Software Requirements Specification

for the

MDO Application:

Aircraft Sizing

Revision 12-19-95

CS5984/CS4984

Independent Verification and Validation

1. Application Scope

The MDO application described below represents the vehicle through which we will explore the use of Independent Verification and Validation (IV&V) techniques applicable throughout the software development lifecycle. To promote additional relevance to NASA's interests, the selected MDO application reflects an aeronautical flavor. More specifically, the application computes performance, aerodynamic and weight characteristics based on design parameters such as wing area, cruise altitude, mach speed, and various weight factors. The computational process is iterative and stresses the convergence to an ideal aircraft weight.

As detailed later in this document, three distinct data sets are computed:

- Performance characteristics:
  - Takeoff distance, and
  - Landing distance,
- Aerodynamic characteristics:
  - Cruise lift coefficient,
  - Cruise drag coefficient, and
  - Rotational velocity,

and finally

- Weight characteristics:
  - Weight of the fuel in the cruise segment of the flight,
  - Weight of the fuel in the climb segment of the flight,
  - Wing weight,
  - Engine weight,
  - A "fixed weight" overhead, and
  - A cargo weight.

2. Computation Overview

The computational procedure for computing aircraft performance, aerodynamic and weight characteristics is a convergent one that employs an iterative process. An initial set of data ($DSet_0$) is provided and contains two classes of data items: (a) design parameters that remain fixed throughout the convergent process, and (b) a set of weights that change with each iterative computation. An initial data set, $DSet_0$, is used during the first set of computations (or iteration) and results in an initial estimate of performance, aerodynamic and weight characteristics. In turn, a subset of the newly computed weight estimates is used to update corresponding elements in $DSet_0$, yielding a new input data set ($DSet_1$) for the subsequent iteration. Hence, each successive iteration (a) computes a more precise (or refined) set of performance, aerodynamic and weight characteristics, and (b) provides refined weight measures for updating the data set to be used in the next iteration.

Effectively, for the i^{th} set of iterative computations $(1 \le i)$,

$$\{Performance\ Characteristics\}_i = f_{Perf}\left(DSet_{i-1}\right)$$

$$\{Aerodynamic\ Characteristics\}_i = f_{Aero}\left(DSet_{i-1}\right)$$

$$\{Weight\ Characteristics\}_i = f_{Wght}\left(DSet_{i-1}\right),$$

where

$$DSet_0 \equiv \left\{\langle Design\ Parameters\rangle_{Fixed}, \langle Weights\rangle_{Initial}\right\}$$

$$DSet_i \equiv \left\{\langle Design\ Parameters\rangle_{Fixed}, \langle Weights\rangle_{i-1}\right\} \text{ and } \langle Weights\rangle_{i-1} \subset \langle Weight\ Characteristics\rangle_{i-1}$$

2.1 Performance Characteristics

Two performance characteristics are computed: *Takeoff distance* (S_{to}) and *Landing distance* (S_{ldg}). In computing the takeoff distance four components are considered: distance covered on the ground(S_g), during rotation (S_r), during transition (S_t) and during climbout (S_c). Landing distance is computed as the sum of two components: the air distance from an altitude of 50 feet to the touchdown point (S_{air}), and the subsequent ground roll (S_{lg}). The formula for computing S_{lg} factors in antiskid braking, thrust reversing, ground spoilers and speed brakes.

2.2 Aerodynamic Characteristics

Three aerodynamic characteristics are computed: *Cruise lift coefficient* (C_{lift}), *Rotation velocity* in knots ($V_{rot_{knots}}$) and the *Cruise drag coefficient* (C_{drag}). The cruise lift coefficient is computed as a function of the dynamic pressure for a given cruise altitude and the weight of the aircraft once cruise altitude has been attained. Rotation velocity is expressed as a function of the maximum cruise lift for the aircraft, the wing area and the total takeoff weight. The cruise drag coefficient is computed as a sum of three components: the cruise drag coefficient under zero-lift conditions (C_{D_0}), the transonic wave drag coefficient ($C_{D_{wave}}$), and C_{D_i} - which is a coefficient reflecting the impact of the Oswald Efficiency factor on cruise lift coefficient.

