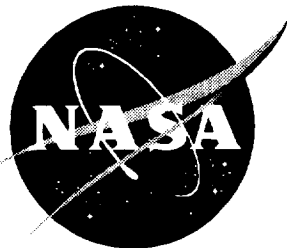


NASA/TM-2001-210867



HSCT4.0 Application

Software Requirements Specification

*A. O. Salas, J. L. Walsh, B. H. Mason, R. P. Weston, J. C. Townsend,
J. A. Samareh, and L. L. Green
Langley Research Center, Hampton, Virginia*

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

May 2001

The use of trademarks or names of manufacturers in this report is for accurate reporting and does not constitute an official endorsement, either expressed or implied, of such products or manufacturers by the National Aeronautics and Space Administration.

Available from:

NASA Center for AeroSpace Information (CASI)
7121 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

Foreword	1
1.0 Introduction	2
1.1 Purpose	2
1.2 Scope	2
1.3 Acronyms, Abbreviations, and Nomenclature	3
1.3.1 Acronyms and Abbreviations	3
1.3.2 Nomenclature	3
1.4 Overview	5
2.0 Overall Description	5
2.1 Product Perspective	5
2.2 Product Functions and Constraints	6
3.0 Specific Requirements	6
3.1 HSCT4.0 Application Input and Output	6
3.1.1 Optimization Data	6
3.1.2 Fixed Input Data	11
3.1.3 Generated Data	13
3.2 Functions	17
3.2.1 Optimization Process	21
3.2.2 Analysis Process	25
3.2.3 Geometry Process	26
3.2.4 Weights Process	28
3.2.5 Nonlinear Corrections Process	33
3.2.6 Rigid Trim Process	34
3.2.7 Polars Process	35
3.2.8 Performance Process	36
3.2.9 Ground Scrape Process	36
3.2.10 Displacements Process	37
3.2.11 Loads Convergence Process	38
3.2.12 Stress and Buckling Process	40
3.3 Data Storage Requirements	43
3.4 Performance Requirements	43
3.5 System Design Constraints	43
3.6 Software System Attributes	44
Appendix	45
4.0 References	48

List of Tables

Table 1. HSCT4.0 Design Variable Data Summary	7
Table 2. Shape Design Variables	8
Table 3. Structural Design Variables	9
Table 4. Geometry Constraints	10
Table 5. Weights Constraints	10
Table 6. Performance Constraints	11
Table 7. Structural Constraints	11
Table 8. HSCT4.0 Fixed Data Summary	12
Table 9. Load Condition Data	13
Table 10. HSCT4.0 Generated Data Summary	14
Table 11. Weight Types Assigned to Each Weight Mesh	29
Table 12. Structural Weights Hierarchy	30
Table 13. Systems, Payload, and Fuel Weights Hierarchy	31

List of Figures

Figure 1.	High-Speed Civil Transport.	50
Figure 2.	Baseline HSCT4.0 Model.....	51
Figure 3.	Optimization.....	52
Figure 4.	Gradient-Based Optimizer.....	53
Figure 5.	Analysis.....	54
Figure 6.	Geometry.....	55
Figure 7.	Weights.....	56
Figure 8.	Nonlinear Corrections.....	57
Figure 9.	Rigid Trim.....	58
Figure 10.	Polars.....	59
Figure 11.	Performance.....	59
Figure 12.	Ground Scrape.....	60
Figure 13.	Displacements.....	61
Figure 14.	Loads Convergence.....	62
Figure 15.	Stress and Buckling.....	63
Figure 16.	Shape Design Variables - Planform.....	64
Figure 17.	Planform Control Points.....	64
Figure 18.	Shape Design Variables - Thickness, Camber, Twist, and Shear.....	65
Figure 19.	Structural Design Variable Zones.....	66
Figure 20.	Composite Laminate Stacking Sequence.....	67
Figure 21.	Ply Orientations.....	67
Figure 22.	Miscellaneous Geometry.....	68
Figure 23.	Airfoil Interior Thickness Constraints in Airfoil Cross-section.....	68
Figure 24.	Weights Meshes.....	69

Foreword

During the 1990s, the High Performance Computing and Communications Program (HPCCP) sponsored a series of application projects that were based on the High Speed Civil Transport (HSCT). With each new HSCT project, the application was made more complex by increasing both the number and fidelity of the engineering analyses used. As the application complexity increased, it became clear to the software developers that a more formal approach to software development was needed in these projects.

This document presents the requirements for the HSCT4.0 application. The HSCT4.0 application project was the only project in the HSCT series that documented the application requirements. This was a much more difficult task than was anticipated by most project members, and not all requirements were clearly documented before design and implementation; however, all project members developed an appreciation for the benefits gained from documenting project requirements. The lessons learned from the HSCT4.0 requirements effort has formed an experience base that will be applied towards future HPCCP projects.

1.0 Introduction

This document specifies the software requirements for the High Performance Computing and Communication Program (HPCCP) High Speed Civil Transport (HSCT, fig. 1) application project referred to as HSCT4.0. Information about the HSCT4.0 computing environment can be found in references 1 and 2.

1.1 Purpose

This requirements document serves several purposes. First, it serves to establish an agreement between the suppliers (Multidisciplinary Optimization Branch, Computational Modeling and Simulation Branch, Analytical and Computational Methods Branch, and Computer Sciences Corporation) and the customer (HPCCP/Computational AeroSciences Team) as to what the HSCT4.0 application should do. Second, the document provides descriptions of the various functions (and the relationships among them) that make up the multidisciplinary application. Any constraints on the software design are included. The goal is to provide to the software developers the information necessary to design and implement the system. Finally, the document provides a baseline for verification and validation activities and for later enhancements.

1.2 Scope

The objective of the HPCCP HSCT4.0 application project is to demonstrate the application of high-performance computing techniques to the problem of multidisciplinary design optimization (MDO) of a supersonic transport configuration, using high-fidelity analysis simulations. The products of this project include a set of high-fidelity tools suitable for use in a conceptual to early preliminary design environment. The use of a realistic baseline model will have the added benefit that the results generated can be used to derive physically meaningful trends. The requirements set forth in this document are for an HSCT configuration. (A different HSCT configuration will require changes to the problem data defined later in this document.)

The project deliverable initially called for a software system capable of multidisciplinary analysis and optimization of a typical configuration using shape and structural sizing variables. The system was to provide the user a menu of options that include analysis, sensitivity analysis, and optimization for each contributing discipline as well as multidisciplinary analysis, sensitivity analysis and optimization. The goals of the project were too ambitious for the project's time period; therefore, this document primarily covers the requirements for the multidisciplinary analysis that are suitable for an optimization process; this multidisciplinary analysis is the main project deliverable. (The specific requirements can be found in section 3.0, in subsections 3.1.1, 3.2, 3.4, 3.5, and 3.6.)

The HSCT4.0 application will be implemented within a framework. Although this document primarily focuses on the analysis capabilities, a set of requirements related to user interaction with the framework were specified in early stages of the project. These requirements were later deemed beyond the scope of the project and therefore were removed from the software requirements.

1.3 Acronyms, Abbreviations, and Nomenclature

1.3.1 Acronyms and Abbreviations

AIT	Airfoil Interior Thickness
BC	Boundary Condition
BLF	Buckling Load Factor
CFD	Computational Fluid Dynamics
CORBA	Common Object Request Broker Architecture
FEM	Finite Element Model
FIDO	Framework for Interdisciplinary Design Optimization
GLW	Gross Landing Weight
GTOW	Gross Takeoff Weight
HPCCP	High Performance Computing and Communication Program
HSCT	High Speed Civil Transport
MDO	Multidisciplinary Design Optimization
OEW	Operational Empty Weight
SFI	Stress Failure Index
SLP	Sequential Linear Programming
c. g.	center of gravity

1.3.2 Nomenclature

1-direction	Fiber (0°) direction
2-direction	Transverse to fiber (90°) direction
A direction	Direction in which N_A acts
B direction	Direction in which N_B acts
C_L	Coefficient of lift
C_{Di}	Coefficient of drag due to lift
C_{Dv}	Coefficient of viscous drag
C_{Dw}	Coefficient of wave drag
C_m	Pitching moment coefficient
D_{ij}	Stiffness matrix terms (row i , column j)
F	Approximate objective function
F_0	Objective function
K_0	FEM stiffness matrix for unloaded shape

K_C	FEM stiffness matrix for cruise shape
N	Number of design variables
N_{11}	In-plane normal stress resultant in 1-direction
N_{22}	In-plane normal stress resultant in 2-direction
N_{12}	In-plane shear stress resultant
N_A	Largest compressive normal stress resultant
N_B	Smallest compressive normal stress resultant
N_{mn}	Bi-axial buckling load
N_{Shear}	Shear buckling load
PM_0	Ply mixture ratio for 0° plies
PM_{45}	Ply mixture ratio for 45° plies
PM_{90}	Ply mixture ratio for 90° plies
S	Allowable shear stress in the principal material system
T	Average thickness of upper and lower wing surfaces at a given wing station
X	Design variable vector
ΔX	Design variable increment
X_C	Allowable compressive stress in the 1-direction
X_T	Allowable tensile stress in the 1-direction
Y_C	Allowable compressive stress in the 2-direction
Y_T	Allowable tensile stress in the 2-direction
a	Length of the side of a finite element in the A direction
a_C	Aerodynamic cruise loads
a_M	Aerodynamic maneuver loads
b	Length of the side of a finite element in the B direction
f_C	Cruise loads (aerodynamic + inertia)
f_M	Maneuver loads (aerodynamic + inertia)
g	Approximate constraint vector
g_0	Constraint vector
h_{min}	Minimum AIT
r	Distance between upper and lower wing surfaces ($z^{Upper} - z^{Lower}$)
t_0	Thickness of 0° ply
t_{45}	Thickness of 45° ply
t_{90}	Thickness of 90° ply
t_{Core}	Thickness of core
t_{Face}	Thickness of composite face sheet ($t_0 + 2t_{45} + t_{90}$)

s_0	Aircraft unloaded shape (FEM node location vector)
s_C	Aircraft cruise shape (FEM node location vector)
s_M	Aircraft maneuver shape (FEM node location vector)
u_0	Cruise displacements using unloaded shape stiffness matrix (K_0)
u_C	Cruise displacements using cruise shape stiffness matrix (K_C)
u_M	Maneuver displacements using cruise shape stiffness matrix (K_C)
x, y, z	Principal coordinate axes (see fig. 2)
$\frac{\partial F}{\partial \bar{X}}$	Vector of objective function gradients
$\frac{\partial g}{\partial \bar{X}}$	Vector of constraint gradients
σ_{11}	In-plane stress in 1-direction
σ_{22}	In-plane stress in 2-direction
τ_{12}	In-plane shear stress

1.4 Overview

The remainder of the document is organized as follows. Section 2 describes general factors that affect the HSCT4.0 software requirements. Section 3 lists specific software requirements for the HSCT4.0 application.

2.0 Overall Description

2.1 Product Perspective

The HSCT4.0 application is a natural continuation of earlier applications referred to as HSCT2.1 and HSCT3.5. The previous applications have made use of increasingly refined versions of the Framework for Integrated Design Optimization (FIDO) system^{3, 4}. The major components of the FIDO system included the Parallel Virtual Machine based communications library, Data Manager, Graphical User Interface, Executive module, and discipline modules.

During earlier stages of the project, the iSIGHT^{5, 6, 7} software was considered as an alternative framework to FIDO for implementing the HSCT4.0 application. Because a number of difficulties were encountered when applying the iSIGHT 3.0 software to the HSCT2.1 problem, a decision was made to develop a follow-on to FIDO that would be based on Common Object Request Broker Architecture⁸ (CORBA) and Java. Specific requirements for the CORBA/Java framework, referred to as CJOpt, are not included in this document. See references 1 and 2 for information on CJOpt.

The HSCT4.0 application is a multidisciplinary system that is to be assembled from an existing set of legacy codes; therefore, the software development focuses on the integration (rather than internal details) of the legacy codes.

2.2 Product Functions and Constraints

The primary product of the HSCT4.0 application project is a multidisciplinary analysis that includes the following capabilities (described in section 3.2):

- geometric modeling and parameterization with consistent aerodynamic and structural models (see fig. 2)
- weights analysis
- aerodynamics analysis (linear and nonlinear)
- performance analysis
- ground scrape analysis
- structural analysis (displacements, stress, and buckling)
- aeroelastic loads analysis

The HSCT4.0 application will execute in a distributed environment. In order to reduce the computation time, portions of the application will be executed in parallel.

3.0 Specific Requirements

3.1 HSCT4.0 Application Input and Output

In this section, the HSCT4.0 application data are described by three categories. First, the optimization data (design variables and constraints) are described. Second, the initial, fixed data (data which does not change during the HSCT4.0 analysis) are described. Finally, the data generated from an HSCT4.0 analysis are summarized. The design variables, initial and generated data items are shown on the process diagrams, fig. 3 - fig. 15. These figures are described in detail in section 3.2.

3.1.1 Optimization Data

This section describes the design variables and constraints to be used in the HSCT4.0 application. The requirements for the optimization data are listed below. The requirements refer to the tables that follow in this section.

Requirement #1 - Twenty-seven shape design variables defined in table 2 shall be used to control the planform and airfoil shapes.

Requirement #2 - Two hundred forty-four structural design variables defined in table 3 shall be used to size the ply thicknesses and core thicknesses in 61 design variable zones.

Requirement #3 - The geometry constraints defined in table 4 shall be applied in the optimization.

Requirement #4 - The weights constraints defined in table 5 shall be applied in the optimization.

Requirement #5 - The performance constraints defined in table 6 shall be applied in the optimization.

Requirement #6 - The structural constraints defined in table 7 shall be applied in the optimization.

3.1.1.1 Design Variables

The design variables to be used in the HSCT4.0 optimization include shape variables (table 2) and structural variables (table 3). The HSCT4.0 application has 271 design variables for optimization – 244 structural variables and 27 shape variables.

The design variables are updated by the *Gradient-Based Optimizer* process (fig. 4). Table 1 below identifies the subprocesses that use the shape and structural design variables.

Table 1. HSCT4.0 Design Variable Data Summary

Data Description	Used By:
design variables (shape) - 27 shape variables	<i>Linear Aero Model Update, Miscellaneous Geometry Update, Nonlinear Aero Surface Model Update, FE Model Update</i>
design variables (structural) - 244 structural variables	<i>Section Property Update, Structural Geometry, Buckling</i>

Shape Design Variables. All analysis geometry models (i.e., aerodynamics and structures) are parameterized based on the locations of the shape design variables that are varied relative to the reference geometry. Table 2 lists all of the shape design variables. Note that the ranges provided are the initial estimates that can be used for optimization. The twenty-seven shape design variables consist of two sets. The first set contains the nine planform variables shown in fig. 16: the root chord C_r , the outer break chord C_2 , the tip chord C_3 , the semispan distance to the outer break B_2 , the leading edge sweep of the two outer wing panels SLE_2 and SLE_3 , the total projected area of the three wing panels A_r , and the fuselage nose and tail lengths L_n and L_r . Note that the root chord also sets the length of the center fuselage section, and that the wing semispan variable B_3 is dependent on other planform variables, including the total projected area.

Table 2. Shape Design Variables

Shape Design Variables (See Fig. 16)		
Name	Description (units)	Estimated Range
Wing Planform Variables (See Fig. 16)		
L_1	Tail length (distance from wing trailing edge to the first tailframe) (inches)	$\pm 5\%$ of baseline value
L_n	Fuselage nose length (inches)	$\pm 5\%$ of baseline value
C_r	Root chord (also panel 1 inboard chord) (inches)	$\pm 5\%$ of baseline value
B_2	Semispan distance to the outer break (panel 2) (inches)	$\pm 5\%$ of baseline value
C_2	Outer break chord (panel 2) (inches)	$\pm 5\%$ of baseline value
SLE_2	Leading-edge sweep (panel 2) (degrees)	$\pm 5\%$ of baseline value
A_1	total projected wing area (inches ²)	$\pm 5\%$ of baseline value
C_3	Tip chord (panel 3) (inches)	$\pm 5\%$ of baseline value
SLE_3	Leading-edge sweep (panel 3) (degrees)	$\pm 5\%$ of baseline value
Airfoil Shape Variables (See Fig. 18a)		
Thickness 1	(at 10% chord, 29.17% span) (inches)	± 10 inches
Thickness 2	(at 10% chord, 52.17% span) (inches)	± 10 inches
Thickness 3	(at 10% chord, 70.09% span) (inches)	± 10 inches
Thickness 4	(at 50% chord, 29.17% span) (inches)	± 10 inches
Thickness 5	(at 50% chord, 52.17% span) (inches)	± 10 inches
Thickness 6	(at 50% chord, 70.09% span) (inches)	± 10 inches
Camber 1	(at 10% chord, 29.17% span) (inches)	± 10 inches
Camber 2	(at 10% chord, 52.17% span) (inches)	± 10 inches
Camber 3	(at 10% chord, 70.09% span) (inches)	± 10 inches
Camber 4	(at 50% chord, 29.17% span) (inches)	± 10 inches
Camber 5	(at 50% chord, 52.17% span) (inches)	± 10 inches
Camber 6	(at 50% chord, 70.09% span) (inches)	± 10 inches
Airfoil Shape Variables (See Fig. 18b)		
Twist 1	(at 52.17% span) (degrees)	± 5 degrees
Twist 2	(at 93.40% span) (degrees)	± 5 degrees
Twist 3	(at 100% span) (degrees)	± 5 degrees
Shear 1	(at 52.17% span) (inches)	± 10 inches
Shear 2	(at 93.40% span) (inches)	± 10 inches
Shear 3	(at 100% span) (inches)	± 10 inches

The changes in all planform locations under the influence of the planform design variables are governed by a set of equations defining the locations of 22 control points (points 15 - 22 are used for the engines). The control point locations are shown in fig. 17. For convenience, all points are referenced to the aft end of the fuselage (point 0), because the fuselage end and the horizontal tail will not change during the optimization.

The second set of shape design variables (see fig. 18) consists of airfoil control shape points that define the wing thickness, camber, twist, and shear at a set of airfoil shape definition points. For the HSCT4.0 application, the definition points for thickness and camber are identical (fig. 18a), and the points for the wing twist line and the wing shear definition are identical (fig. 18b). The 18 airfoil shape variables are the vertical (z) perturbations of the camber, thickness, and shear from the wing baseline shape and the wing twist perturbation from the baseline shape in constant y planes. Note that the airfoil camber and thickness perturbations are smooth globally, while the twist and shear perturbations are linear between the line definition points.

Structural Design Variables. The structural design variables are associated with the ply thicknesses and core thicknesses of the laminated composite finite elements in the skins of the fuselage and the wing. Sixty-one design variable zones are sized by the design variables. As shown in fig. 19, 39 zones are defined on the fuselage, and 22 zones are defined on the wing. The elements in each zone are associated with a unique section property set. Each section property set is defined as a 9-layer laminated composite (four ply sheets surrounding a layer of core material) as shown in fig. 20. Four design variables are defined for each laminate. One design variable is the thickness of the 0° plies in each face sheet. Another design variable is the thickness of the 45° plies in each face sheet. The -45° plies have the same thickness as the 45° plies. The third design variable is the thickness of the 90° plies in each face sheet. The last design variable in each design variable zone is the thickness of the core. Local orientations for the composite laminate 0° ply direction on the wing and fuselage are shown in fig. 21. The same ply thickness limits (given in table 3) are used for all design variable zones.

Table 3. Structural Design Variables

Structural Design Variables per Zone (61 Zones)		
Description (units)	Minimum Thickness	Maximum Thickness
0° ply thickness (inches)	0.00740	0.1000
45° ply thickness (inches)	0.00740	0.1000
90° ply thickness (inches)	0.00370	0.1000
Core thickness (inches)	0.20000	5.0000

3.1.1.2 Constraints

The tables in this section describe the geometry, weights, performance, and structural constraints to be applied to the aircraft design.

Table 4. Geometry Constraints

Geometry Constraints			
# of type	Description (units)	Range	
		Lower Bound	Upper Bound
11	Fuel volume of tank (inches ³)	0	NA
30	Airfoil Interior Thickness (AIT) ratio	0	NA
61	0° ply-mixture ratio	0.1	NA
61	45° ply-mixture ratio	0.4	0.6
61	90° ply-mixture ratio	0.1	NA
1	Takeoff scrape lift (pounds)	Gross Takeoff Weight (GTOW)	NA
1	Landing scrape lift (pounds)	Gross Landing Weight (GLW)	NA

For details on the calculation of the fuel volume, AIT ratios and ply-mixture ratios, see section 3.2.3 “Geometry Process”. For details on the scrape lift constraints, see section 3.2.9 “Ground Scrape Process”.

Table 5. Weights Constraints

Weights Constraints			
# of type	Description (units)	Range	
		Lower Bound	Upper Bound
14	Mesh Weights (pounds)	0	NA
9	Fuel Tank Weights (pounds)	0	NA
1	Operational Empty Weight (OEW) (pounds)	0	NA

For details on the weights constraints, see section 3.2.4 “Weights Process”.

Table 6. Performance Constraints

Performance Constraints			
# of type	Description (units)	Range	
		Lower Bound	Upper Bound
1	Range (nautical miles)	5,000	NA
1	Balanced takeoff field length (feet)	0	10,000
1	Landing field length (feet)	0	10,000
1	Approach speed (knots)	0	155
1	Time to climb to cruise (hours)	0	1
1	Takeoff flyover noise (decibels)	NA	baseline flyover noise level
1	Takeoff sideline noise (decibels)	NA	baseline sideline noise level
1	Combined figure of merit noise (decibels)	NA	baseline figure of merit noise level

For more information on the performance constraints, see section 3.2.8 “Performance Process”.

