

NASA/CR-2008-214902



Modeling and Optimization for Morphing Wing Concept Generation II

Part I: Morphing Wing Modeling and Structural Sizing Techniques

*Michael D. Skillen and William A. Crossley
School of Aeronautics and Astronautics
Purdue University, West Lafayette, Indiana*

February 2008

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers, but having less stringent limitations on manuscript length and extent of graphic presentations.
- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or co-sponsored by NASA.
- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.
- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results ... even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at <http://www.sti.nasa.gov>
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA STI Help Desk at (301) 621-0134
- Phone the NASA STI Help Desk at (301) 621-0390
- Write to:
NASA STI Help Desk
NASA Center for AeroSpace Information
7115 Standard Drive
Hanover, MD 21076-1320

NASA/CR-2008-214902



Modeling and Optimization for Morphing Wing Concept Generation II

Part I: Morphing Wing Modeling and Structural Sizing Techniques

*Michael D. Skillen and William A. Crossley
School of Aeronautics and Astronautics
Purdue University, West Lafayette, Indiana*

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

Prepared for Langley Research Center
under Grant NNL06AA04G

February 2008

Trade names and trademarks are used in this report for identification only. Their usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Available from:

NASA Center for AeroSpace Information (CASI)
7115 Standard Drive
Hanover, MD 21076-1320
(301) 621-0390

National Technical Information Service (NTIS)
5285 Port Royal Road
Springfield, VA 22161-2171
(703) 605-6000

Table of Contents

Table of Contents.....	1
Nomenclature	3
I. Introduction and Project Objectives.....	3
II. Implementation and Comparison of Morphing-Wing Structural Optimization Strategies	4
A. Motivation and Background.....	4
B. Morphing Wing Structural Optimization Strategies – Process Architecture and Implementation	4
1. Fixed-Geometry Wing Sizing.....	4
2. “Simultaneous Analysis” Approach	5
3. “Sequential” Approach.....	6
4. “Aggregate” Approach	7
C. Developing Morphing Wing Finite Element Models – An Object-oriented Preprocessor	8
5. Facets of a Morphing Wing System	8
6. Morphing Wing Modeling - Process Architecture.....	8
7. Information Flow and Major Data Structures.....	11
8. Process Demonstration Using a Folding Wing Concept.....	14
D. Case study – Structural Sizing of the “Shear-Wing” Concept.....	19
9. Overview of the Concept.....	19
10. Model schematic and related system / sizing information	19
11. Experimental Datum for Structural Sizing Strategy Comparison.....	22
12. “Simultaneous Analysis” Strategy.....	28
13. “Sequential” Optimization Strategy	30
14. “Aggregate” Optimization Strategy.....	33
15. Summary of Results / Conclusions.....	34
E. Suggested Future Efforts	34
Appendix	36
A. User-Defined Folding Wing Template.....	36
B. User-Defined Scripts to Define and Morph the Control Point Set	37
References	40

Nomenclature

AR	=	Wing aspect ratio
BDF	=	Bulk Data File
DOE	=	Design of Experiments
OML	=	Outer Mold Line
S	=	Wing area
n	=	Number of primary wing configurations
n_z	=	Maximum vertical load factor
t/c	=	Wing thickness to chord ratio
$\{x\}$	=	Structural design variables
β	=	Wing fold angle for the folding wing concept
λ	=	Wing taper ratio
Λ	=	Wing sweep angle

I. Introduction and Project Objectives

As a precursor to the contents of this report, this author would like to establish clear meaning to the word “morphing” as it will be applied here. Namely, a “morphing wing” is an aircraft wing able to drastically change planform shape during flight – perhaps a 200% change in aspect ratio, 50% change in wing area, and a 20 degree change in wing sweep.¹ Furthermore, a “morphing aircraft” refers to a conventional fixed-geometry aircraft structure with the fixed wing replaced with a morphing one. Other types of “morphing”, such as variable geometry airfoils, are not considered here.

The effort documented here is an extension of the work presented in a NASA technical report submitted in December of 2006.² Previously, morphing-wing weight predictors were developed using simple beam-model representations of various morphing concepts. Wing weight data was generated by structurally sizing several beam models with different outer mold lines (OML) and extents of actuation; a face-centered central composite Design of Experiments (DOE) was used to prescribe these geometric models. Because no formal/industrial software exists to structurally size this type of wing platform, several structural optimization strategies were described that aimed to incorporate conventional toolsets while also including the facets of a reconfigurable structure. Using these techniques, the optimal structural design[‡] of a given wing was shown to be a function of the structural optimization strategy. This effort will revisit these structural optimization techniques to demonstrate and further substantiate their performance using intermediate complexity finite element models. A description of these techniques, as well as the means to develop intermediate complexity wing models will be the subject of Part I of this report. The reader is referred to references 3, 4, and 5 for more in-depth discussion of these topics.

Two conceptual morphing-aircraft sizing models were also previously presented and compared. The objective of these routines was to predict the major system-level aircraft parameters (i.e. thrust-to-weight ratio, wing area, wing sweep, thickness-to-chord ratio, wing aspect ratio, and wing taper) and wing shapes (morphed wing state / wing configuration) for each leg of the design mission that enabled the lowest aircraft takeoff gross weight. Both treated the morphed states of the wing as top-level design variables for each segment of the design mission. As this creates a relatively large design variable set compared to a fixed-geometry aircraft (and a corresponding increase in optimization runtime), a new technique was developed that removes the wing-state variables from the top-level design variable set. This new sizing model will be described in detail and demonstrated through various case studies using a NextGen-like morphing aircraft concept in Part II of this document.

[‡] Note that “optimal structural design” refers to an “optimal” or near optimal structural mass distribution only; the structural layout of the wing in any given configuration is assumed to be known *a priori* and remain fixed during the structural sizing task. Optimization of the structural layout / substructure topology or the morphing mechanism design is not considered in this work.

II. Implementation and Comparison of Morphing-Wing Structural Optimization Strategies

A. Motivation and Background

The idea of morphing aircraft is not a new one, but has only recently gained strength as a viable technology and an active research topic; the DARPA-funded Morphing Aircraft Structures (MAS) program⁶ is the most recent initiative to provide a technology thrust in this area. During Phase II of this program, two flight-traceable morphing wing concepts were developed. The first, a “folding wing” concept, was developed by Lockheed Martin and enables variations of span length, aspect ratio, and effective sweep angle.⁷ The second, a variable sweep / variable root chord concept, was developed by NextGen Aeronautics and enables direct variations in root chord length and sweep angle; indirectly varying the planform area and aspect ratio.⁸ Both of these are illustrated in the following figure.

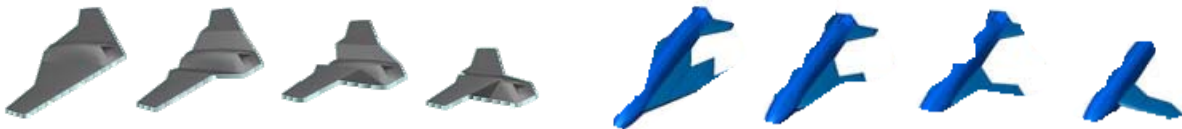


Figure 1 Recently Developed Morphing Wing Concepts (Left – Lockheed Martin’s Folding Wing Concept, Right – NextGen Aeronautics Variable Sweep / Variable Root Chord Concept)

While Phase II of the MAS program concentrated on development and testing of scaled wind tunnel models to determine system feasibility in flight equivalent environments (wind tunnel experiments at the TDT at NASA, Langley), Phase III will pursue the development of these concepts as flight demonstrational vehicles.⁹ Clearly, a need exists to be able to structurally size these kinds of reconfigurable systems to a high degree of structural efficiency to reduce unnecessary wing weight. The work presented here will focus on such design strategies.

B. Morphing Wing Structural Optimization Strategies – Process Architecture and Implementation

1. Fixed-Geometry Wing Sizing

As a reference point for the discussion to follow, the general FEM-based process to structurally size a conventional fixed-geometry wing will be described. Currently, ASTROS¹⁰ and NASTRAN¹¹ enable MDA/MDO capabilities necessary for wing design,^{12,13} and form the software basis set to which comments in this section apply. Assuming the outer mold line and structural layout of a fixed-geometry wing have been defined, an intermediate complexity FEM is likely to be developed; this will be a medium-fidelity model probably lacking detail in joints, fittings, cutouts, etc. that naturally arise in the detail design phase. This model will be structurally sized to establish a second-order weight estimate, determine primary or optimal load paths, and understand aeroelastic characteristics / tendencies to enable more rigorous design trade studies.

The fixed-wing structural optimization task has the following information flow:

- Define the fixed-wing’s OML and sub-structural layout
- Define design load conditions that tend to produce maximum structural constraint conditions
- Develop an intermediate complexity FEM (typically facilitated w/ a graphical preprocessor)
 - i. Generate the structural model
 - ii. Generate the aerodynamic surface model
 - iii. Spline the aerodynamic surface to the structural grid points
 - iv. Define and link element design variables as appropriate (web thicknesses, spar cap areas, etc.)
 - v. Define design constraints (structural strength, aeroelastic characteristics, maximum strain / displacements, design variable bounds, and manufacturing constraints)

- Establish the bulk data file (BDF) analysis/optimization cards to evaluate design load conditions
- Establish the BDF cards to perform the design optimization task
- Run optimization / Analyze results / Perform trade studies

Most commercial FEA codes have been setup to directly handle this type of fixed-geometry structural optimization task in a standalone environment. However, morphing wing systems are by definition a variable-geometry structure and have multiple (if not infinite) structural arrangements characterizing its operational capabilities. Unfortunately, there is no way to define such a system in ASTROS or NASTRAN such that a traditional structural optimization task could be applied within the standalone FEA/MDO environment. Therefore, surrogate strategies that perform similar function as the fixed-geometry wing optimization task will be presented that aim to structurally size morphing wing systems.

2. “Simultaneous Analysis” Approach

The simultaneous analysis approach follows the fixed-geometry wing sizing form as presented above, but incorporates the idea that multiple wing states, or “configurations”, define the wing system as a whole. By selecting a subset of these wing states that are anticipated to drive the structural design task, the morphing wing system can be reduced to a set of “fixed-geometry” wing models. In this case, an intermediate complexity finite element model and an associated design load case set would be developed for all the “primary configurations” of the morphing wing. However, because common FEA/MDO codes can only handle a single fixed-wing system at a time, a further “work-around” is necessary to enable the structural optimization task. The strategy employed by the “simultaneous analysis” approach is to decouple the optimization capabilities of NASTRAN or ASTROS from the FEA capabilities. By implementing the structural optimization algorithm external to the FEA environment (e.g. in Matlab¹⁴ or iSIGHT¹⁵), and treating NASTRAN/ASTROS as the means to evaluate the objective metric, design constraints, and sensitivity information for a given structural design, the morphing wing structural optimization task exactly follows the design optimization strategy given for a fixed-geometry wing.

Figure 2 illustrates the simultaneous analysis process graphically. Notice that elements with common colors have some level of connectivity between them: The light gray boxes indicate I/O processing at the user-level; red boxes indicate tasks directly handled by the optimization environment (iSIGHT in this case); light green indicates an FEA task with the appropriate data files and design load cases. Also notice that this diagram does not illustrate objective function information for the sake of simplicity. However, note that the optimization objective is to minimize wing mass (equivalently, wing weight) and that the objective function value and associated sensitivity information can be obtained from any of the FEA calls. Last, the design variable file “desvar.d” is a bulk data file that is referenced by all FEM_i bulk data files via the “INCLUDE” directive. Therefore, each finite element model is given the same design variable values within the grouping of FEA calls. After output processing from all the FEA calls, the global constraint vector, {g}, is generated by sequentially collocating each of the constraint value vectors from the individual configuration analyses; the same is true for the sensitivity matrices.

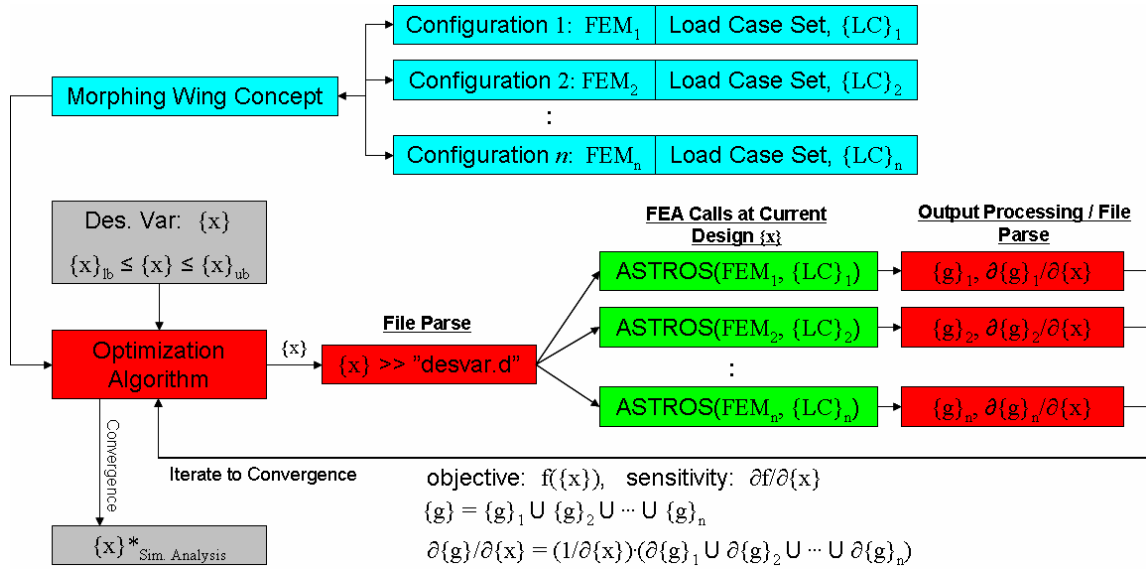


Figure 2. Simultaneous Analysis Process Architecture and Information Flow

3. “Sequential” Approach

The “sequential” structural optimization approach was first documented in Reference 7 and is desirable because it does not require the FEA and optimization tasks to be decoupled. In fact, the ASTROS or NASTRAN-based optimization venues can be used outright, as long as the assumption remains that the morphing wing can be replaced by a collection of “fixed-wing” FEMs. If this approximation seems applicable, then the process proceeds as illustrated in Figure 3. Notice that this design task is initiated by defining the morphing wing concept, its associated primary configurations and load case sets, and the initial design variable values and bounds. A structural optimization task is implemented using ASTROS and initializes the sequential optimization iteration loop. The “optimal” design variable values from FEM₁, are set as lower bound values in the BDF design variable cards, and the initial conditions are set to the same values. Using the updated design variable bounds and initial conditions, another ASTROS-based optimization task is implemented for FEM₂ and results in a new set of “optimal” design variable values. These are again set as the lower bound design variable values and initial conditions, and this iterates until all configurations have been optimized. The “optimal” design for the last configuration is the “sequential-optimal” design.

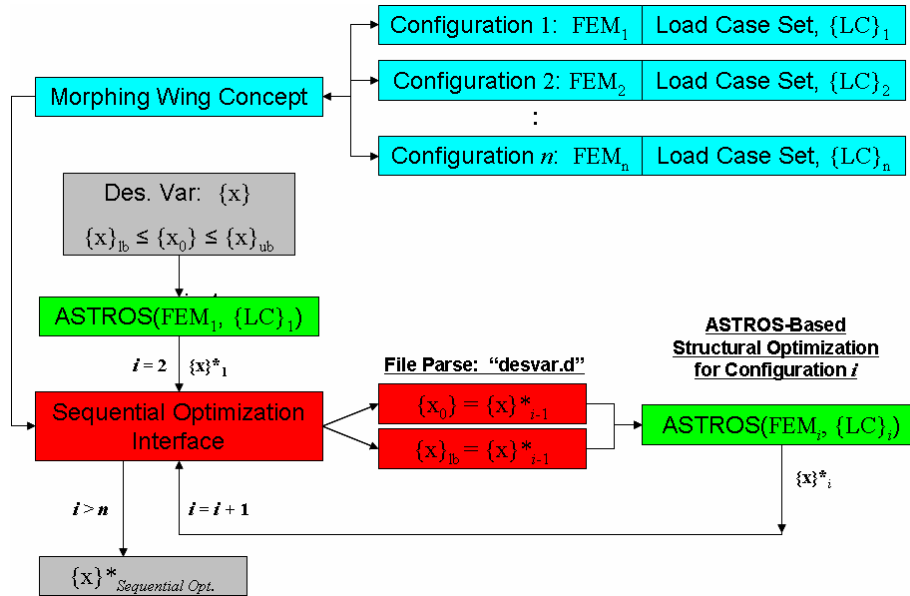


Figure 3. “Sequential” Optimization Process Architecture and Information Flow

4. “Aggregate” Approach

The aggregate approach is similar to the sequential optimization strategy in that the FEA / optimization task remains coupled and implemented within the FEA/MDO environment. This method requires slightly less effort to obtain a final “optimal” structural design for the morphing wing, but the underlying result is far less robust from a standpoint of final design efficiency / optimality. The architecture for this setup is shown in Figure 4. While maintaining constant design variable bounds and initial conditions, each configuration is structurally sized using ASTROS. The resulting “optimal” designs will be indicative of the most efficient load paths for a given configuration (structural arrangement) and its associated design load cases. By integrating the largest element dimensions from the set of “optimal” structural designs, the “aggregate-optimal” structural design is established.

