morphing Tool that was created for use in GMAP Mesh. This vehicle is an RLV (Reusable Launch Vehicle) concept, which demonstrates how small geometric modifications to design variables can have a large impact of stability and thermal characteristics. This example shown below tailors the nose shape in order to improve the tradeoff between lateral directional stability and aerodynamic heating.

A baseline spatular (flat nose) RLV model was generated based on a representative RLV vehicle shown in Figure 6. The nose of the baseline vehicle is shown below in Figure 7. A baseline heating and stability analysis was performed using the hypersonic panel aerodynamic analysis code S/HABP's[8] real gas capability to establish proper boundary conditions.[9]

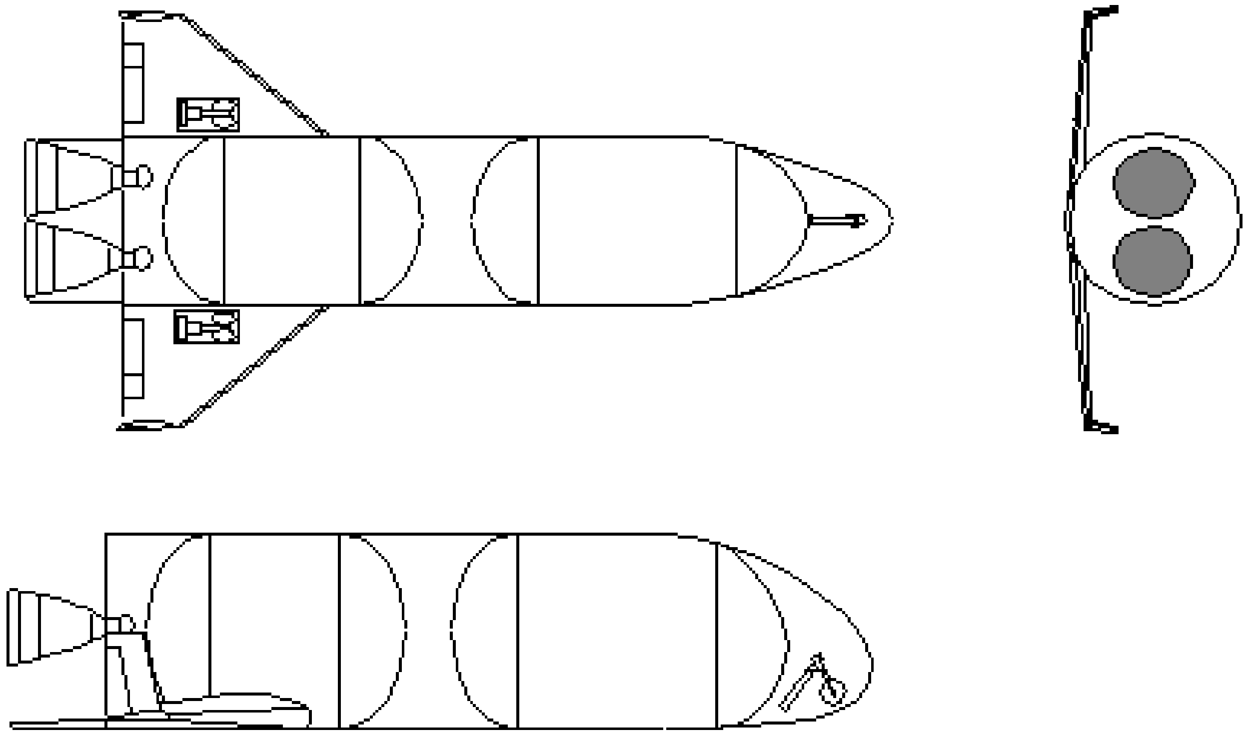

Figure 6: Representative RLV configuration.