Table 7. Structural Constraints

Structural Constraints			
# of type	Description (non-dimensional)	Range	
		Lower Bound	Upper Bound
15,820	Buckling Load Factor (BLF) for each element and load condition in all design variable zones	NA	1
15,820	Hoffman Stress Failure Index (SFI) for each element (maximum layer-wise index) and load condition in all design variable zones	NA	1

The number of each structural constraint type is the number of sized elements (2260) multiplied by the number of load conditions (7).

For details on the SFI and BLF calculations, see section 3.2.12 “Stress and Buckling Process”.

3.1.2 Fixed Input Data

Table 8 summarizes the fixed (would not change during a multidisciplinary optimization) input data required for the various HSCT4.0 analysis processes. Each entry in the table briefly summarizes the data item and identifies the subprocesses that require the data. (Note that a different HSCT configuration will require changes to the problem data defined in this document.)

Table 8. HSCT4.0 Fixed Data Summary

Data Description	Used By:
parameterized linear aero grids - linear aerodynamics grid resulting from the shape parameterization	<i>Linear Aero Model Update</i>
parameterized miscellaneous grids - wireframe surface grid resulting from the shape parameterization	<i>Miscellaneous Geometry Update</i>
parameterized nonlinear aero surface grids - nonlinear aerodynamics surface grid resulting from the shape parameterization	<i>Nonlinear Aero Surface Model Update</i>
parameterized FEM grid - set of finite element model (FEM) node locations resulting from the shape parameterization	<i>FE Model Update</i>
airfoil surface node data - paired upper and lower airfoil surface FEM node numbers and minimum distance between them	<i>Structural Geometry</i>
finite element data and BC's - material properties, nonsized section properties, element connectivities, rigid bars and elastic elements, boundary conditions (BC's)	<i>Theoretical FEM Weight, FEM Displacements, FEM Stress, Buckling</i>
reference weights data - as-built reference structural and nonstructural weights and weight distributions	<i>Built Up Weights</i>
Mach number, angle of attack, tail deflection table - sets of values used for the linear aerodynamics calculations; (different sets are used for the <i>Linear Aero</i> subprocesses called from <i>Nonlinear Corrections, Rigid Trim, Polars, and Ground Scrape</i>)	<i>Linear Aero</i>
aero reference data - reference span, mean aerodynamic chord, and wing area	<i>Linear Aero, Calculate C_L, Nonlinear Aero, Performance, Scrape Lift</i>
baseline nonlinear aero volume and surface grids - three distinct computational fluid dynamics (CFD) volume grids that are used for various load conditions and the surface grids that correspond to them	<i>Volume Grid Adjustment</i>
load condition data - See table 9	<i>Calculate C_L, Aero Moment Equilibrium, Induced Drag, Add Inertia Loads</i>
zero lift angle of attack - angle of attack at which no lift occurs for the configuration	<i>Wave Drag</i>
polars conditions - sets of C_L , Mach, and altitude values at which the drag calculations are evaluated	<i>Wave Drag, Viscous Drag, Induced Drag</i>
performance reference data - engine, mission, and performance analysis parameters	<i>Performance</i>
scrape angle data - specified roll angle, minimum allowed clearance distance, landing gear strokes	<i>Scrape Angle</i>
augment loads data - internal cabin pressure, taxi forces, factor of safety	<i>Augment Loads</i>

3.1.2.1 Load Condition Data

Table 9 shows the load condition data used in the HSCT4.0 application. These eight load conditions represent critical design load conditions along with the associated wing flap settings, mass case, load factor, Mach number, altitude, and equivalent air speed. Three flap settings are considered: (1) undeflected flaps set for flight at the cruise Mach number, (2) flaps set for flight at a high subsonic Mach number, and (3) flaps set for flight at a low supersonic Mach number. The masses are practical construction (i.e., realistic or “as-built”) weights that represent variations in payload and fuel ranging from the OEW to maximum GTOW. These load conditions are used to define the loads for structural analysis.

Table 9. Load Condition Data

Load Condition Number	VE	Mass Case	Load Factor (limit g's)	Mach Number	Altitude	Wing Flap Settings
1	Cruise	Cruise	Cruise	Cruise	Cruise	Cruise
2	V_C	GTOW	MIN	High Subsonic	A2	High Subsonic
3	V_D	GTOW	MAX	High Subsonic	A3	High Subsonic
4	+HAA	GTOW	MAX	High Subsonic	A4	High Subsonic
5	V_C	GTOW	MIN	Low Supersonic	A5	Low Supersonic
6	+HAA	GTOW	MAX	Low Supersonic	A6	Low Supersonic
7	-HAA	GTOW	MIN	Low Supersonic	A7	Low Supersonic
8	Taxi	GTOW	Taxi	-	-	-

V_C = cruise velocity; V_D = dive velocity; VE = equivalent air speed; +HAA = positive high angle of attack; -HAA = negative high angle of attack; A2 - A7 = load condition altitudes.

3.1.3 Generated Data

The following table summarizes the data generated by the HSCT4.0 analysis subprocesses. Each entry in the table briefly summarizes the data item and identifies the subprocess that generates the data and the subprocesses that require the data for input. Note that table entries are not included for data items shown on the optimization process figures (figs. 3 and 4). Also, note that the *Linear Aero* subprocess is used in several of the higher level *Analysis* processes (*Nonlinear Corrections*, *Rigid Trim*, *Polars*, and *Ground Scrape*). The data generated by and used by *Linear Aero* are determined by the process from which *Linear Aero* is invoked. If the data item is not applicable to *Linear Aero* in all situations, the relevant parent processes are identified in parentheses.

Table 10. HSCT4.0 Generated Data Summary

Data Description	Generated By:	Used By:
derived linear aero grids - grids (contained in the wavedrag decks) that provide discrete representations of the configuration surfaces in the form of coordinates at each discrete point	<i>Linear Aero Model Update</i>	<i>Apply Linear Delta Displacements, Linear Aero (Rigid Trim, Polars, Ground Scrape), Wave Drag, Viscous Drag</i>
derived miscellaneous model - data for miscellaneous processes	<i>Miscellaneous Geometry Update</i>	<i>Fuel Geometry, Performance Geometry, Scrape Geometry</i>
derived nonlinear aero surface grids - surface grids used to generate the derived nonlinear aerodynamics volume grids	<i>Nonlinear Aero Surface Model Update</i>	<i>Apply Nonlinear Delta Displacements</i>
derived FEM - finite element node locations which define the cruise shape of the aircraft	<i>FE Model Update</i>	<i>Structural Geometry, Theoretical FEM Weight, Built Up Weights, Loads Transfer, FEM Displacements, FEM Stress, Buckling</i>
derived section properties - sized element (composite) section properties	<i>Section Property Update</i>	<i>Theoretical FEM Weight, FEM Displacements, FEM Stress</i>
derived fuel tank geometry - coordinates of fuel tank reference points	<i>Fuel Geometry</i>	<i>Built Up Weights</i>
derived performance geometry - a coarse resolution, wire frame geometry description	<i>Performance Geometry</i>	<i>Performance</i>
derived scrape geometry - number of airframe test points, coordinates of landing-gear ground-touch points and of selected airframe test points	<i>Scrape Geometry</i>	<i>Scrape Angle</i>
ply mixture and airfoil interior thickness ratios - data used for constraints	<i>Structural Geometry</i>	<i>Calculate Objective and Constraints</i>
FEM weight - theoretical FEM nodal weights	<i>Theoretical FEM Weight</i>	<i>Built Up Weights</i>
fuel weights - weight of fuel in tanks (for available fuel volume and tank fill ratios)	<i>Built Up Weights</i>	<i>Performance</i>
cruise weight (total) - total weight at cruise; cruise weight with aft center of gravity (c. g.) limit (OEW + payload + cruise fuel weight)	<i>Built Up Weights</i>	<i>Calculate C_L, Calculate Nonlinear Corrections, Aero Moment Equilibrium</i>
cruise weight (nodal) - nodal weight at cruise	<i>Built Up Weights</i>	<i>Add Inertia Loads</i>

Table 10. (Continued) HSCT4.0 Generated Data Summary

Data Description	Generated By:	Used By:
cruise c. g. - axial location of c. g. at cruise	<i>Built Up Weights</i>	<i>Calculate Nonlinear Corrections, Aero Moment Equilibrium, Linear Aero (Polars)</i>
GTOW (total) - total weight at takeoff; maximum GTOW with forward c. g. limit (OEW + payload + GTOW fuel weight)	<i>Built Up Weights</i>	<i>Calculate C_L, Calculate Nonlinear Corrections, Aero Moment Equilibrium, Performance</i>
GTOW (nodal) - nodal weight at takeoff	<i>Built Up Weights</i>	<i>Add Inertia Loads</i>
GTOW c. g. - axial location of c. g. at takeoff	<i>Built Up Weights</i>	<i>Calculate Nonlinear Corrections, Aero Moment Equilibrium</i>
OEW (total) - the sum of structural, nonstructural, and systems weights	<i>Built Up Weights</i>	<i>Calculate Objective and Constraints</i>
displaced linear aero grid - derived linear aerodynamics grid modified by the delta displacements	<i>Apply Linear Delta Displacements</i>	<i>Linear Aero (Nonlinear Corrections, Rigid Trim)</i>
displaced nonlinear aero surface grid - derived nonlinear aerodynamics surface grid modified by the delta displacements	<i>Apply Nonlinear Delta Displacements</i>	<i>Volume Grid Adjustment</i>
displaced nonlinear aero volume grid - CFD volume grid based on the displaced nonlinear aerodynamics surface grids	<i>Volume Grid Adjustment</i>	<i>Nonlinear Aero</i>
target C_L - required lift coefficient based on total weight	<i>Calculate C_L</i>	<i>Nonlinear Aero</i>
linear pressures, forces, and moments - table of linear aerodynamics surface pressure coefficients and total force and moment coefficients for a set of angles of attack	<i>Linear Aero (Nonlinear Corrections)</i>	<i>Calculate Nonlinear Corrections</i>
nonlinear pressures, forces, and moments - nonlinear aerodynamics surface pressure coefficients and total force and moment coefficients	<i>Nonlinear Aero</i>	<i>Calculate Nonlinear Corrections</i>
nonlinear corrections - nodal surface pressure coefficient differences between the nonlinear aerodynamics and the matching-force linear aerodynamics pressures	<i>Calculate Nonlinear Corrections</i>	<i>Apply Nonlinear Corrections</i>
linear pressures - table of linear aerodynamics surface pressure coefficients for a set of angles of attack	<i>Linear Aero (Rigid Trim)</i>	<i>Apply Nonlinear Corrections</i>
corrected pressures - table of linear aerodynamics surface pressure coefficients, modified by nonlinear corrections, for a set of angles of attack	<i>Apply Nonlinear Corrections</i>	<i>Aero Moment Equilibrium</i>

Table 10. (Continued) HSCT4.0 Generated Data Summary

Data Description	Generated By:	Used By:
trimmed C_L, C_{Di} (cruise) - trimmed cruise target C_L and resulting C_{Di} from linear aerodynamics with non-linear corrections.	<i>Aero Moment Equilibrium</i>	<i>Induced Drag</i>
trimmed aero loads - trimmed aerodynamics panel forces generated for target C_L , resulting from linear aerodynamics with nonlinear corrections	<i>Aero Moment Equilibrium</i>	<i>Loads Transfer</i>
table of C_{Dw} - wave drag coefficients for a set of Mach numbers	<i>Wave Drag</i>	<i>Assemble Polars</i>
table of C_{Dv} - viscous drag coefficients for a set of Mach numbers and altitudes	<i>Viscous Drag</i>	<i>Assemble Polars</i>
table of C_L, C_{Di}, C_m - linear aerodynamics lift and drag force coefficients and pitching moment coefficient for a set of angles of attack and Mach numbers	<i>Linear Aero (Polars)</i>	<i>Induced Drag</i>
table of trimmed C_L, C_{Di} - linear aerodynamics lift and drag force coefficients at the trim angle of attack for a set of Mach numbers	<i>Induced Drag</i>	<i>Assemble Polars</i>
mission tables - C_{Di} , C_{Dv} , and C_{Dw} at the polars conditions	<i>Assemble Polars</i>	<i>Performance</i>
performance metrics - range, balanced takeoff field length, landing field length, takeoff speed, approach speed, landing speed, time to climb to cruise, takeoff noise (flyover, sideline, and combined)	<i>Performance</i>	<i>Calculate Objective and Constraints, Scrape Lift</i>
scrape angles of attack - computed angles of aircraft at scrape conditions on takeoff and landing	<i>Scrape Angle</i>	<i>Linear Aero (Ground Scrape)</i>
scrape C_L - computed lift coefficients at scrape conditions on takeoff and landing	<i>Linear Aero (Ground Scrape)</i>	<i>Scrape Lift</i>
ground scrape responses - takeoff scrape lift, landing scrape lift	<i>Scrape Lift</i>	<i>Calculate Objective and Constraints</i>
aero loads - aerodynamic forces distributed to FEM nodes	<i>Loads Transfer</i>	<i>Add Inertia Loads</i>
displacement loads - applied forces (combined aerodynamic and inertia) at FEM nodes	<i>Add Inertia Loads</i>	<i>FEM Displacements</i>
displacements - FEM nodal displacements	<i>FEM Displacements</i>	<i>Calculate Delta Displacements</i>
delta displacements - delta displacements at FEM nodes (displacements for maneuver conditions minus cruise displacements)	<i>Calculate Delta Displacements</i>	<i>Apply Linear Delta Displacements, Apply Nonlinear Delta Displacements</i>
converged loads (2-7) - converged displacement loads for load conditions 2-7	<i>Loads Convergence</i>	<i>Augment Loads</i>
augmented loads (2-8) - combination of converged loads and augment loads data for load conditions 2-8	<i>Augment Loads</i>	<i>FEM Stress</i>

Table 10. (Continued) HSCT4.0 Generated Data Summary

Data Description	Generated By:	Used By:
stress resultants - stress resultants on sized elements	FEM Stress	<i>Buckling</i>
buckling load factors - buckling load factors on sized elements	<i>Buckling</i>	<i>Calculate Objective and Constraints</i>
stress failure indices - maximum layer-wise stress failure indices on sized elements	FEM Stress	<i>Calculate Objective and Constraints</i>

3.2 Functions

The functional requirements for the HSCT4.0 application are defined in this section. In the subsections that follow, the application functions (or processes) are summarized with a series of process diagrams and diagram descriptions. These diagrams include fig. 3 - fig. 15. The diagrams show the basic information flow among processes included in the multidisciplinary analysis and optimization. A description is provided for each figure in the subsections to follow.

In the diagrams, circles are used to indicate processes (or functions) and arrows show the principal data that is passed between processes. A shaded circle refers to a process that may be further expanded into a set of subprocesses. An unshaded circle represents a lowest level subprocess (or function). A process outlined with dashes indicates a process that may or may not execute during an analysis. By convention, this document will use italics for process names.

The data are distinguished several ways in the diagrams. Data items enclosed within two solid lines indicate fixed data that do not change throughout the optimization. The data generated during each optimization cycle are labeled in two different notations on the diagrams. If the data item is generated by another process that is not shown on an individual diagram, that data item is enclosed within a shaded rectangular box. Otherwise, the data item is labeled beside the arrow exiting the process that generated it. For example, in the *Weights* process diagram (fig. 7), the “derived FEM” data item is computed by the *Geometry* process and therefore is enclosed within a shaded box. The *Analysis* process diagram (see fig. 5) has been simplified to show only enough data to indicate the sequencing among the highest level processes.

Note that fig. 3 gives an overview of the optimization process, where the two most significant processes are the *Analysis* and the *Sensitivity Analysis* processes. The details of the *Sensitivity Analysis* process are not covered in this document, because the final project deliverables did not include the optimization. This document gives a description of the optimization process, but the focus of the document is to describe the requirements for the *Analysis* process.

The following list summarizes the functional requirements for the HSCT4.0 application processes and subprocesses. Each requirement refers to the sections that describe that requirement in more detail. When appropriate, the existing engineering analysis code selected to perform a particular function is identified. The appendix provides descriptions for each of the codes selected, except for the optimization codes JOPT and CONMIN. If no codes are specified for a particular requirement, the code must be developed according to that requirement.

Requirement #7 - The HSCT4.0 optimization process (see section 3.2.1 “Optimization Process”) shall minimize GTOW subject to the constraints and design variable limits defined in section 3.1.1. This requirement has been deferred.

Requirement #8 - The *Calculate Objective and Constraints* subprocess (see section 3.2.1 “Optimization Process”) shall gather the responses from the analysis processes and calculate the objective function and constraints for the optimizer. This requirement has been deferred.

Requirement #9 - The *Accept Design and Set Move Limits* subprocess (see section 3.2.1 “Optimization Process”) shall determine the move limit factor for the design variables. The move limits will be determined manually based on the criteria for design acceptance and rejection. This requirement has been deferred.

Requirement #10 - The *Gradient-Based Optimizer* process (see section 3.2.1 “Optimization Process”) shall determine the values for the set of design variables that optimize the objective function, subject to the set of constraints. The JOPT and CONMIN codes shall be used for these calculations. This requirement has been deferred.

Requirement #11 - The *Linear Aero Model Update* subprocess (see section 3.2.3 “Geometry Process”) shall deform the linear aerodynamics grids to incorporate the changes in the shape design variables. The GP code will be used for this calculation.

Requirement #12 - The *Nonlinear Aero Surface Model Update* subprocess (see section 3.2.3 “Geometry Process”) shall deform the nonlinear aerodynamics grids to incorporate the changes in the shape design variables. The GP code will be used for this calculation.

Requirement #13 - The *Miscellaneous Geometry Update* subprocess (see section 3.2.3 “Geometry Process”) shall deform the miscellaneous geometry grid to incorporate the changes in the shape design variables. The GP code will be used for this calculation.

Requirement #14 - The *FE Model Update* subprocess (see section 3.2.3 “Geometry Process”) shall derive the FEM grid to incorporate the changes in the shape design variables. The GP code will be used for this calculation.

Requirement #15 - The *Scrape Geometry* subprocess (see section 3.2.3 “Geometry Process”) shall derive the scrape geometry.

Requirement #16 - The *Fuel Geometry* subprocess (see section 3.2.3 “Geometry Process”) shall derive the fuel tank volumes.

Requirement #17 - The *Performance Geometry* subprocess (see section 3.2.3 “Geometry Process”) shall derive the performance geometry.

Requirement #18 - The *Section Property Update* subprocess (see section 3.2.3 “Geometry Process”) shall derive the laminated composite shell property sets for the sixty-one design variable zones from the structural design variables.

Requirement #19 - The *Structural Geometry* subprocess (see section 3.2.3 “Geometry Process”) shall derive the ply-mixture ratios and AIT ratios.

Requirement #20 - The *Theoretical FEM Weight* subprocess (see section 3.2.4 “Weights Process”) shall compute the theoretical FEM weights at the nodes. The GENESIS® code (Vanderplaats Research & Development, Inc.) shall be used for this calculation.

Requirement #21 - The *Built Up Weights* subprocess (see section 3.2.4 “Weights Process”) shall compute the as-built total weight, nodal weights, and the c. g. for the cruise and GTOW mass cases.

Requirement #22 - The *Nonlinear Corrections* subprocess, *Apply Linear Delta Displacements* (see section 3.2.5 “Nonlinear Corrections Process”), shall deform the linear aerodynamics surface grid for a given load condition. The S2W code shall be used for this calculation.

Requirement #23 - The *Nonlinear Corrections* subprocess, *Linear Aero* (see section 3.2.5 “Nonlinear Corrections Process”), shall calculate linear aerodynamics characteristics for the current aircraft surface shape at a given load condition for specified flight conditions. The USSAERO code shall be used for this calculation.

Requirement #24 - The *Apply Nonlinear Delta Displacements* subprocess (see section 3.2.5 “Nonlinear Corrections Process”) shall deform the nonlinear aerodynamics surface grid for a given load condition. The CSCMDO code shall be used for this calculation.

Requirement #25 - The *Volume Grid Adjustment* subprocess (see section 3.2.5 “Nonlinear Corrections Process”) shall adjust the nonlinear aerodynamics volume grid for the new derived surface grid. The CSCMDO code shall be used for this calculation.

Requirement #26 - The *Calculate C_L* subprocess (see section 3.2.5 “Nonlinear Corrections Process”) shall calculate the aircraft lift coefficient, C_L , for a given aircraft weight and load condition.

Requirement #27 - The *Nonlinear Aero* subprocess (see section 3.2.5 “Nonlinear Corrections Process”) shall solve the Euler/Navier-Stokes equations for a given load condition on the derived nonlinear aerodynamics volume grid. The CFL3D code shall be used for this calculation.

Requirement #28 - The *Calculate Nonlinear Corrections* subprocess (see section 3.2.5 “Nonlinear Corrections Process”) shall calculate the nodal-pressure difference (on the linear aerodynamics grid) between the nonlinear and linear aerodynamics calculations. The L2NL code shall be used for this calculation.

Requirement #29 - The *Rigid Trim* subprocess, *Linear Aero* (see section 3.2.6 “Rigid Trim Process”), shall calculate linear aerodynamics characteristics for the current aircraft sur-

face shape at a given load condition for specified flight conditions. The USSAERO code shall be used for this calculation.