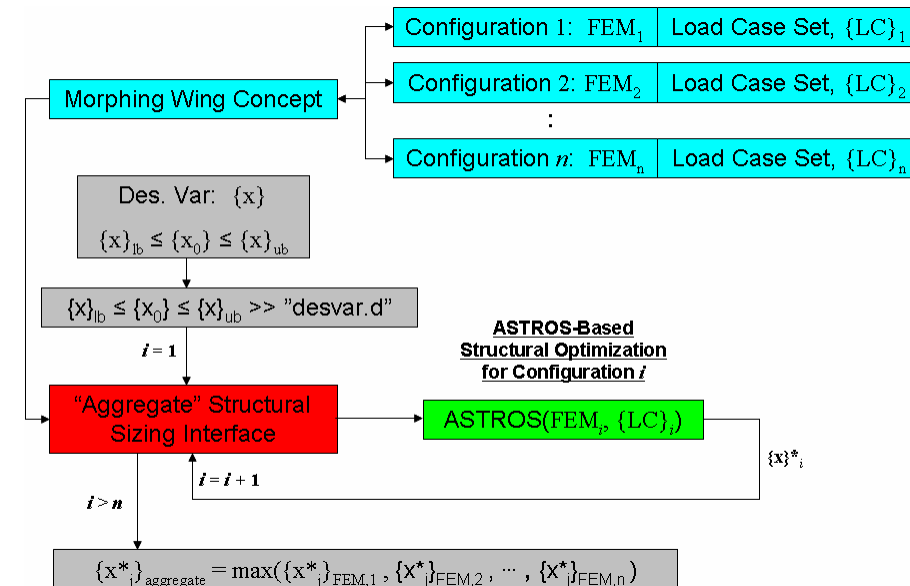


Figure 4. “Aggregate” Optimization Process Architecture and Information Flow

C. Developing Morphing Wing Finite Element Models – An Object-oriented Preprocessor

In order to expedite trade studies at the conceptual or preliminary design level, a preprocessor was developed to generate intermediate complexity morphing wing models based on top-level wing parameterizations. This idea has been previously implemented for fixed-geometry type wing structures and is documented in References 16 and 17. In their work, Sensmeier, *et al.* have developed an algorithm that maps a generic structural topology defined in a natural coordinate system onto a continuous or piecewise continuous wing planform topology. This sort of mapping allows spar and rib locations to be initially defined as a percentage of span and chord lengths (i.e. a spar runs the length of the wing span $\{s: [0 \ 1]\}$ at the 20% chord position $\{c: [0.20 \ 0.20]\}$); this idea is illustrated in Figure 5. This generic topology can then be mapped onto the topological domain of the wing's OML and then extruded to the wing's upper and lower skin domain creating the 3-D representation of the finite element model. Reference 17 further alludes to the applicability of this process to morphing wing concepts and demonstrates the idea with a folding wing design. However, treatment of the morphing capabilities did not appear to be an original priority during their core software development and a different approach has been taken in this work to incorporate the facets of general morphing wing characteristics. Note, however, that while the parametrically defined structural model will be discussed here, this topic is the original work of Sensmeier, Samareh, and Stewart. The deviations from their approach will be clearly defined when applicable and represent original work by this author.

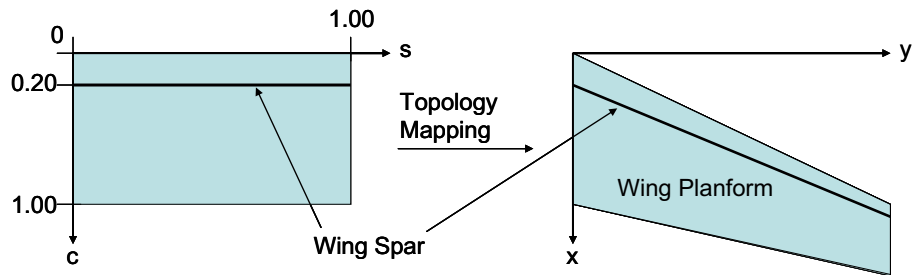


Figure 5. Illustration of Structural Topology Mapping from a Natural Coordinate Frame to the Wing Planform Domain

5. Facets of a Morphing Wing System

Recent development of various morphing wing concepts has led to the generalization of potential morphing wing characteristics. For example, Lockheed Martin's folding wing design incorporates hinged joints at two spanwise stations enabling rigid body motion of four primary wing sections. The NextGen Aeronautics concept incorporates a reconfigurable shell structure with an elastic skin to achieve wing shape variation. From these, the generalized facets of a morphing wing constitute the following design elements:

- A collection of traditional wing substructure / superstructure components mechanized with movable joints or hinges allowing rigid body motion of the constituent wing surfaces (e.g. a conventional variable sweep wing, or LM's folding wing design)
- A reconfigurable / movable substructure coupled with an elastic skin enabling in-flight wing shape variation (e.g. NextGen's morphing wing concept)
- A combination of these types of wing sections having both fixed structural components as well as reconfigurable substructure components

In addition to incorporating these characteristics, the preprocessor architecture will naturally reduce and apply to fixed-geometry wing structures.

6. Morphing Wing Modeling - Process Architecture

The basic premise behind the developed preprocessor is that the wing structure will initially be defined as a mechanism and, from this, be parametrically defined as aero/structural components. More specifically, the initial

mechanism-based model will consist of a set of control points that govern spatial orientation of primary structural elements and/or OML domain bounds. The spatial arrangement of the control point set will be a function of user-defined morphing parameters that describe the morphing domain / shape-changing capabilities of the wing and the top-level wing parameters that govern overall wing shape geometry (AR, taper, sweep, etc.). Once the control point set has been established, the topology of the primary and secondary structural elements will be defined based on a parametric, user-defined, structural template similar to that used in References 16 and 17. The established 2-D structural topology will then be used to generate the intermediate complexity FEM by extrusion to the OML domain. Note that the control point set can be used to define / segment the wing into a number of wing sections; each section will be defined by the wing's aero/structural template and need not be coplanar. In this case, each wing section will have an associated local reference frame that is defined relative to the global frame. The topology mapping function will be applied to each wing section in its local coordinate system and then converted to the global frame once the extrusion has occurred and the section's FEM generated.

In the case of multiple wing sections, the structural connectivity between the sections will be of major importance. Treatment of this is facilitated via structural "bridge". The "bridge template" is a user-defined object that describes how to join two wing sections by defining: Required modifications to the associated wing sections' topology; the topology mesh to be inserted between the sections; and / or the finite element layout/mesh scheme used to connect the primary structure of section i to section j . Several pre-defined bridge-template examples are illustrated in Figure 6. The first is a "continuous" bridge that essentially has little function; the spars are piecewise continuous, so no topology modifications are needed. The "hinge" bridge modifies the original sections' topology

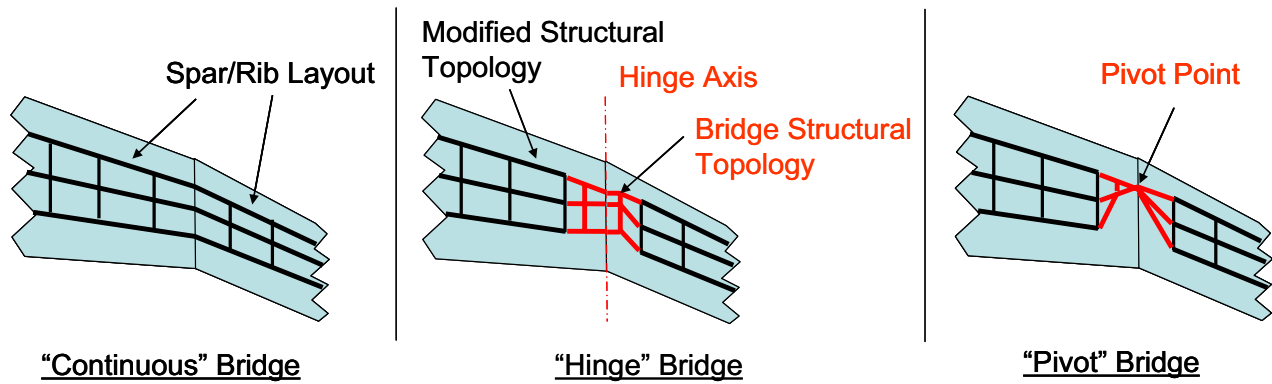


Figure 6 Several Examples of Structural "Bridges"

and inserts its own structural grid to facilitate a hinge mechanism that enables out-of-plane rotation about the hinge axis. The "pivot" bridge has similar features of the hinge bridge, but converges the sections' primary structures to a pivot joint enabling conventional variable sweep type motion. This set of bridge templates is not exhaustive, but serves to demonstrate the basic premise of this object.

Overall, the basic preprocessor architecture is an object-oriented framework implemented entirely within the Matlab environment. During execution, all processes will act to populate two main variables: The 'wing' structure and the 'DB_FEM' structure. The former contains all geometric information of the model including aerodynamic surface geometry, structural topology, 3-D structural geometry, coordinate frame information, structural bridge information, etc. that is structured in an object-oriented fashion to facilitate traceability (e.g. wing.section(1).str_geom.spar(1) would request information associated with the LE spar of wing section #1). The 'DB_FEM' structure contains the same information, but in a form necessary to write all BDF entries. For example, the associated data structure 'nodes' would be a primary component of 'DB_FEM', and the variables 'ID_node', 'X', 'Y', 'Z', and 'ID_ref_frame' would be a subcomponent of 'nodes'. To write the grid point data associated with the FEM into the BDF, a script would loop through all entries in the 'node' structure and write the variable information in the format required by the specific FEA environment. Note that the components of the 'DB_FEM' variable are cross-referenced in the 'wing' structure to further trace BDF information back to the original wing model components. For example, the LE spar will be discretized into a number of finite elements; one of which will be a 'shear' element. After the 'shear' element data has been written into the 'DB_FEM' structure, a corresponding element identification number (EID) will be returned and then cross-referenced in the 'wing' structure. A more

thorough discussion of information flow and the major form of the data structures is presented in Section II.C.7: *Information Flow and Major Data Structures*.

A formal description of the process flow will be presented now that the basic underlying ideas have been established. Note that a process entry without color indicates that some form of user-level input is required to execute the process. A green process entry indicates a “core program function” associated with structural model generation, and is executed automatically without a user request. Similarly, a blue process entry indicates a “core program function” associated with the BDF generation.

Process / Associated Function File(s)	Information
Define Control Point Set in Reference Configuration <i>PARAM_define_cp_set();</i>	User-defined function file; uses top-level wing geometry information (e.g. AR, λ , Λ , etc.)
Morph Control Point Set <i>MORPH_cp_set();</i>	User-defined function file; uses the control point set defined in the reference config. and the requested state of the morphing parameters (e.g. increase OB sweep 30 [deg]) to spatially rearrange the control point set.
Define Wing Sections <i>SECTION_add(..)</i>	Specify wing structure type: {BoxBeam, Shell} Specify section connectivity: {WingRoot, WingTip, Bridge}
Define Bridges (If model has multiple sections) <i>SECTION_add_bridge(...)</i>	Specify bridge type: {'Continuous', 'Hinge', etc.} Specify connected sections: [ID_sec1, ID_sec2] Specify array of pertinent data: [DATA]
Define Aerodynamic Surface for each Wing Section <i>SECTION_add_aero(...)</i> NOTE: This is to define the OML for topology extrusion	Specify Section ID Specify Airfoil (e.g. NACA0012) Specify input data plan: [integer] (e.g. 7 = derive planform from a specified control point set) Specify input data: [DATA] depends on plan chosen Specify reference frame: position / orientation
Define Aerodynamic Paneling Information <i>AERO_add_steady(...)</i> <i>AERO_add_steady_panel(...)</i>	Specify Section ID Specify Num. Spanwise Panels for CFD surface Specify Num. Chordwise Panels for CFD surface
Define Structural Topology Layout <i>PARAM_MODEL_topology()</i> <i>PARAM_add_comp(...)</i>	User-defined function file Define 2-D layout of spars and ribs (or shell mesh) as a function of control point locations, planform topology information, or existing spar/rib data
Define Skin Panel Superelements <i>SECTION_add_struct_skin(...)</i>	Specify area over specific spars and ribs to attach skin panels. If this skin panel superelement is discretized later, the preprocessor will link the associated elements making up this panel to a common design variable / thickness
Add Boundary Conditions <i>SECTION_add_support(...)</i>	Specify B.C. type: {cantilever, SUPPORT pt.} Specify structural components to which BC applies/links Add concentrated mass (aircraft mass if SUPPORT)
Develop / Implement the Bridge Structure (if applicable) <i>BRIDGE_initialize_data()</i>	Operate on the user-specified topology information to accommodate the requested bridge type. Add bridge structural topology information if applicable
Discretize the User-Defined Topology Information <i>STRUCT_grid_topology()</i>	Given user-defined topology information, discretize the spars and ribs at any intersection points
Extrude the Structural Topology to the OML Domain <i>STRUCT_extrude_topology()</i>	Extrude the structural topology information to the OML domain to get a 3-D representation of the components.
Initialize the 'DB_FEM' structure <i>INITIALIZE_DB()</i>	Initializes the major constituents of the 'DB_FEM' structure.
Write Material Data into the FEM Database <i>DB_add_material(...)</i>	Pre-defined Material Database Types: {ALUM2024} The material associated with the primary structure defaults to the pre-defined ALUM2024 properties but can reference other user-defined materials if desired

Write Object-Oriented Spar Information into DB_FEM <i>DB_add_spars()</i> <i>DB_add_node()</i> <i>DB_add_el_shear()</i> <i>DB_add_pshear()</i> <i>DB_add_el_bar()</i> <i>DB_add_bar1()</i>	Takes the extruded (3-D) spar data from the 'wing' structure and translates the information into 'nodes', 'shear', and 'bar' entries in the 'DB_FEM' structure. Associated GRID IDs and element IDs are cross-referenced in the 'wing' structure
Write Object-Oriented Rib Information into DB_FEM <i>DB_add_ribs()</i>	Same idea as for spars
Translate Other Object-Oriented Info. into DB_FEM <i>DB_add_skins()</i> <i>DB_add_supports()</i> <i>DB_add_aerospline()</i>	<i>DB_add_bridges()</i> <i>DB_add_coord2r()</i> <i>DB_add_aero_steady()</i>
Add Design Variable Information into DB_FEM <i>DB_add_desvar_spar(...)</i> <i>DB_add_desvar_rib(...)</i> <i>DB_add_desvar_skin(...)</i>	Writes 'DESVARP' elements into the 'DB_FEM' structure. Design variable linking (elements linked to a common design variable) is facilitated based on a user-defined scheme (e.g. link shear webs in groups of two along the sparlength)
Write All NODE Info. into the User-defined BDF <i>FEM_write_nodes()</i>	Calls the function file <i>FEM_write_nodes()</i> which loops through all 'DB_FEM.nodes' data and writes the properly formatted cards into the BDF
Write All Remaining Info. from the DB_FEM Structure into the BDF <i>FEM_write_coord2()r</i> <i>FEM_write_el_cshear()</i> <i>FEM_write_el_cquad4()</i> <i>FEM_write_el_ctria3()</i> <i>FEM_write_el_cbar()</i>	Same approach as the nodes except applied to all other relations within the DB_FEM structure <i>FEM_write_el_crod()</i> <i>FEM_write_el_rbar()</i> <i>FEM_write_materials()</i> <i>FEM_write_aero_caero6()</i> <i>FEM_writeetc.</i>
Program Complete / BDF written	At the end of execution, the 'DB_FEM' and 'wing' structure remain in the Matlab workspace allowing the user to trace any desired information

7. Information Flow and Major Data Structures

Information storage during the pre-processing task is considered to be the crux of this architecture. As discussed above, the transition of the information from the object-oriented 'wing' structure into the BDF relation-oriented 'DB_FEM' structure prior to the BDF output dump enables a very traceable information architecture with minimal coding chaos within the main source code. The purpose of this section is to more accurately describe the primary variables, 'DB_FEM' and 'wing', and their component hierarchies.

The following table defines the top-level components of the 'wing' structure (i.e. these would be accessed by typing **wing.x** in the command window console, where **x** is an associated top-level component):

wing.[]	Description
morph. cp_set.cp	Contains the set of user-defined control points and their spatial position in global coordinates
section. TYPE_STRUCT connect ID_bridge ID_coord aero_geom str_topology str_geom	Contains all wing section information: Type of structural design {BoxBeam, Shell} Connectivity at [section root, section tip]: {'WingRoot', 'WingTip', 'Bridge'} Index ID of wing.bridge component associated with the bridge of this section Index ID of wing.coord comp. associated with this section's coordinate frame Top-level aerodynamic information of this wing panel [AR, λ , Λ , b, S, t/c, etc.] Topology information associated with this wing panel Object-oriented 3-D structural info. (e.g. spars, ribs, skins)
coord G_pos orient	Contains information for all defined coordinate frames (# frames = # sections) Position of this coordinate frame's origin in global coordinates: [x,y,z] String describing orientation of this frame: {'GLOBAL', 'SECTION'}

axis_x_G axis_y_G axis_z_G T L to G	Vector defining the x-axis of the local frame in global coordinates [Δx , Δy , Δz] “ y-axis “ “ z-axis “ Transformation matrix \rightarrow Local coordinate frame to global frame
bridge type ID_sections DATA nodes connectivity	Contains the relevant information associated with the bridge objects Bridge type: {'Continuous', 'Hinge', 'Pivot'} Index ID's of the wing.section components that are connected with this bridge A vector of any required data for this bridge (dependent on bridge type) A structure containing node data that are added to the DB_FEM structure and referenced exclusively by this bridge 3-D Structural geometry generated by this bridge to connect the associated wing sections
aero_model_steady	Contains information associated with the full wing aerodynamic model (e.g. reference area, reference chord length, reference span length, SUPPORT node, etc.)
support	Contains information regarding the structural boundary conditions such as which structural components to constrain or link via rigid bar connections

Although the previous table demonstrated the basic information hierarchy of the 'wing' structure, subcomponents of this structure need to be more intimately discussed as they constitute the most important aspects of this pre-processing architecture; the 'wing.section.str_geom' structure will be described in this section.

wing.section.str_geom.[]	Description
spar pos L r_vect div_pts comp x y z nodes r pos L div_pts EID_shear PID_shear EID_rod_vert EID_bar_caps	A collection of 'spar' objects associated with this wing section The position of the spar's origin defined in local coordinates The length of the spar A vector defining the orientation of the spar in local coordinates A vector defining the intersection points with other structural members A collection of discretized panels making up the 3-D spar A vector defining the x-ordinates of this 3-D spar panel component “ y-ordinates “ “ z-ordinates “ A vector of node ID's associated with the [x,y,z] panel ordinates A vector defining the 2-D orientation of the spar panel The starting position of the 2-D spar layout defined in local coordinates The length of this 3-D spar panel component A vector of points describing further discretization of this spar panel The element ID # of this shear element in the DB_FEM relation The property ID # of the associated shear element's property card in the DB_FEM relation The element ID #'s of the vertical dummy rods in the DB_FEM relation The element ID #'s of the spar caps ('bar' el.) in the DB_FEM relation
rib	A collection of 'rib' objects associated with this wing section. The information hierarchy is the same as the 'spar' objects
skin EID_quad4_upr EID_quad4_lwr nodes_upr nodes_lwr	A collection of 'skin' superelements containing associated element information A vector of element ID #'s associated with the 'quad4' elements making up the upper surface of this skin superelement A vector of element ID #'s associated with the 'quad4' elements making up the lower surface of this skin superelement A vector of unique node #'s making up the upper surface of the superelement “ lower “

Next, the 'DB_FEM' structure will be described. Again, this database structure is essentially a reproduction of the data within the 'wing' structure but in an organizational form more conducive to the task of writing the BDF. However, this also facilitates traceability between the BDF, the 'DB_FEM' structure, and the 'wing' structure. The

following table illustrates the primary components of the 'DB_FEM' database structure and the most relevant subcomponents of these objects. Note that these primary and secondary components have one lower component beneath them that describes the major data for the BDF relation; this will be demonstrated momentarily.