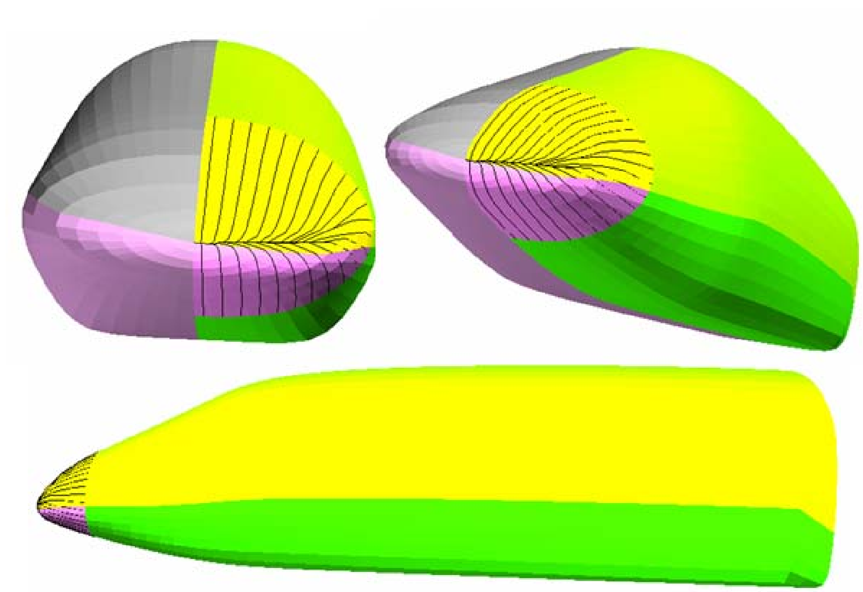

Figure 7: Nose of spatular RLV baseline configuration.

For a constant area nose, the aerodynamic heating will be higher for a spatular nose than for a blunted nose. In order to optimize this tradeoff, the Constant Area Tool was developed for use in GMAP Mesh. This tool (shown below in Figure 8) uses five cross-sectional cuts, which have a specified area. The tool has its greatest geometric influence at the foremost cross section, with the amount of influence decreasing linearly all the way back to the rearmost cross section, which has no influence. This allows for a smooth transition between the nose and the body. Only a single design variable, the Scaling Aspect Ratio, is exposed to the user. This parameter adjusts the vertical and horizontal aspect ratio of the nose tip while keeping its cross-sectional area constant.