Requirement #30 - The *Apply Nonlinear Corrections* subprocess (see section 3.2.6 “Rigid Trim Process”) shall add the nonlinear correction nodal pressures to the linear aerodynamics nodal pressures, resulting in a corrected pressure distribution. The TRIM_NLC code shall be used for this calculation.

Requirement #31 - The *Aero Moment Equilibrium* subprocess (see section 3.2.6 “Rigid Trim Process”) shall calculate the angle of attack and tail deflection that trim the aircraft for a given load condition and weight. The induced drag coefficient and the loads are interpolated for these angles. The TRIM_NLC code shall be used for this calculation.

Requirement #32 - The *Wave Drag* subprocess (see section 3.2.7 “Polars Process”) shall calculate the supersonic wave drag contribution to the total configuration drag coefficient. The AWAVE code shall be used for this calculation.

Requirement #33 - The *Viscous Drag* subprocess (see section 3.2.7 “Polars Process”) shall calculate the surface skin friction drag contribution to the total configuration drag coefficient. The CDF1 code shall be used for this calculation.

Requirement #34 - The *Polars* subprocess, *Linear Aero* (see section 3.2.7 “Polars Process”), shall calculate linear aerodynamics characteristics for the aircraft surface shape at cruise for specified flight conditions. The USSAERO code shall be used for this calculation.

Requirement #35 - The *Induced Drag* subprocess (see section 3.2.7 “Polars Process”) shall calculate the lift-dependent drag contribution to the total configuration drag coefficient.

Requirement #36 - The *Assemble Polars* subprocess (see section 3.2.7 “Polars Process”) shall generate the drag polar mission tables.

Requirement #37 - The *Performance* subprocess (see section 3.2.8 “Performance Process”) shall perform a mission analysis. The FLOPS code shall be used for this calculation.

Requirement #38 - The *Scrape Angle* subprocess (see section 3.2.9 “Ground Scrape Process”) shall calculate the maximum pitch angle that can be attained for each specified clearance, stroke and roll angle.

Requirement #39 - The *Ground Scrape* subprocess, *Linear Aero* (see section 3.2.9 “Ground Scrape Process”), shall calculate linear aerodynamics characteristics for the current aircraft surface shape at takeoff and landing conditions. The USSAERO code shall be used for this calculation.

Requirement #40 - The *Scrape Lift* subprocess (see section 3.2.9 “Ground Scrape Process”) shall calculate the lift forces available at the maximum angles of attack for takeoff and landing conditions.

Requirement #41 - The *Loads Transfer* subprocess (see section 3.2.10 “Displacements Process”) shall transfer the z direction aerodynamic loads to the FEM using a method that conserves the total force and moments. The A2S code shall be used for this calculation.

Requirement #42 - The *Add Inertia Loads* subprocess (see section 3.2.10 “Displacements Process”) shall add inertia loads to aerodynamic loads for a given load condition. The A2S code shall be used for this calculation.

Requirement #43 - The *FEM Displacements* subprocess (see section 3.2.10 “Displacements Process”) shall calculate displacements for a given load condition. The GENESIS® code shall be used for this calculation.

Requirement #44 - The *Loads Convergence* subprocess, *Apply Linear Delta Displacements* (see section 3.2.11 “Loads Convergence Process”), shall add delta displacements at aerodynamic grid nodes to the derived linear aerodynamics grid to produce the displaced linear aerodynamics grid for the given load condition. The S2W code shall be used for this calculation.

Requirement #45 - The *Rigid Trim* process invoked from the *Loads Convergence* subprocess (see section 3.2.11 “Loads Convergence Process”) shall generate trimmed aerodynamics pressures for a given load condition. Refer to section 3.2.6 “Rigid Trim Process” for a description of the *Rigid Trim* process.

Requirement #46 - The *Displacements* process invoked from the *Loads Convergence* subprocess (see section 3.2.11 “Loads Convergence Process”) shall generate the FEM nodal displacements for a given load condition. Refer to section 3.2.10 “Displacements Process” for a description of the *Displacements* process.

Requirement #47 - The *Calculate Delta Displacements* subprocess (see section 3.2.11 “Loads Convergence Process”) shall subtract the FEM nodal displacements for the cruise condition from nodal displacements for a given load condition.

Requirement #48 - The *Augment Loads* subprocess (see section 3.2.12 “Stress and Buckling Process”) shall augment the converged loads for the *FEM Stress* subprocess.

Requirement #49 - The *FEM Stress* subprocess (see section 3.2.12 “Stress and Buckling Process”) shall perform a finite element analysis to compute SFIs and stress resultants for a set of augmented loads. The GENESIS® code shall be used for this calculation.

Requirement #50 - The *Buckling* subprocess (see section 3.2.12 “Stress and Buckling Process”) shall compute BLFs for FEM elements using stress resultants for a set of augmented loads.

3.2.1 Optimization Process

Although it is not required that optimization be implemented, it is described here to provide context for the HSCT4.0 analysis. The HSCT4.0 optimization problem is described as:

Minimize: GTOW

Subject to:

Side Constraints (upper and lower limits for design variables) (tables 2 and 3)

Geometry Constraints (table 4)

Weights Constraints (table 5)

Performance Constraints (table 6)

Structural Constraints (table 7)

The optimization procedure is shown in fig. 3. Refer to section 3.5 “System Design Constraints” for a discussion on the optimization method choice for the HSCT4.0 application. The *Gradient-Based Optimizer* process and the subprocesses, *Calculate Objective and Constraints* and *Accept Design and Set Move Limits*, are described in this section. The *Analysis* process is described in section 3.2.2. No requirements for the *Sensitivity Analysis* process have been defined for this project.

After an initialization process, the optimization proceeds using a Sequential Linear Programming (SLP) technique, where the outer loop shown in fig. 3 represents a “design cycle”. A design cycle uses the following processes: *Analysis*, *Calculate Objective and Constraints*, *Accept Design and Set Move Limits*, *Sensitivity Analysis*, and *Gradient-Based Optimizer*. The *Analysis* process produces the multidisciplinary responses, which are used by the *Calculate Objective and Constraints* subprocess to form the objective function and constraints. The *Accept Design and Set Move Limits* subprocess either accepts or rejects the current design and may adjust the design variable move limits; this subprocess requires manual intervention by the designer. The *Sensitivity Analysis* process evaluates the gradients of the objective function and constraints, which are used by the *Gradient-Based Optimizer* process to compute an approximate objective function and approximate constraints. Errors which may be introduced by use of the approximate analysis are controlled by imposing “move limits” on each design variable during the iteration process. A move limit, which is specified as a fractional change of each design variable value, is imposed as an upper and lower design variable bound.

The remainder of this section briefly describes the overall *Optimization* process. Section 3.2.1.1 describes each subprocess in more detail. The HSCT4.0 optimization procedure is conducted as follows. Prior to the first cycle, initial values for the design variables and move limits are set, and the analysis, objective function, and constraints are calculated. Then this design is reviewed for acceptance and possible move limit adjustments. (See criteria discussion in the next section for more detail on decision making.) Note that on cycle=0, even if this first design is infeasible, the design is “accepted” and the move limits are not adjusted.

If the design is accepted, the current (accepted) set of design variables along with the corresponding function and gradient data will be used in the next execution of the *Gradient-Based Optimizer* process. The *Accept Design and Set Move Limits* subprocess may or may not adjust the move limits. The cycle number is updated and the solution is checked for convergence. If the optimization process has converged, the HSCT4.0 application calculation is terminated. Otherwise, the cycle

continues with a gradient calculation. The gradients are used to form the approximate functions that are used in the *Gradient-Based Optimizer* process.

On the other hand, if the design is rejected, the cycle number is not updated, the move limits are decreased, and the most recently accepted design variables, along with the corresponding function and gradient data, are used as input to the *Gradient-Based Optimizer* process. This process is repeated until the optimization convergence criteria are met. The HSCT4.0 optimization procedure will be considered converged when the change in the objective function over three consecutive cycles is within a specified tolerance and all of the constraints are satisfied.

3.2.1.1 Optimization Subprocesses

The *Calculate Objective and Constraints* subprocess gathers responses from the *Analysis* process to calculate the objective function, F , and the constraint vector, g . The constraints described in section 3.1.1.2 are used to form g . The *Sensitivity Analysis* process computes the gradients of the objective function and constraints.

The *Accept Design & Set Move Limits* subprocess is an interactive subprocess in which the designer chooses whether to accept or reject the current set of design variable values and adjusts the move limit values if necessary according to the criteria described below. Note that the software must have a stop and restart capability to perform this function.

The criteria for design acceptance and rejection involves the designer examining the current objective function and constraints, the approximate objective function and constraints, and the previous cycle's objective function and constraints. Note that the term "current objective function and constraints" refers to the values obtained from the *Analysis* process based on the current set of design variables. The term "previous objective function and constraints" refers to the values obtained from the *Analysis* process based on the previous set of design variables. The term "approximate objective function and constraints" refers to the values obtained from the *Gradient-Based Optimizer* process based on the current set of design variables. The first priority for acceptance will be improved constraint satisfaction followed by reduction in objective function. The objective function and constraint approximation values (equations 1 and 2 below) are compared to the actual values and the move limits are adjusted according to the criteria for design acceptance and rejection outlined below. (Recall that the objective function will be minimized.)

Assuming all constraints are satisfied at the beginning of the current cycle, then at the end of the current cycle:

1. If all current constraints are satisfied and the current objective function decreases, then the current set of design variables is accepted, and the move limits are not adjusted.
2. If all current constraints are satisfied and the current objective function increases, then the current set of design variables is rejected, move limits are decreased, and the cycle is repeated with the previous set of design variables.

3. If there are current constraint violations and the current objective function increases, then the current set of design variables is rejected, move limits are decreased, and the cycle is repeated with the previous set of design variables.
4. If there are small current constraint violations and the current objective function decreases, then the violated constraints need to be examined in greater depth. If it is felt that it would be easy for the optimizer to satisfy the violated constraints, then the current set of design variables is accepted. The approximate objective function and approximate constraints are also examined to see if the move limits need to be reduced.

Assuming some constraints are violated at the beginning of the current cycle, then at the end of the current cycle:

1. If all current constraints are satisfied and the current objective function decreases, then the current set of design variables is accepted, and move limits are not adjusted.
2. If all current constraints are satisfied and the current objective function increases, then the current set of design variables is accepted, and move limits are not adjusted.
3. If there are current constraint violations and the current objective function increases, then if the largest current constraint violations are less than at the beginning of the cycle, the current set of design variables is accepted (even if the number of violated constraints increases), and move limits are not adjusted.
4. If the current constraint violations are greater and more numerous than at the beginning of the cycle, then the current set of design variables is rejected, the move limits are decreased, and the cycle is repeated with the previous set of design variables.

Note that in the above criteria, in cases where move limits are not adjusted by the designer, the designer may make a judgement call based on his previous optimization experience to decrease the move limits. This is usually the case when the designer feels that the approximate constraints and objective function are no longer “good” approximations of the current constraints and objective function.

The *Gradient-Based Optimizer* process, expanded in fig. 4, consists of a general purpose optimization program and an approximate analysis, which is used to reduce the number of full analyses during the optimization procedure. The approximate analysis is used to approximate the objective function and constraints with linear Taylor Series expansions using gradients of the objective function and constraints (computed in the *Sensitivity Analysis* process in each design cycle). The approximate objective function value and approximate constraint values are described by the following two equations, respectively.

$$(1) \quad F = F_0 + \sum_{i=1}^N \left. \frac{\partial F}{\partial X_i} \right|_0 \Delta X_i$$

$$(2) \quad g = g_0 + \sum_{i=1}^N \left. \frac{\partial g}{\partial X_i} \right|_0 \Delta X_i$$

where F_0 is the objective function (computed by the *Analysis* process), F is the approximate objective function, g_0 is the constraint vector (computed by the *Analysis* process), g is the approximate constraint vector, X is the design variable vector, ΔX is the design variable increment (controlled by the move limits), N is the number of design variables, $\frac{\partial F}{\partial X}$ is the vector of objective function gradients, and $\frac{\partial g}{\partial X}$ is the vector of constraint gradients. The assumption of linearity will not introduce a large error into the analysis, provided the changes ΔX are small. The loop shown in fig. 4 represents an “iteration”.

3.2.2 Analysis Process

A high level view of the procedure required for HSCT4.0 multidisciplinary analysis is shown in fig. 5. Descriptions of the *Analysis* processes appear in the sections to follow. For the HSCT4.0 analysis, there are eight load conditions (see table 9), but these are applied to only three distinct geometry models. The models are distinguished by flap settings: cruise, high subsonic, and low supersonic. These flap settings have been modeled for linear and nonlinear aerodynamics analyses; however, the FEM analysis uses only the cruise flap settings for all load conditions.

The primary engineering disciplines used in the HSCT4.0 analysis include geometry, weights, aerodynamics, structures, performance, and ground scrape. All discipline models are derived from a single CAD model. Weight is calculated using a combination of reference as-built weights and theoretical FEM weights. The aeroelastic calculation uses a linear structural analysis coupled with a variable-fidelity aerodynamics analysis. The variable-fidelity aerodynamics analysis uses a combination of linear aerodynamics with corrections based on nonlinear aerodynamics. Linear aerodynamics, without nonlinear corrections, are also used in both the performance and ground scrape disciplines. The performance computation uses the linear aerodynamics calculations in addition to empirical data to perform a mission analysis. The ground scrape computations are a simple algebraic computation based on the linear aerodynamics in ground effect.

The *Analysis* process begins with the *Geometry* process, which derives the updated geometries, grids, and section properties from the baseline geometries and grids using the design variable values. Once the *Geometry* process has completed, the *Weights* process may proceed.

The *Weights* process computes the theoretical FEM weight, the total as-built weights, the nodal weights, and the c. g. locations for the cruise and takeoff mass cases. These data are needed before the *Nonlinear Corrections* and *Rigid Trim* processes can be executed.

The *Nonlinear Corrections* process is the first of two stages in what is called a variable-fidelity aerodynamics analysis approach. For efficiency during the *Analysis* process, this approach uses only one computationally intensive, nonlinear CFD calculation per load condition. A nonlinear correction is then calculated relative to an appropriate linear aerodynamics calculation. In the second stage of the approach, this correction will be applied each time a linear aerodynamics calculation

tion is made during the subsequent *Rigid Trim* and *Loads Convergence* processes. Because of the large time for the nonlinear CFD computation, the *Nonlinear Corrections* process will not be executed in every *Analysis* cycle after the first cycle. (This is indicated by the dashed lines in fig. 5.)

Using the most recent nonlinear corrections, the *Rigid Trim* process calculates trim conditions for the cruise condition that will be used as reference conditions for subsequent calculations. The trim conditions are the angle of attack and tail deflection that provide lift equal to weight and no net pitching moment. (Note that the *Rigid Trim* process is used again for other load conditions in the *Loads Convergence* process.) Once the *Rigid Trim* process has completed, the *Analysis* process divides into two branches, which may be computed in parallel.

In the left branch of the *Analysis* process, the *Polars* process is followed by the *Performance* and *Ground Scrape* processes. The *Polars* process uses the cruise shape for all of its aerodynamic calculations. The cruise results from *Rigid Trim* are augmented by a set of linear aerodynamics drag polars calculated for a set of Mach numbers and lift coefficients. The *Polars* process provides input for the range calculation in the *Performance* process. The *Performance* process also calculates the takeoff and landing speeds, in addition to other metrics. These speeds are used in the *Ground Scrape* process to calculate the takeoff and landing lift that are achievable with the tail at the minimum clearance from the runway.

In the right branch of the *Analysis* process, the *Displacements* process is followed by the *Loads Convergence* and *Stress and Buckling* processes. The *Displacements* process calculates the FEM displacements under the trimmed cruise loads for use in the *Loads Convergence* process. Then, the *Loads Convergence* process performs the aeroelastic trim calculations for the six noncruise load conditions (2-7). (Within the *Loads Convergence* process, the *Rigid Trim* and *Displacements* processes are used again at each of these load conditions.) This resulting set of loads, along with the load representing taxi, is used in the *Stress and Buckling* process to provide stress and buckling response values.

3.2.3 Geometry Process

The *Geometry* process is displayed in fig. 6. The *Geometry* process calculates the geometry constraints and provides shape parameterization for the HSCT4.0 application. An important feature of any shape optimization formulation is the means to parameterize the geometry in terms of a set of user-defined design variables that can be systematically varied during the optimization to improve the design. The parameterization is done once after the shape design variables have been selected. (Reference 9 provides a survey of shape parameterization techniques for multidisciplinary optimization and highlights some emerging ideas.) As shown in fig. 6, the *Geometry* process consists of nine subprocesses: *Linear Aero Model Update*, *Miscellaneous Geometry Update*, *Nonlinear Aero Surface Model Update*, *FE Model Update*, *Fuel Geometry*, *Performance Geometry*, *Scrape Geometry*, *Section Property Update*, and *Structural Geometry*. Each subprocess is described in the next section.

A discussion of the shape design variables is provided in section 3.1.1.1. The shape design variables and geometry constraints are listed in table 2 and table 4, respectively.

3.2.3.1 Geometry Subprocesses

The *Linear Aero Model Update*, *Miscellaneous Geometry Update*, *Nonlinear Aero Surface Model Update*, and *FE Model Update* subprocesses, shown in fig. 6, use the approach described in reference 9 to modify the geometry of the analysis models. This approach provides internal FEM grids consistent with aerodynamic surface grids. All analysis geometry models (i.e., aerodynamics and structures) are parameterized based on the locations of the shape design variables that are varied relative to the reference geometry. For each design cycle, the “derived grids/models” are generated by applying the shape design variable values to each of the parameterized grids. Recall that there are three aerodynamic shapes, one for each flap setting.

The output from the *Miscellaneous Geometry Update* process, shown in fig. 22, consists of a set of curves that defines a wire-frame description of the model for the various miscellaneous geometry subprocesses. The *Fuel Geometry* subprocess uses the locations of sets of points at the corners of the fuel tanks to calculate the total and individual fuel volumes of the tanks. The *Performance Geometry* subprocess uses discretized curves to calculate a wide variety of geometric information needed as input for the *Performance* process; examples are the wing span, sweep angles, and aspect ratio, the wing chords and maximum thickness at several span stations, and the fuselage dimensions. The *Scrape Geometry* subprocess uses the location of selected points on the aircraft surface to calculate the pitch angle for which one or more of these points touches the ground.

Two *Geometry* subprocesses generate the FEM data. The *FE Model Update* subprocess updates the finite element node locations, and the *Section Property Update* subprocess updates the finite element section properties. The *FE Model Update* subprocess uses the 27 shape design variables to produce a set of finite element node locations. The *Section Property Update* subprocess uses the 244 structural design variables (table 3) to produce 61 laminated composite shell property sets.

Both the ply-mixture and AIT ratios are computed by the *Structural Geometry* subprocess. (Refer to section 1.3.2 for definitions of the symbols used below.) The ply-mixture ratios are computed at each of the sixty-one design variable zones (see fig. 19 and fig. 20) using equations 3 - 5.

$$(3) \quad PM_0 = t_0/t_{Face}$$

$$(4) \quad PM_{45} = 2t_{45}/t_{Face}$$

$$(5) \quad PM_{90} = t_{90}/t_{Face}$$

The AIT ratios are computed at each of 30 wing stations (each with a corresponding upper and lower airfoil surface node, see fig. 23) from the equation below:

$$(6) \quad AIT = 1 + \frac{h_{min} - r}{T}$$

where $h_{min} = 0$ for HSCT4.0.

3.2.4 Weights Process

The *Weights* process is displayed in fig. 7. The *Weights* process computes the as-built nodal and total weights, fuel weights, and the c. g. location of a given aircraft configuration.

A particular aircraft geometry is specified with the shape design variables (table 2), which define the aircraft cruise shape. A particular mass distribution is defined using shape design variables, structural design variables, and fuel loading. The term “fuel loading” implies both the weight and the location of the fuel. The shape design variables may affect the takeoff and cruise fuel loadings by changing the available fuel volume.

The total theoretical FEM weights represent only about 20 percent of the total aircraft as-built weight. As such, the total theoretical FEM weight is incomplete for use as an optimization objective function, because the percentage change in the theoretical FEM weight is much greater than the percentage change in the aircraft total as-built weight. Therefore, it is required that the *Weights* process estimate other aircraft weights to provide a total as-built weight.

The *Weights* process uses as-built nodal weights for a reference aircraft and theoretical FEM weights to provide a reasonable estimate of the as-built weight. The process uses a set of as-built nodal weights for the reference airplane geometry, including the takeoff and cruise fuel loadings that were generated by the process described in reference 10. The reference data includes fuel weight, tank fill factors, c. g. locations for nine tanks, and two mass cases. The process also uses the FEM for the reference aircraft geometry for the calculation of the theoretical FEM weight. The weights process must yield the reference weight results for the reference airplane geometry, using the appropriate reference aircraft fuel loadings.

The reference weights data are a set of fixed data files that assign as-built nodal weights to each of the 14 weights meshes seen in fig. 24 for several types of weights. See table 11 for a complete listing of the weight mesh names and the types of weights assigned to each mesh. The as-built nodal weights in this data set were defined using the reference weights process described in reference 10 for the specific geometry shown in fig. 24. All 14 meshes are assigned as-built structural nodal weights. Some of the 14 meshes are also assigned nodal weight increments for nonstructural, systems, payload, and fuel weights.