DB_FEM.[]	Description
coord2r	Collection of coordinate frame information
nodes	Top-level node structure
node	Collection of 'node' objects
pos	Node position in global coordinates
elements	Top-level components referencing finite elements
shear	Collection of 'shear' element objects
rod	Collection of 'rod' element objects
quad4	Collection of 'quad4' element objects
tria3	Collection of 'tria3' element objects
rbar	Collection of 'rbar' element objects
bar	Collection of 'bar' element objects
pbar1	Collection of bar element property card objects
prod	Collection of rod element property card objects
pshear	Collection of shear element property card objects
pshell	Collection of shell element property card objects
mat	Collection of material objects
celas2	Collection of structural spring element objects
support	Collection of structural support B.C. objects
conmass	Collection of concentrates mass objects
aeros, caero, airfoil, aerfact, desvarp, elist, elistm, dconvm, spline, set, mpc	... etc.

Now, an example of the sub-level data beneath one of the primary 'DB_FEM' components will be demonstrated. In NASTRAN and ASTROS, a bar element is defined in the BDF by specifying the following information set:

CBAR	EID	PID	GA	GB	X1	X2	X3
------	-----	-----	----	----	----	----	----

The EID is the element ID #, the PID is the ID # of a PBAR property entry in the BDF, GA is the grid point ID # used to define the starting point of the bar, GB is the grid point ID # specifying the end of the bar, and [X1,X2,X3] is a vector defining the orientation of the bar's local coordinate system. The following table shows the information stored in a 'DB_FEM.element.bar' object:

DB_FEM.elements.bar.[]	Description
EID	The element ID # of this bar element
PID	The property ID # of the PBAR card associated w/ this element
nodes	A [2x1] vector of grid point ID # specifying GA and GB respectively
X1	The x-component of the orientation vector defined in the global frame
X2	The y-component of the orientation vector defined in the global frame
X3	The z-component of the orientation vector defined in the global frame

Notice that the information required in the BDF entry is nearly one-to-one with the data storage within the 'DB_FEM' structure, and in most cases is equivalent. As a further demonstration of the pre-processing task, the autonomous core-program function file used to write the 'bar'-object information into the BDF is shown below:

<i>FEM_write_el_cbar(...).m</i>
<pre>function [] = FEM_write_el_cbar(DB_FEM, fid_out); script ID = 'FEM write el cbar';</pre>

```

msg = sprintf('Adding CBAR data information into the FEM bulk data file...');
ECHO_show_msg(script_ID, msg);

fprintf(fid_out, '\n$\n$$$$$$$$$\n$ BAR ELEMENTS\n$$$$$$$');

bar_el = DB_FEM.elements.bar;
for(i = 1:length(bar_el))
    EID = bar_el(i).EID;
    PID = bar_el(i).PID;
    GA = bar_el(i).nodes(1);
    GB = bar_el(i).nodes(2);
    X1 = bar_el(i).X1;
    X2 = bar_el(i).X2;
    X3 = bar_el(i).X3;

    fprintf(fid_out, '\nCBAR,%3i,%3i,%3i,%3i,%6.4f,%6.4f,%6.4f', i, PID, GA, ...
                                                    GB, X1, X2, X3);
end

return

```

Note that this routine is a Matlab-based function file. The input variable ‘fid_out’ is a FILE pointer to the pre-opened BDF. A desirable utility that comes from this approach is that if the user needs to switch from one FEA environment to another, and these require different BDF formatting techniques, then this file can simply be exchanged by one with the proper formatting scheme. This is desirable because the user does not have to alter or become familiar with the core-program code to make the modifications. Ideally, the suite of *FEM_write_x()* functions could be defined for a number of FEA environments, and based on a user-level switch identifying the FEA tool, the appropriate files could be copied into the working directory at run-time so that the task would be transparent to the user.

8. Process Demonstration Using a Folding Wing Concept

The subtleties of model generation using this approach are best demonstrated by example. Note that the user-defined script defining this wing template can be found in Appendix A. Consider Lockheed Martin’s folding wing concept as illustrated in Figure 7. Because this is a symmetric wing concept, a halfspan model will be used to demonstrate the process. First, assume the following major top-level parameters are used to define the wing geometry in its unfolded configuration: $\{\lambda_{IB} = 0.4, \lambda_{OB} = 0.65, b_{\text{halfspan}} = 15 \text{ [ft]}, \Lambda_{LE} = 40 \text{ [deg]}, c_{\text{root}} = 15 \text{ [ft]}, b_{\text{hinge,IB}} = 0.2, b_{\text{hinge,OB}} = 0.4, t/c = 12 \text{ [%]}\}$. This information will be used to establish a control point set (via user-defined control point template) that defines the OML extents of three primary wing surfaces. This is demonstrated in the left illustration of Figure 8 where the control points are indicated by circled numbers. Note that the control point set is always defined initially in the morphing wing’s reference configuration. Now the morphing parameters are tied into the algorithm. For this concept, the fold angle of the intermediate wing section, β , is the only associated morphing parameter. The control point set is then updated / morphed using a user-defined template that uses the value of β and the position of the control points in their reference configuration as input. Note that if the fold angle is zero, the control point set is unchanged and the model will be developed in the reference configuration. In the right illustration of Figure 8, the fold angle is set to 135 [deg] and the morphed positions of the control points are indicated with primes to denote the morphed state. The actual Matlab scripts used to generate this data can be found in Appendix B. Now, the morphed control point set can be used to build-up the aerodynamic and structural topology models. First, three wing surfaces are defined as ‘BoxBeam’ type structures with ‘Bridge’ connectivity between sections [1,2] and [2,3].



Figure 7 Lockheed Martin’s Folding Wing Concept

Next, the bridges are defined more fully by specifying a 'BridgeType' of 'Hinge' and [DATA] that pushes the basic structural topology away from the hinge joint by the length equal to the thickness of the airfoil at the hinge location. The basic planform topology and OML is

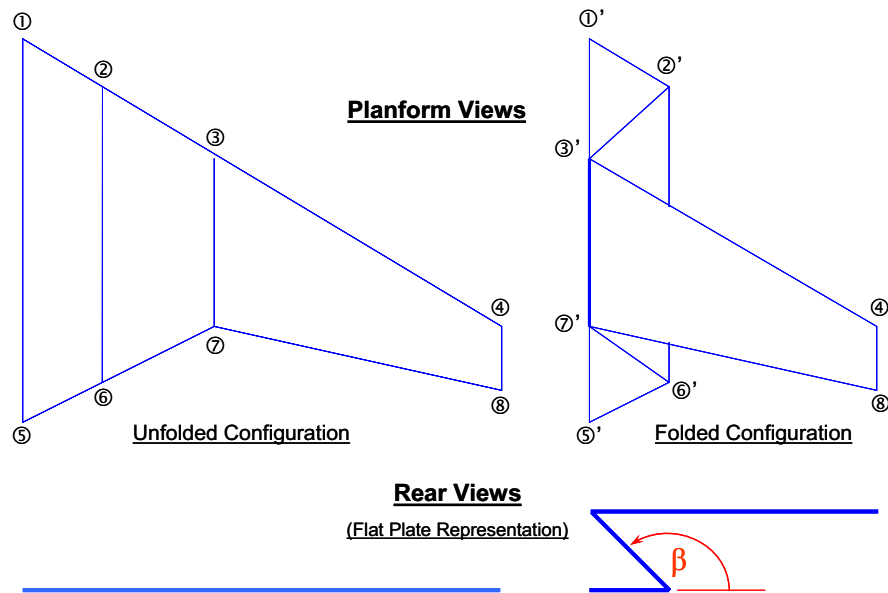


Figure 8. Orthographic Views of the Folding Wing Concept Illustrating Control Points and Morphing Parameter, β

established next by calling the *SECTION_add_aero()* routine for each wing section while specifying the control points associated with that particular section. During this call, the preprocessor will also establish a local coordinate frame associated with the wing section with respect to the global frame; a transformation matrix from local to global coordinates is also saved in the 'wing.coord' object. At this point in the preprocessor, a plot request is implemented to verify the position and orientation of the wing section planform geometry and the associated control points; this is shown in Figure 9. In this case, the preprocessor was run twice to demonstrate the wing in two morphed states: Run #1 specified a fold angle of zero, which produced the reference configuration layout; run #2 specified a fold angle of 110 [deg] which produced the morphed state as illustrated on the right side of the figure.

Now, the CFD paneling information is defined by specifying the number of chordwise and spanwise panels for each wing section; this will be evident in later plots. Next, the structural topology is defined for each section. Section 1 (most IB section) is defined to have 4 spars and 3 ribs with a LE aero/structural offset of 10% and a TE aero/structural offset of 20%. The same is true for section 2 (mid-section) but will have 5 ribs. Furthermore, section 3 will have similar aero/structural offsets for the LE and TE spars, but will have 3 spars and 9 ribs. Note that the number of ribs and spars for each section is a user-defined parameter and is not hard-coded into the wing template. Also note that although the number of ribs have been specified outright, constant length rib spacing is an option. At this point, when the structural topology information is being processed, the inclusion of the 'Hinge' bridge will trigger a

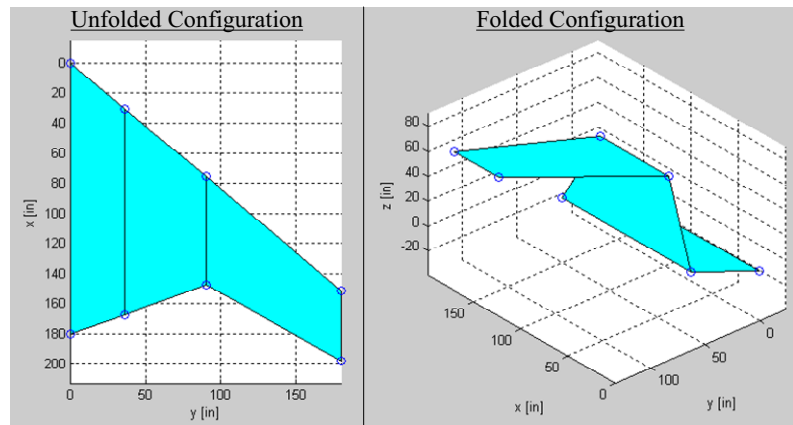


Figure 9. Illustration of the Control Points and Planform Geometry in the Unfolded Configuration (Left) and Folded Configuration (Right)

modification to the structural topology layout; specifically, the function will offset the ribs away from the hinge joint by an amount specified by the user (a length equal to the thickness of the airfoil at the hinge axis station in this case). The topology layout is applied to the wing section in its local reference frame, so that no unnecessary or overbearing data manipulation and “record keeping” is taking place to keep track of local and global information. Via plot request, the structural topology is shown for each wing section in its local frame in Figure 10. However, a plot request is also available that renders the basic planform geometry and structural topology information for the “assembled” structure in the global frame (see Figure 11). At this point, skin panel superelements are defined by the user and, in this case, are defined by the area spanning the region bounded by consecutive ribs and the LE to TE

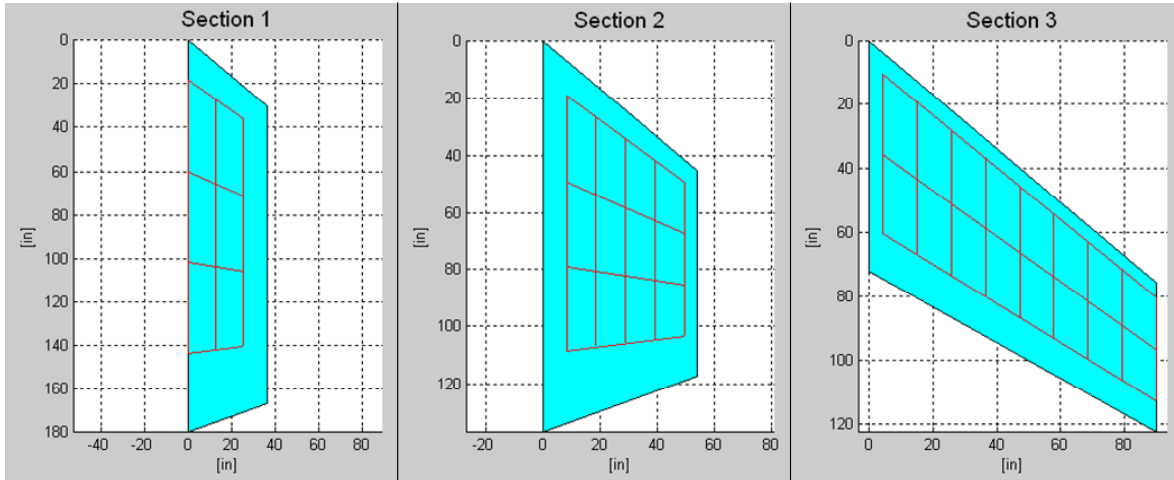


Figure 10. Structural Topology Layout on each Wing Section

spars. This will be evident when the design variable linking plot is shown; the skin panels (QUAD4 finite elements) making up the region spanned by the superelements, as defined here, will be rendered in similar color (Figure 15).

The preprocessor will now switch from the user-defined scripts to the “core program architecture” that is essentially a suite of autonomous function calls that will fully generate the 3-D FEM, further populate the ‘wing’ structure with relevant information, transfer the ‘wing’ structure information into the ‘DB_FEM’ structure, and translate the ‘DB_FEM’ data into a working bulk data file. First, the structural topology information is extruded to the OML domain for each section; this is again done in the local frame, but a plot routine exists to render the fully assembled wing structure. The extruded geometry is shown in Figure 13 for the individual wing sections in their local reference frames. Furthermore, any associated ‘bridge’ components of the wing are operated on by the preprocessor to develop any required 3-D structural elements. The 3-D ‘hinge’ bridge is illustrated in

Figure 12. Notice that structural membrane elements (CTRIA3) are connected from the wing section’s spars to a point on the hinge axis. At each hinge point, two nodes are defined at the same spatial position; the triangular element from section i connects to one of the nodes while the element from section j connects to the other. Associated DOF of these nodes are linked to have a common deflection except for the rotational DOF about the chordwise axis. In this case, a torsional spring is added between the two nodes to carry bending loads at the hinge.

Next, all relevant data within the ‘wing’ structure will be autonomously transferred into the ‘DB_FEM’ structure (an FEM-oriented data format). Upon completion of this task, a plot request can be implemented to visualize all major geometric entities within the ‘DB_FEM’ structure, including the aerodynamic surface paneling (Figure 14). An important note is that this illustration is generated via information from the ‘DB_FEM’ data structure only; data from the ‘wing’ structure is unnecessary at this point because both structures contain the same information, but use different organizational schemes. Another helpful plot

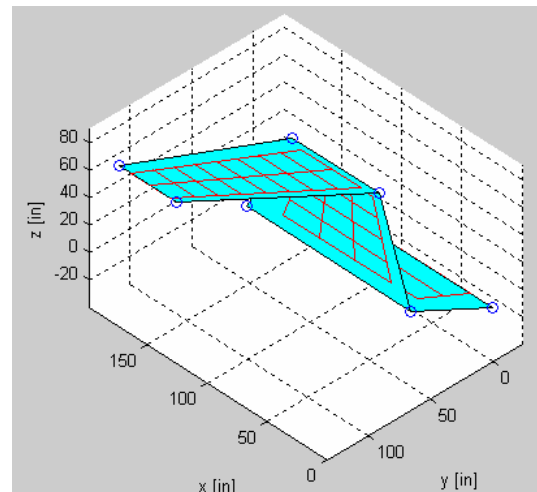


Figure 11. Illustration of Structural Topology for all Sections Rendered in the Global Reference

aids in the visualization of linked elements sharing a common design variable (Figure 15). In this example, the design variable linking scheme for the spars is as follows:

- The thickness of every two shear webs are linked to a common design variable starting from the root
- The spar cap areas of every two consecutive bar elements (including upper and lower spar caps) are linked to a common design variable starting from the root; this translates into 4 bar elements being linked to a common design variable
- The skin panels associated with a skin superelement have thicknesses linked to a common design variable

After the 'DB_FEM' has been fully populated via *DB_add 'x'()* function calls, the suite of *FEM_write 'x'()* functions are called to transfer all data within the 'DB_FEM' structure into the actual BDF.