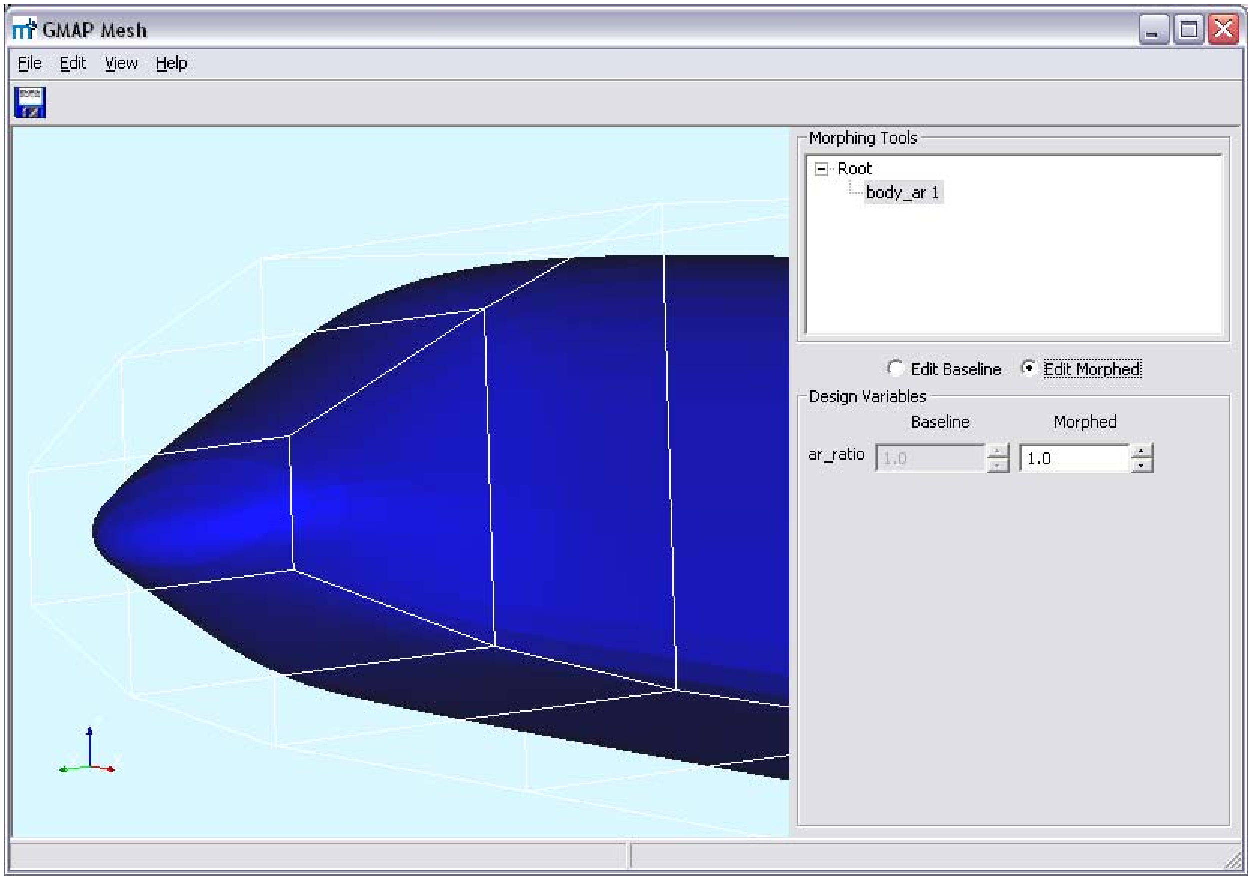

Figure 8: GMAP Morphing Tool for RLV baseline spatular configuration; Scaling Aspect Ratio=1.0.

Once the baseline S/HABP geometry and the morphing tool were provided, GMAP Mesh and S/HABP were then run for 100 different RLV configurations with Scaling Aspect Ratios ranging from 1.0 to 3.0. For each iteration in this process, an automated script input a new Scaling Aspect Ratio to the Morphing Tool, re-ran GMAP to generate a new geometry, re-ran S/HABP on the new geometry, and saved the results. The entire 100-step run executed automatically with no user interaction.

A Scaling Aspect Ratio of 1.0 corresponds to the baseline spatular configuration while a Scaling Aspect Ratio of 3.0 corresponds to a fully blunted configuration (see Figure 9).

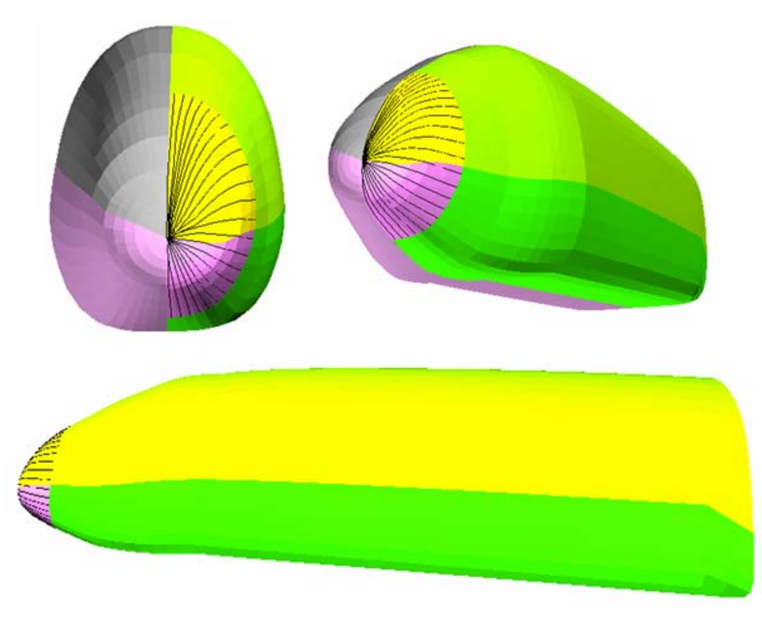

Figure 9. Nose of blunted RLV morphed configuration; Scaling Aspect Ratio=3.0.