The structural weights are understood to include a theoretical FEM weight, as well as as-built structural weight increments for production splices, local pad-ups, side-of-body joints, adhesives, paints, materials for damage tolerance, sealants, and fasteners essential in building the aircraft. The nonstructural weights include weight increments for windows, landing gear doors, access doors, seat tracks, fuel tank baffles, passenger doors, and system attachment fittings. As shown in table 11, the nonstructural weights are applied only to the five meshes (1 through 5) representing the primary aircraft structure. For the HSCT4.0 application, the systems weights are provided as fixed nodal loads.

Table 11. Weight Types Assigned to Each Weight Mesh

Weight Mesh Number	Weight Mesh Name	Structural Weights	NonStructural Weights	Systems Weights	Payload Weights	Fuel Weights
1	Inboard Wing	Yes	Yes	Yes	No	Yes
2	Outboard Wing	Yes	Yes	Yes	No	No
3	Fwd-Fuselage	Yes	Yes	Yes	Yes	No
4	Mid-Fuselage	Yes	Yes	Yes	Yes	No
5	Aft-Fuselage	Yes	Yes	Yes	Yes	Cruise only
6	Vertical Tail	Yes	No	Yes	No	No
7	Horizontal Tail	Yes	No	Yes	No	No
8	Inboard Strut	Yes	No	Yes	No	No
9	Outboard Strut	Yes	No	Yes	No	No
10	Inboard Control Surfaces	Yes	No	Yes	No	No
11	Outboard Control Surfaces	Yes	No	No	No	No
12	Rudder	Yes	No	No	No	No
13	Elevator	Yes	No	No	No	No
14	Nose Cone	Yes	No	No	No	No

The as-built total weight of the aircraft can be considered to be a hierarchy of summations of smaller weight groupings, as illustrated in tables 12 and 13. Mesh numbers are shown in parentheses. In general, moving from right to left in these two tables for each component, each vertical bar in the table represents a summation of several items from the right column to form the item in the left column. The fuel weight in table 13 is an exception; the fuel component is either cruise or GTOW fuel.

For example, in table 12, nodal theoretical FEM weights for a given mesh are summed to find a total theoretical FEM weight for the same mesh. The summation of nodal theoretical FEM weights for a given mesh into a total theoretical FEM weight for the same mesh is defined as a subassembly weight. Likewise, the summation of nodal as-built structural weight increments for a given mesh to find a total as-built structural weight increment for the same mesh is also defined as a subassembly weight. A similar subassembly weight is defined as the summation of nodal non-structural weights for a given mesh (of meshes 1 through 5 only) to find a total nonstructural weight for the same mesh.

Table 12. Structural Weights Hierarchy

Component	Reference Aircraft Assembly (Weight Mesh Number, fig. 24)	Reference Aircraft Subassembly
Wing	Inboard Wing Structure (1)	Theoretical FEM Weight
		As-built Structural Weight Increment
		Nonstructural Weight
	Outboard Wing Structure (2)	Theoretical FEM Weight
		As-built Structural Weight Increment
		Nonstructural Weight
	Inboard Strut Structure (8)	Theoretical FEM Weight
		As-built Structural Weight Increment
	Outboard Strut Structure (9)	Theoretical FEM Weight
		As-built Structural Weight Increment
	Inboard Control Surface Structure (10)	Theoretical FEM Weight
As-built Structural Weight Increment		
Outboard Control Surface Structure (11)	Theoretical FEM Weight	
	As-built Structural Weight Increment	
Fuselage	Nose Cone Structure (14)	Theoretical FEM Weight
		As-built Structural Weight Increment
	Fwd-Fuselage Structure (3)	Theoretical FEM Weight
		As-built Structural Weight Increment
		Nonstructural Weight
	Mid-Fuselage Structure (4)	Theoretical FEM Weight
		As-built Structural Weight Increment
		Nonstructural Weight
	Aft-Fuselage Structure (5)	Theoretical FEM Weight
As-built Structural Weight Increment		
Nonstructural Weight		
Vertical Tail	Vertical Tail Structure (6)	Theoretical FEM Weight
		As-built Structural Weight Increment
	Rudder Structure (12)	Theoretical FEM Weight
		As-built Structural Weight Increment
Horizontal Tail	Horizontal Tail Structure (7)	Theoretical FEM Weight
		As-built Structural Weight Increment
	Elevator Structure (13)	Theoretical FEM Weight
		As-built Structural Weight Increment

Table 13. Systems, Payload, and Fuel Weights Hierarchy

Component	Reference Aircraft Assembly (Weight Mesh Number, fig. 24)	Reference Aircraft Subassembly (Weight Mesh Number, fig. 24)
Engines	Engines (8 and 9)	Inboard Engine Weight (8)
		Outboard Engine Weight (9)
Systems	Systems (1 through 10)	Inboard Wing Systems Weight (1)
		Outboard Wing Systems Weight (2)
		Fwd-Fuselage Systems Weight (3)
		Mid-Fuselage Systems Weight (4)
		Aft-Fuselage Systems Weight (5)
		Vertical Tail Systems Weight (6)
		Horizontal Tail Systems Weight (7)
		Inboard Strut Systems Weight (8)
		Outboard Strut Systems Weight (9)
		Inboard Control Surface Systems Weight (10)
Landing Gear	Landing Gear (1 and 3)	Body Landing Gear Weight (3)
		Wing Landing Gear Weight (1)
Payload	Payload (3, 4, and 5)	Fwd-Fuselage Payload Weight (3)
		Mid-Fuselage Payload Weight (4)
		Aft-Fuselage Payload Weight (5)
*Fuel	Cruise Fuel (1 and 5)	Inboard Wing Cruise Fuel Weight (1)
		Aft-Fuselage Cruise Fuel Weight (5)
	GTOW Fuel (1)	Inboard Wing GTOW Fuel Weight (1)

* Fuel is either cruise or GTOW fuel: i.e., there is no summation.

The summation of the subassembly weights for a given mesh to find the total mesh structural and nonstructural weight is defined as an assembly weight. For example, the summation of the total theoretical FEM weight, as-built structural weight increment, and nonstructural weight, all for mesh 1, is defined for the assembly known as the Inboard Wing Structure. Several assembly weights are summed to find the total weight of aircraft components, such as the wing or fuselage. The OEW is the summation of the weights for the wing, fuselage, vertical tail, horizontal tail, engines, systems, and landing gear components. The cruise weight is the summation of the OEW, payload weight, and cruise fuel weight. The GTOW is the summation of the OEW, payload weight, and GTOW fuel weight.

3.2.4.1 Weights Subprocesses

The Weights process has two subprocesses, *Theoretical FEM Weight* and *Built Up Weights* (see fig. 7). Both processes use data computed by the *Geometry* process. The *Theoretical FEM Weight* subprocess computes the nodal theoretical FEM weights for both the reference (a one time only

calculation) and current aircrafts. The theoretical FEM weights are computed on an element-by-element basis from first principles relating the structural dimensions of elements in the FEM to their volumes, densities, and weights. The theoretical FEM element weights are then translated to nodal weights by using an element-to-node translation algorithm specific to the finite element analysis code. The structural design variables size only the finite elements representing the skins of the primary structural components (meshes 1 through 5). The shape design variables affect the theoretical FEM weights of all meshes except for meshes 6, 7, 12, and 13.

The *Built Up Weights* subprocess computes the as-built nodal structural weight increments and adds them to the theoretical FEM nodal weights. This subprocess also computes fuel weights, airplane total weight, and the c. g. location based on the design variable values and fuel fill factors. Nodal as-built weights for the reference aircraft, computed by the reference 10 process, are provided to the *Built Up Weights* subprocess from the reference weights data set.

As-built structural weight increments are determined by subtracting the reference theoretical FEM weight from the reference aircraft as-built structural weight at every corresponding node. These as-built structural increments are then applied as a constant vector of increments to the theoretical FEM weights for the current aircraft. Systems and payload weights are applied as constant vectors to the theoretical FEM weights for the current aircraft. Three mass cases are considered: 1) cruise weight, 2) GTOW, and 3) OEW.

The next step in the process is to calculate the fuel weight for both the cruise and the GTOW mass cases. Because the exact location and geometry of the nine reference aircraft fuel tanks are not known, they are modeled as a set of 11 six-faced, straight-edged polygons. These tanks are defined as part of the HSCT4.0 geometry parameterization and are all contained within the aircraft wing and fuselage outer mold lines. The 11 tank definitions include all the locations of FEM nodes of the reference aircraft for which fuel can be assigned, considering the different fuel loads and distribution for the takeoff and cruise conditions.

Fuel volumes for each of 11 tanks are computed for the reference and current aircrafts. These fuel volumes are multiplied by the density of JP-4 aircraft fuel to obtain 11 full fuel weights for each of the reference and current aircrafts. These 11 full fuel weights for the reference and current aircraft geometries must then be associated with nine fuel weights provided as part of the reference aircraft data. The association is done by comparing the reference geometry tank c. g.'s. Some of the tanks may be combined or renumbered to associate with the nine reference geometry fuel tanks.

The 11 tank geometries from the HSCT4.0 application parameterization and the nine fuel tank weights and fill factors from the reference aircraft data must be reconciled so that the HSCT4.0 application fuel geometries contain the right amount of fuel weight for the reference geometry. Three sets of scale factors are used to determine the correct fuel tank weights. The first two scale factor sets are geometric factors and apply to any mass case. The third set of scale factors is mass case specific, i.e., there is a scale factor set for both the cruise and the GTOW mass case. The first set of scale factors is applied to the 11 HSCT4.0 application computed fuel weights to ensure that the correct reference fuel weight is obtained for the reference geometry. The second set of scale factors is applied to the HSCT4.0 application fuel weights to account for changes in the fuel volume between the current and reference geometries. The third set of scale factors is used to change

the full fuel weight within each tank, based upon the reference aircraft tank fill factors for each tank between the takeoff and cruise conditions. When the current aircraft is geometrically and structurally identical to the reference aircraft, the as-built weights computed by the *Built Up Weights* subprocess identically reproduce the reference weights data set.

3.2.5 Nonlinear Corrections Process

The process for calculating nonlinear corrections is displayed in fig. 8. The purpose of the *Nonlinear Corrections* process is to compute a correction term to be added to each linear aerodynamics surface pressure coefficient to account for nonlinear aerodynamics effects. This computation is the first stage in the variable-fidelity aerodynamics process. The second stage applies this set of correction terms each time a linear aerodynamics calculation is made during the *Rigid Trim* process.

Although the optimization requirements have been deferred as stated in section 3.2.1, for completeness, the role of the *Nonlinear Corrections* process in the optimization process is described here. The computational time for calculating the nonlinear aerodynamics is much longer than the combined computational time for all the other processes in the *Analysis* process. In an effort to accelerate the overall computation time, the *Nonlinear Corrections* process shall be computed in parallel with several cycles of the optimization process. The *Nonlinear Corrections* process will be executed during the first optimization cycle using zero delta displacements. In subsequent optimization cycles, the most recently calculated nonlinear corrections will be used until the updated nonlinear corrections are available. Note that the lag in computation will make the corrections inconsistent with the current set of design variable values. In the later stages of the optimization, the nonlinear corrections will be updated every cycle to increase accuracy. For the cycles in which the nonlinear corrections are computed, the updated nonlinear aerodynamics volume grids and linear aerodynamics grids are used in recalculating the nonlinear correction terms for each load condition.

Calculations of the *Nonlinear Corrections* process are required for each load condition. Because the process is independent for each load condition, these calculations can be performed in parallel. In addition, within each load condition calculation, the right and the left branches of this process (refer to fig. 8) are independent of each other and can be performed in parallel before synchronizing at the *Calculate Nonlinear Corrections* subprocess.

3.2.5.1 Nonlinear Corrections Subprocesses

The right branch in fig. 8 calculates the nonlinear aerodynamics for a given load condition. To produce the displaced nonlinear aerodynamics surface grid, the *Apply Nonlinear Delta Displacements* subprocess adds the delta displacements from the most recent *Calculate Delta Displacements* (shown in fig. 14) subprocess to the derived nonlinear aerodynamics surface grid. The *Volume Grid Adjustment* subprocess then uses this displaced nonlinear aerodynamics surface grid and the baseline nonlinear aerodynamics volume and surface grids to produce the displaced nonlinear aerodynamics volume grid, which is used as input to the *Nonlinear Aero* subprocess. The appropriate altitude and the total weight (computed within the *Weights* process) are used by the *Calculate C_L* subprocess to determine the target C_L that is input to the *Nonlinear Aero* subprocess.

The *Nonlinear Aero* subprocess calculates the nonlinear aerodynamics surface pressures, forces, and moments for the input C_l by iteratively solving the three-dimensional Euler equations with a finite-volume formulation on the structured volume grid.

The left branch in the figure calculates linear aerodynamics for the same load condition. To produce the displaced linear aerodynamics surface grid, the *Apply Linear Delta Displacements* subprocess adds the delta displacements from the most recent *Calculate Delta Displacements* subprocess to the derived linear aerodynamics surface grid. Independent executions of the *Linear Aero* subprocess are made for a predetermined set of angles of attack and tail deflections. The angles are preselected by the aerodynamics specialist to bracket the range of expected angles for the given load condition. The *Linear Aero* subprocess provides surface pressure distributions, forces, and moments to the *Calculate Nonlinear Corrections* subprocess.

The *Calculate Nonlinear Corrections* subprocess determines the linear aerodynamics angle of attack that generates the same configuration normal force that was calculated in *Nonlinear Aero*. The normal force was chosen as the matching condition, rather than lift force, because only the normal force components are used in the aeroelastic *Loads Convergence* process. The linear pressures are interpolated at the angle of attack that matches the configuration normal force. The nonlinear pressures are transferred to the linear aerodynamics grid using a method that conserves the total forces and moments. The nonlinear corrections are then computed as the panelwise difference between the resulting linear pressures and nonlinear pressures. The integral of these corrections over the entire configuration is zero because of the imposed normal force matching condition. As a result, there is no net normal force contributed when the corrections are applied in the *Rigid Trim* process; however, there will be a net pitching moment for the configuration. This moment is accounted for in the *Rigid Trim* process.

3.2.6 Rigid Trim Process

The *Rigid Trim* process is shown in fig. 9. The *Rigid Trim* process represents the second stage in the variable-fidelity aerodynamics analysis approach. The purpose of this process is to provide surface pressures and total aerodynamic forces at a trim condition. The *Rigid Trim* process executes within two different processes; both of these process executions involve different load conditions. The first execution of *Rigid Trim* occurs after the *Weights* and *Nonlinear Corrections* (when computed) processes have completed; during this execution, the aircraft is trimmed for the cruise load condition only. The *Rigid Trim* process is also executed from within the *Loads Convergence* process for each of the noncruise load conditions two through seven.

3.2.6.1 Rigid Trim Subprocesses

The *Rigid Trim* process computes the trimmed aerodynamic loads independently for each load condition. The input grid for the *Linear Aero* subprocess is the current geometry for the load condition being considered (i.e., the derived grid for the cruise load condition or the displaced grids for the noncruise load conditions). The *Linear Aero* subprocess computes the surface pressures for a prescribed set of angles of attack and tail deflections for this load condition. The table of angles of attack and tail deflections is preselected by the aerodynamics specialist to bracket the range of expected angles for each load condition.

In the *Apply Nonlinear Corrections* subprocess, the most recent nonlinear corrections for the current load condition are added, node-by-node, to the linear aerodynamics pressures. These corrected pressures provide input to the *Aero Moment Equilibrium* subprocess, which computes trimmed aerodynamic loads for the load condition. The trimmed aerodynamic loads and total force coefficients are interpolated using the calculated angle of attack and the tail deflection that together yield the target C_L with no net pitching moment about the c. g. location. (The input weight and c. g. are calculated in the *Weights* process and depend on the specific load condition. This target C_L is calculated from this input weight and the specific load condition.) The trimmed aerodynamic loads will be used in the subsequent *Displacements* process executions. The trimmed force coefficients (C_L and C_{Di}), for the cruise load condition only, will be used in the *Polars* process.

3.2.7 Polars Process

The *Polars* process is shown in fig. 10. Aircraft drag polars are calculated for the current set of design variable values over a range of flight conditions. The *Polars* subprocesses calculate the lift-dependent and lift-independent drag components. All of these subprocesses use the derived linear aerodynamics grid for the 1g cruise shape (from the *Geometry* process) and the polars conditions. These conditions are preselected by the performance specialist and consist of a fixed set of Mach numbers, altitudes, and lift coefficients. No nonlinear corrections are used in the *Polars* process.

3.2.7.1 Polars Subprocesses

The right side of fig. 10 shows the lift-dependent drag subprocesses. First, the *Linear Aero* subprocess is executed for combinations of angle of attack and tail deflection. These combinations are preselected by the aerodynamics specialist at Mach numbers matching those from the polars conditions table. Then, in the *Induced Drag* subprocess, for each Mach number and angle of attack, the trimmed drag due to lift is interpolated at the tail deflection that produces no net pitching moment. This produces an intermediate table of lift and trimmed drag coefficients at the specified angles of attack. This table is extended by including the trimmed cruise condition data computed within the *Rigid Trim* process. Finally, the data in this table are interpolated at the polars conditions C_L values to provide the lift-dependent drag coefficient (C_{Di}) table.

The left side of fig. 10 shows the lift-independent drag subprocesses, *Wave Drag* and *Viscous Drag*. The *Wave Drag* subprocess is executed at the zero-lift angle of attack for the specified Mach numbers; the computed wave drag coefficients (C_{Dw}) are only non-zero for supersonic Mach numbers. The *Viscous Drag* subprocess calculates all viscous drag components (C_{Dv}) including skin friction, roughness, and profile (form) drag. The *Viscous Drag* subprocess is executed for the specified Mach numbers and altitudes.

The *Assemble Polars* subprocess produces the mission tables of drag coefficients, which will be used by the *Performance* process. First, the lift-independent drag coefficient tables are assembled from the wave drag and viscous drag coefficients. Then the lift-dependent drag coefficient table is combined with the lift-independent drag coefficient tables to produce the mission tables.

3.2.8 Performance Process

The *Performance* process shown from the *Analysis* process (Fig. 5) contains no subprocesses. However, a more detailed view of the inputs and outputs of this process are shown in fig. 11. The *Performance* process calculates the aircraft range and several other performance metrics (see definition, table 10).

The *Performance* process requires numerous inputs from a variety of sources. The derived performance geometry (a coarse resolution, wire frame geometry description) is obtained from the *Geometry* process. The wing and fuselage fuel weights and GTOW are obtained from the *Weights* process. Mission tables of drag coefficients are obtained from the *Polars* process. A variety of other reference aircraft data (engine, mission, and performance analysis parameters) are obtained from prior analyses of the reference configuration.

Given these inputs, the *Performance* process repeatedly solves the equations of motion until a mission analysis is obtained that is consistent with the input geometry, weights, aerodynamics, and engine tables. The mission analysis considers the takeoff, landing, climb, cruise, descent, and reserve portions of a specified mission profile, while requiring that the various Federal Aviation Regulations (FAR) and Federal Aviation Agency (FAA) flight operational procedures, required for certification, are satisfied. These regulations are summarized in references 11-13.

3.2.9 Ground Scrape Process

The *Ground Scrape* process is shown in fig. 12. The purpose of the *Ground Scrape* process is to calculate the maximum available lift forces at ground scrape conditions on takeoff and landing. These forces will be used in the formulation of the ground scrape constraints (see table 4). The takeoff lift force must be greater than the GTOW, and the landing lift force must be greater than the GLW (usually defined as GTOW minus one-half takeoff fuel weight). Weights exceeding these maximum available lift forces would require higher angles of attack, resulting in the aircraft tail scraping the ground. Although the implemented *Ground Scrape* process is not realistic in fully constraining the aircraft design to avoid all ground scrape conditions, it is included as a simple model of more realistic ground scrape processes that may be used by industry.

3.2.9.1 Ground Scrape Subprocesses

The *Ground Scrape* process consists of three subprocesses. The first subprocess, *Scrape Angle*, calculates the takeoff and landing pitch angles with the landing gear just touching the runway at zero roll angle and the aircraft tail at the specified minimum ground clearance (6 inches) to avoid tail strike. For the takeoff condition, the landing gear is assumed to be at the static length (-6 inches stroke), and for the landing condition, the gear is at the fully stroked length (-24 inches stroke). The tail strike point comes from the derived scrape geometry, which is produced by the *Geometry* process.

The second subprocess, *Linear Aero*, computes the lift coefficients at the maximum angles of attack for takeoff and landing (the pitch angles just calculated) at a low subsonic Mach number. For simplicity, the *Linear Aero* process uses the 1g cruise shape (the derived linear aerodynamics

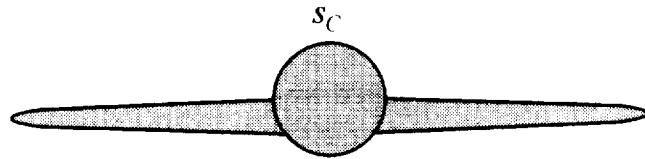
grid from *Geometry*) for the aircraft, ignoring realistic takeoff and landing flap and tail deflections.

The third subprocess, *Scrape Lift*, uses standard atmosphere data for a high altitude airport (5000 feet) and the takeoff and landing speeds from the performance metrics (generated by the *Performance* process) to convert the lift coefficients into the maximum available takeoff and landing lift forces.