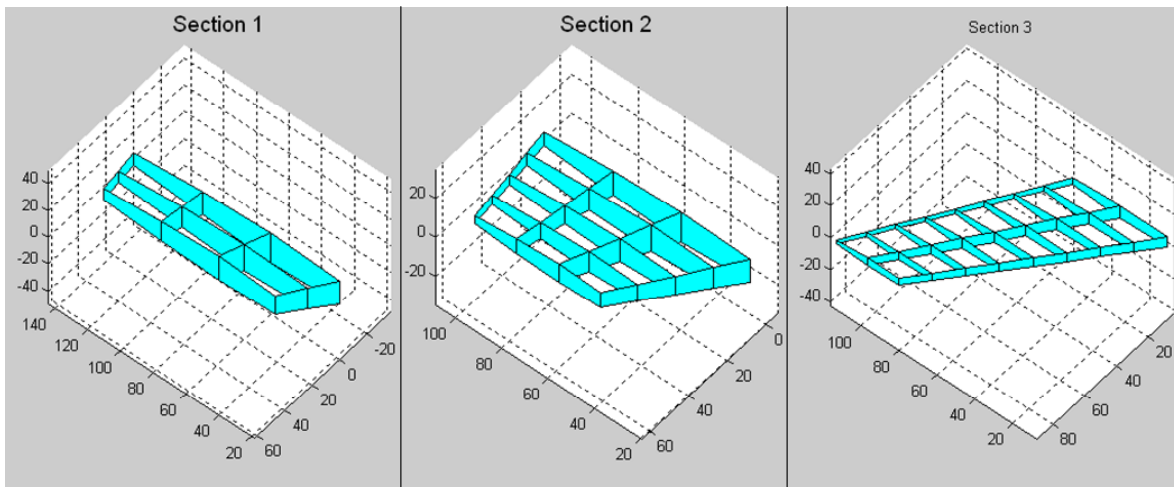


Figure 13 Structural Topology Layout on each Wing Section

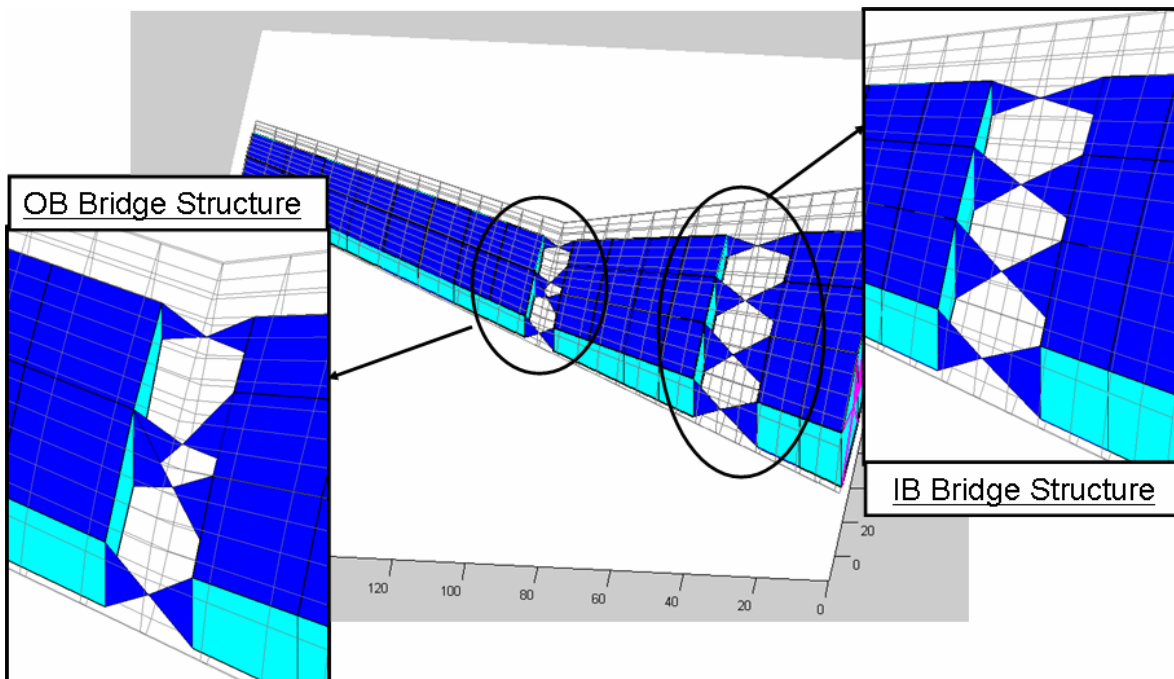


Figure 12 Illustration of the 'Hinge' Bridge Structures on the Fully Assembled Wing Model (Unfolded)

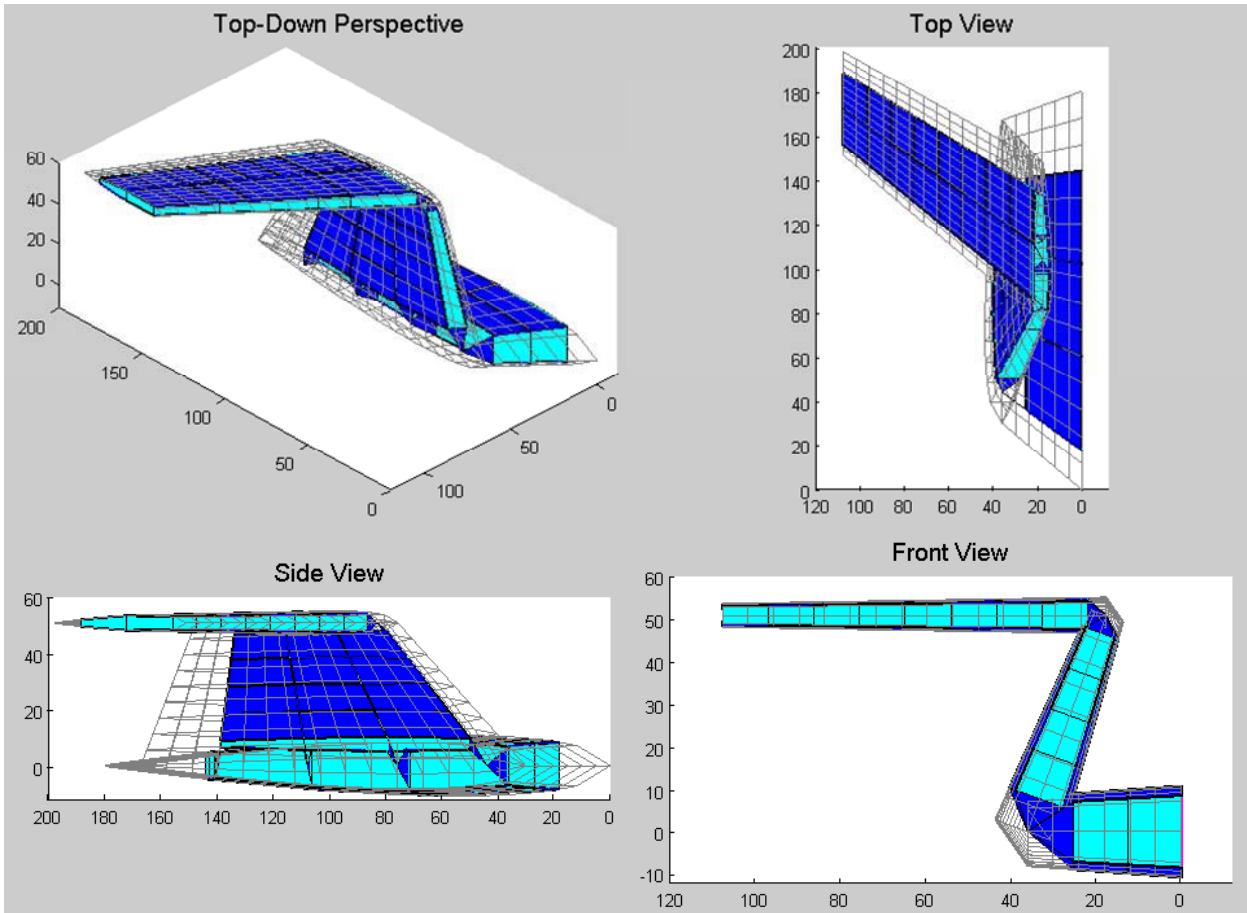


Figure 14 Processor Plot Request Using Data Exclusive from the 'DB FEM' Structure

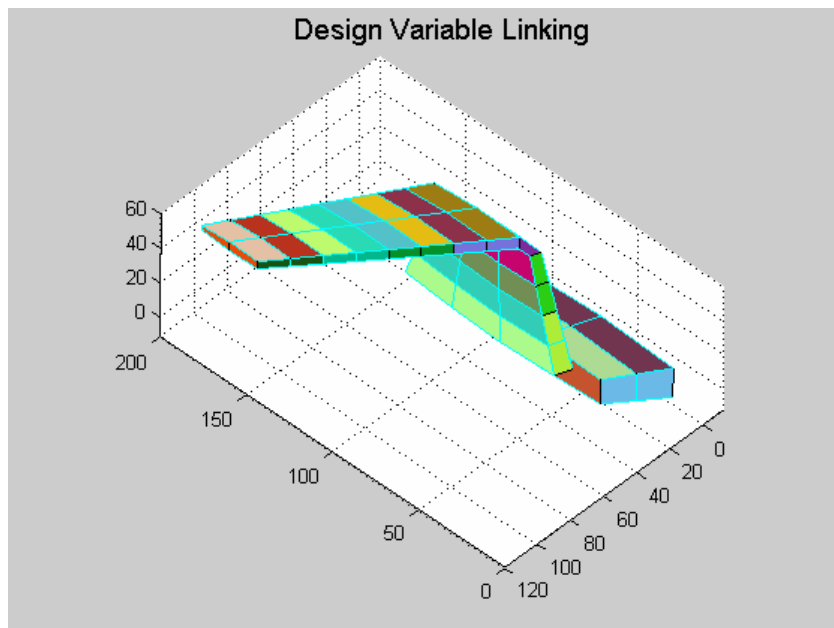


Figure 15. Elements Linked to a Common Design Variable are Rendered with Common Colors

D. Case study – Structural Sizing of the “Shear-Wing” Concept

Now that the preprocessor is in place, a simple “shear-wing” template will be described and used to generate two primary configuration FEMs. These will then be used to demonstrate the structural sizing strategies presented in Section II.B.

9. Overview of the Concept

The “shear-wing” concept is based upon the idea that a rectangular cell can be distorted into a parallelogram via shearing motion without straining the perimeter walls. If the cell walls represent traditional aluminum-construction spar webs, shear webs, and spar caps, and the enclosed area is assumed to be covered with a highly elastic material similar to that used on NextGen Aeronautics morphing wing concept, a collection of these cells can form the primary structure of the shear-wing concept; Figure 16 illustrates this idea. Beyond this basic premise, one could imagine that a tapered wing could be modeled using a discontinuous collection of these cells. Furthermore, conventional ailerons and/or trailing edge flaps could be connected into the trailing edge spar without much compromise; especially if the connecting spar is continuous over the length of the span. A leading edge device could be attached in a similar manner.

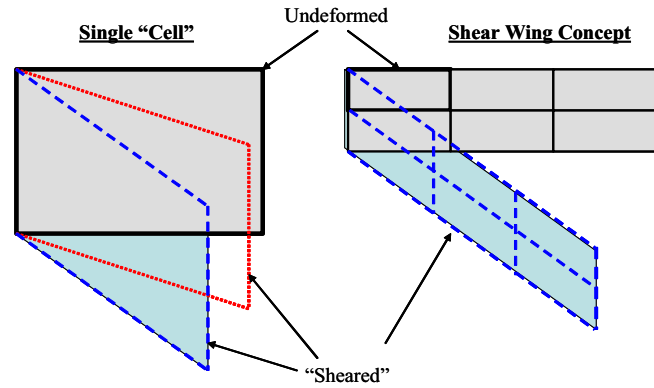


Figure 16. Planform View of the “Shear-Wing” Concept

10. Model schematic and related system / sizing information

a) Model Schematic

Figure 17 illustrates the basic planform and structural topology of the shear-wing concept that will be used in the subsequent case studies; this is presented as a halfspan model and is assumed symmetric about the x-z plane. Notice that the extent of the aerodynamic model is highlighted in cyan, while the substructure topology is outlined in black within this area. Also notice the guide strut attached to the internal trailing edge spar; while only modeled as an external spar, this support is intended to have similar aerodynamic form and structural function as a brace-strut on a braced, high-wing transport. In this case, the brace will also function as the means to control the sweep angle of the wing; the angle, β , as indicated in the figure will be equal to the leading edge sweep of the wing during morphing. A quantitative description of these models is also presented in Table 1. Note that the structural model incorporates an all-aluminum construction and uses general material properties for ALUM-2024.

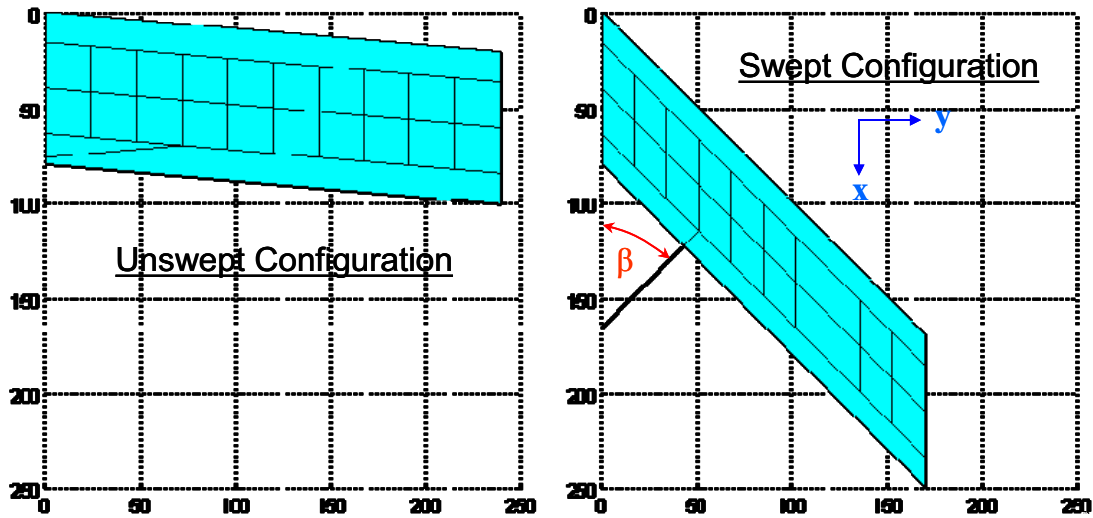


Figure 17. Planform and Structural Topology of the Shear-Wing Concept: Halfspan Model of the Unswep Configuration (Left); Halfspan Model of the Swept Configuration (Right)

Wing Parameter	Unswep Configuration		Note	Swept Configuration	
	Value	Unit		Value	Unit
Λ (LE)	5	[deg]		45	[deg]
AR	5.98	[]	Full Wing Model	4.24	[]
S	38254	[in ²]	Full Wing Model	27152	[in ²]
	265.7	[ft ²]	Full Wing Model	188.6	[ft ²]
c_root	80	[in]		80	[in]
λ	1	[]		1	[]
b (body axis)	239.1	[in]	Halfspan Wing Model	169.7	[in]
b (elastic axis)	240	[in]	Halfspan Wing Model	240	[in]
Structural Offset	20	[%]	LE Aero/Structural Offset	20	[%]
	20	[%]	TE Aero/Structural Offset	20	[%]
Airfoil	NACA0012	---	Constant Over Spanlength	NACA0012	---
			Zero Twist Distribution		

Table 1 Quantitative Description of Major Wing Parameters for the Shear-wing Concept

b) Load Cases

The design load-case set for this study will be limited to a single symmetric pull-up maneuver in each configuration. While the actual design specifications might prescribe different flight conditions and maximum maneuvering envelopes for each configuration, this case study will not bias the structural design conditions in this way; each configuration will be designed to a 5-g pull-up maneuver (this translates to a 7.5-g ultimate load factor that will be used to size the wing structure). Because the full aircraft has not been modeled, the aircraft mass has been concentrated at the support point of the finite element model; the support point is located at the 25 percent chord position at the root of the wing. To perform the symmetric aeroelastic trim analysis, the pitch and vertical displacement DOF are free while all others are fixed at the support point. Note that a reaction load is applied to the support point to prevent a rigid body singularity while the pitch DOF is determined during the aeroelastic trim

analysis. Inertia relief due to wing structural mass in the accelerated frame is also intrinsically incorporated during the analysis.

<u>Load Case Param.</u>	<u>Unswept Configuration</u>		<u>Note</u>	<u>Swept Configuration</u>	
	<u>Value</u>	<u>Unit</u>		<u>Value</u>	<u>Unit</u>
Aircraft Wt.	20000	[lb]	TOGW assumed ~25000	20000	[lb]
Wing Loading	75.3	[psf]	Steady Level Flight @ TOGW	106.0	[psf]
Altitude	15000	[ft]		15000	[ft]
Mach #	0.7			0.7	
q	2.85	[psi]		2.85	[psi]
Load Factor, nz	5	[g's]	Limit Load Factor	5	[g's]
	7.5	[g's]	Ultimate (Design) Load Factor	7.5	[g's]

Table 2 Load Case Information for each Configuration

c) Design Variables

The design variables are setup and linked in an object-oriented manner. For example, the skin panels are linked in such a way that all panels along the chord at a given spanwise station are linked to a common design variable; namely, the thickness of the skin panels. Each of the spars are considered separately. For a given spar, the shear web thickness and spar cap areas appear as design variables. However, starting from the root, all elements within a 20% span-length distance are linked to a common design variable. This constitutes sequentially linking every two elements along the spar in this case. For example, the two most inboard shear webs associated with the trailing edge spar are linked to a common design variable describing the web thickness. The upper and lower spar caps associated with these webs are also linked to a common design variable describing the cross-sectional area of the caps. This idea is illustrated in Figure 18 where elements linked to a common design variable are rendered with a common color.