2.3 Weight Characteristics

Within this application the aircraft gross takeoff weight, W_{to}, will play a key role in the iterative computational process. W_{to} is defined as the sum of six distinct components: The wing weight (W_{wing}), the fuel weight during the cruise segment (W_{fuel}), the fuel weight apportioned to the climb (ascent) segment of the flight (W_{fclm}), the combined weight for all engines (W_{eng_n}), a "fixed" weight component associated with the aircraft (W_{fixed}), and a fixed cargo weight (W_{cargo}). Among these weights, only the W_{eng_n} and W_{cargo} remain constant throughout all iterative computations. W_{fixed} and W_{fclm} are computed based on a percentage of the total gross weight (derived from the immediately preceding iteration). W_{fuel} is computed as a function of the weight of the aircraft during the cruise segment as well as the cruise lift and cruise drag coefficients. Like W_{to}, W_{fuel} and W_{fclm} both play crucial roles in the iterative process. More specifically, W_{to} and its two components, W_{fuel} and W_{fclm}, are precisely the three weight components used to update $DSet_{i-1}$, and thereby, forming the new set of input parameters, $Dset_i$,for the subsequent iterative computation.

3. Requirements

3.1 External Input Data Requirements

The MDO application shall read its input parameters from a single data input file containing the following six (6) logical subsections:

(1) Control Flags,
(2) Initial Points,
(3) Parametric Mach Number Setup
(4) Baseline Weights (Fixed / Fract / Comput),
(5) Propulsion Variables, and
(6) Aerodynamic/Geometric Variables.

Note: All data item values from the input file shall remain constant throughout the computational processes with exception to those items included in the *Baseline Weights* subsection.

The physical structure of the input file shall have the following properties. Each data item is placed on a separate line. That data item may be followed on the same line by descriptive text. One or more spaces separate the descriptive text from the data item. This descriptive text is ignored for computational purposes. There shall be no more than 80 characters of text, including the data item, per line.

27 data items must be present in the input data file. If too few or too many data items are detected, an appropriate error message is printed and the program terminates. Data items with stipulated range values shall be validated. An appropriate error message shall be printed on data item values found to be out of range, the program shall stop execution after displaying the error message.

The filename of the input file shall be “mdo.inp”. All characters of the filename shall be non-capitalized to allow for portability between Unix and PC machines.

To make it easier to refer to individual data items later in the requirements document, each data item in the following description of the logical subsections will have an abbreviated neunomic associated with it. A sample input file with the logical subsections present, as well as an actual physical sample input file that the application should be able to process are displayed in Appendicies B & C.

Data types indicated as double precision real shall be treat as the highest precision real data type available within the ANSI C language and platform the program is implemented on.

3.1.1 Subsection 1: Control Flag

The control flag subsection includes one control flag that should be changeable by the user through the input file and not hard wired in the application. This control flag may be used to indicate if certain outputs shall be produced by the application.

Data Item 1: Detailed Weight Convergence Print Flag (IPTDET), indicates if the program is to print intermediate aircraft characteristic values during the iterative (convergent) process. If this flag is set to one (1) output of all relevant data (described in the output section) shall be produced after each iteration. If set to zero (0) no intermediate values are printed.

Data Type: Integer

Data Range: 1 for on (print intermediate values), 0 for off (do not print intermediate values).

3.1.2 *Subsection 2: Parametric Mach Number Setup*

The data items processed by this application as well as the results produced depend to a large extend on the initial mach parameter provided. In this subsection an initial mach (speed) value is provided along with an incremental stepping value and a value for the number of mach increment iterations (counter). The combination of these three data items allows the application to produce several airplane sizings, starting with the initial mach value, and for mach values computed by adding the product of the current mach iteration and the incremental mach stepping value to the initial mach value.

The Parametric Mach Number Setup subsection has 3 data item values, each provided on a separate line.

Data Item 1: Number of Mach Iterations ($MInc_{Cnt}$).