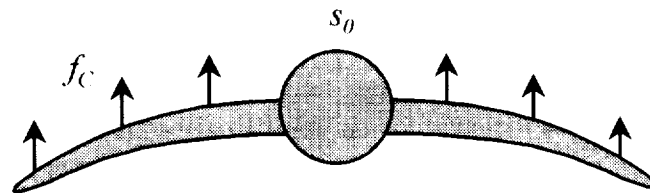
3.2.10 Displacements Process

The *Displacements* process is shown in fig. 13. Structural displacements due to displacement loads are generated by the *Displacements* process. When *Displacements* is called from within the *Analysis* process, the displacements are calculated for the cruise load condition only. When *Displacements* is called from within the *Loads Convergence* process, the displacements are calculated for load conditions two through seven. In this section, background for the cruise displacement calculations is presented. Refer to section 1.3.2 for the definitions of symbols used in this section.

Both the geometry of the aerodynamics model used in the *Rigid Trim* process and the geometry of the FEM used in the *Displacements* process are based on the 1g cruise shape (s_c , see sketch below), which is described by the shape design variables.



In order to obtain the correct stresses from a structural analysis, the displacements must be computed relative to the unloaded shape, s_0 (see sketch below). The unloaded shape is the aircraft shape that deforms to the cruise shape when cruise loads (f_c) are applied.



The cruise loads are computed from the following equation:

$$(7) \quad f_c = a_c - \text{nodal cruise weight}$$

where a_c represents the aerodynamic cruise loads. The inertia loads (load factor, which is 1.0 for the cruise case, multiplied by the nodal cruise weight) are subtracted from a_c , because the weight acts in the $-z$ direction.

The shape change (displacements, u_0) from the unloaded shape to the cruise shape is given by the following equation:

$$(8) \quad s_c - s_0 = u_0 = [K_0]^{-1} f_c$$

The stiffness matrix of the unloaded shape (K_0) is unknown. Assuming that the FEM is geometrically linear, the differences between K_0 and the stiffness matrix of the cruise shape (K_c) are negligible. Using this assumption, f_c can be applied to K_c to produce a set of cruise displacements (u_c).

$$(9) \quad u_c = [K_c]^{-1} f_c = [K_0]^{-1} f_c = u_0$$

thus, the unloaded shape is given by equation 10.

$$(10) \quad s_0 = s_c - u_0 = s_c - u_c$$

3.2.10.1 Displacements Subprocesses

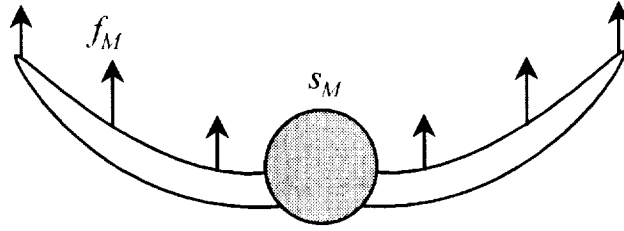
As shown in fig. 13, the *Displacements* process uses data calculated in the *Weights* process (nodal cruise weight and nodal GTOW) and the *Geometry* process (derived FEM and derived section properties). In the first step of the *Displacements* process, the *Loads Transfer* subprocess transfers the z direction aerodynamic loads to finite element nodes. *Loads Transfer* is executed once for the cruise condition (when *Displacements* is called from within the *Analysis* process) and once per iteration for load conditions two through seven in the *Loads Convergence* process (fig. 14).

The *Add Inertia Loads* subprocess calculates the displacement loads by subtracting the inertia loads (acting in the $-z$ direction) from the aerodynamic loads. The final subprocess to be executed is *FEM Displacements*, which computes structural displacements at the finite element nodes due to the displacement loads. For the HSCT4.0 application, the *FEM Displacements* subprocess performs a linear static finite element analysis.

3.2.11 Loads Convergence Process

The *Loads Convergence* process is shown in fig. 14. Before the *Loads Convergence* process is discussed, it is necessary to explain the need for an iterative aeroelastic convergence process. In this section, load conditions two through seven are referred to as maneuver conditions, and loads are considered to be forces acting on nodes. Refer to section 1.3.2 for the definitions of symbols used in this section.

At maneuver conditions, the aircraft has maneuver shape s_M and is subjected to maneuver loads f_M (illustrated in the following sketch).

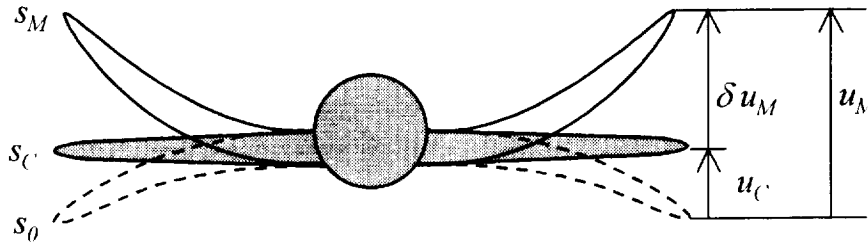


The maneuver loads are computed from the following equation:

$$(11) \quad f_M = a_M - LF \cdot \text{nodal GTOW}$$

where a_M represents the aerodynamic maneuver loads, and LF represents the maneuver load factor. The inertia loads (LF multiplied by *nodal GTOW*) are subtracted from a_M , because the weight acts in the $-z$ direction. Two unknowns, s_M and a_M , are determined using the *Loads Convergence* process. The maneuver shape is the net vehicle shape used for the aerodynamic analysis and is consistent with the deformed structural shape produced by the aerodynamic loads.

The *Rigid Trim* process is used to compute a_M using s_M as input. The aerodynamic grids used in *Rigid Trim* are based on s_C (see following sketch).



To compute a_M , an aerodynamic grid based on s_M is required. The aerodynamic grid for s_M is computed by deforming the s_C grid by δu_M , the delta displacements. The delta displacements are computed from equation 12:

$$(12) \quad \delta u_M = s_M - s_C = (s_M - s_0) - (s_C - s_0) = u_M - u_C$$

Recall that equation 9 is used to compute u_C . The other unknown in computing δu_M is u_M (the maneuver displacements), which is computed using equation 13:

$$(13) \quad u_M = [K_0]^{-1} f_M = [K_C]^{-1} f_M$$

(The u_M is computed from a finite element analysis.) The maneuver displacements are thus a function of the maneuver loads, and the aerodynamic loads (a component of the maneuver loads) are a function of the maneuver shape of the aircraft.

$$(14) \quad u_M = u_M(f_M, K_C) = u_M(a_M, \text{nodalGTOW}, K_C)$$

$$(15) \quad a_M = a_M(u_M, u_C, s_C)$$

Because of this mutual dependency between aerodynamic maneuver loads and maneuver displacements, a converging iterative process is used to determine the converged maneuver loads for load conditions two through seven. This process is called the *Loads Convergence* process in the HSCT4.0 application.

3.2.11.1 Loads Convergence Subprocesses

As shown in fig. 14, the *Loads Convergence* process uses data calculated in the *Weights* process and the *Geometry* process. In the first step of the *Loads Convergence* process, the *Apply Linear Delta Displacements* subprocess uses the delta displacements (see equation 12) to modify the derived linear aerodynamic grids, generating displaced linear aerodynamic grids. For the first iteration in the loads convergence loop, a vector of zero delta displacements is used, and the resulting displaced aerodynamic grids represent the cruise shape of the aircraft.

In the next step, the *Rigid Trim* process uses the displaced linear aerodynamics grids and the GTOW and c. g. to compute trimmed aerodynamic loads for load conditions two through seven. The *Rigid Trim* process is explained in detail in section 3.2.6 “Rigid Trim Process”.

The *Displacements* process uses the trimmed aerodynamic loads and the nodal GTOW to compute new displacements for load conditions two through seven. The *Displacements* process is explained in detail in section 3.2.10 “Displacements Process”.

The *Loads Convergence* process iterates until convergence. Convergence is achieved when the net vehicle shape used for the aerodynamic calculations is consistent with the structural displacements caused by the structural loads.

For each load condition, the *Calculate Delta Displacements* subprocess computes the δu_M using equation 12 above. This subprocess is only invoked if the convergence criterion is not met. When the convergence criterion is met, the final set of delta displacements corresponds to the net vehicle shape that was used for the final aerodynamic calculations.

The output of the *Loads Convergence* process is a set of converged loads and converged delta displacements.

3.2.12 Stress and Buckling Process

The *Stress and Buckling* process is shown in fig. 15. Refer to section 1.3.2 for the definitions of symbols used in this section.

The structural responses are listed in table 7 and include SFI and BLF for each quad element and triangle element in each design variable zone for each load condition. As shown in fig. 19, there

are 61 design variable zones. All 61 design variable zones contain quad elements, and 35 of these zones also contain triangle elements.

The *Augment Loads* subprocess prepares load conditions two through eight (see table 9) and assembles them into augmented loads. The *FEM Stress* subprocess then calculates an SFI, stress resultant, and stress for each quad and triangle element in the design variable zones for each load condition. The stress resultants are used to calculate the BLF for each load condition in each quad and triangle element in the design variable zones.

3.2.12.1 Stress and Buckling Subprocesses

The *Augment Loads* subprocess generates a set of seven augmented loads to be used to calculate the SFI's and BLF's. The first six augmented loads (based on load conditions two through seven) are obtained from the converged loads generated by the *Loads Convergence* process. Internal forces representing a cabin pressure of 10.78 psi are added to selected nodes on the fuselage for all of the trimmed loads from the *Loads Convergence* process. The seventh augmented load is based on the taxi condition (load condition 8). The forces for the taxi condition were obtained by multiplying the inertia forces (nodal masses multiplied by -1.0) by the appropriate load factor (given in table 9) and adding reaction forces at two nodes. The values for these two reaction forces were determined by the following rationale. The aft landing gear is attached to the wing at three locations. If all three points were constrained in the upward z direction, there is the possibility of getting unrealistic reactions at these three points because of the redundancy of the supports as the wing displaces. For example, as the wing stiffness changes there is the possibility that a force at the gear point could be in the downward instead of upward direction; therefore, only one of the three gear points was constrained and forces equal to one-third of the total gear load were applied to the other two nodes. The seven augmented loads are then multiplied by a factor of safety (1.5) and are then assembled into the augmented loads for the *FEM Stress* subprocess.

Three in-plane stresses (σ_{11} , σ_{22} , and τ_{12}) are computed from a linear static finite element analysis. An SFI is used to normalize these stresses by material stress allowables (X_C , X_T , Y_C , Y_T , and S) defined by the material properties. An SFI is also used to reduce the output from three values per element to one value. The Hoffman SFI is computed for all layers in the composite face-sheets (a total of eight layers) for each of the 2260 elements in the design variable zones for each of seven load conditions. For the constraints, only the maximum layer-wise SFI in each element is retained. The equation used to compute the SFI is given below¹⁴:

$$(16) \quad SFI = \sigma_{11} \left(\frac{1}{X_T} + \frac{1}{X_C} \right) + \sigma_{22} \left(\frac{1}{Y_T} + \frac{1}{Y_C} \right) - \frac{\sigma_{11}^2}{X_T X_C} - \frac{\sigma_{22}^2}{Y_T Y_C} + \frac{\tau_{12}^2}{S^2} + \frac{\sigma_{11} \cdot \sigma_{22}}{X_T X_C}$$

For the in-plane stresses, the 1- and 2-directions correspond to the 0° and 90° ply orientations in fig. 21, respectively.

The *Buckling* subprocess uses analytical solutions for the buckling load in simply-supported square plates. This is referred to as a local buckling constraint, as opposed to a global (eigenvalue-based) buckling constraint. The BLF is computed for each element in the design variable zones using the equations below:

$$(17) \quad BLF = \frac{N_A}{\min(N_{mn})} + \left(\frac{N_{12}}{N_{shear}} \right)^2$$

$$(18) \quad N_A = -1 \cdot \min(N_{11}, N_{22}, 0)$$

$$(19) \quad N_B = -1 \cdot \max(N_{11}, N_{22})$$

In the above equations, the in-plane stress resultants (N_{11} , N_{22} , and N_{12}) are computed from a linear static finite element analysis. N_A is the largest compressive (negative) normal stress in the element. If both N_{11} and N_{22} are in tension (positive), then N_A is zero, otherwise, it is positive. The buckling allowables (N_{mn} and N_{shear}) are computed from the material properties and the element dimensions.

The bi-axial buckling load (N_{mn}) is obtained from the non-trivial solutions for stability of a simply-supported rectangular plate under uniform bi-axial compression¹⁵, as given in the equation below (the N_{mn} used is the lowest of the 25 combinations of $m, n=1$ to 5):

$$(20) \quad N_{mn} = \frac{\pi^2 \left[D_{AA} m^4 + 2(D_{12} + 2D_{66}) m^2 \left(\frac{nb}{a} \right)^2 + D_{BB} \left(\frac{nb}{a} \right)^4 \right]}{a^2 \left(m^2 + \left(\frac{nb}{a} \right)^2 \frac{N_B}{N_A} \right)}$$

Assuming the elements are square, then a is equal to b , and equation 20 becomes:

$$(21) \quad N_{mn} = \frac{\pi^2 [D_{AA} m^4 + 2(D_{12} + 2D_{66}) m^2 n^2 + D_{BB} n^4]}{a^2 \left(m^2 + n^2 \frac{N_B}{N_A} \right)}$$

A constant value (30 in.) is used for the element edge length (a). This edge length represents a standard aircraft frame spacing. The element bending stiffness matrix terms (D_{AA} , D_{BB} , D_{12} , and D_{66}) are computed from the material properties and the structural design variables.

The N_{shear} term in equation 22 is obtained from the shear buckling interaction equation¹⁶:

$$(22) \quad N_{shear} = \frac{32.5 \sqrt{D_{AA} D_{BB}} + 10.1 [2(D_{12} + 2D_{66})]}{b^2} \sqrt[4]{\frac{D_{AA}}{D_{BB}}} \quad \text{for } D_{AA} \geq D_{BB}$$

$$(23) \quad N_{shear} = \frac{32.5 \sqrt{D_{BB} D_{AA}} + 10.1 [2(D_{12} + 2D_{66})]}{b^2} \sqrt[4]{\frac{D_{BB}}{D_{AA}}} \quad \text{for } D_{BB} > D_{AA}$$

3.3 Data Storage Requirements

Specific data storage requirements have been deferred to the HSCT4.0 application project design phase.

3.4 Performance Requirements

Because the HSCT4.0 application project is a high-performance computing project, the following requirement applies:

Requirement #51 - The HSCT4.0 application shall demonstrate high-fidelity analysis calculations using high-performance techniques, exploiting coarse-grained parallelism where appropriate.

3.5 System Design Constraints

The requirements phase for the HSCT4.0 application project included selecting a set of existing codes to perform the computations described earlier in section 3.2. In the remaining portion of this section, the reasons for the major discipline code selections are briefly stated. Additional codes selected are briefly described in the appendix. The functional requirements indicated where these codes would be used. The project's requirements phase also selected the approach for implementing distributed computing.

Requirement #52 - The HSCT4.0 application shall be designed, using the legacy codes described in the appendix, to execute in a distributed, heterogeneous Unix environment using the Java programming language and CORBA-compliant software. If a cost-effective CORBA based product is not available for a target architecture, the Java remote method invocation language feature will be used to facilitate distributed computation.

The conventional multidisciplinary feasible formulation¹⁷, which requires a complete multidisciplinary analysis for each function evaluation, was chosen as the HSCT4.0 optimization approach. The SLP technique (using the CONMIN¹⁸ feasible directions optimizer and piecewise linear approximations) was chosen as the specific optimization algorithm. An important factor in the decision to use SLP was the amount of past team experience accumulated using this approach.

During the requirements phase of the project, the decision was made to use the GENESIS® software for structural analyses (see ref. 19). Among the other finite element codes considered for use were MSC-NASTRAN²⁰, COMET-AR²¹, and EAL²². GENESIS® and MSC-NASTRAN were considered top choices primarily due to derivative computation features. At the time the decision was made, the team considered MSC-NASTRAN to be superior in several factors (flutter analysis, eigenvalue and buckling analysis, team member familiarity, industry acceptance and use); however, GENESIS® was chosen due to the code's additional derivative capability and parallelism. (At the time this decision was made, MSC-NASTRAN did not have parallel capability.)

The USSAERO code^{23, 24} was chosen as the linear aerodynamics analysis code for calculating the pressure distribution and aerodynamic characteristics of wing-body-tail combinations in both sub-

sonic and supersonic potential flow. The USSAERO code provides a unified approach to the aerodynamic analysis of wing-body-tail configurations in both subsonic and supersonic flow. This code permits the analysis of noncircular bodies, provides a more accurate representation than other linear aerodynamics codes of the rounded wing leading edges, and allows the determination of wing interference effects in the presence of the body.

For nonlinear aerodynamics analysis, the CFL3D²⁵ code was chosen. This code has a very general structured-grid capability for complex geometries and has been used extensively within NASA and in industry for analysis of high-speed flows. The particular version used has been processed through the automatic differentiation tool ADIFOR²⁶ to provide aerodynamic sensitivities and can run on both sequential and parallel processors.

The FLOPS²⁷ mission analysis code was chosen as the performance code for a variety of reasons. First, the code executes quickly. Second, it has been used in prior NASA-industry studies. Third, its accuracy for mission analysis has been validated against similar analysis codes.

3.6 Software System Attributes

This section defines several requirements related to numerical precision and error handling.

Requirement #53 - All analysis codes shall be compiled as double precision, unless otherwise specified.

Requirement #54 - All exchange of data between analysis codes shall be in double precision format, unless otherwise specified.

Requirement #55 - All analysis codes shall indicate if errors have occurred. If an error occurs, the code shall report the cause of the error by returning an integer flag. A negative flag shall indicate a fatal error. A positive flag shall indicate a warning. A flag set equal the number 0 shall indicate normal execution.

Requirement #56 - The integrated HSCT4.0 system shall report any analysis code errors and take the appropriate actions.

Appendix

This appendix briefly describes the codes that have been selected for implementing HSCT4.0 application functions. The codes and the functions that they implement were identified in the summary of functional requirements, listed in section 3.2.

A2S. The Aerodynamics-to-Structures, A2S, code transfers the aerodynamic loads to the structural elements using a distribution process that preserves the total aerodynamic normal (z -component) force and moments. The surface pressure on each aerodynamic panel is first converted into a single force normal to the panel at its center. Only the configuration normal component of each panel force is then distributed among the nodes of the closest structural element. The distribution is done so that the total normal force and the total x -moment and y -moment at the structural nodes is the same as that of the aerodynamic panel force. This process is repeated for all the aerodynamic panels to transfer all the aerodynamic loads to the structure.

AWAVE. The AWAVE code computes the cross-sectional area and volume distributions as functions of the axial distance along the fuselage for an aircraft configuration input to the code in wave drag format. From the computed area and volume distributions, the wave drag coefficient is computed for the input configuration and also for an optimized, equivalent body of the same volume. The wave drag computations are performed at the configuration zero-lift angle of attack for a set of specified Mach numbers. The resulting wave drag is zero for subsonic Mach numbers and is some positive value for supersonic (greater than 1.0) Mach numbers. The user of the code inputs information to control the fidelity of the area and volume computations.

CDF1. The CDF1 program calculates several aircraft configuration viscous drag coefficient components, including the friction, form, and roughness drag coefficients, at user-specified Mach numbers and altitudes. The aircraft geometry is specified as a series of wetted-area segments, each with distinct viscous drag characteristics. The friction drag is computed by a strip theory method. The form drag is calculated as a function of friction drag and the roughness drag as a function of both friction and form drag.

CFL3D²⁵. The CFL3D code solves the three-dimensional, time-dependent Euler and thin-layer Navier-Stokes equations with a finite-volume formulation on structured grids. The equations are advanced in time implicitly with the use of 3-factor approximate factorization. It can employ grid sequencing, multigrid, and local time-stepping to accelerate convergence to steady state. It can also utilize a wide variety of grid multiple block connection strategies-including point matched, patched, and overset grid connections-in order to handle complex geometric configurations. Second-order upwind-biased spatial differencing is used for the inviscid terms, and flux limiting is used to obtain smooth solutions in the vicinity of shock waves. Viscous terms, if present, are centrally differenced. Several turbulence models of varying complexity are available. The particular version of the code used here is known as CFL3dv4.1hp. This version has been ported to parallel computer architectures via the use of MPI message passing protocols. Furthermore, the automatic differentiation tool ADIFOR²⁶ has been applied to this version of the CFL3D code. The resulting code is able to provide a numerical solution to the Euler (or Navier-Stokes) equations as well as consistent derivatives of the numerical solution with respect to shape design variables.

CSCMDO²⁸. The Coordinate and Sensitivity Calculator for Multidisciplinary Design Optimization (CSCMDO) code is a general purpose multi-block three-dimensional volume grid generator which is suitable for MDO. The code is fast, robust, highly automated, and written in ANSI "C" for platform independence. Algebraic techniques are used to generate and/or modify block face and volume grids to reflect geometric changes resulting from design optimization. Volume grids are generated/modified in a batch environment and controlled via an ASCII user input deck. This allows the code to be incorporated directly into the design loop. Volume grids have been successfully generated/modified for a wide variety of configurations.

FLOPS²⁷. The Flight Optimization System (FLOPS) code is a multidisciplinary system of computer programs for conceptual and preliminary design and evaluation of advanced aircraft concepts. It consists of nine primary modules: Weights, Aerodynamics, Engine cycle analysis, Propulsion data scaling and interpolation, Mission performance, Takeoff and landing, Noise footprint, Cost analysis, and Program control.