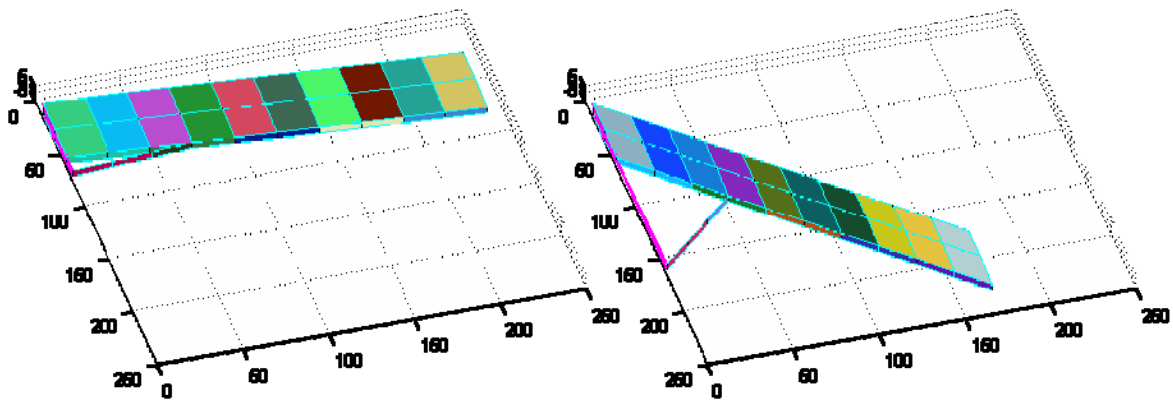


Figure 18. Structural Design Variable Linking of the Shear-Wing Concept: Halfspan Model of the Unswept Configuration (Left); Halfspan Model of the Swept Configuration (Right)

d) Design Constraints

The design constraint set for this wing model currently consists of two types: static strength constraints and maximum tip deflection constraints. All primary structural elements (shear webs, spar caps, and skin panels) are constrained to a VonMises strength constraint; buckling constraints have not yet been incorporated. Due to a runtime bug in ASTROS, a lift effectiveness constraint can not be directly imposed. As a current “work-around” to this situation, a set of wing-tip displacement constraints has been enforced. More specifically, a vertical displacement constraint is applied to the wing-tip node of the mid-spar such that the vertical deflection is less than or equal to 12% of the span length, using maximum operational wing-tip travel for the B-52 as a guideline.¹⁸

Furthermore, a secondary constraint is imposed such that the relative vertical displacement between the wing-tip nodes of the leading and trailing edge spars is less than or equal to a +/- 5 degree angle of attack relative to the horizontal. Ultimately, this constraint set acts as a surrogate for the aeroelastic lift effectiveness constraint.

11. Experimental Datum for Structural Sizing Strategy Comparison

This goal of this section is to develop an experimental datum for the morphing wing structural optimization strategies that will ensue. Each wing configuration will be structurally optimized as a fixed-geometry wing using ASTROS, and then using the alternate sizing architecture where iSIGHT handles the optimization task and ASTROS becomes the means to evaluate the objective function, constraint values, and sensitivity information. This will enable a comparison of primary load paths for the “optimal” mass distribution, provide a reference for subsequent results using the morphing wing structural optimization strategies, and validate that the iSIGHT sizing architecture produces similar or better optimal design results as when ASTROS is run as a standalone, fixed-wing, sizing venue.

e) Sizing Results in the Unswept Configuration

The results presented here are for the unswept configuration using ASTROS as a standalone optimization venue. Figure 19 illustrates the “optimal” shear web thickness distribution as well as the skin thickness distribution. Notice that the dominant load path is primarily through the forward spar with the mid-spar being loaded at the root. The skin thickness distribution decreases from maximum thickness at the root to a minimal value at the tip; an expected trend.

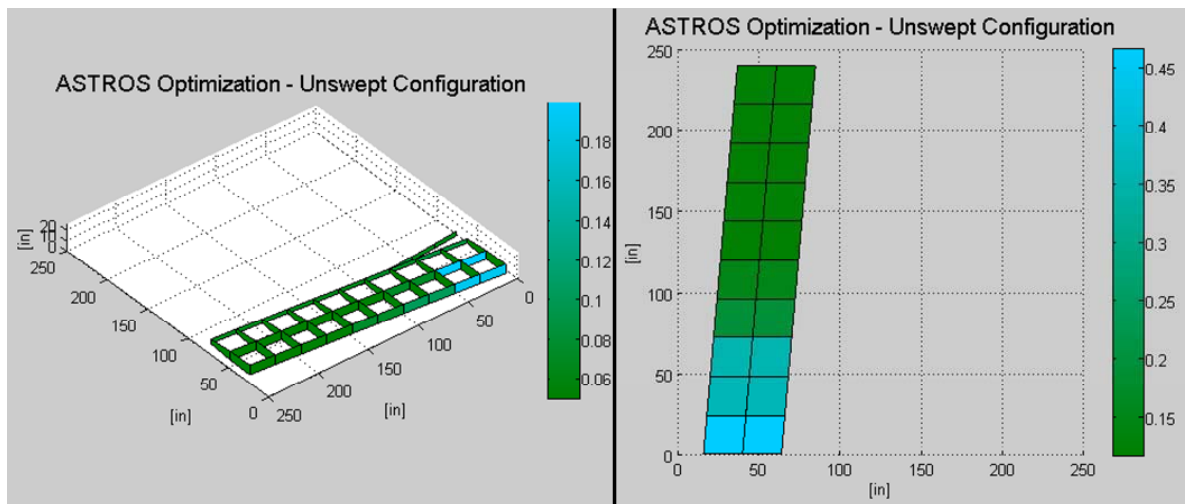


Figure 19. Structural Optimization Results for the Unswept Configuration Using ASTROS: Spar Web Thickness (Left); Upper Skin Thickness (Right)

Figure 20 proceeds to illustrate the resulting spar cap areas after optimization. As with the shear webs, a similar trend is seen regarding the material distribution. Notice that the brace strut has been heavily loaded in this configuration. Figure 20 also illustrates the vertical displacement of the structure at trimmed flight conditions with the optimal material distribution.

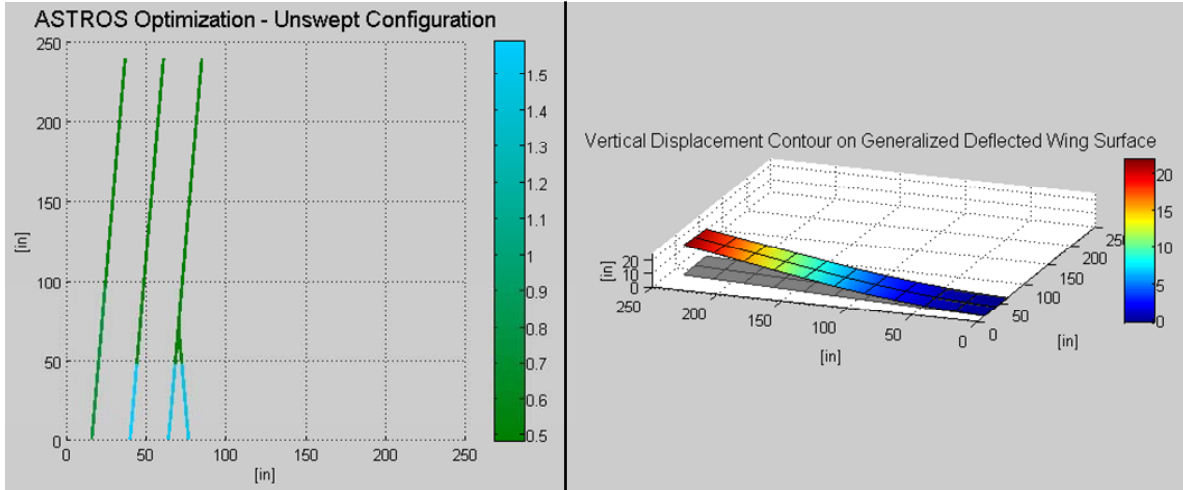


Figure 20. Structural Optimization Results for the Unswept Configuration Using ASTROS: Spar Cap Area (Left); Vertical Displacement Contour at Trim Conditions (Right)

Next, iSIGHT will be used as an optimization platform while ASTROS becomes a means to analyze the objective function value, constraint values, and sensitivity information; this sizing architecture will be referred to as the “iSIGHT optimization venue”. The left illustration of Figure 21 shows the relative percent change of iSIGHT venue $\{x^*\}$ results with respect to the ASTROS venue $\{x^*\}$ results. Notice that iSIGHT found a design variable set that substantially reduced the designed element dimensions in nearly all elements; five element dimensions were increased, but less than 25% in all cases. The formula used to calculate relative percent change as demonstrated in this plot is shown in the following equation:

$$\text{Rel. \% Change of D.V. } \{x_i\} = 100 \cdot \frac{\{x_i\}_{iSIGHT}^* - \{x_i\}_{ASTROS}^*}{\{x_i\}_{ASTROS}^*} \quad (1)$$

Also shown in Figure 21 is a “constraint profile” for the iSIGHT venue optimization results. Because there are 1019 constraints for this configuration, the constraint value data was plotted using a linear interpolation between consecutive data points to aid in visualization. In this case, the maximum constraint value was $-3.844689e-006$ so the design is feasible. However, several constraints appear to be active; in fact, 27 constraints have values greater than (-0.0001) while 31 have values greater than (-0.001) .

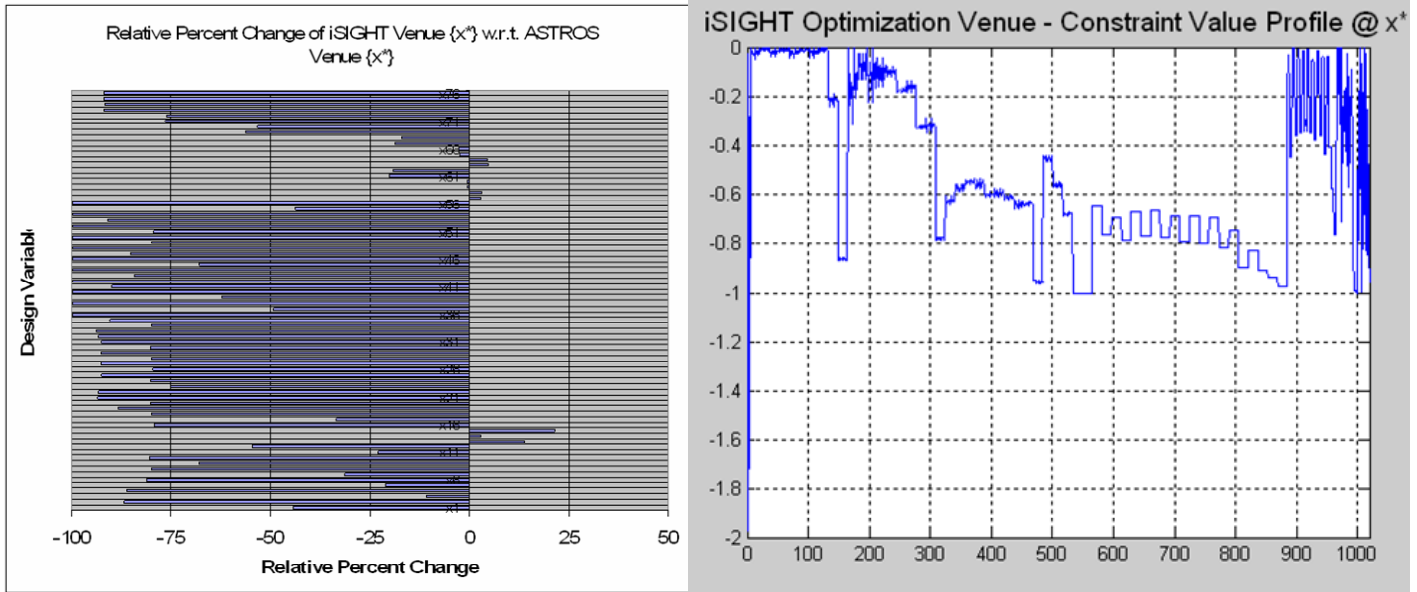


Figure 21. Comparison of Structural Optimization Results for the Unswept Configuration: Relative Percent Change of iSIGHT Venue $\{x^*\}$ w.r.t. ASTROS Venue $\{x^*\}$ (Left); Constraint Profile of iSIGHT Venue Results (Right)

Basic optimization performance metrics are evidenced in Table 3. Notice that the wing weight was substantially reduced when using the iSIGHT optimization architecture. Recall that iSIGHT is setup to use SQP for the initial search and then the method of feasible directions to bring any violated constraints back into the feasible active set; ASTROS uses a version of modified method of feasible directions and differences in results are attributed to the different optimization techniques. Also note that the runtime performance of ASTROS (1 minute) is substantially better than optimization using iSIGHT (78 minutes). This is attributed to the fact that for every function evaluation required by iSIGHT, design variable values must be written (parsed) into the associated bulk data file cards, ASTROS must be called, data files read, required analysis performed, and requested output files written; this is a very costly, but viable, procedure.

Unswept Configuration	ASTROS Venue	iSIGHT Venue	
Perform. Metric			Unit
Wing Wt. (Halfspan)	667.6	428.6	[lb]
Wing Wt. (Full Wing)	1335.2	857.2	[lb]
Run Time	1.0	78	[min]
Active Constraints	4	27	---

Table 3 Optimization Performance Metrics

Because the difference between the ASTROS and iSIGHT optimization tasks vary significantly in “optimal” designs, a further validation case was implemented to verify these results. The optimal solution from iSIGHT was given as the starting design point for an ASTROS venue optimization. Upon execution, ASTROS performed one design iteration and reported the initial design variable values to be an optimal condition; no step was made to a better design. Furthermore, ASTROS also reported no constraint violations at the given starting point. This case validates the setup and performance of the iSIGHT venue optimization procedure.

Because the iSIGHT optimization venue resulted in a substantially better design than ASTROS, the “optimal” material distribution plots will be revisited and used as a reference for subsequent studies. Notice in Figure 22 that the brace-strut has not been heavily loaded in this design; this is contrary to the ASTROS results. The skin thickness distribution is similar to the ASTROS case however.

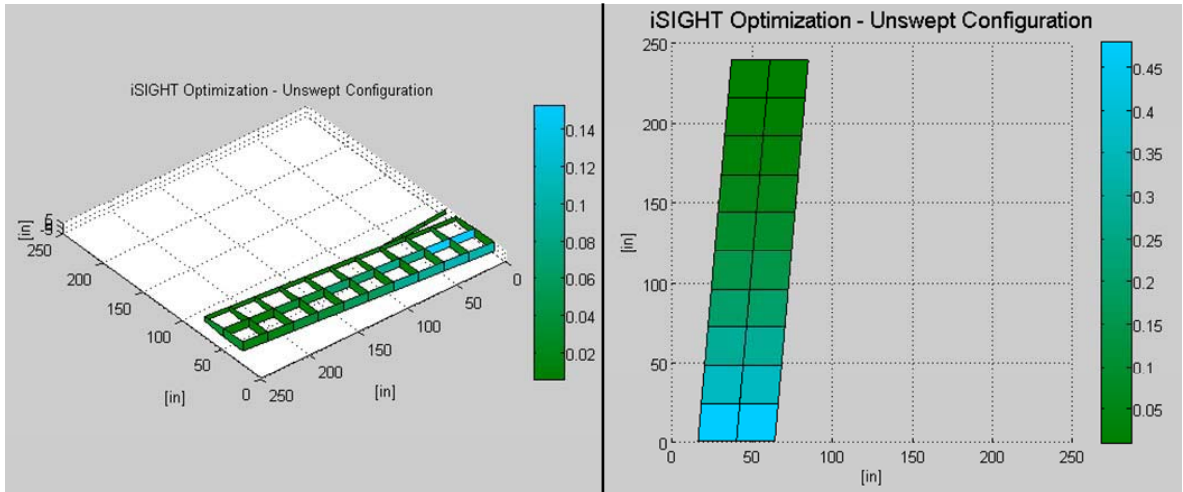


Figure 22. Structural Optimization Results for the Unswept Configuration Using iSIGHT: Spar Web Thickness (Left); Upper Skin Thickness (Right)

Figure 23 illustrates the “optimal” material distribution for the spar caps. The caps associated with the mid-spar have been designed with greater area at the root than the surrounding caps; this is also contrary to the ASTROS results.

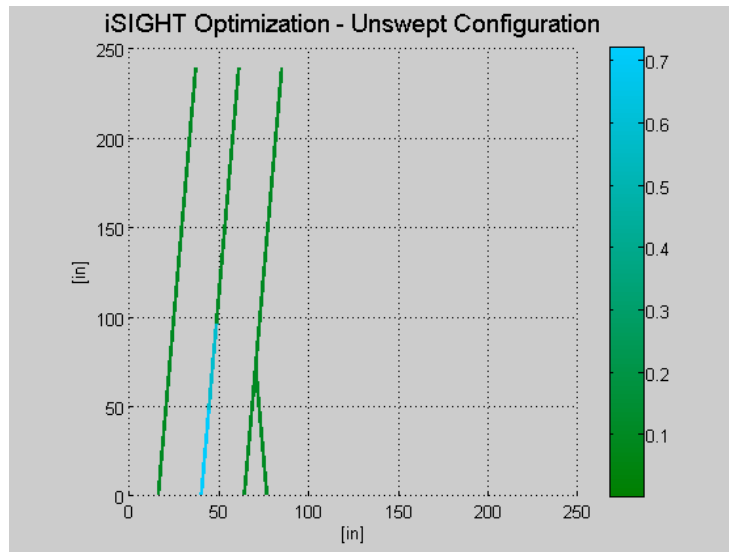


Figure 23. Structural Optimization Results for the Unswept Configuration Using iSIGHT: Optimal Spar Cap Areas

f) Sizing Results in the Swept Configuration

The same approach is now taken for the swept configuration study as was done for the unswept case. Table 4 shows the trends between an ASTROS venue optimization versus and iSIGHT approach. Again, the optimal solution reported by iSIGHT is of lesser weight (94.0 lb less) than that prescribed by the ASTROS run. The runtime is again significantly longer for iSIGHT, but more constraints are in the active set at the optimal conditions; 39 constraints are active with a constraint value greater than or equal to (-0.0001) while 39 are active at a value greater than (-0.001). Figure 24 illustrates further the relative differences in design variable values; the relative percentage difference between the iSIGHT and ASTROS venue optimization results are presented in the same manner as was done before. Also shown is the constraint profile for the swept configuration using the iSIGHT optimal design. Notice that the profile shape has changed significantly from the unswept configuration results and is attributed to the

variation in shape of the wing causing a reduced wing area, increased sweep, shifted aerodynamic load distribution, and different aeroelastic behavior.