Data Type: Integer

Data Item 2: Initial Mach Value (Mach).

Data Type: Double Precision Real

Data Range: 0 - 1

Units: Dimensionless

Data Item 3: Mach Value Increment per Iteration (Inc_{mach}).

Data Type: Double Precision Real

3.1.3 *Subsection 3: Initial Points*

A set of seven (7) initial design parameters are required to perform the sizing of the aircraft.

Data Item 1: Aspect Ratio (AR)

Data Type: Double Precision Real
Data Range: 0 - 100,000
Units: Dimensionless

Data Item 2: Wing Area (S_W)

Data Type: Double Precision Real
Data Range: 10 - 100,000
Units: ft^2

Data Item 3: Cruise Altitude (Height)
Data Type: Double Precision Real
Data Range: 0 - 100,000
Units: ft

Data Item 4: Mid-Chord Wing Sweep ($Sweep_{deg}$)
Data Type: Double Precision Real
Data Range: 0 - 85
Units: degrees

Data Item 5: Wing Thickness to Chord Ratio (TC)
Data Type: Double Precision Real
Data Range: 0 - 1
Units: Dimensionless

Data Item 6: Wing Taper Ratio (TPR)
Data Type: Double Precision Real
Data Range: 0 - 1
Units: Dimensionless

Data Item 7: Flight Range (Range)
Data Type: Double Precision Real
Data Range: 0 - 100,000
Units: nm

3.1.4 Subsection 4: Baseline Weights

The main objective of this application is to find the ideal gross takeoff weight of an aircraft. As described, this is performed by iteratively sizing several components of the aircraft to yield the best size of components to weight ratio. To start the initial sizing computations reference weights have to be provided as input. (**Note:** The choice of these reference weights is not totally arbitrary since a “wrong” choice may lead to an early exit from the application if it is impossible to size the aircraft for a given set of reference weights.) Within the first iteration of the sizing process it can be expected that new weights are calculated for the different weight components that differ greatly from the reference weights provided by the input file. **Note:** For each new mach value iteration the original reference weights will have to be used since the newly computed weights may not be usable for a new mach value.

Data Item 1: Reference Gross Takeoff Weight ($W_{to_{ref}}$), is used in the first iteration of the sizing computation. A new takeoff weight is computed with every iteration until the computed gross takeoff weights converge.

Data Type: Double Precision Real
Data Range: 0 - 1,000,000
Units: lb

Data Item 2: Reference Fuel Weight ($W_{fuel_{ref}}$), is also used for the first iterative computation. A fraction of fuel weight is allocated to takeoff and landing of the aircraft with the remainder used as cruise fuel weight. (Note: Landing fuel weight computation performed implicitly through formulas, not specifically identified in SRS)

Data Type: Double Precision Real
Data Range: 0 - 1,000,000
Units: lb

Data Item 3: Fixed Cargo Weight (W_{cargo}), is a fixed weight of cargo the aircraft should be able to carry.

Data Type: Double Precision Real
Data Range: 0 - 1,000,000
Units: lb

Data Item 4: Engine Weight per Engine (W_{eng}).

Data Type: Double Precision Real
Data Range: 0 - 1,000,000
Units: lb

Data Item 5: Climb Fuel Fraction (F_{clm}), indicates that fraction of fuel weight used to climb to the cruising altitude.

Data Type: Double Precision Real
Data Range: 0 - 1
Units: Dimensionless

Data Item 6: Aircraft Fixed Weight Fraction (C_{fix}), used to computed a fraction of the gross takeoff weight with every iteration.

Data Type: Double Precision Real
Data Range: 0 - 1
Units: Dimensionless

Data Item 7: Structural Load Factor for Wing Weight (N), is a constant used in computing the wing weight.

Data Type: Double Precision Real
Data Range: 1 - 10
Units: Dimensionless

3.1.5 Subsection 5: Propulsion Variables

The next set of parameters deals with the engines of the aircraft. The amount of fuel (fuel weight) required for an airplane to fly a given range is related to the amount of fuel burned by the engines, the thrust produced by the engines and the number of engines.