The FLOPS code may be used to analyze a point design, parametrically vary certain design variables, or optimize a configuration with respect to these design variables (for minimum gross weight, minimum fuel burned, maximum range, minimum cost, or minimum NO_x emissions) using nonlinear programming techniques. The configuration design variables are wing area, wing sweep, wing aspect ratio, wing taper ratio, wing thickness-chord ratio, gross weight, and thrust (size of engine). The performance design variables are cruise Mach number and maximum cruise altitude. The engine cycle design variables are the design point turbine entry temperature, the maximum turbine entry temperature, the fan pressure ratio, the overall pressure ratio, and the bypass ratio for turbofan and turbine bypass engines. The aircraft configuration, engine cycle and size, and the flight profile may be optimized simultaneously.

GENESIS¹⁹. The GENESIS® code is a fully integrated finite-element analysis/design software package. Analyses are available for static, normal modes, direct and modal frequency analysis, and heat transfer. Shape, sizing and topology optimization are the design options available to the user. Sensitivity derivatives can also be computed.

GP⁹. The GP code is a parameterization tool for complex shapes suitable for a multidisciplinary design optimization application. The approach consists of three basic concepts: 1) parameterizing the shape perturbations rather than the geometry itself, 2) exploiting Soft Object Animation algorithms used in computer graphics, and 3) relating the deformation parameters to aerodynamics shape design variables such as thickness, camber, twist, shear, and planform. The GP code formulation is independent of grid topology, which makes it suitable for use with a variety of analysis codes such as CFD and computational structural mechanics. This algorithm is suitable for low-fidelity (e.g., linear aerodynamics and equivalent laminated plate structures) and high-fidelity analysis tools (e.g., nonlinear CFD and detailed finite-element modeling). The analytical sensitivity derivatives are available for use in a gradient-based optimization.

L2NL. The L2NL code calculates the nonlinear corrections as the nodal difference between the nonlinear aerodynamics surface pressures, transferred to the linear aerodynamics grid, and the linear aerodynamics surface pressures that have been interpolated at a matching condition. The lin-

ear pressures are interpolated at the angle of attack at which the configuration normal force matches that from the nonlinear aerodynamics calculation.

S2W. The Structures-to-Wavedrag, S2W, code transfers the computed displacements from the structures grid to the linear aerodynamics grid. The transfer is accomplished by infinite-plate splines. This method is based on a superposition of the solutions for the partial differential equation of equilibrium for an infinite plate. The details of the method can be found in reference 29.

TRIM_NLC. The TRIM_NLC code adds the nonlinear corrections, node by node, to the linear aerodynamics surface pressures for a prescribed set of angles of attack and tail deflections. The code interpolates to determine the angle of attack and tail deflection angle that together yield the target lift coefficient with no net pitching moment about the c. g. location.

USSAERO^{23, 24}. The Unified Subsonic and Supersonic Aerodynamic analysis (USSAERO) code is a linear aerodynamics panel code that has incorporated a symmetrical singularity method to provide surface pressure distributions on a fuselage and wings in subsonic and supersonic flow. This method extends the range of application of the program to include the analysis of multiple engine nacelles or finned external stores. In addition, nonlinear compressibility effects in high subsonic and supersonic flows are approximated by using a correction based on the local Mach number at panel control points.

4.0 References

1. Sistla, R., Dovi, A. R., Su, P., and Shanmugasundaram, R.: "Aircraft Design Problem Implementation Under the Common Object Request Broker Architecture." *40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, St. Louis, MO, 1999, pp. 1296-1305B.
2. Sistla, R., Dovi, A. R., and Su, P.: "A Distributed Heterogeneous Computing Environment for Multidisciplinary Design & Analysis of Aerospace Vehicles." *5th National Symposium on Large-Scale Analysis, Design and Intelligent Synthesis Environments*, Oct 12-15, 1999, Williamsburg, VA.
3. Weston, R. P., Townsend, J. C., Eidson, T. M., and Gates, R. L.: "A Distributed Computing Environment for Multidisciplinary Design." *5th AIAA/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Panama City Beach, FL, AIAA-94-4372-CP, Sept. 1994, pp. 1091-1097.
4. Krishnan, R., Sistla, R., and Dovi, A. R.: "High-Speed Civil Transport Design Using FIDO," NASA CR-1999-209693, Oct. 1999.
5. Engineous Software, Inc.: *iSIGHT Developer's Guide, Version 3.1*, Morrisville, NC, Apr. 1998.
6. Engineous Software, Inc.: *iSIGHT Advanced Designer's Guide, Version 3.1*, Morrisville, NC, Apr. 1998.
7. Engineous Software, Inc.: *iSIGHT Designer's Guide, Version 3.1*, Morrisville, NC, Apr. 1998.
8. Vinoski, S.: "CORBA: Integrating Diverse Applications Within Distributed Heterogeneous Environments." *IEEE Communications Magazine*, Vol. 14, No. 2, 1997, pp. 46-55.
9. Samareh, J. A.: "A Novel Shape Parameterization Approach." NASA / TM-1999-209116, May 1999.
10. Mitchell, P. M.: "Advanced Finite Element Weight Estimation Process on the High Speed Civil Transport." SAWE Paper No. 2169, Presented at the 52nd Annual Conference of the Society of Allied Weight Engineers, Inc. May 24-25, 1993, Biloxi, MS.
11. Roskam, J., and Lan, C.-T. E.: *Airplane Aerodynamics and Performance*. Design, Analysis, and Research Corporation (DARcorporation), Lawrence, KS, 1997.
12. Roskam, J.: *Airplane Design, Part I: Preliminary Sizing of Airplanes*. Design, Analysis, and Research Corporation (DARcorporation), Lawrence, KS, 1997.
13. Nicolai, L. M.: *Fundamentals of Aircraft Design*. METS, Inc., Distributor, Xenia, OH, and E. P. Domincone Printing Services, Fairborn, OH, 1975.
14. Haftka, R. T., and Gurdal, Z.: "Elements of Structural Optimization." Third revised and expanded edition, Kluwer Academic Publishers, Boston, 1992, pp. 416-420.
15. Whitney, J. M.: "Structural Analysis of Laminated Anisotropic Plates." Technomic Publishing Company, Inc., Lancaster, PA, 1987, p. 104.

16. Stroud, W. J., Agranoff, N.: "Minimum-Mass Design of Filamentary Composite Panels Under Combined Loads: Design Procedure Based on Simplified Buckling Equations." NASA TN D-8257, 1976, Table III.
17. Dennis, J. E., and Lewis, R. M.: "Problem Formulations and Other Optimization Issues in Multidisciplinary Optimization." Center for Research on Parallel Computation, Rice University, CRPC-TR94469, Houston, TX, Apr. 1994.
18. Vanderplaats, G. N.: "CONMIN—A Fortran Program for Constrained Function Minimization." NASA TM-62282, Aug. 1973, plus addendum, May 1978.
19. Vanderplaats, G. N.: *GENESIS User's Manual, Version 4.0*. Colorado Springs, CO, 1997.
20. Moore, G. J.: "MSC/NASTRAN Design Sensitivity and Optimization User's Guide." The MacNeal Schwendler Corp., April 1994.
21. Stanley, G., Hurlbut, B., Levit, I., Loden W., Stehlin, B., and Swenson, L.: "Computational Mechanics Testbed with Adaptive Refinement User's Manual." Lockheed Palo Alto Research Laboratory, Feb. 1995.
22. Whetstone, W. D.: EISI-EAL Engineering Analysis Reference Manual. Engineering Information Systems, Inc. July 1983.
23. Woodward, F. A.: "An Improved Method for the Aerodynamic Analysis of Wing-Body-Tail Configurations in Subsonic and Supersonic Flow, Part I -Theory and Application." NASA CR-2228, Part I, May 1973.
24. Woodward, F. A.: "USSAERO Computer Program Development, Versions B and C." NASA CR-3228, April 1980.
25. Krist, S. L., Biedron, R. T., and Rumsey, C. L.: "CFL3D User's Manual (Version 5.0)." NASA TM-208444, June 1998.
26. Bischof, C., Carle, A., Hovland, P., Khademi, P., and Mauer, A.: "ADIFOR 2.0 User's Guide (Revision D)." CRPC-TR95516-S, March 1995, Rev June 1998.
27. McCullers, A.: "Aircraft Configuration Optimization Including Optimized Flight Profiles." *Proceedings of the Symposium on Recent Experiences in Multidisciplinary Analysis and Optimization*, Sobieski, J., ed. NASA CP-2327, Hampton, VA, April 1984, pp. 396-412.
28. Jones, W. T., and Samareh-Abolhassani, J.: "A Grid-Generation System for Multidisciplinary Design Optimization." *Proceedings of the 12th AIAA Computational Fluid Dynamics Conference*, San Diego, CA, 1995, pp. 657-669.
29. Harder, R. L., and Desmarais, R. N.: "Interpolation Using Surface Splines." *Journal of Aircraft*, Vol. 9, No. 2, 1972, pp. 189-191.

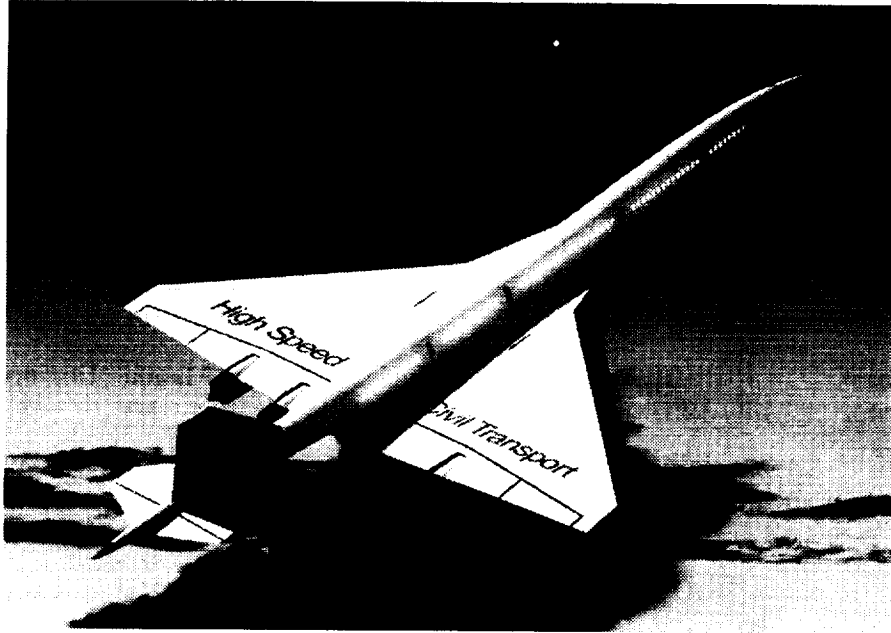
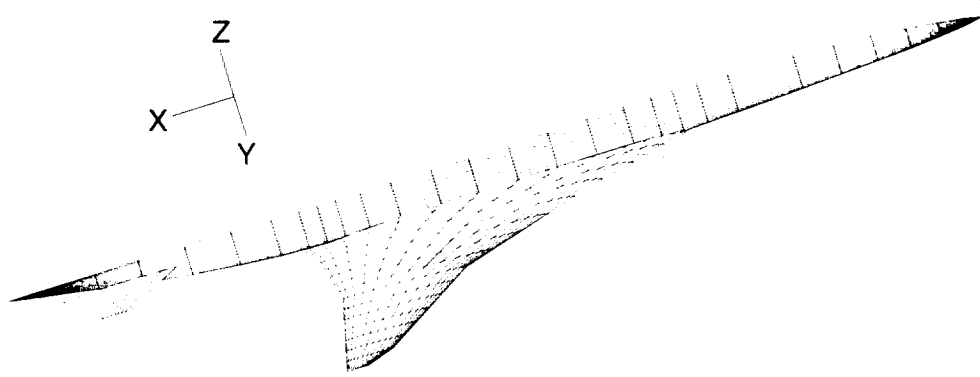
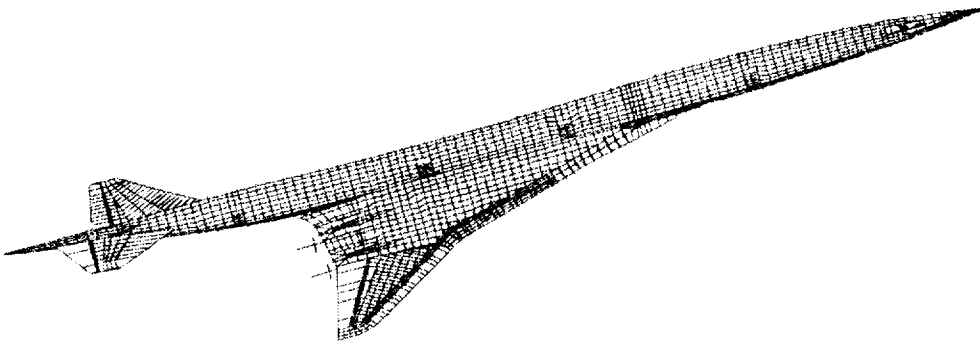


Figure 1. High-Speed Civil Transport.



a) Linear aerodynamics grid



b) Finite element model

Figure 2. Baseline HSCT4.0 Model.

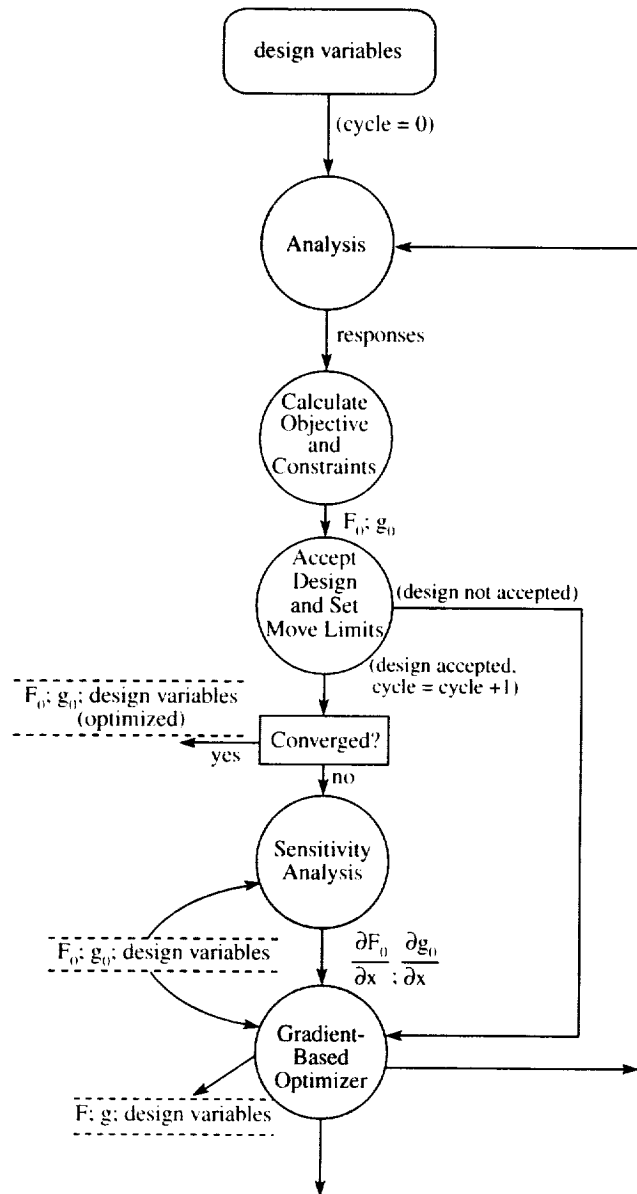


Figure 3. Optimization.

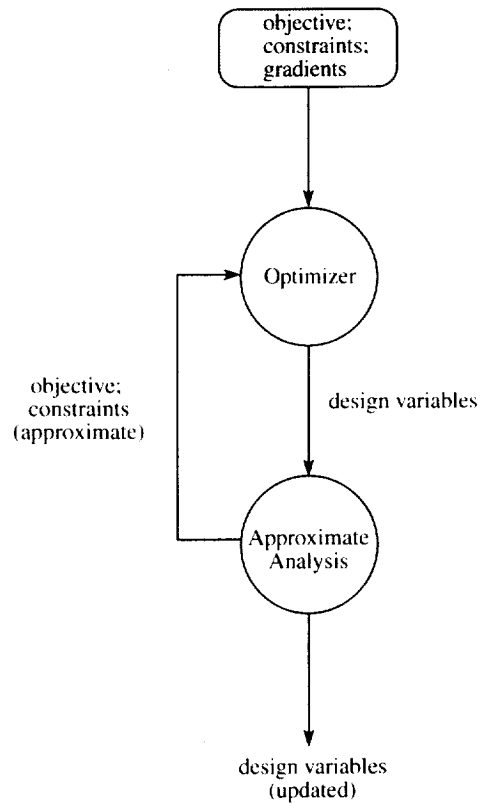


Figure 4. Gradient-Based Optimizer.

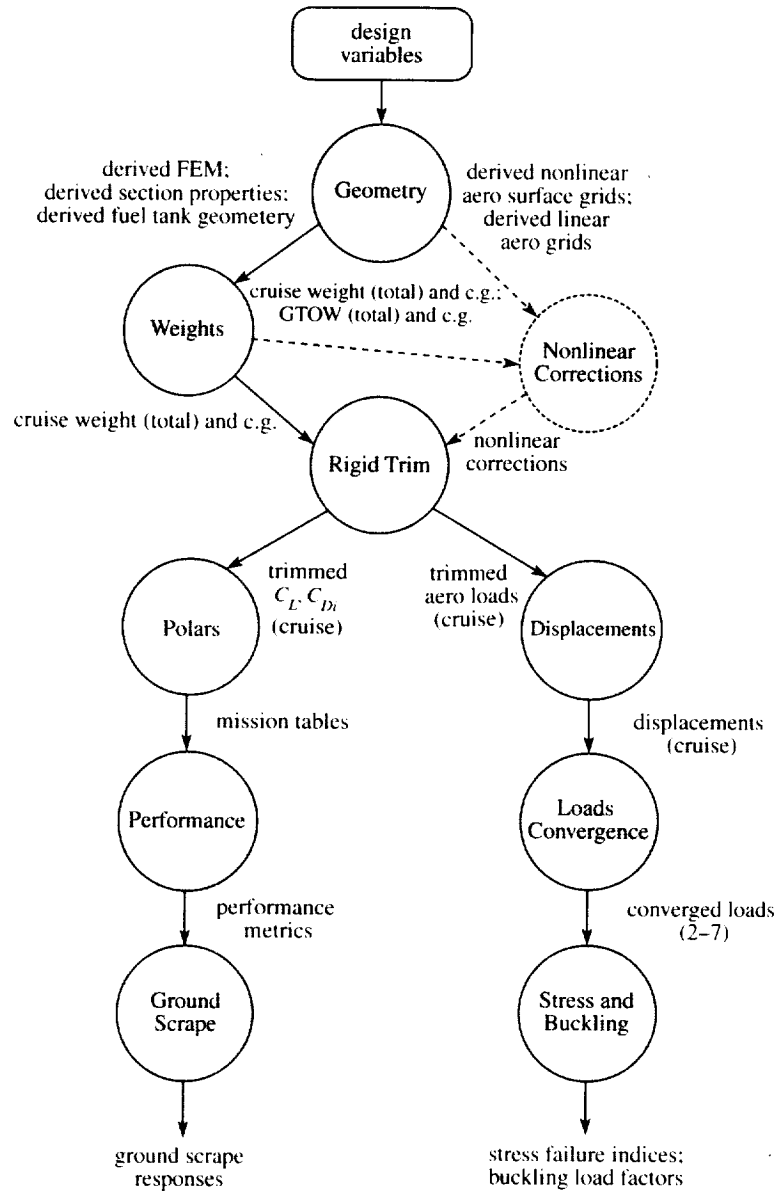


Figure 5. Analysis.

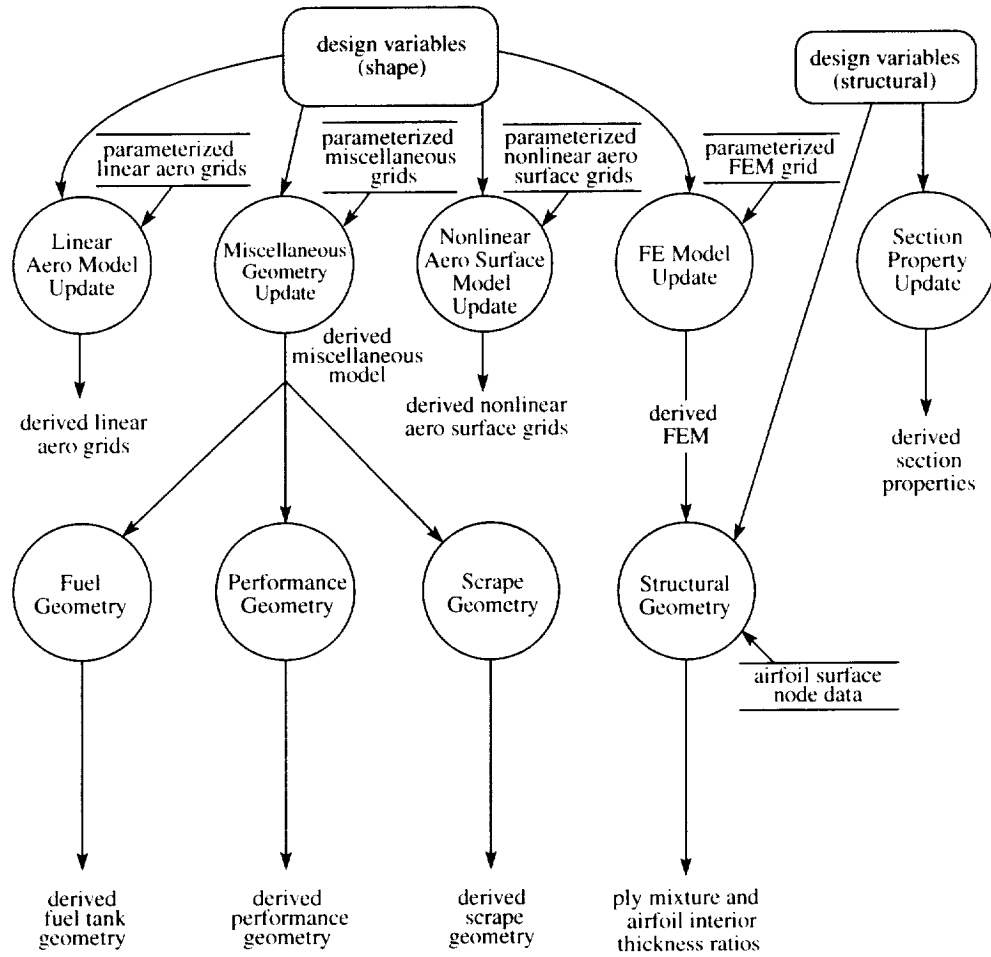


Figure 6. Geometry.