The Stability Derivatives ($C_{m\alpha}$ and $C_{\ell\beta}$) were computed using a morphed S/HABP model of the full configuration (see Figure 10). A reference length, a reference span, and then CG/moment reference point were assumed.

Stability Coefficients without any reference to where the center of gravity is are not an easily understood quantity. By incorporating the center of gravity, along with the $C_{m\alpha}$ and $C_{\ell\beta}$ curves, the aerodynamic center can be plotted for different Minimum Nose Radii (see Figure 11). As expected, increasing the Minimum Nose Radius (blunting the nose of the vehicle) decreases lateral stability. For a constant area, the Maximum Nose Radius must similarly be increased. This results in a corresponding increase in the longitudinal stability.

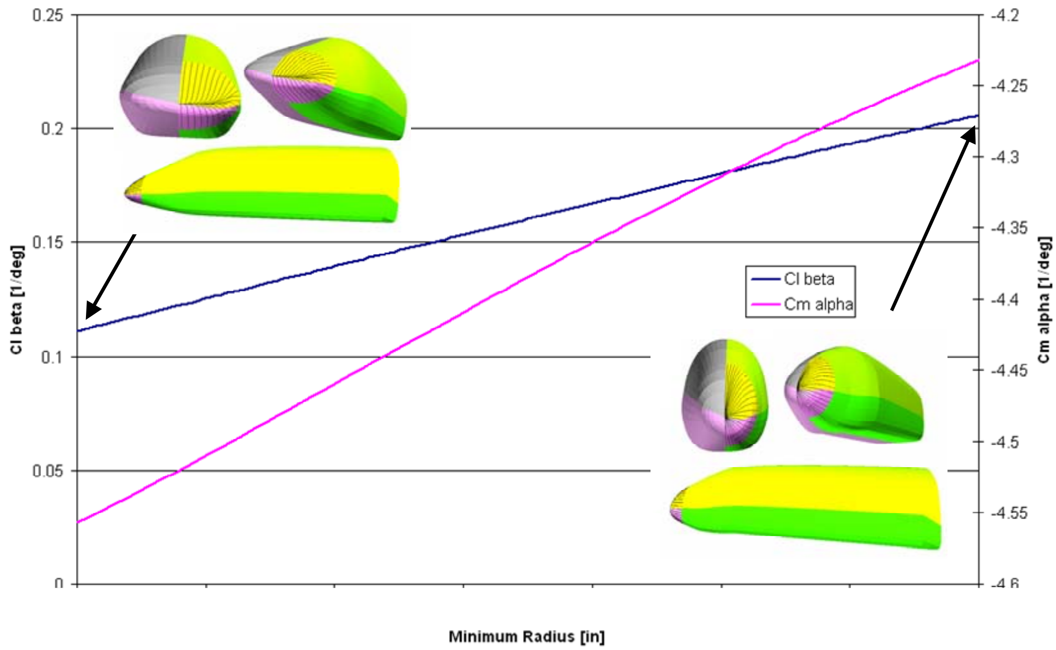

Figure 10. Stability derivatives for 100 RLV configurations.