Data Item 1: Maximum Engine Thrust per Engine (T_{max}).

Data Type: Double Precision Real
Data Range: 0 - 500,000
Units: lb

Data Item 2: Cruise Fuel Consumption Fraction per Engine (sfc).

Data Type: Double Precision Real
Data Range: 0 - 5
Units: $\frac{lb}{\left(\frac{lb}{hr}\right)}$

Data Item 3: Number of Engines (N_{eng}).

Data Type: Integer
Data Range: 0 - 100

Units: None

3.1.6 Subsection 6: Aerodynamic/Geometric Variables

The aerodynamics and flying abilities of an aircraft are not only dependent on the weights of the aircraft, but also on those areas exposed to airflow. Aircraft engineers call the aircraft areas exposed to airflow the “wetted” area, thinking of all those areas that would get wet if the aircraft was immersed in water. In flight these areas generate either a drag or lift depending on their exposure to airflow. The following parameters are used in the computations of the aerodynamics including cruise lift and cruise drag.

Data Item 1: Maximum Lift Coefficient ($C_{L_{max}}$).

Data Type: Double Precision Real
Data Range: 0 - 5
Units: Dimensionless

Data Item 2: Oswald efficiency factor (E).

Data Type: Double Precision Real
Data Range: 0 - 1
Units: Dimensionless

Data Item 3: Fuselage Wetted Area (S_{fuse}).

Data Type: Double Precision Real
Data Range: 10 - 100,000
Units: ft^2

Data Item 4: Horizontal Tail Wetted Area (S_{tail}).

Data Type: Double Precision Real
Data Range: 10 - 100,000
Units: ft^2

Data Item 5: Vertical Tail Wetted Area (S_{vtail}).

Data Type: Double Precision Real
Data Range: 10 - 100,000
Units: ft^2

Data Item 6: Pod Wetted Area (S_{pod}).

Data Type: Double Precision Real
Data Range: 10 - 100,000
Units: ft^2

3.2 Terminating the Iterative Process

As implied in Sections 1 and 2, given a set of input parameters, the MDO application employs an iterative computational process in an attempt to converge to an ideal aircraft weight. During each iteration specific performance, aerodynamic and weight characteristics are recomputed based on input derived from the immediately preceding iterative computation.

The iterative process terminates when either of three (3) following conditions occur:

- 500 iterations have been performed $\left(effectively, 1 \le i \le 500\right)$,
- $\left|W_{to_{i-1}} - W_{to_i}\right| \le 1.0 \times 10^{-7}$ *lbs.* (the criteria for convergence), or
- $W_{to_i} \ge 9{,}000{,}000$ *lbs.* (the maximum allowed aircraft gross takeoff weight).

3.3 Incorporating Mach (Speed) Increments

The convergent process assumes an initial set of *fixed* design parameters, and an initial estimate of the total weight of the aircraft (this weight changes, of course, as the iterative process is employed). Clearly, one can change any of the initial input parameters and re-invoke the application to determine additional ideal aircraft weights. What is often the case, however, is that one desires the ideal weight computation over a range of mach (speed) values. Consequently, the MDO application shall

- read an increment count ($MInc_{cnt}$), a mach value (*Mach*) and a mach increment value (Inc_{mach}) and and
- compute the ideal aircraft weight (through the iterative process) over the set of mach values:

  $$mach + \left(j \bullet Inc_{mach}\right), \quad 0 \le j \le MInc_{cnt},$$

  also

  $$mach + \left(j \bullet Inc_{mach}\right) \le 1 \ .$$

3.4 The Computational Process and Formulas.

The following subsections provide the mathematical formulas to compute three (3) sets of aircraft characteristics: performance, aerodynamic and weight. Each set of characteristics can be computed independent of each other. (**Note:** Close scrutiny will reveal that duplicate computations are often performed, and can be avoided if a proper design is implemented. We encourage such a design. Several duplicate computations are noted.) Although not expected, should the program detect a possibility for a division by zero error, an appropriate error message shall be generated, indicating as closely as possible the area in the program in which the error would have occurred. The program shall also detect possible negative values under square root functions (although not expected) and shall generate an appropriate error message, indicating as closely as possible the area in the program in which the error would have occurred. Overflow or underflow errors are not expected and shall not be handled.