	<u>ASTROS Venue</u>	<u>iSIGHT Venue</u>	
<u>Perform. Metric</u>			<u>Unit</u>
Wing Wt. (Halfspan)	599.9	505.9	[lb]
Wing Wt. (Full Wing)	1199.8	1011.8	[lb]
Run Time	1.3	56	[min]
Active Constraints	1	39	---

Table 4 Optimization Performance Metrics

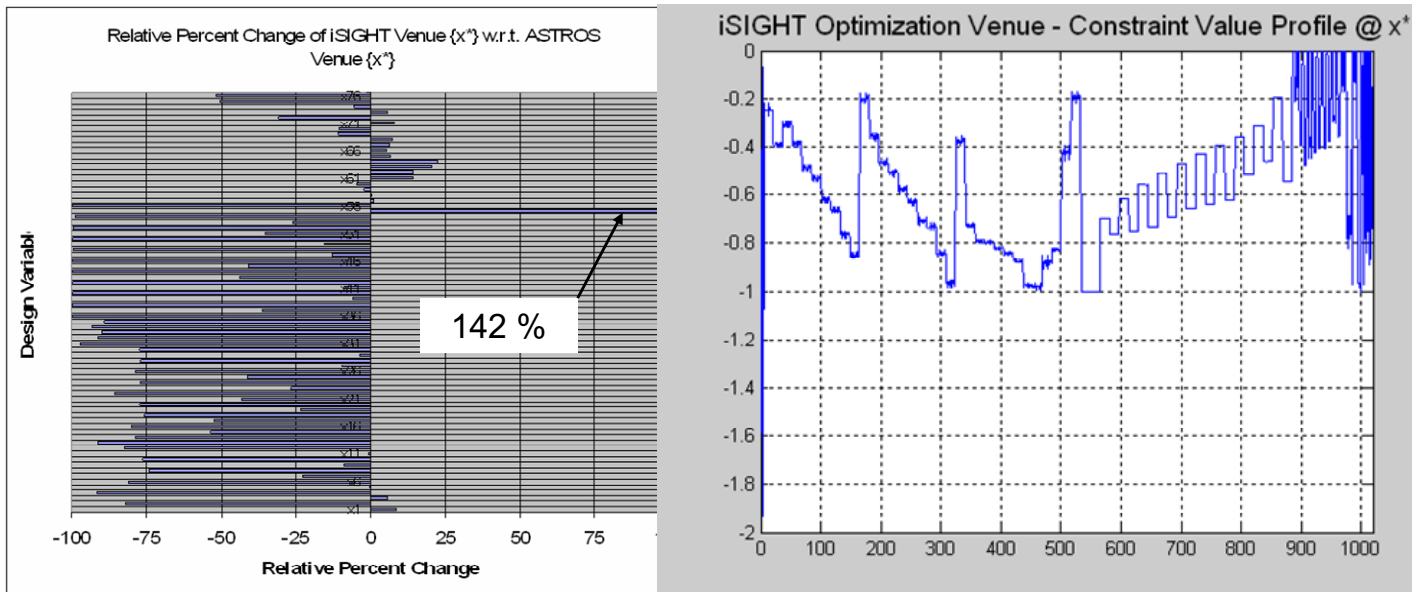


Figure 24. Comparison of Structural Optimization Results for the Swept Configuration: Relative Percent Change of iSIGHT Venue {x*} w.r.t. ASTROS Venue {x*} (Left); Constraint Profile of iSIGHT Venue Results (Right)

Instead of reporting first the ASTROS results and then the more optimal iSIGHT results as was done for the unswept configuration study, the iSIGHT results will only be considered here. Figure 25 illustrates the “optimal” shear web material distribution as well as the skin thickness distribution for the upper wing surface. While the material distribution of the skin shows the same trend as before, the shear webs have increased in mass. Notice that the dominant load path is still primarily through the forward spar, but the thickness of the mid-spar and aft spar have increased substantially near the wing root. Figure 26 shows the designed spar cap areas. Notice that all spar caps have been sized to ~0.1 [in], the lower bound value. This is most likely attributed to the fact that the skin is being designed to carry both the bending and shear loads.

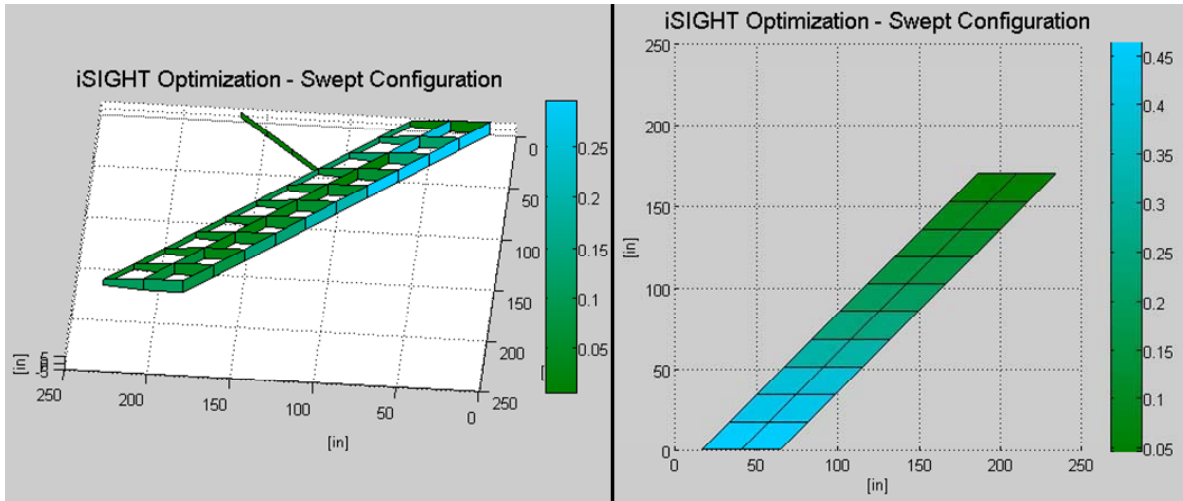


Figure 25. Structural Optimization Results for the Swept Configuration Using iSIGHT: Spar Web Thickness (Left); Upper Skin Thickness (Right)

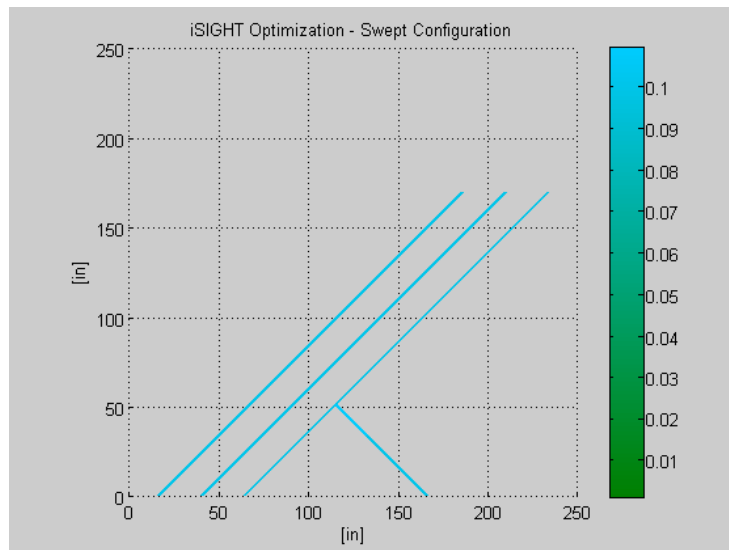


Figure 26. Structural Optimization Results for the Swept Configuration Using iSIGHT: Designed Spar Cap Areas

g) Summary of Individual Sizing Results

Table 5 summarizes the optimal wing weight results as obtained for the experimental datum case studies. The iSIGHT optimization venue found a better feasible design than the ASTROS venue, but at the expense of increased runtime. Notice that a third column appears in this table: “Wing Wt. Eqn.”. By considering these wings to be of fixed geometry, Raymer’s¹⁹ wing weight predictor for fighter/attack type aircraft was used (as well as relevant information from Table 1 and Table 2) to predict the flight-ready wing weight. In both cases, the weight of the unswept and swept wing configurations provide a weight less than the predicted flight-ready wing weight; this is desirable because it allows for the addition of non-optimal structural weight without compromising the conceptual-level weight prediction. For example, joints, fittings, and access panels have not been included in this intermediate complexity wing model and will have the impact of weight addition later in the design process. In fact, Shanley²⁰ indicates that non-optimal structural weight can increase the optimal weight prediction by 20 to 80 percent; this might be a fairly conservative estimate, though, taking into account the period of this publication (1960). The last column in Table 5 indicates how much weight growth is available to exactly match the predicted fielded wing

weight. Notice that the prediction for the unswept configuration might be unreasonable. Because typical fighter/attack aircraft have large sweep angles to increase aerodynamic efficiency at high speeds, the wing geometry / weight database used to develop the predictor is probably inaccurate for wings well outside the original scope of the geometric data. Therefore, the non-optimal weight growth estimate for the unswept configuration should be approached with some skepticism. However, if the wing weight equation for cargo / transport-type aircraft is used instead, the predicted flight-ready wing weight of the unswept configuration is 896.4 pounds allowing a growth margin of 39.2 pounds; this prediction seems more credible / applicable in this case.

	ASTROS Venue	iSIGHT Venue	Wing Wt. Eqn.	Growth
Configuration	Wing Wt [lb]	Wing Wt [lb]	Wing Wt [lb]	Non. Opt. Wt. [lb]
Config 1: Unswept	1335.2	857.2	1269.4	412.2
Config 2: Swept	1199.8	1011.8	1103.2	91.4

Table 5 Results Summary for the Experimental Datum Case Study

12. “Simultaneous Analysis” Strategy

The shear-wing concept will now be considered as a morphing wing system and the different optimization strategies will be demonstrated. The “simultaneous analysis approach will be considered first, and used as a baseline to compare the other optimization strategies. Recall that the simultaneous analysis strategy was explained in detail in section B.2. Here, each configuration will be analyzed at a given set of design variable values and constraint values / gradient information from each will be used to determine a viable search direction.

Figure 27 and Figure 28 illustrate the optimal mass distribution in the primary structural members. The trends are similar to those for the fully swept configuration sizing datum. Notice that the wing is shown in an intermediate configuration ($\Lambda_{LE} = 25$ [deg]) to reinforce the idea that the material distribution shown applies to all morphed states; structural mass will not be added or lost during morphing.

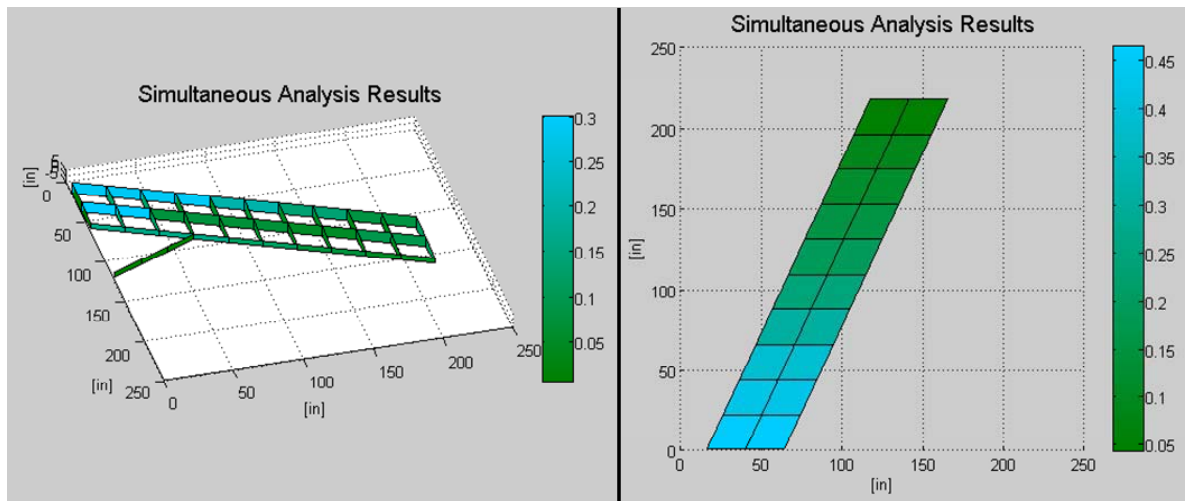


Figure 27. Structural Optimization Results Shown for an Intermediate Wing Configuration Using the Simultaneous Analysis Approach: Spar Web Thickness (Left); Upper Skin Thickness (Right)

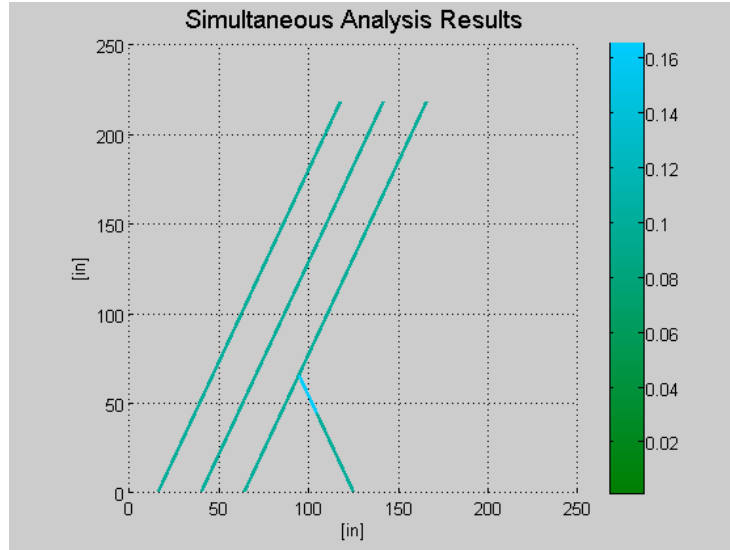


Figure 28. Structural Optimization Results Shown for an Intermediate Configuration Using the Simultaneous Analysis Approach: Designed Spar Cap Areas

To aid in the comparison of this structural design, Figure 29 illustrates the relative percent change of the simultaneous optimization design, $\{x^*\}_{Sim. Analysis}$, with respect to the iSIGHT-optimal fixed-wing sizing datums presented in Section II.0.11.e) and f). For both configurations, notice that the dominant trend was an increase in structural mass, especially relative to the fixed-geometry unswept design.

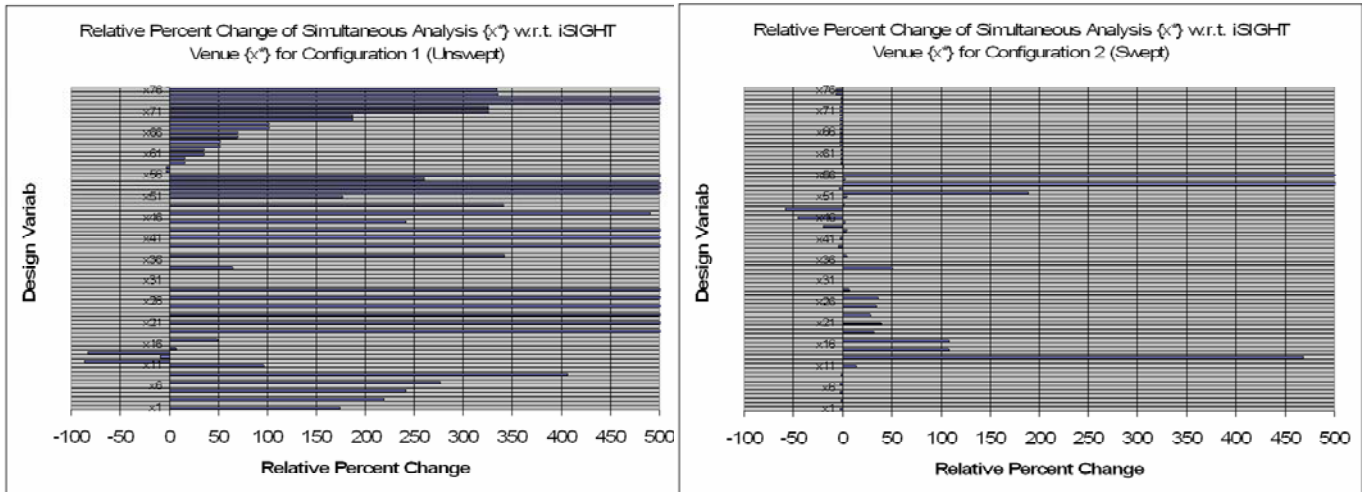


Figure 29. Relative Percent Change of Optimal Simultaneous Analysis Design Variable Values Relative to the Experimental Datum Cases: Unswept Configuration D.V. Values (Left); Swept Configuration (Right)

Figure 30 further illustrates data for the simultaneous analysis optimization approach. In this case, the constraint value profiles are shown for each configuration at the optimal structural design. Notice that the unswept configuration has 2 active constraints while the swept configuration has 31 (constraint considered active if $g(x) \geq -0.0001$). This is important because it suggests that the design is a compromise between the optimal load paths if the wing were sized separately in each configuration. This result further suggests that one configuration has not completely dominated the structural design. Although the swept configuration has 29 more active constraints than the unswept configuration, the 2 active constraints in the unswept configuration influence the structural design of the morphing wing as a system. Therefore, a sequential or aggregate optimization strategy will probably not produce a structural design as efficient as this one.