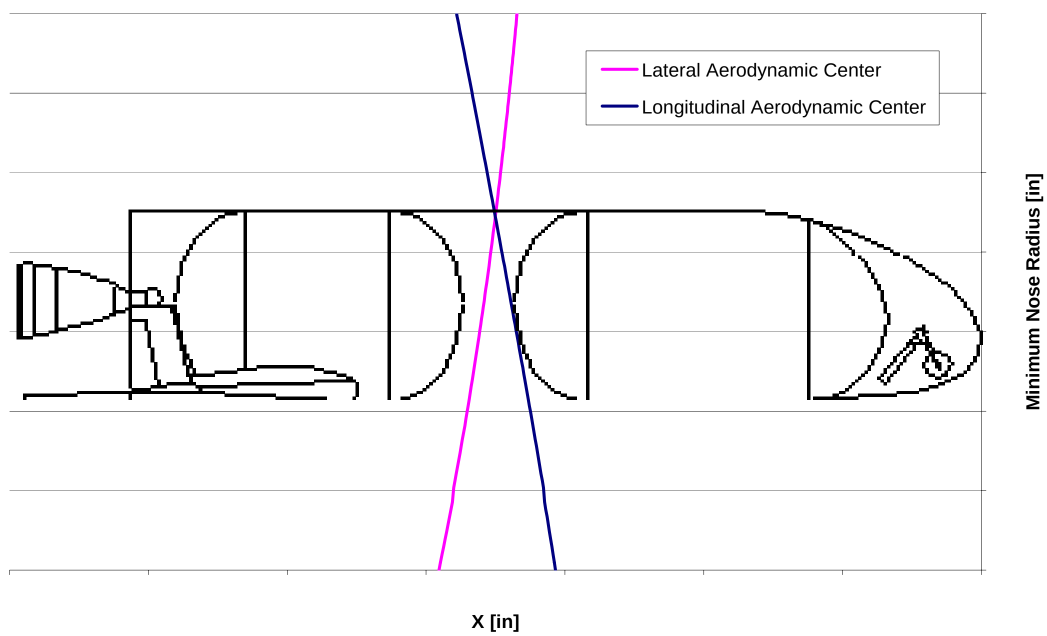

Figure 11. Aerodynamic center travel for 100 RLV configurations.

The peak heating (thermal loading) was computed along the critical streamline using the morphed S/HABP nose model. This was done to ensure adequate streamline coverage on the nose of the model. Due to the fact that the vehicle was analyzed at a Flight Condition of Mach 24.25 with an Altitude of 280,000 feet, Real Gas Effects and Rare Gas Effects are important.

The baseline geometry and a morphed (blunted) variation is shown in Figures 7 and 9. The baseline vehicle has a spatular nose with high curvature (small nose radius), whereas the blunted vehicle has a lower curvature (large nose radius). As expected, in the critical vertical direction, the blunted RLV has a lower peak heat flux.[10] This is shown below in Figure 12.