We specifically note that total aircraft weight (W_{to}), the weight of the fuel during the climb (W_{fclm}) and cruise (W_{fuel}) segments are recomputed during each iteration and then used as updated values in the immediately successive iterative set of computations. Subsequently, for **iteration$_{\mathbf{i}}$** you will see W_{to}, W_{fclm} and W_{fuel} being referred to as $W_{to_{i-1}}$, $W_{fclm_{i-1}}$ and $W_{fuel_{i-1}}$, indicating that the values being used for these items were computed during **iteration$_{\mathbf{i-1}}$**. For iteration$_1$ the following initial values will be used for W_{to_0}, W_{fuel_0} and W_{fclm_0}:

$$W_{to_0} \equiv W_{to_{ref}} \qquad (W_{to_{ref}} \textit{ is provided as input})$$

$$W_{fuel_0} \equiv W_{fuel_{ref}} \qquad (W_{fuel_{ref}} \textit{ is provided as input})$$

$$W_{fclm_0} \equiv F_{clm} \bullet W_{to_{ref}} \qquad (F_{clm} \textit{ and } W_{to_{ref}} \textit{ are provided as input}).$$

Please note further that, with the exception of W_{to} and W_{fuel} all computational data items shown in **bold face** are input parameters.

3.5 Computing the Performance Characteristics

S_{to} is the takeoff distance (S_{to} is a function of $W_{to_{i-1}}$)

S_{ldg} is the landing distance (S_{ldg} is a function of $W_{fuel_{i-1}}$ and $W_{to_{i-1}}$)

$$S_{to} = S_g + S_r + S_t + S_c$$

where S_g is the ground distance component

S_r is the rotational distance component

S_t is the transition distance component, and

S_c is the climbout distance component

$$S_g = \int_0^{V_{rot_{mph}}} \left\{ \frac{\boldsymbol{W_{to_{i-1}}}}{32.174} \bullet \frac{v}{Thrust - Drag_v - 0.06 \bullet \left(\boldsymbol{W_{to_{i-1}}} - Lift_v\right)} \right\} dv$$

$$V_{rot_{mph}} = 1.1 \bullet \sqrt{\frac{2.0 \bullet \boldsymbol{W_{to_{i-1}}}}{\boldsymbol{C_{L_{max}}} \bullet 0.00273 \bullet \boldsymbol{S_w}}}$$

$$Thrust = 0.95 \bullet \boldsymbol{N_{eng}} \bullet \boldsymbol{T_{max}}$$

$$Drag_v = C_{drag_{to}} \bullet 0.001365 \bullet v^2 \bullet \boldsymbol{S_w}$$

$$C_{drag_{to}} = C_{D_0} + \frac{\left(0.8 \bullet \boldsymbol{C_{L_{max}}}\right)^2}{\boldsymbol{p} \bullet \boldsymbol{AR} \bullet \boldsymbol{E}}$$

$$C_{D_0} = \left(0.0032 \bullet \left(\left(\frac{S_{tot} - S_{wing}}{\boldsymbol{S_w}} \right) + \left(FF \bullet \frac{S_{wing}}{\boldsymbol{S_w}} \right) \right) \right) + 0.0045$$

$$S_{wing} = 1.8 \bullet \boldsymbol{S_w}$$

$$S_{tot} = S_{wing} + \boldsymbol{S_{fuse}} + \boldsymbol{S_{tail}} + \boldsymbol{S_{vtail}} + \boldsymbol{S_{pod}}$$

$$FF = 1.0 + (0.891 \bullet \boldsymbol{TC}) + 100.0 \bullet (0.495 \bullet \boldsymbol{TC})^4$$