Table 6 quantifies various performance metrics for the simultaneous analysis optimization strategy. Notice that the wing weight of the full-span model is 1344.5 pounds which deviates from the fixed-wing unswept sizing result (857.2 lb) by 487.3 pounds, and from the fixed-wing swept sizing result (1011.8 lb) by 332.7 pounds. Furthermore, because the runtime for this strategy was approximately 2 hours, this approach seems to be a viable one.

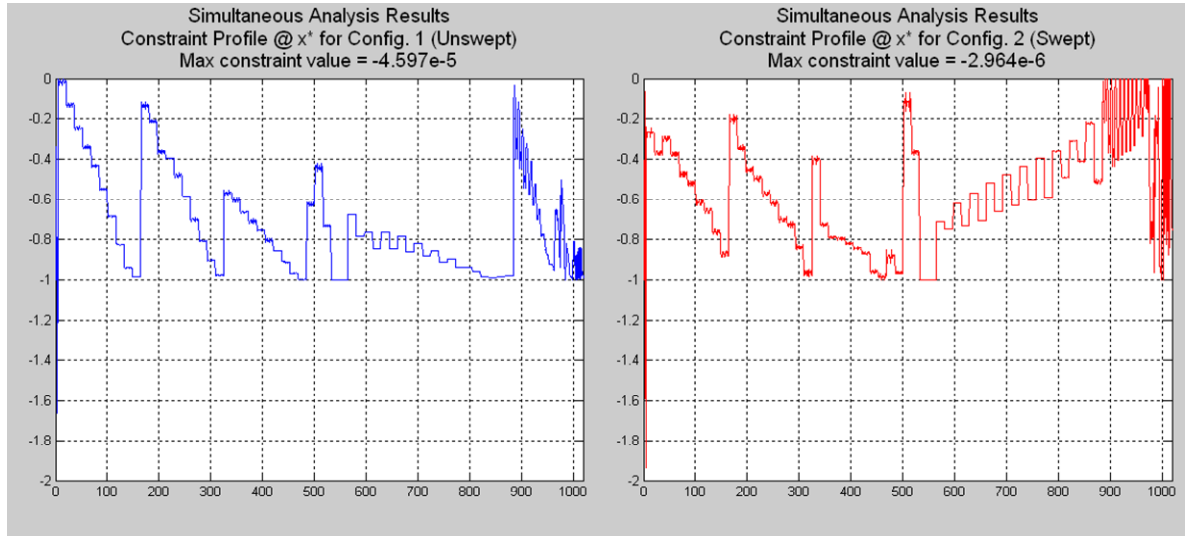


Figure 30. Constraint Profiles of Configuration 1 (Left) and Configuration 2 (Right) at $\{x\}^*$

Simultaneous Analysis	iSIGHT Venue	
Perform. Metric		Unit
Wing Wt. (Full Wing)	1344.5	[lb]
Run Time	131	[min]
Active Constraints	2	Unswept
	31	Swept

Table 6 Optimization Performance Metrics

13. “Sequential” Optimization Strategy

The sequential optimization strategy as described in Section II.B.3 has the advantage that the ASTROS optimization venue can be directly incorporated into the morphing wing design process without the need for other supporting software. Furthermore, this approach is also desirable because the ASTROS optimization runtime performance is substantially better than methods decoupling the optimization / FEA process. Because the “optimal” design produced by this strategy is potentially path dependent (i.e. the optimal design is a function of the order in which the configurations are sized), all possible sizing sequences should, ideally, be considered (# possible sizing sequences = (num configs.)!). In this case, for a wing with two primary configurations, the 2 possible sequences will be investigated and compared.

h) Sizing Results – Unswept Configuration Sized First

In this sequential optimization task, the unswept configuration is structurally sized, the resulting optimal design variable values set as new lower bound values, and then the swept configuration is sized with the new variable bounds. The design variable value set for the swept configuration, or equivalently the second sizing task, is considered to be the “optimal” design, $\{x\}^*$. Figure 31 illustrates the optimal design variable values for this sequential optimization task as a relative percent change from the simultaneous analysis optimum design. The dominant trend in this case is an increase in structural mass. This is further evidenced in Table 7 where the full-span

wing weight is reported as 1776.8 pounds; this corresponds to a 432.3 pound increase (32.2%) relative to the simultaneous analysis design.

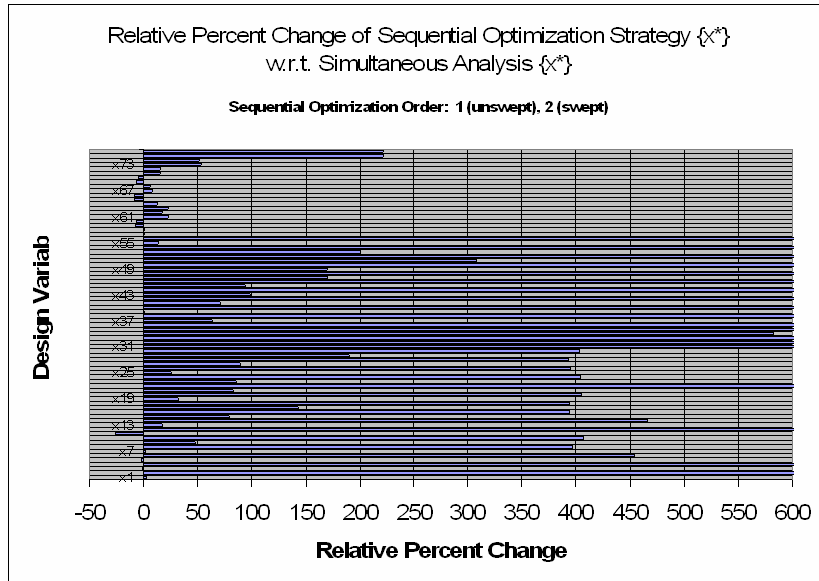


Figure 31. Relative Percent Change of Optimal Sequential Strategy (Order: 1,2) Design Variable Values Relative to the Optimal Simultaneous Analysis Values

Another important consideration is the design history of the active constraint set. Table 7 indicates that after the first structural sizing task in the unswept configuration, the structural design leads to 4 active constraints in the unswept configuration, while 70 are violated in the swept configuration. Upon executing the second structural sizing task in the swept configuration, all constraints are satisfied in the unswept configuration, while 4 constraints are active in the swept configuration. This suggests that the final structural design is non-optimal / over-designed / inefficient for the unswept configuration.

Sequential (Config 1 First)	ASTROS Venue		Unit
	After Config 1	After Config 2 (x*)	
Perform. Metric			
Wing Wt. (Full Wing)	1335.2	1776.8	[lb]
Run Time	1.0	2.2	[min]
Active Constraints	4	0	Unswept
	70 (violated)	4	Swept

Table 7 Optimization Performance Metrics

Figure 32 illustrates the constraint profiles for both configurations at the “optimal” structural design. This will be used as a reference to compare the subsequent sequential strategy in which the structural sizing order is reversed.

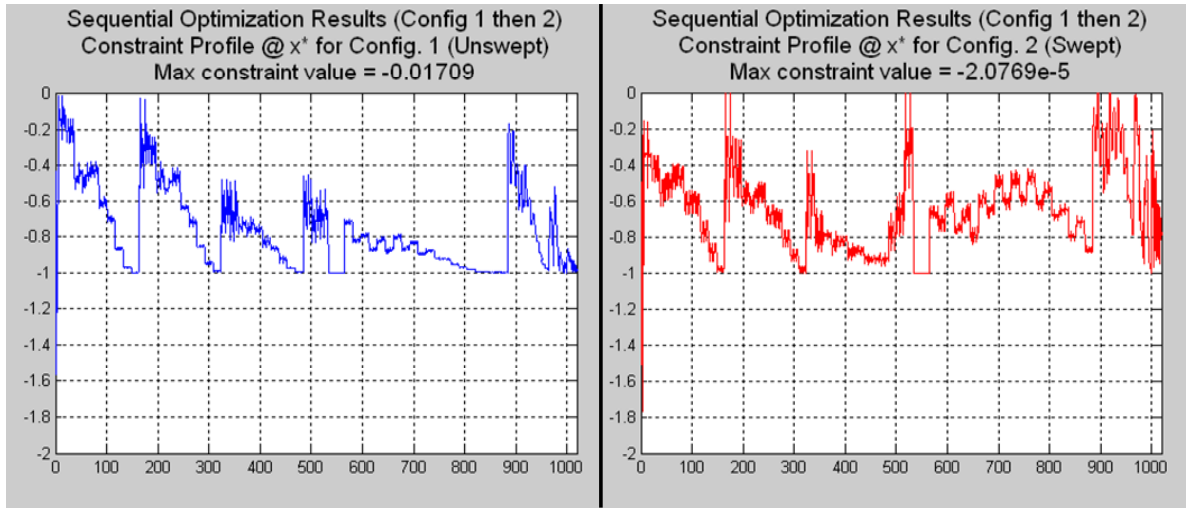


Figure 32. Constraint Profiles of Configuration 1 (Left) and Configuration 2 (Right) at $\{x\}^*$

i) Results – Swept Configuration Sized First

In this sequential optimization task, the swept configuration is structurally sized, the resulting optimal design variable values set as new lower bound values, and then the unswept configuration is sized with the new variable bounds. The design variable value set for the unswept configuration, or equivalently the second sizing task, is considered to be the “optimal” design, $\{x^*\}$. Figure 33 shows the optimal design variable values as a relative percent change from the sequential optimization strategy results; again, the dominant trend is an increase the structural mass. This is further evidenced in Table 8 where the full-span wing weight is reported as 1587.4 pounds; this corresponds to a 242.9 pound increase (18.1%) relative to the simultaneous analysis design.

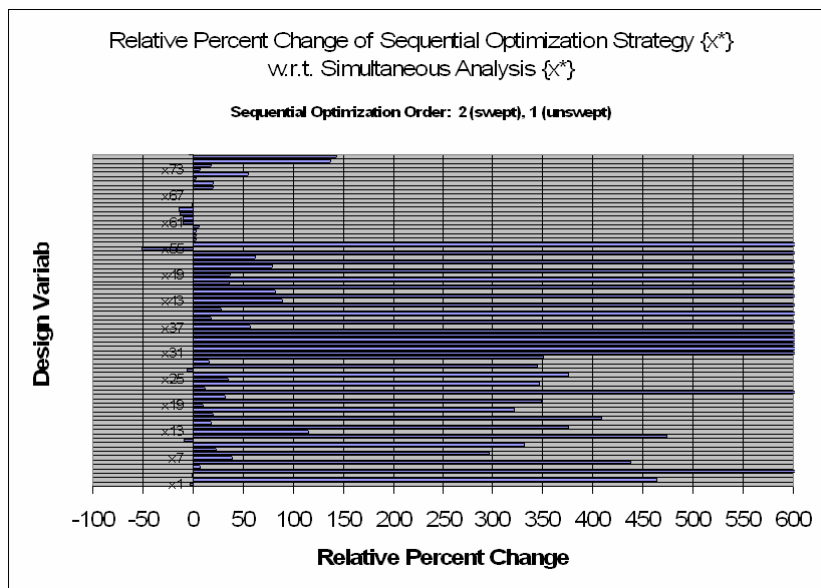


Figure 33. Relative Percent Change of Optimal Sequential Strategy (Order: 2,1) Design Variable Values Relative to the Optimal Simultaneous Analysis Values

As in the previous sequential optimization case (Order: 1,2), structural sizing of the first configuration (the swept state) led to active constraints in that configuration but violated ones in the unswept state. Upon structural sizing in the unswept configuration, the violated constraints were brought back into the active/feasible set, and the

originally active constraints in the swept configuration were pushed into the feasible domain past the active set boundary. This again suggests that the initial configuration is, to some extent, structurally inefficient. The resulting constraint profiles are shown in Figure 34 for both configurations at the sequential-optimal design variable values.

Sequential (Config 2 First)	ASTROS Venue		Unit
	After Config 2	After Config 1	
Perform. Metric			
Wing Wt.	1527.8	1587.4	[lb]
Run Time	1.3	1.2	[min]
Active Constraints	2 (violated)	2	Unswept
	1	0	Swept

Table 8 Optimization Performance Metrics

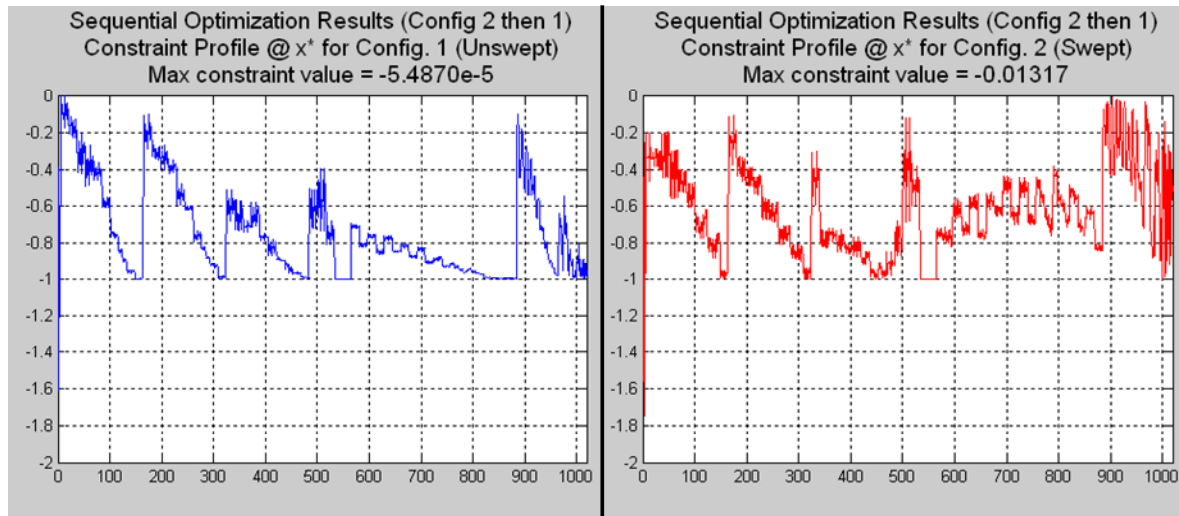


Figure 34. Constraint Profiles of Configuration 1 (Left) and Configuration 2 (Right) at $\{x\}^*$

14. “Aggregate” Optimization Strategy

The aggregate optimization strategy is the simplest to implement, yet the most questionable from a standpoint of structural design efficiency. The intent is to demonstrate this technique for completeness. Upon executing individual sizing tasks for each configuration, the maximum element dimension is taken from the collection of optimal design variable sets and integrated into the final “optimal” structural design. The resulting wing weight will be at least the maximum reported value from the individual optimizations, but will most likely weigh substantially more. In this case, the maximum reported weight for the individual configurations was 1011.8 pounds whereas the reported full-span wing weight of the “aggregate” structural design is 1657.4 pounds, as indicated in Table 9. Although this value is “desirable” because it falls between the predicted weight values using the sequential approach, the final design is infeasible. This is most likely attributed to the fact that simply integrating the elements with the largest mass values has an effect on the aeroelastic characteristics of the wing. More specifically, the variation in the structural stiffness from the “optimal” single-configuration designs might change the aeroelastic response at trim conditions with the aggregate-optimal design. This, in turn, could vary the aerodynamic load distribution on the wing enough to produce major load paths in the wing structure where originally there was not; if the structure is undersized in this region, the wing will not have adequate strength to carry the loads.

Aggregate Strategy	<u>ASTROS Venue</u>	
<u>Perform. Metric</u>		<u>Unit</u>
Wing Wt.	1657.4	[lb]
Run Time	2.3	[min]
Active Constraints	0	Unswept
	6 (All violated)	Swept

Table 9 Optimization Performance Metrics for the Aggregate Optimization Strategy

15. Summary of Results / Conclusions

Table 10 summarizes the resulting optimal morphing wing weight for the various structural optimization strategies investigated here. Notice that the simultaneous analysis approach produced the lightest weight structure. The sequential analysis procedures resulted in structural designs that were 32% and 18% greater than the simultaneous analysis approach. Furthermore, it is worth noting that the optimal design weight from this particular strategy is highly path dependent; the order in which the configurations are sized influences the optimal design weight. How this trend progresses for morphing wings with more than two primary configurations remains to be seen. The aggregate approach resulted in a wing weight 23% greater than the simultaneous analysis results but the resulting design was infeasible. This was likely to be attributed to the idea that integration of optimal designs changes the aeroelastic response of the wing relative to the original, individual configuration designs. This could also be problematic in the sequential approach, but was not evident in this case study. Ultimately, this author feels that the reduction of wing weight, or equivalently, the increased structural efficiency, enabled by the sequential optimization strategy outweighs the increased runtime relative to the other strategies, and should be considered as the primary structural design approach for morphing wing systems.