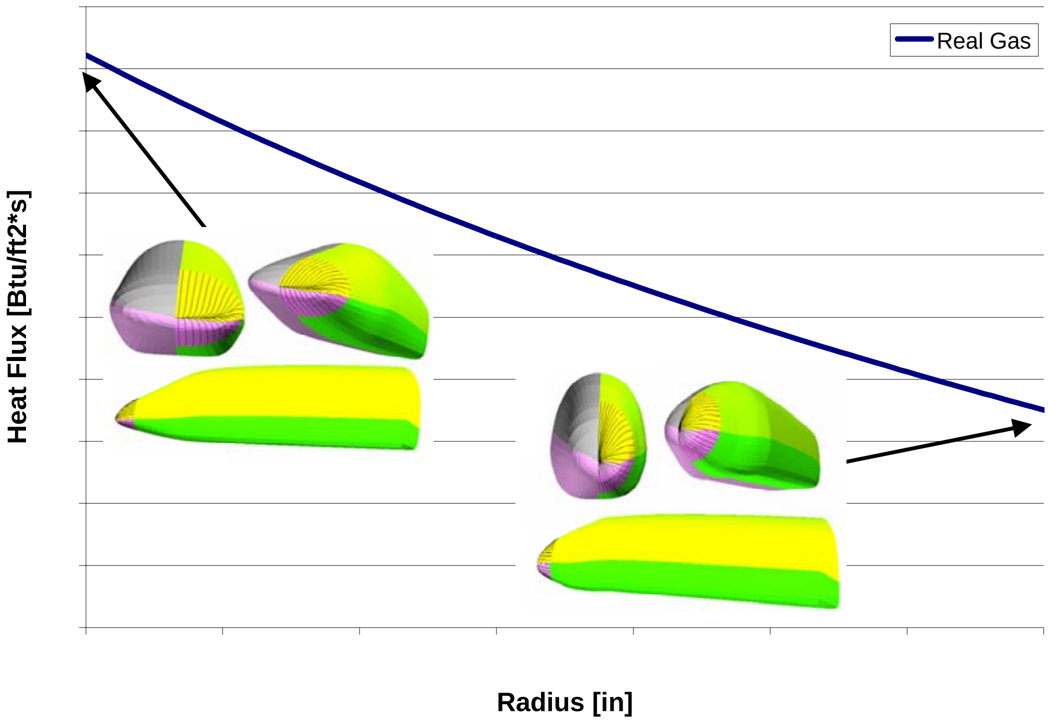

Figure 12. Thermal loading for 100 RLV configurations.

IV. Application: Fatigue Optimization for a Hydraulic Actuator

One of the main benefits of the GMAP system is the applicability to multidisciplinary analysis and optimization, because the morphing approach applies equally well to models of any discipline. While the previous result shows an application to the aerodynamic and aerothermal disciplines, this section deals with a complex structural problem.

Structural analysis of a component often involves static, dynamic, and fatigue analyses. One of the most challenging types of analyses is fatigue analysis, since the overall fatigue behavior of a component depends on the complete history of loading over a parts life, which requires analysis of many different load conditions, and combining the results into a single fatigue life calculation. GMAP Mesh was used as part of a fatigue optimization process used at M4 Engineering to improve the fatigue life of a metallic structural component. The process is outlined below in Figure 13. A FEM of the baseline geometry is created and the model is solved for each of the load conditions that occur in the fatigue spectrum. Fatigue margins are then determined for every node in the model. The optimization algorithm (in this case a genetic algorithm) changes the design variables to attempt to increase the fatigue life of the part. When updated optimization parameters are provided, the GMAP system is used to update the FEM to the new shape. The process repeats until the design with the highest fatigue margin possible is found.[11]