$$\boldsymbol{p} = \arccos(-1.0)$$

$$Lift_v = \left(0.8 \bullet \boldsymbol{C_{L_{max}}}\right) \bullet 0.001365 \bullet v^2 \bullet \boldsymbol{S_w}$$

$$S_r = 3.0 \bullet V_{rot_{mph}}$$

$$S_t = \begin{bmatrix} \sqrt{R^2 - (R - H_{tr})^2} & if\ H_{tr} \geq 50.0 \\ R \bullet \sin(\Gamma) & otherwise \end{bmatrix}$$

$$H_{tr} = R \bullet (1 - \cos(\Gamma))$$

$$R = 0.205 \bullet \left(V_{stall_t}\right)^2$$

$$V_{stall_t} = \sqrt{\frac{2.0 \bullet \boldsymbol{W_{to_{i-1}}}}{\boldsymbol{C_{L_{max}}} \bullet 0.00273 \bullet \boldsymbol{S_w}}}$$

$$\Gamma = \arcsin\left(\frac{Thrust - Drag_{V_{rot_{mph}}}}{\boldsymbol{W_{to_{i-1}}}}\right)$$

$$Thrust = 0.95 \bullet \boldsymbol{N_{eng}} \bullet \boldsymbol{T_{max}}$$

Note: $Drag_{V_{rot_{mph}}}$ is $Drag_V$ with $V = V_{rot_{mph}}$

$$S_c = \begin{bmatrix} 0 & if\ H_{tr} \geq 50.0 \\ \dfrac{50.0 - H_{tr}}{\tan(\Gamma)} & otherwise \end{bmatrix}$$

$$S_{ldg} = S_{air} + S_{\lg}$$

Where S_{air} is the air distance from the point over a 50ft obstacle to the touchdown point, and

S_{lg} is the ground roll distance

$$S_{air} = \frac{1.0}{0.1} \bullet \left(\frac{V_a^2 - V_{Td}^2}{64.348} + 50.0\right)$$

$$V_a = 1.2 \bullet \left(V_{stall_{ldg}}\right)$$

$$V_{stallldg} = \sqrt{\frac{2.0 \bullet W_{ldg}}{\boldsymbol{C_{L_{max}}} \bullet 0.00273 \bullet \boldsymbol{S_w}}}$$

$$W_{ldg} = \boldsymbol{W_{to_{i-1}}} - \left(0.2 \bullet \boldsymbol{W_{fuel_{i\text{-}1}}}\right)$$

$$V_{Td} = V_a \bullet \sqrt{0.9}$$

$$S_{\mathrm{lg}} = \frac{V_{Td}^2}{38.6088}$$

Note: To compute the integral defining S_g, the programmer shall use Simpson's rule:

$$\int_a^b f(x)\,dx \cong \frac{b-a}{3n} \bullet \begin{pmatrix} f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \dots \\ + 2f(x_{n-2}) + 4f(x_{n-1}) + f(x_n) \end{pmatrix}$$

where *n*, the number of partitions, is 200.

3.6 Computing the Aerodynamics Characteristics

C_{lift} is the cruise lift coefficient	(C_{lift} is a function of $W_{to_{i-1}}$ and $W_{fclm_{i-1}}$).
$V_{rot_{knots}}$ is the rotational velocity	($V_{rot_{knots}}$ is a function of $W_{to_{i-1}}$).
C_{drag} is the cruise drag coefficient	(C_{drag} is a function of $W_{to_{i-1}}$ and $W_{fclm_{i-1}}$).

$$C_{lift} = \frac{\boldsymbol{W_{to_{i-1}}} - \boldsymbol{W_{fclm_{i-1}}}}{DynamicPressure}$$

$$DynamicPressure = f\,(\boldsymbol{Height}, \boldsymbol{Mach}, \boldsymbol{S_w}) \quad \text{- Subroutine provided}$$

$$V_{rot_{knots}} = V_{rot_{mph}} \left(\frac{3600.0}{6080.0}\right)$$

$$V_{rot_{mph}} = 1.1 \bullet \sqrt{\frac{2.0 \bullet \boldsymbol{W_{to_{i-1}}}}{\boldsymbol{C_{L_{max}}} \bullet 0.00273 \bullet \boldsymbol{S_w}}}$$