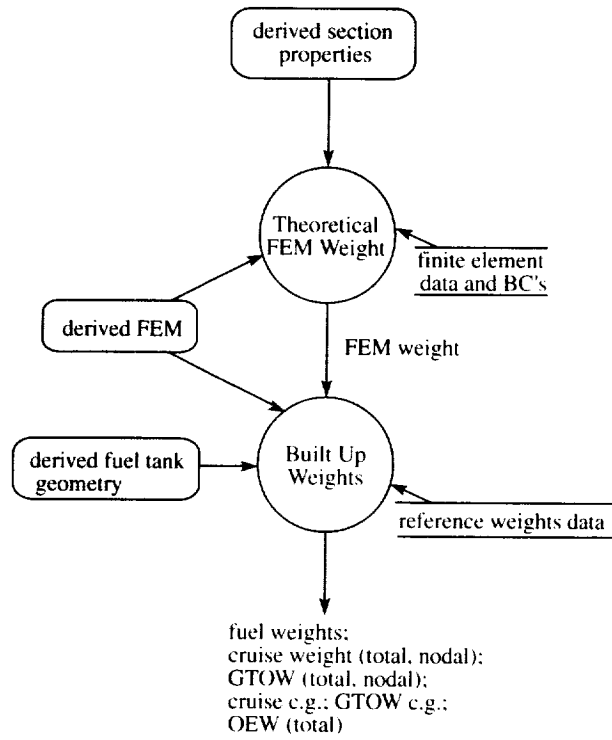
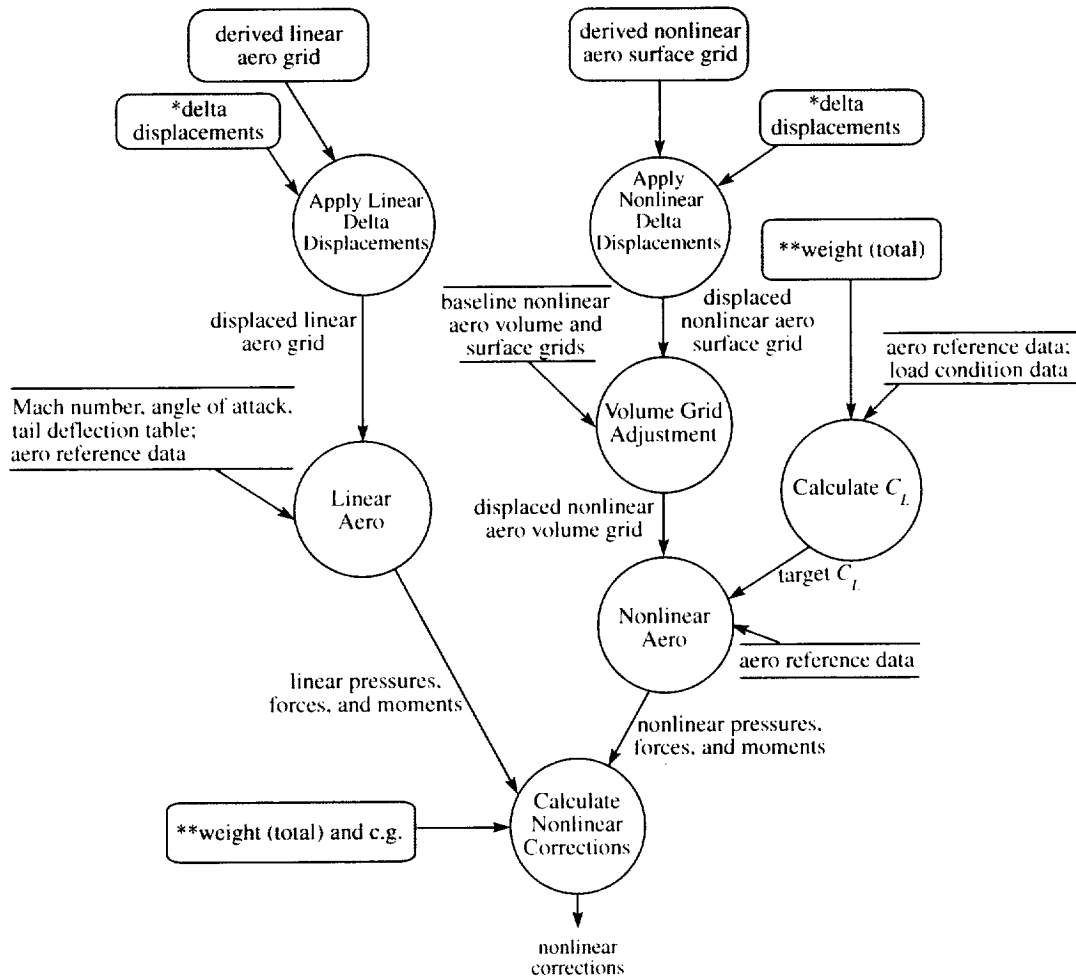


Figure 7. Weights.

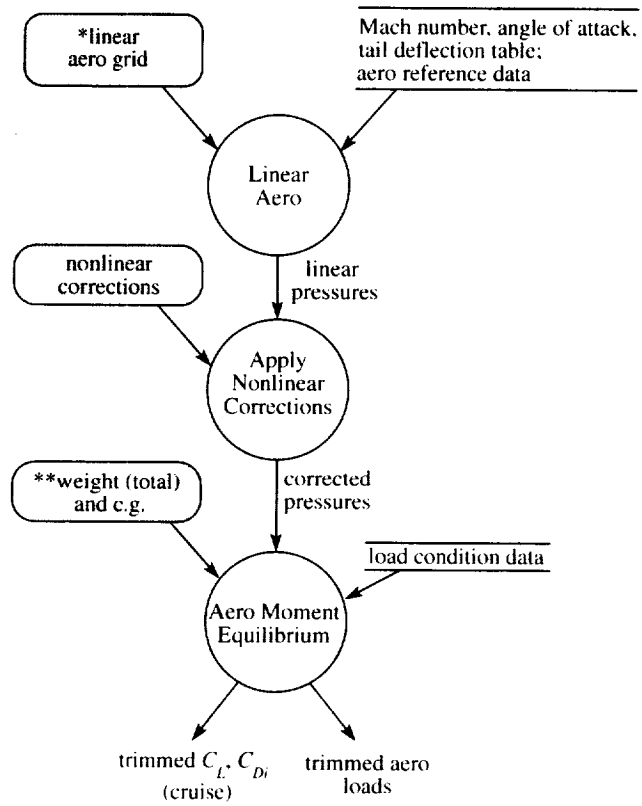


* For design cycle one, delta displacements are zero.

** For load condition 1, weight (total) and c.g. are "cruise".
 For load conditions 2-7, weight (total) and c.g. are GTOW.

The process is performed for load conditions 1-7.

Figure 8. Nonlinear Corrections.



* From *Analysis*, linear aero grid is "derived". From *Loads Convergence*, the grid is "displaced".

** From *Analysis*, weight (total) and c.g. are "cruise" for load condition 1. From *Loads Convergence*, weight (total) and c.g. are GTOW for load conditions 2-7.

In *Loads Convergence*, this process is performed for load conditions 2-7.

Figure 9. Rigid Trim.

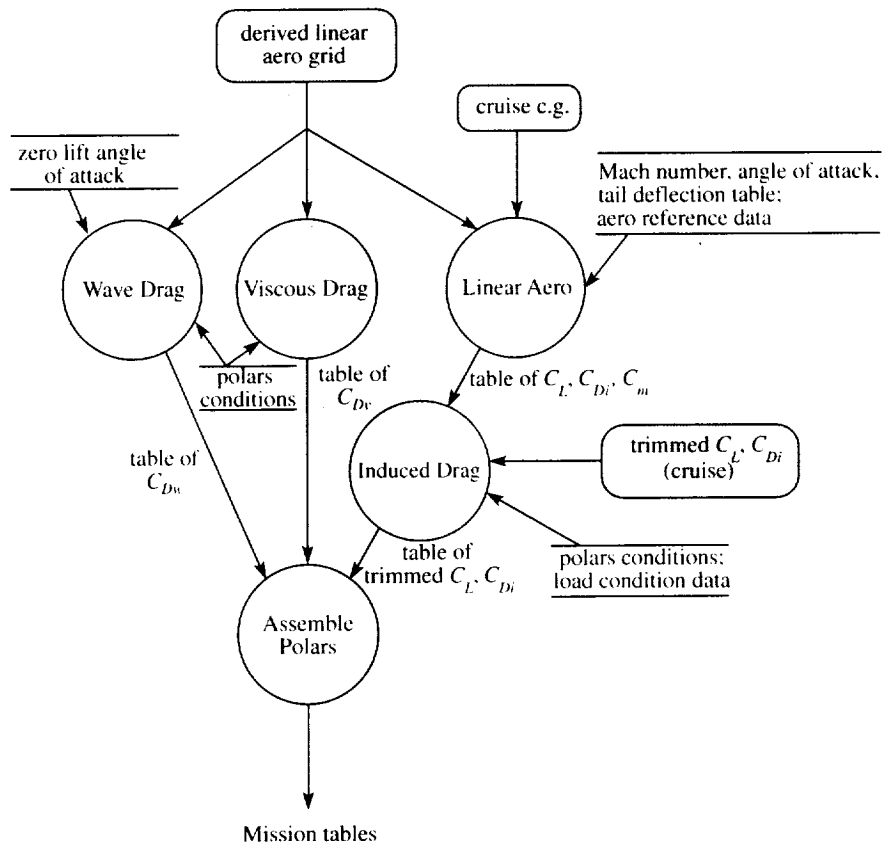


Figure 10. Polars.

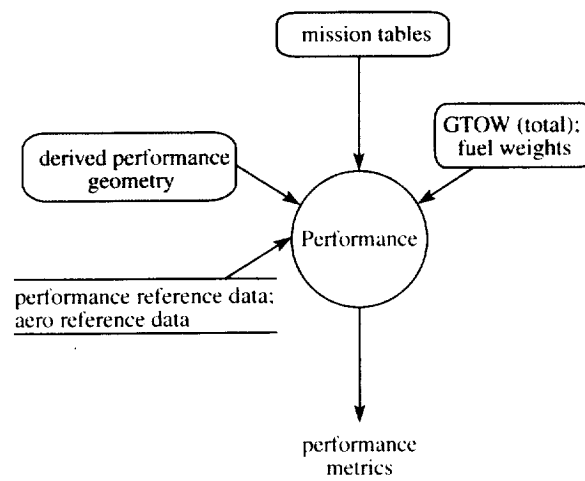


Figure 11. Performance.

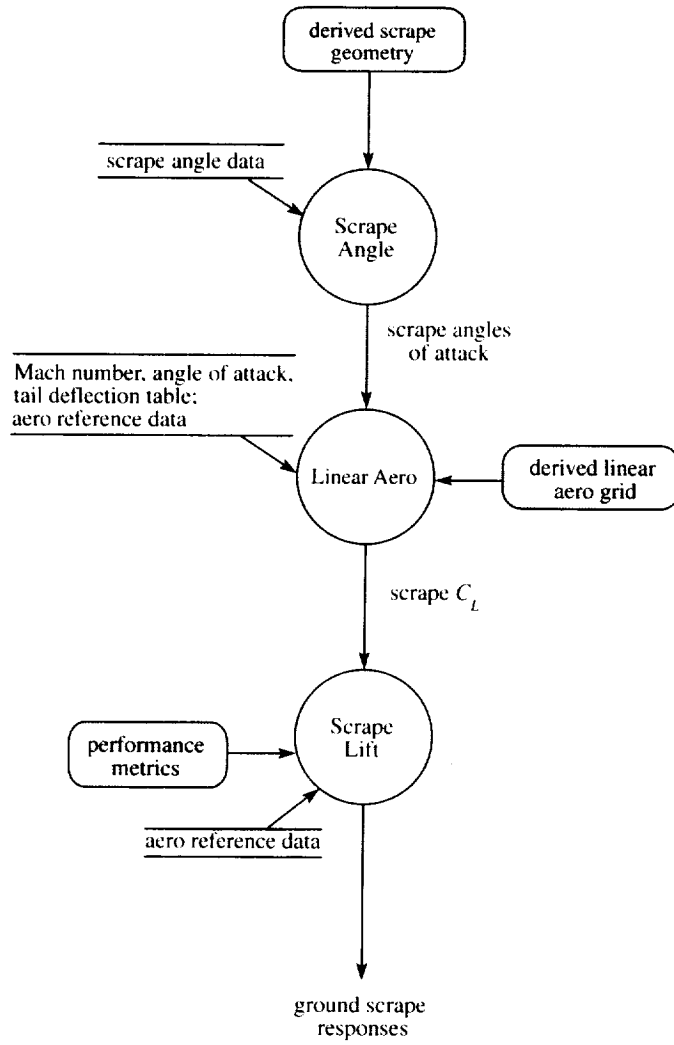
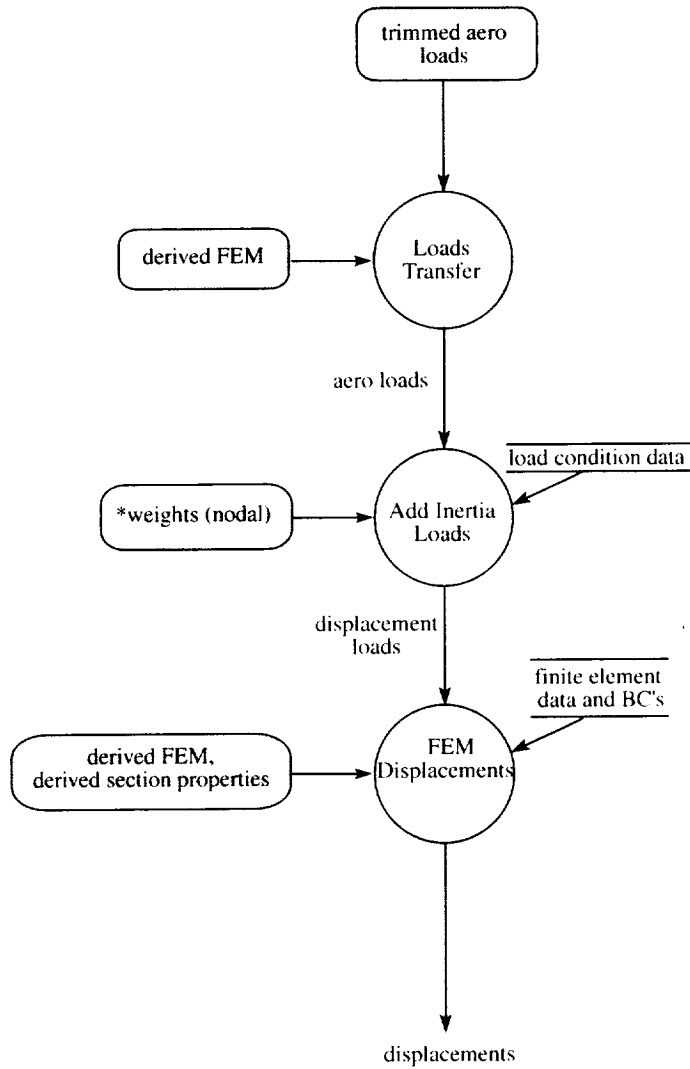
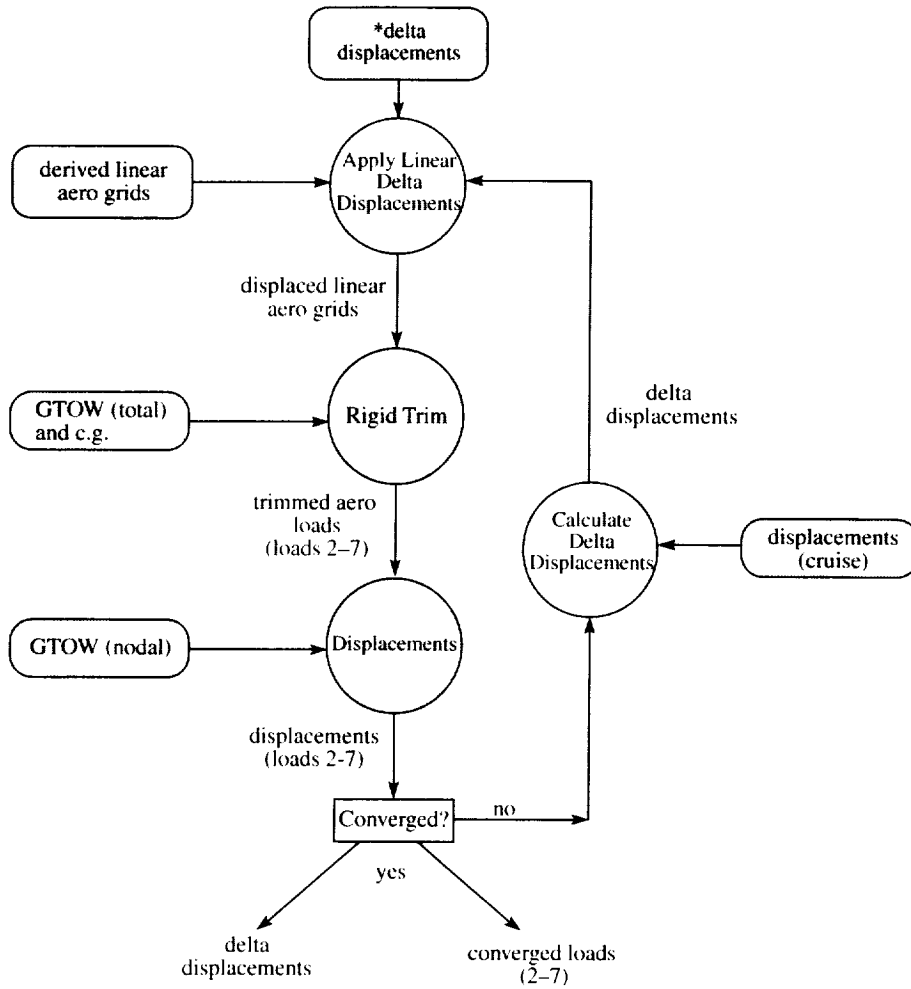


Figure 12. Ground Scrape.



* In *Analysis*, weight (nodal) is "cruise" for load condition 1.
 In *Loads Convergence*, weight (nodal) is "GTOW" for load conditions 2-7.
 In *Loads Convergence*, this process is performed for load conditions 2-7.

Figure 13. Displacements.



* For design cycle one, delta displacements are zero.

Figure 14. Loads Convergence.

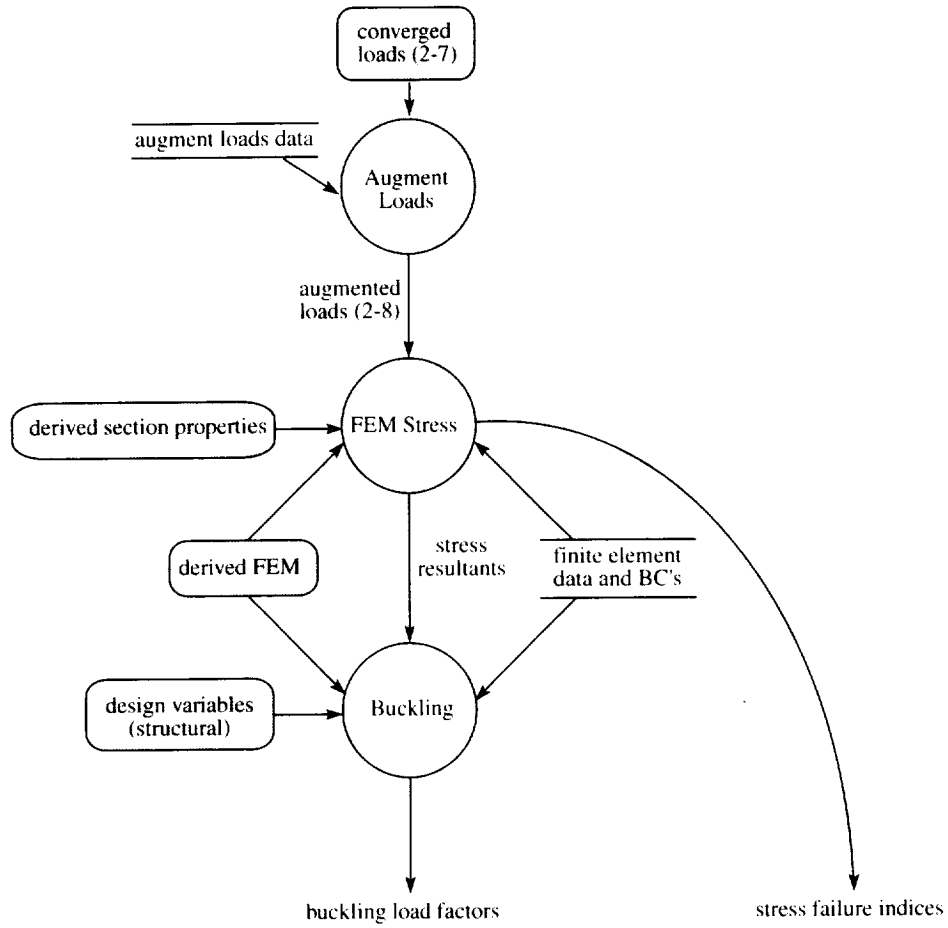


Figure 15. Stress and Buckling.