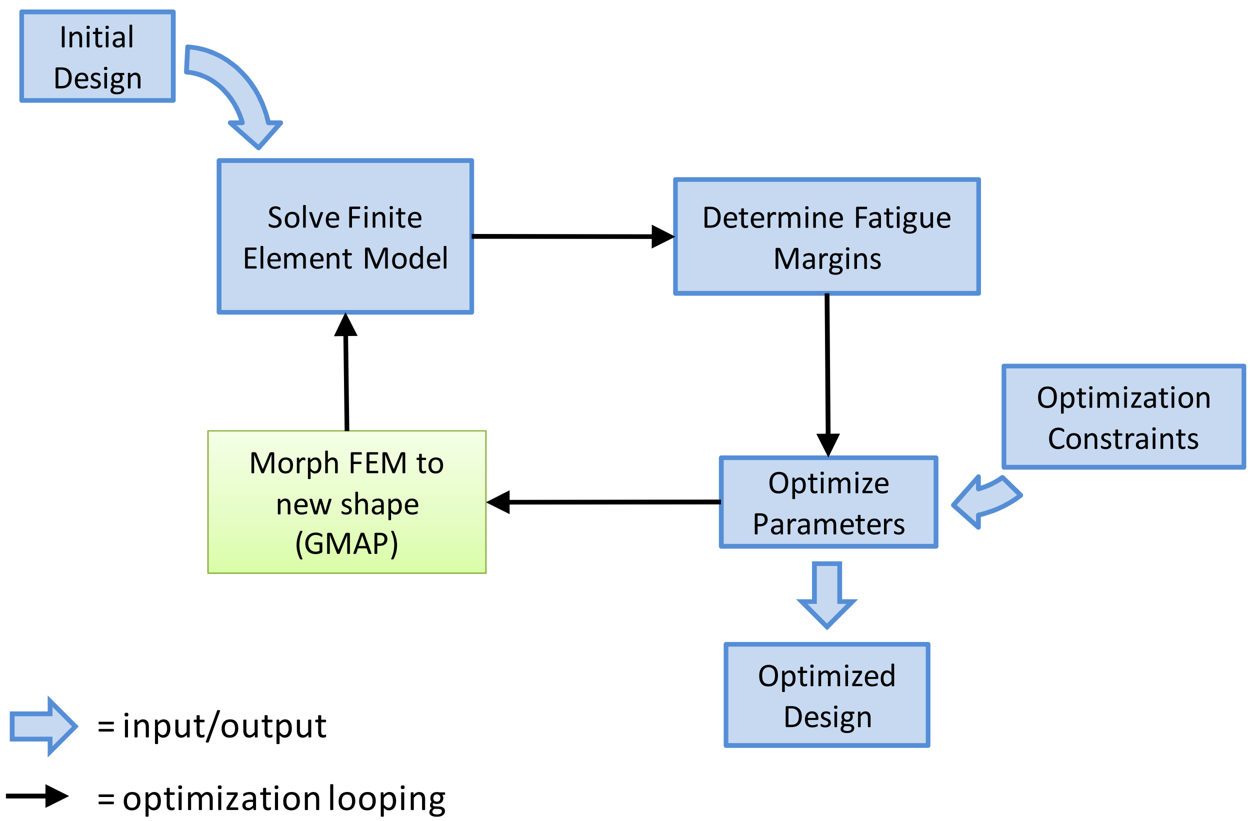

Figure 13. Fatigue optimization process.

The process was run on a hydraulic actuator using three design variables. The goal of the optimization was to maximize fatigue margin. The design variables for this problem were the semimajor and semiminor axes a and b of the elliptical port hole, and t (thickness).

A GMAP Mesh Morphing Tool was created that exposed these three design variables. Using this morphing tool greatly simplified this optimization process because the optimizer only needed to work with those three design variables. If we instead had to optimize the locations of each control point defining the individual morphing volumes, the optimization would be much more complex.

Figure 15 shows a significant reduction of 47% in max principal stress due to the geometry change after analysis in NASTRAN.[12]

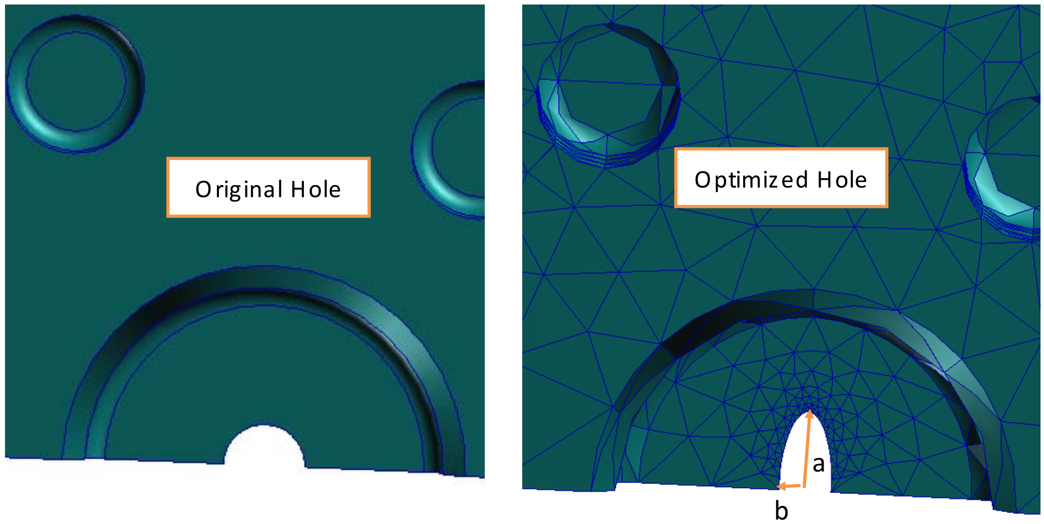

Figure 14. Baseline (left) and optimized actuator port (right) geometry.