$$C_{drag} = C_{D_0} + C_{D_{wave}} + C_{D_i}$$

$$C_{D_i} = \frac{(\mathrm{C}_{\mathrm{lift}})^2}{\boldsymbol{p} \bullet \boldsymbol{AR} \bullet \boldsymbol{E}}$$

$$C_{D_{wave}} = \left[\begin{array}{ll} \dfrac{20.0 \bullet (\boldsymbol{Mach} - Mach_{crit})^4}{\cos^3(Sweep_{rad})} & if\ \boldsymbol{Mach} \ge Mach_{crit} \\ & \\ 0 & otherwise \end{array} \right]$$

$$Mach_{crit} = \frac{0.9}{\cos(Sweep_{rad})} - \frac{\boldsymbol{TC}}{\cos^2(Sweep_{rad})} - \frac{C_{lift}}{10.0 \bullet \cos^3(Sweep_{rad})} - \left(\frac{0.1}{80.0}\right)^{1/3}$$

$$Sweep_{rad} = \frac{\boldsymbol{Sweep_{deg}} \bullet p}{180.0}$$

$$p = \arccos(-1.0)$$

$$C_{D_0} = \left(0.0032 \bullet \left(\left(\frac{S_{tot} - S_{wing}}{\boldsymbol{S_w}}\right) + \left(FF \bullet \frac{S_{wing}}{\boldsymbol{S_w}}\right)\right)\right) + 0.0045$$

$$S_{tot} = S_{wing} + \boldsymbol{S_{fuse}} + \boldsymbol{S_{tail}} + \boldsymbol{S_{vtail}} + \boldsymbol{S_{pod}}$$

$$S_{wing} = 1.8 \bullet \boldsymbol{S_w}$$

$$FF = 1.0 + (0.891 \bullet \boldsymbol{TC}) + 100.0 \bullet (0.495 \bullet \boldsymbol{TC})^4$$

3.7 Computing the Weight Characteristics

$$W_{to_i} = W_{wing} + W_{fuel_i} + W_{eng_n} + W_{fixed} + W_{fclm_i} + W_{cargo}$$

where $\mathrm{W}_{\mathrm{to_i}}$ is the total takeoff weight of the aircraft ,

$\mathrm{W}_{\mathrm{wing}}$ is the wing weight,
$\mathrm{W}_{\mathrm{fuel_i}}$ is the fuel weight,
$\mathrm{W}_{\mathrm{eng_n}}$ is the combined weight for all engines
$\mathrm{W}_{\mathrm{fixed}}$ is a fixed weight component associated with the aircraft
$\mathrm{W}_{\mathrm{fclm_i}}$ is the weight of the fuel needed to climb to the cruise altitude (this is not included in $\mathrm{W}_{\mathrm{fuel}}$)

W_{cargo} is a fixed cargo weight component.

Each weight component is computed as follows:

$$W_{wing} = 0.0051 \bullet \boldsymbol{S_w}^{0.649} \bullet S_{csw}^{0.1} \bullet \frac{\boldsymbol{AR}^{0.5}}{\boldsymbol{TC}^{0.4}} \bullet \frac{(\boldsymbol{N} \bullet \boldsymbol{W_{to_{i-1}}})^{0.557} \bullet (1.0 + \boldsymbol{TPR})^{0.1}}{\cos(Sweep_{rad})}$$

$$S_{csw} = 0.10 \bullet \boldsymbol{S_w}$$

$$Sweep_{rad} = \frac{\boldsymbol{Sweep_{deg}} \bullet p}{180.0}$$

$$p = \arccos(-1.0)$$

$$W_{fuel_i} = W_{cruise} - \left(\frac{W_{cruise}}{RE}\right)$$

$$W_{cruise} = W_{to_{i-1}} - W_{fclm_i}$$

$$RE = e^{\left(\frac{(\boldsymbol{Range} \bullet \boldsymbol{sfc})}{\left(V_{cruise_{knots}} \bullet LD\right)}\right)}$$

$$V_{cruise_{knots}} = V_{cruise_{mph}} \bullet \left(\frac{3600.0}{6080.0}\right)$$