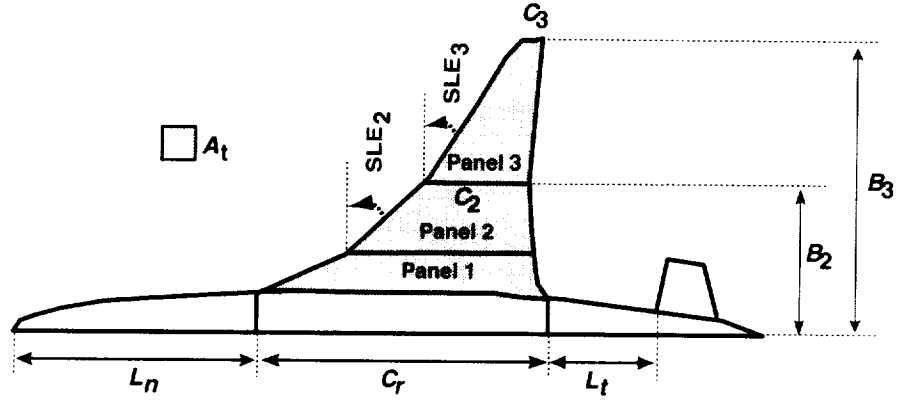


Figure 16. Shape Design Variables - Planform.

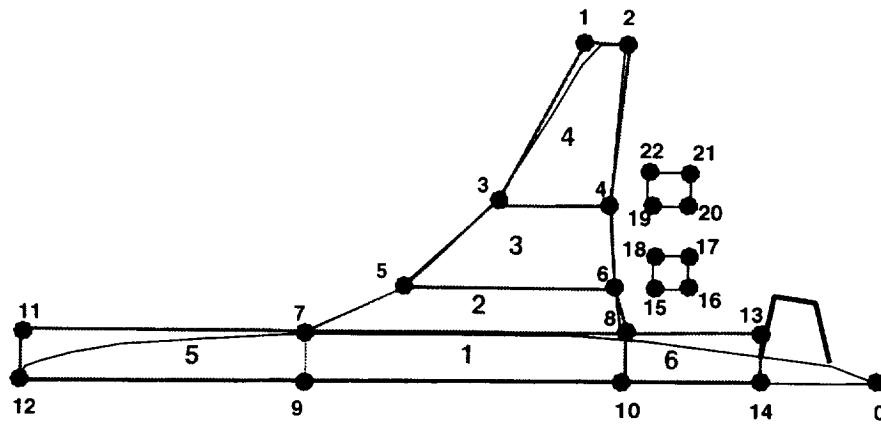
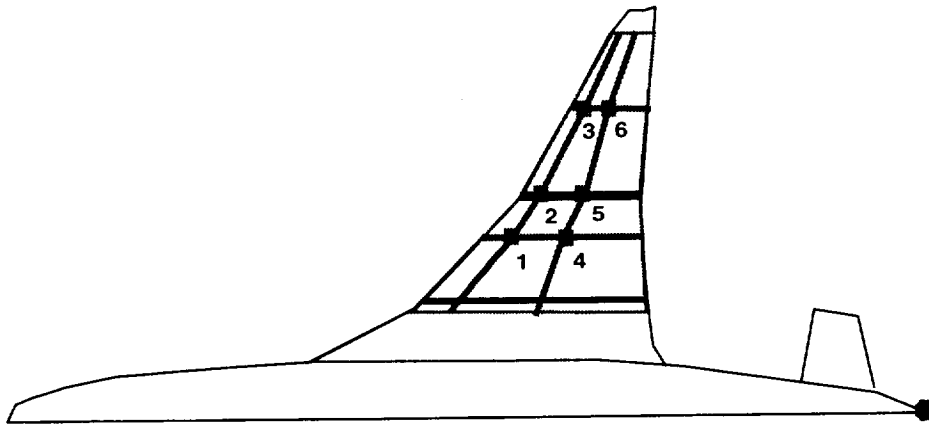
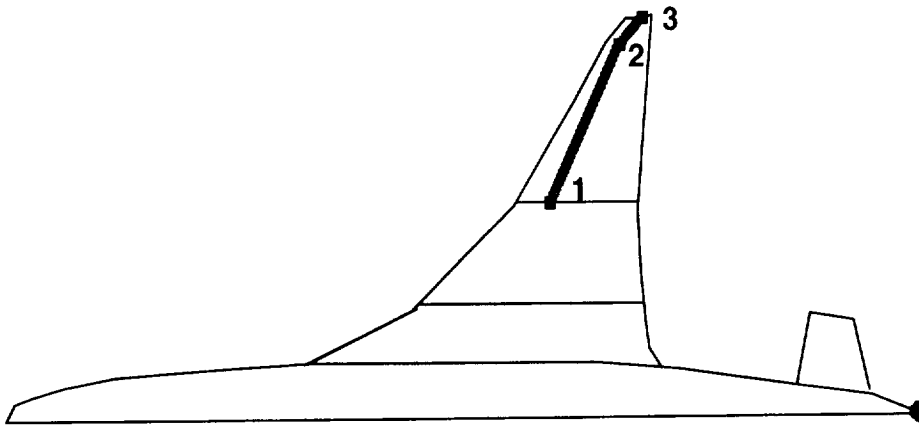


Figure 17. Planform Control Points.



(a) Thickness and Camber Locations



(b) Twist and Shear Locations

Figure 18. Shape Design Variables - Thickness, Camber, Twist, and Shear.

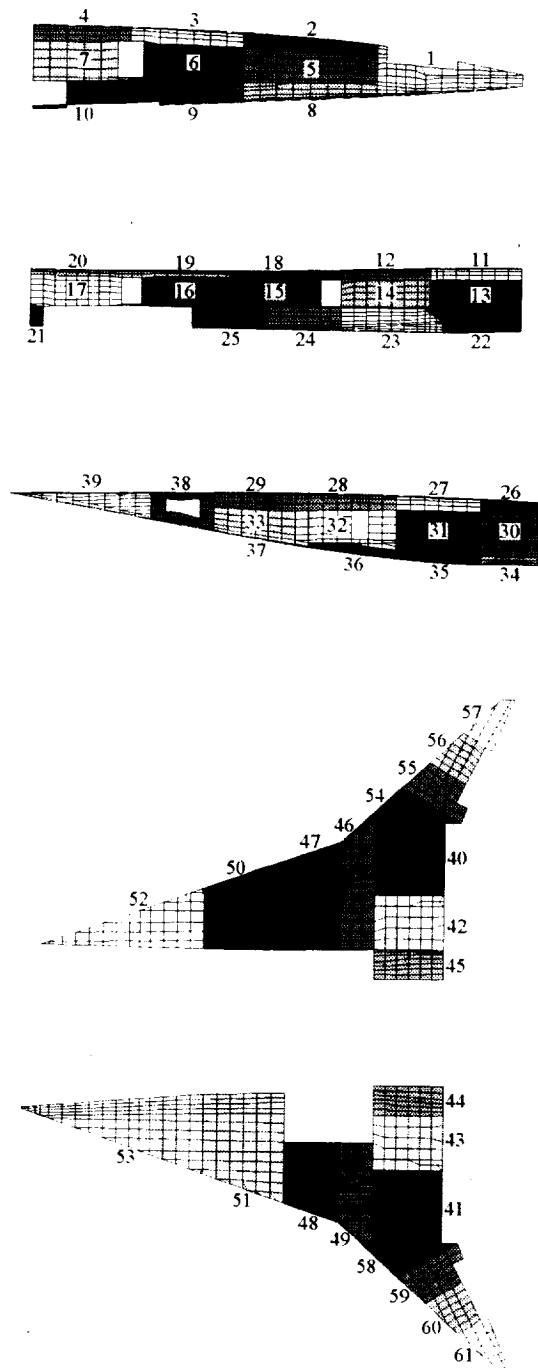


Figure 19. Structural Design Variable Zones.

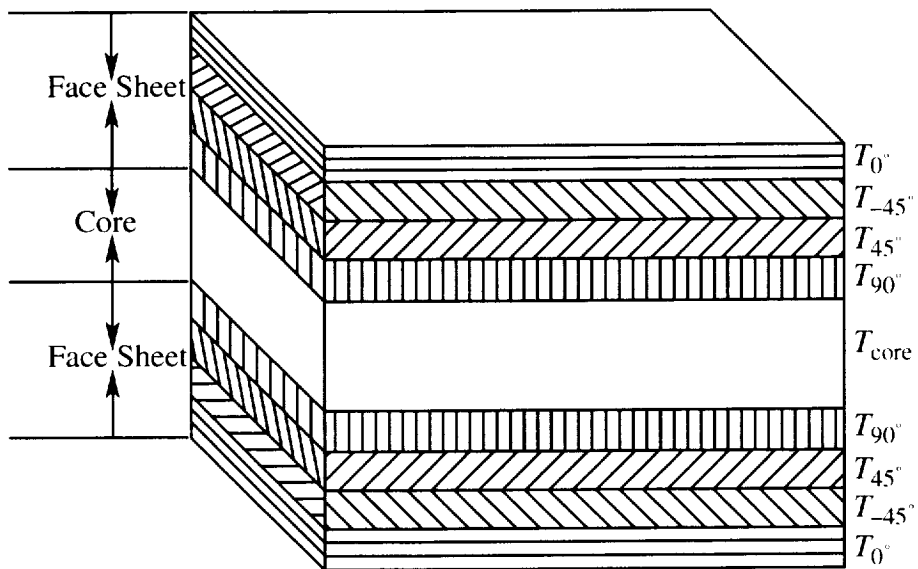


Figure 20. Composite Laminate Stacking Sequence.

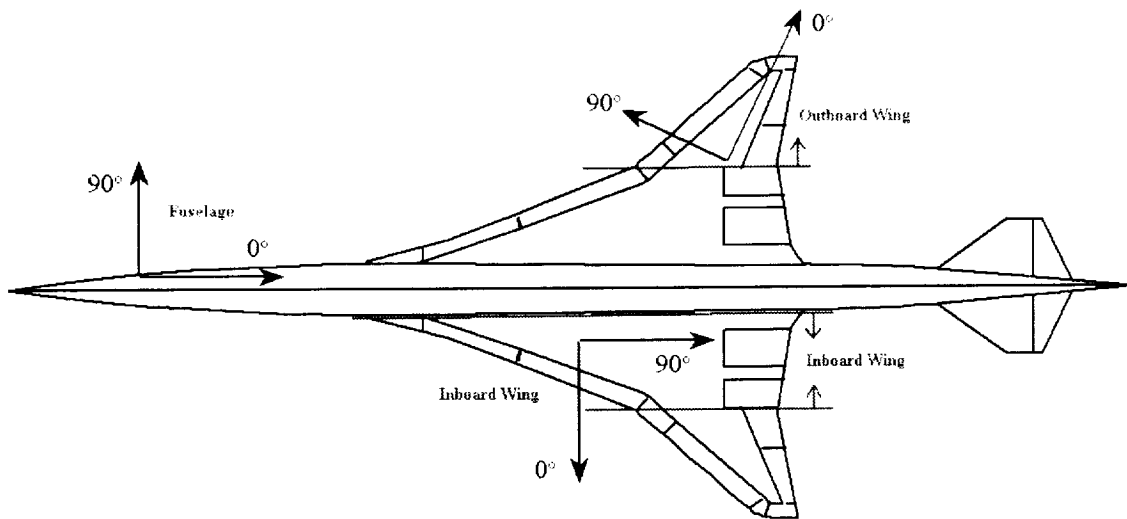


Figure 21. Ply Orientations.

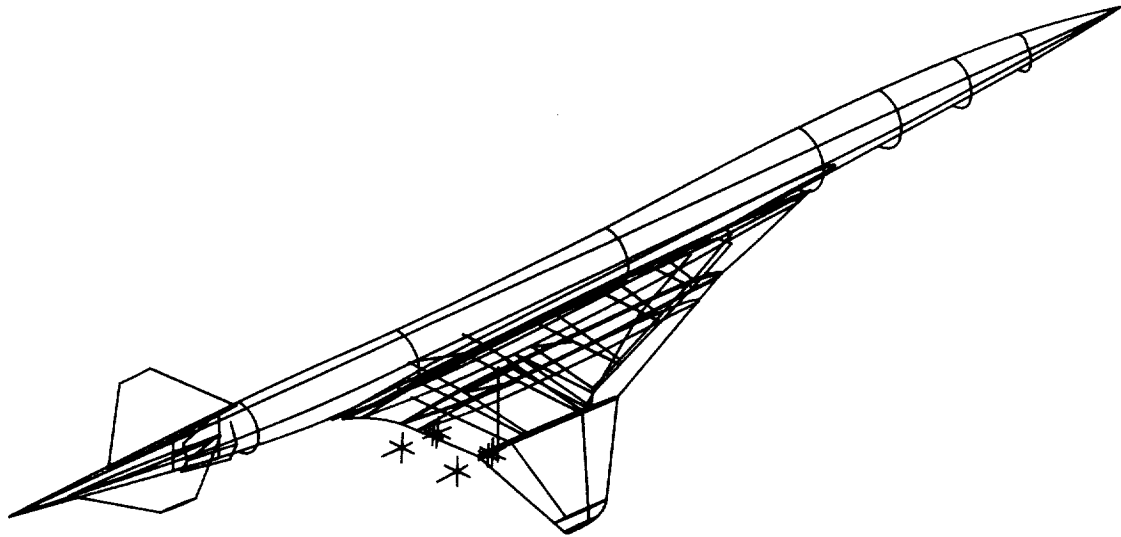


Figure 22. Miscellaneous Geometry.

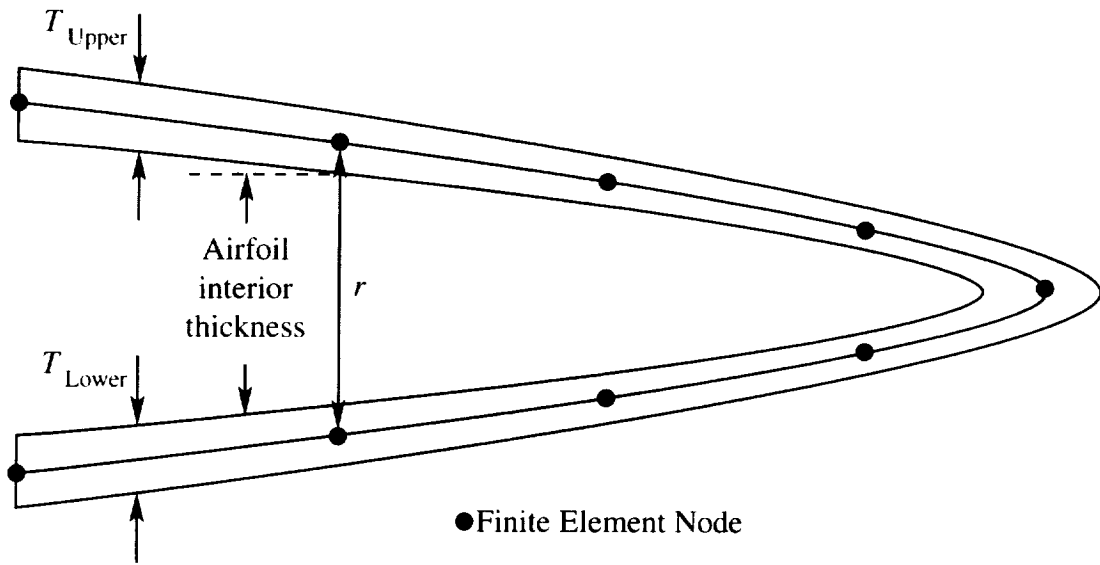


Figure 23. Airfoil Interior Thickness Constraints in Airfoil Cross-section.

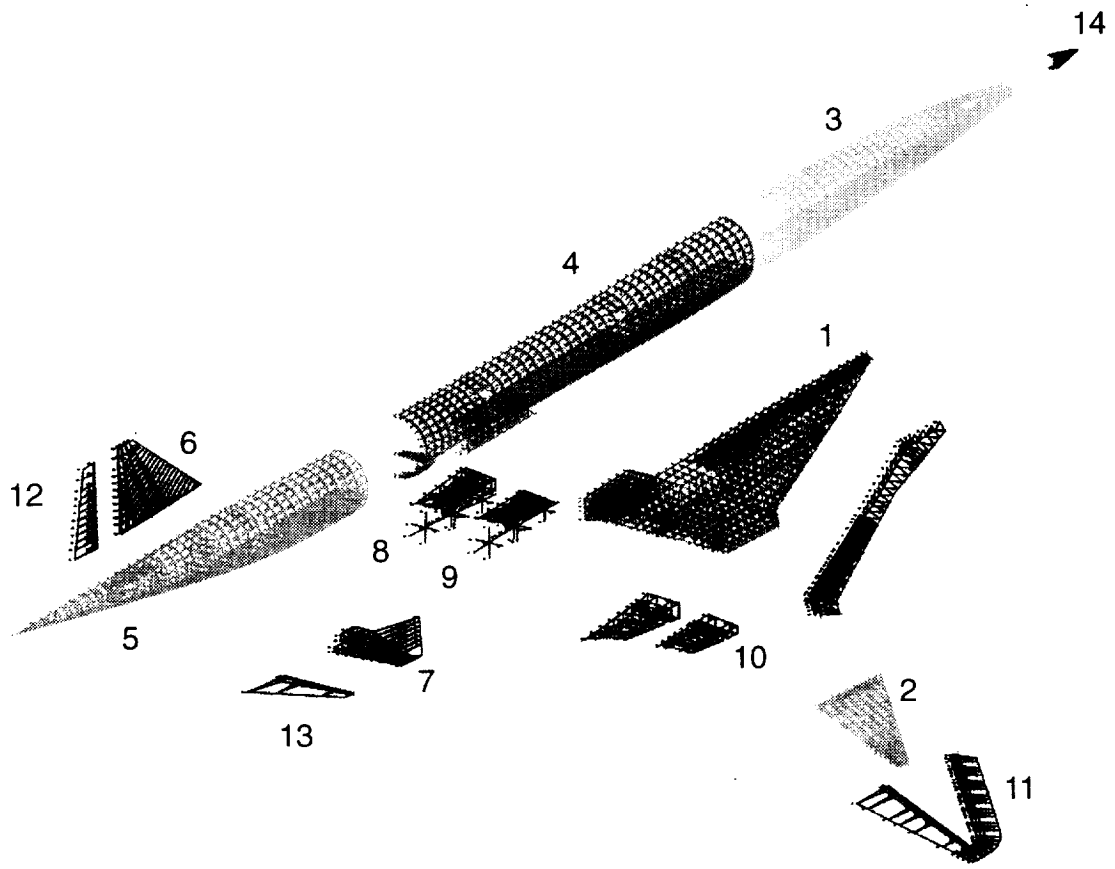


Figure 24. Weights Meshes.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
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 Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE May 2001	3. REPORT TYPE AND DATES COVERED Technical Memorandum		
4. TITLE AND SUBTITLE HSCT4.0 Application Software Requirements Specification			5. FUNDING NUMBERS 725-10-11-01	
6. AUTHOR(S) A. O. Salas, J. L. Walsh, B. H. Mason, R. P. Weston, J. C. Townsend, J. A. Samareh, L. L. Green				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Langley Research Center Hampton, VA 23681-2199			8. PERFORMING ORGANIZATION REPORT NUMBER L-18072	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA/TM-2001-210867	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified--Unlimited Subject Category: 05 Distribution: Nonstandard Availability: NASA CASI (301) 621-0390			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) The software requirements for the High Performance Computing and Communication Program High Speed Civil Transport application project, referred to as HSCT4.0, are described. The objective of the HSCT4.0 application project is to demonstrate the application of high-performance computing techniques to the problem of multidisciplinary design optimization of a supersonic transport configuration, using high-fidelity analysis simulations. Descriptions of the various functions (and the relationships among them) that make up the multidisciplinary application as well as the constraints on the software design are provided. This document serves to establish an agreement between the suppliers and the customer as to what the HSCT4.0 application should do and provides to the software developers the information necessary to design and implement the system.				
14. SUBJECT TERMS Software requirements, High Speed Civil Transport, High performance computing, Multidisciplinary design optimization			15. NUMBER OF PAGES 77	
			16. PRICE CODE A05	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