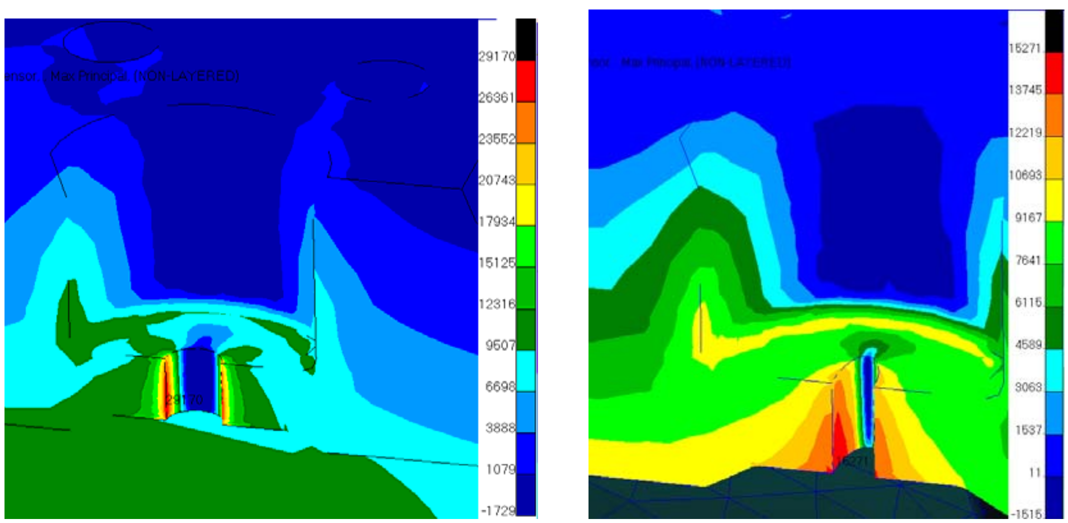

Figure 15. Baseline (left) and optimized actuator port (right) unit pressure max principal stress.

V. Conclusion

GMAP Mesh has provided immense time savings in our multidisciplinary optimization efforts. Development of the GMAP Mesh application is ongoing. In addition to adding morphing support for additional CAE file formats, we are exploring ways to make creation and customization of Morphing Tools easier for a novice user. This will primarily be accomplished by providing intuitive GUI elements to assist the user with tool creation.

Acknowledgements

Funding for the development of GMAP is provided by the Air Force Research Laboratory.

References

[1]Baker, M. L., Munson, M. J., and Alston, K. Y. "Integrated Hypersonic Aeromechanics Tool (IHAT)," *AIAA Missile Sciences Conference,* Monterey, CA, 2002.

[2]Rogers, D. F. *An Introduction to NURBS,* Academic Press, San Diego, CA, 2001.

[3]Sederberg, T. W. *Computer Graphics Method for Changing the Shape of a Geometric Model Using Free-Form Deformation, 4,821,214* April 11, 1989.

[4]Samareh, J. "Aerodynamic Shape Optimization Based on Free-Form Deformation," *AIAA-2004-4630,* 2004.

[5]Raymondson, C. T., Baker, M. L. *GMAP Mesh User's Manual,* M4 Engineering, 2008.

[6]Rappin, N., Dunn, R. *wxPython in Action,* Manning Publications Co, Greenwich, CT, 2006.

[7]Schroeder, W., Martin, K., Lorensen, B. *The Visualization Toolkit, 3rd Edition,* Kitware, 2004.

[8]Gentry, A. E., Smyth, D. N., Oliver, W. R. *The Mark IV Supersonic-Hypersonic Arbitrary-Body Program. Volume I - User's Manual,* Douglas Aircraft Company, McDonnell Douglas Corporation, 1973.

[9]Burns, K. A., Deters, K. J., Haley, C. P., Kihlken, T. A. *Viscous Effects on Complex Configurations: Software User's Manual,* McDonnell Douglas Aerospace, 1995. WL-TR-95-3060

[10]Bertin, J. Hypersonic Aerothermodynamics, 3rd Edition, AIAA, 1994.

[11]Young, S., Tai, J., Doyle, S., Oberschmidt, A. *SBIR Phase II Final Report: Weapon Weight Reduction Using Evolutionary Algorithms,* s.l. : M4 Engineering, 2008.

[12] *MSC. Nastran 2005 Quick Reference Guide: Volume1,* MSC.Software Corporation, Santa Ana, CA, 2004.