Results Summary	Wing Weight	% Difference	Feasibility
<u>Optimization Strategy</u>	<u>Full Wing [lb]</u>	<u>Rel. to Sim. Anl.</u>	
iSIGHT (config 1 - unswept)	857.2		Feasible
iSIGHT (config 2 - swept)	1011.8		Feasible
Simultaneous Analysis	1344.5	---	Feasible
Sequential (Order: 1, 2)	1776.8	32.15%	Feasible
Sequential (Order: 2, 1)	1587.4	18.07%	Feasible
"Aggregate" Approach	1657.4	23.27%	Infeasible

Table 10 Summary of Optimal Design Weights for the Various Structural Sizing Strategies

E. Suggested Future Efforts

While this work has demonstrated the performance of several structural optimization strategies for morphing wing systems, several design characteristics are unaccounted for. The following list represents the most influential characteristics that should be addressed in future studies:

- A surrogate lift effectiveness constraint was imposed here to incorporate some level of desired aeroelastic response. Ultimately, unsteady flutter constraints should be directly imposed.
- Morphing wing systems with reconfigurable substructures will likely have a wing skin material capable of extreme in-plane deformation, but having stiff out-of-plane characteristics to transfer aerodynamic loads. In this case study, the skin was assumed to have orthotropic properties consistent with aluminum 2024. Increased modeling fidelity in this capacity should be incorporated in future studies.
- Although discussed in previous work by this author, the underlying mechanisms that enable / correspond to the shape changing capabilities of a morphing wing – actuators, pivot joints, hinges,

friction – have not been modeled here. These were shown to increase wing weight by an average of 5 – 25 percent and should not be neglected when making conceptual-level weight predictions.^{4,5}

The demonstrated preprocessor used to develop intermediate complexity wing finite element models was primarily geared towards morphing wing concepts. However, treatment of the morphing wing system characteristics tends to apply and reduce naturally to other unconventional wing configurations. For example, a multiply-connected wing configuration like the joined-wing concept could be easily modeled as a collection of three wing surfaces and an appropriate structural bridge to join the sections. The template-based wing modeling approach further enables a generic architecture that facilitates user-level interaction with the preprocessor and precludes the need for “core source code” interaction. Because the program was developed in a Matlab-based environment, there is no need to recompile source code when modifications are made. However, more advanced skin-meshing techniques might be investigated, such as that presented in Reference 17, to increase FEA accuracy.

Appendix

A. User-Defined Folding Wing Template

```
% foldingwing_model.m

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% User Defined Parameters
sweep_LE_d = 40;      % [deg]
taper_ib   = 0.4;    % []
taper_ob   = 0.65;   % []
b          = 15*12;   % [in]
c_root     = 15*12;   % [in]
b_hinge_IB = 0.2;    % [x/b] Spanwise position for IB hinge
b_hinge_OB = 0.5;    % [x/b] Spanwise position for OB hinge as a frac. of the spanlength
t_c        = 12;     % [%] Thickness-to-chord ratio

USER_num_spars_sec_1 = 4;
USER_num_spars_sec_2 = 4;
USER_num_spars_sec_3 = 3;
USER_num_ribs_sec_1  = 3;
USER_num_ribs_sec_2  = 5;
USER_num_ribs_sec_3  = 9;

MP_fold_angle = 110; % [deg] Fold angle of intermediate wing section

WT_AIRCRAFT = 20000; % [lb] Aircraft weight during analysis

LE_AEROSTRUCT_OFFSET = 0.1;
TE_AEROSTRUCT_OFFSET = 0.2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Control Point Set

% Define Control Points for this Model (User-defined script)
PARAM_define_cp_set();

% Morph the control points using the foldingwing MORPH template (User-defined script)
MORPH_cp_set();

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Wing Structural Topology and Aerodynamic Geometry

% Build-Up the Reference Wing Geometry (Populate the SECTION and BRIDGE variables)
% Define Wing Sections
% {Str. Type, Morphing Mechanism Type, IB section connectivity, OB connectivity}
wing = SECTION_add(wing, 'BoxBeam', 'Fixed', 'WingRoot', 'Bridge');
wing = SECTION_add(wing, 'BoxBeam', 'Fixed', 'Bridge', 'Bridge');
wing = SECTION_add(wing, 'BoxBeam', 'Fixed', 'Bridge', 'WingTip');

% Define Structural Bridges
wing = SECTION_add_bridge(wing, 'Hinge', [1,2], [t_c/100]);
wing = SECTION_add_bridge(wing, 'Hinge', [2,3], [t_c/100]);

% Aerodynamic Surface Geometry
CP2_pos = wing.morph.cp_set.cp(2).pos;
CP3_pos = wing.morph.cp_set.cp(3).pos;

airfoil = sprintf('NACA00%i', t_c);

wing = SECTION_add_aero(wing, 1, airfoil, 7, [1,5,6,2], [0,0,0], 'GLOBAL');
wing = SECTION_add_aero(wing, 2, airfoil, 7, [2,6,7,3], CP2_pos, 'CP_DATA');
wing = SECTION_add_aero(wing, 3, airfoil, 7, [3,7,8,4], CP3_pos, 'CP_DATA');

% Populate the 'wing.coord' object with required information
wing = COORD_GetPosOrient(wing);
PLOT_wing_planform(wing);
```



```

axis equal; grid;

% Steady Aerodynamic Panel Model Information
wing = AERO_add_steady(wing, [1,2,3], [1,2,3], 'SUPPORTNODE', 1);
wing = AERO_add_steady_panel(wing, 1, [6, 15]);
wing = AERO_add_steady_panel(wing, 2, [9, 15]);
wing = AERO_add_steady_panel(wing, 3, [15, 15]);

% Structural Topology / Layout (Populate the STR_TOPOLOGY variable)
wing = SECTION_add_struct_top(wing, 1, 'Linear_Dist', [LE_AEROSTRUCT_OFFSET, (1-
TE_AEROSTRUCT_OFFSET), USER_num_spars_sec_1], 'Linear_Dist', [0.0, 1.0, USER_num_ribs_sec_1],
'WingBox');
wing = SECTION_add_struct_top(wing, 2, 'Linear_Dist', [LE_AEROSTRUCT_OFFSET, (1-
TE_AEROSTRUCT_OFFSET), USER_num_spars_sec_2], 'Linear_Dist', [0.0, 1.0, USER_num_ribs_sec_2],
'WingBox');
wing = SECTION_add_struct_top(wing, 3, 'Linear_Dist', [LE_AEROSTRUCT_OFFSET, (1-
TE_AEROSTRUCT_OFFSET), USER_num_spars_sec_3], 'Linear_Dist', [0.0, 1.0, USER_num_ribs_sec_3],
'WingBox');

% Structural Skin Panels (a superelement skin panel defined here will have a constant
thickness)
for(i_rib = 2:USER_num_ribs_sec_1);
    wing = SECTION_add_struct_skin(wing, 1, 1, [1, (i_rib - 1)], [USER_num_spars_sec_1,
i_rib]);
end
i_rib = [];
for(i_rib = 2:USER_num_ribs_sec_2);
    wing = SECTION_add_struct_skin(wing, 2, 1, [1, (i_rib - 1)], [USER_num_spars_sec_2,
i_rib]);
end
i_rib = [];
for(i_rib = 2:USER_num_ribs_sec_3);
    wing = SECTION_add_struct_skin(wing, 3, 1, [1, (i_rib - 1)], [USER_num_spars_sec_3,
i_rib]);
end

% Inboard wing support identification
[wing, ID_support_inertial] = SECTION_add_support(wing, 'Inertial', 1, [1, 1, 0.25, 3,
((WT_AIRCRAFT/2)/32.2)]); % Dividing AC wt. by 2 because only half the mass is represented in
the halspan wing model

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Generate structural geometry information

% Generate Structural Topology Information
PARAM_MODEL_topology();

FLAG.PLOT_spars = true;
FLAG.PLOT_ribs = true;
PLOT_wing_section_planform(wing, 1, FLAG);
PLOT_wing_section_planform(wing, 2, FLAG);
PLOT_wing_section_planform(wing, 3, FLAG);

PLOT_wing_planform(wing, FLAG);

```

B. User-Defined Scripts to Define and Morph the Control Point Set

```

% PARAM_define_cp_set.m
%
%-----
%
% PURPOSE: Populates 'wing_section' variable information
%
% CALLED BY:
%
%
```

```

% EXTERNAL CALLS:
%
% ENVIRONMENT:  Matlab 7.1 (R14) Service Pack 3
%
% AUTHOR:  Michael D. Skillen
%          Purdue University, 1282 Grissom Hall, W. Lafayette, IN 47907-1282
%
% DEVELOPMENT HISTORY:
%   DATE      INITIALS      DESCRIPTION
%   08/05/07      MDS          Created
%
script_ID = 'PARAM_define_cp_set';
ECHO_show_msg(script_ID, 'SCRIPT:  Initializing the kinematic Control Point data');

% CP1:  LE / wing root
wing = PARAM_add_cp(wing, 1, [0,0,0]);  % CP1

% CP2:  LE / IB hinge
y = b_hinge_IB*b;
x = y*tan(sweep_LE_d*pi/180);
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP2

% CP3:  LE / OB hinge
y = b_hinge_OB*b;
x = y*tan(sweep_LE_d*pi/180);
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP3

% CP4:  LE / wing tip
y = b;
x = y*tan(sweep_LE_d*pi/180);
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP4

% CP5:  TE / wing root
y = 0;
x = c_root;
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP5

% CP6:  TE / IB hinge
c_hinge_ob = taper_ib*c_root;
dc_dy = (c_hinge_ob - c_root)/(b_hinge_OB*b);
c_hinge_ib = c_root + dc_dy*(b_hinge_IB*b);
y = b_hinge_IB*b;
x = y*tan(sweep_LE_d*pi/180) + c_hinge_ib;
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP6

% CP7:  TE / OB hinge
y = b_hinge_OB*b;
x = y*tan(sweep_LE_d*pi/180) + c_root*taper_ib;
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP7

% CP8:  TE / wing tip
c_tip = taper_ob*(c_root + dc_dy*(b_hinge_OB*b));
y = b;
x = y*tan(sweep_LE_d*pi/180) + c_tip;
z = 0;
wing = PARAM_add_cp(wing, 1, [x,y,z]);  % CP8

```

```

% MORPH_cp_set.m

```

```

%-----
%
% PURPOSE:  Spatially rearranges the control point set based on the state

```

```

%           of the user-defined morphing parameters
%
% CALLED BY:  foldingwing_model()
%
% EXTERNAL REFERENCES:
%
% ENVIRONMENT:  Matlab 7.1 (R14) Service Pack 3
%
% AUTHOR:  Michael D. Skillen
%           Purdue University, 1282 Grissom Hall, W. Lafayette, IN 47907-1282
%
% DEVELOPMENT HISTORY:
%   DATE           INITIALS       DESCRIPTION
%   08/05/07      MDS             Created
%
cp_set = wing.morph.cp_set;

beta_d = MP_fold_angle;  % [deg]

L = cp_set.cp(3).pos(2) - cp_set.cp(2).pos(2);

% Update CP3
CP3_pos_old = cp_set.cp(3).pos;

CP3_pos_new(1) = CP3_pos_old(1);
CP3_pos_new(2) = cp_set.cp(2).pos(2) + L*cos(beta_d*pi/180);
CP3_pos_new(3) = L*sin(beta_d*pi/180);

cp_set.cp(3).pos = CP3_pos_new;

% Update CP7
CP7_pos_old = cp_set.cp(7).pos;

CP7_pos_new(1) = CP7_pos_old(1);
CP7_pos_new(2) = cp_set.cp(2).pos(2) + L*cos(beta_d*pi/180);
CP7_pos_new(3) = L*sin(beta_d*pi/180);

cp_set.cp(7).pos = CP7_pos_new;

% Update CP4
L_3_4 = wing.morph.cp_set.cp(4).pos(2) - wing.morph.cp_set.cp(3).pos(2);
CP4_pos_old = cp_set.cp(4).pos;

CP4_pos_new(1) = CP4_pos_old(1);
CP4_pos_new(2) = cp_set.cp(3).pos(2) + L_3_4;
CP4_pos_new(3) = cp_set.cp(3).pos(3);

cp_set.cp(4).pos = CP4_pos_new;

% Update CP8
CP8_pos_old = cp_set.cp(8).pos;

CP8_pos_new(1) = CP8_pos_old(1);
CP8_pos_new(2) = cp_set.cp(3).pos(2) + L_3_4;
CP8_pos_new(3) = cp_set.cp(3).pos(3);

cp_set.cp(8).pos = CP8_pos_new;

% Update the 'wing' structure object
wing.morph.cp_set = cp_set;

```

References

- ¹ Wall, R., "Darpa Eyes Materials for 'Morphing' Aircraft," *Aviation Week and Space Technology*, April 8, 2002.
- ² Skillen, M., and Crossley, W., "Modeling and Optimization for Morphing Wing Concept Generation," NASA Technical Reports Server (NTRS), [<http://ntrs.nasa.gov>], accessed Aug. 2007.
- ³ Skillen, M., and Crossley, W., "Developing Response Surface Based Wing Weight Equations for Conceptual Morphing Aircraft Sizing," AIAA-2005-1960, 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and 13th AIAA/ASME/AHS Adaptive Structures Conference, Austin, Texas, Apr. 18-21, 2005.
- ⁴ Skillen, M., "Developing Response Surface based Wing Weight Equations for Conceptual Morphing Aircraft Sizing," MS Thesis, School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN, Dec. 2005.
- ⁵ Skillen, M., and Crossley, W., "Developing Morphing Wing Weight Predictors with Emphasis on the Actuating Mechanism" AIAA-2006-2042, 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and 14th AIAA/ASME/AHS Adaptive Structures Conference, Newport, Rhode Island, May 1-4, 2006.
- ⁶ "Morphing Aircraft Structures," DARPA Defense Sciences Office, [<http://www.darpa.mil/dso/trans/mas.htm>], accessed Nov. 2006.
- ⁷ Love, M. H., Zink, P. S., Stroud, R. L., Bye, D. R., Chase, C., "Impact of Actuation Concepts on Morphing Aircraft Structures," AIAA 2004-1724, 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Apr. 2004.
- ⁸ Ashley, S., "Flying on Flexible Wings," *Scientific American*, November 2003, pp. 84-91.
- ⁹ Lawlow, M., "The Shape of Wings to Come," *SIGNAL* [online journal], Oct. 2006, URL: http://www.afcea.org/signal/articles/templates/SIGNAL_Article_Template.asp?articleid=1205&zoneid=56 [cited 31 October 2006].
- ¹⁰ *ASTROS Theoretical Manual for Version 20*. Torrance, CA, Universal Analytics, Inc. 1997.
- ¹¹ MSC Nastran, Software Package, MSC Software, Inc., Santa Ana, CA.
- ¹² Herendeed, D. L., Hoesly, R. L., Johnson, E. H., Venkayya, V.B., "ASTROS - An advanced software environment for automated design," AIAA-1986-856, 27th Structures, Structural Dynamics and Materials Conference, San Antonio, TX, May 19-21, 1986.
- ¹³ Neill, D. J., Johnson, E. H., Canfield, R., "ASTROS - A Multidisciplinary Automated Structural Design Tool," AIAA-1987-713, 28th Structures, Structural Dynamics and Materials Conference, Monterey, CA, Apr 6-8, 1987, Technical Papers. Part 1 (A87-33551 14-39). New York, American Institute of Aeronautics and Astronautics, 1987, p. 44-53.
- ¹⁴ MATLAB, Software Package, Ver. 7.1.0.246, Release 14, The MathWorks, Inc., Natick, MA, 2005.
- ¹⁵ iSIGHT, Software Package, Ver. 9.0.7, Engineous Software, Inc., Cary, NC, 2004.
- ¹⁶ Sensmeier, M.D., and Samareh, J.A., "Rapid Generation and Assessment of Aircraft Structural Topologies for Multidisciplinary Optimization and Weight Estimation," 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Austin, TX, April 18-21, 2005.
- ¹⁷ Sensmeier, M.D., Stewart, B.T., and Samareh, J.A., "Rapid Generation and Assessment of Aircraft Structural Topologies for Multidisciplinary Optimization and Weight Estimation," 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Newport, RI, May 1-4, 2006.
- ¹⁸ Course Notes: Weisshaar, T.A., "Aircraft Aeroelastic Design and Analysis," School of Aeronautics and Astronautics, Purdue University, 1995.
- ¹⁹ Raymer, D., *Aircraft Design: A Conceptual Approach*, AIAA Education Series, American Institute of Aeronautics and Astronautics, Reston VA, 3rd Edition, 1999.
- ²⁰ Shanley, F.R., *Weight-Strength Analysis of Aircraft Structures*, 2nd Ed., Dover Publications, New York, 1960.

REPORT DOCUMENTATION PAGE

*Form Approved
OMB No. 0704-0188*

The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.
PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) 01-02-2008		2. REPORT TYPE Contractor Report		3. DATES COVERED (From - To) 11-04-2006 to 10-31-2007	
4. TITLE AND SUBTITLE Modeling and Optimization for Morphing Wing Concept Generation II - Part I: Morphing Wing Modeling and Structural Sizing Techniques				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER NNL06AA04G	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) Skillen, Michael D.; and Crossley, William A.				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER 659877.02.07.05.9S51.02	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Langley Research Center Hampton, VA 23681-2199 Purdue University School of Aeronautics and Astronautics West Lafayette, IN 47907-2023				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001				10. SPONSOR/MONITOR'S ACRONYM(S) NASA	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S) NASA/CR-2008-214902	
12. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category 05 Availability: NASA CASI (301) 621-0390					
13. SUPPLEMENTARY NOTES Langley Technical Monitor: Jamshid A. Samareh An electronic version can be found at http://ntrs.nasa.gov					
14. ABSTRACT This report documents a series of investigations to develop an approach for structural sizing of various morphing wing concepts. For the purposes of this report, a morphing wing is one whose planform can make significant shape changes in flight - increasing wing area by 50% or more from the lowest possible area, changing sweep 30° or more, and / or increasing aspect ratio by as much as 200% from the lowest possible value. These significant changes in geometry mean that the underlying load-bearing structure changes geometry. While most finite element analysis packages provide some sort of structural optimization capability, these codes are not amenable to making significant changes in the stiffness matrix to reflect the large morphing wing planform changes. The investigations presented here use a finite element code capable of aeroelastic analysis in three different optimization approaches -a "simultaneous analysis" approach, a "sequential" approach, and an "aggregate" approach.					
15. SUBJECT TERMS Morphing; Aircraft design; Wing design; Structural optimization; Simultaneous analysis optimization; Sequential optimization; Aggregate optimization					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			STI Help Desk (email: help@sti.nasa.gov)
U	U	U	UU	45	19b. TELEPHONE NUMBER (Include area code) (301) 621-0390