$V_{cruise_{mph}} = f\,(\boldsymbol{Height}, \boldsymbol{Mach})$ Subroutine provided

$$LD = \frac{C_{lift}}{C_{drag}}$$

$$C_{lift} = \frac{\boldsymbol{W_{to_{i-1}}} - \boldsymbol{W_{fclm_{i-1}}}}{DynamicPressure}$$

$DynamicPressure = f\,(\boldsymbol{Height}, \boldsymbol{Mach}, \boldsymbol{S_w})$ - Subroutine provided

$$C_{drag} = C_{D_0} + C_{D_{wave}} + C_{D_i}$$

$$C_{D_i} = \frac{(\mathrm{C_{lift}})^2}{p \bullet \boldsymbol{AR} \bullet \boldsymbol{E}}$$

$$C_{D_{wave}} = \begin{bmatrix} \dfrac{20.0 \bullet (\boldsymbol{Mach} - Mach_{crit})^4}{\cos^3(Sweep_{rad})} & if\ \boldsymbol{Mach} \geq Mach_{crit} \\ 0 & otherwise \end{bmatrix}$$

$$Mach_{crit} = \frac{0.9}{\cos(Sweep_{rad})} - \frac{\boldsymbol{TC}}{\cos^2(Sweep_{rad})} - \frac{C_{lift}}{10.0 \bullet \cos^3(Sweep_{rad})} - \left(\frac{0.1}{80.0}\right)^{1/3}$$

$$Sweep_{rad} = \frac{\boldsymbol{Sweep_{deg}} \bullet p}{180.0}$$

$$\boldsymbol{p} = \arccos(-1.0)$$

$$C_{D_0} = \left(0.0032 \bullet \left(\left(\frac{S_{tot} - S_{wing}}{\boldsymbol{S_w}}\right) + \left(FF \bullet \frac{S_{wing}}{\boldsymbol{S_w}}\right)\right)\right) + 0.0045$$

$$S_{tot} = S_{wing} + \boldsymbol{S_{fuse}} + \boldsymbol{S_{tail}} + \boldsymbol{S_{vtail}} + \boldsymbol{S_{pod}}$$

$$S_{wing} = 1.8 \bullet \boldsymbol{S_w}$$

$$FF = 1.0 + (0.891 \bullet \boldsymbol{TC}) + 100.0 \bullet (0.495 \bullet \boldsymbol{TC})^4$$

$$W_{eng_n} = \boldsymbol{N_{eng}} \bullet \boldsymbol{W_{eng}}$$

$$W_{fixed} = \boldsymbol{C_{fix}} \bullet \boldsymbol{W_{to_{i-1}}}$$

$$W_{fclm_i} = \boldsymbol{F_{clm}} \bullet \boldsymbol{W_{to_{i-1}}}$$

$W_{cargo} =$ ***Fixed weight***, given as input

3.8 Output Requirements

3.8.1 Printing the Data Sets

All output produced by the application shall be printed to stdout formatted in such manner that individual data items and their corresponding values can easily be identified and read. The exact format of the output is not specified here since it shall be decided on by the developer and shall be specified latest in the low-level design document of this application. It is up to the customer to approve the format when the design is completed. It shall also be possible to redirect output produced from stdout to a file. Any error messages produced shall be directed to stderr, if such is available on the implementation platform, else these shall be directed to stdout as well.

There are three parts of output that the application shall be able to produce, (a) data item values and possible descriptions read from the input data file, (b) performance, aerodynamics, and weight data sets computed at every iteration, (c) a final set of convergence data.

To be able to verify that the data item values are correctly read by the application the values and descriptions (if any) of each data item shall be printed immediately after they have been read. The printout produced may be a line by line reflection of the input data file.

If the detailed weight convergence print flag (IPTDET) is set to one (1), the data sets computed for performance, aerodynamics and weights shall be printed at every iteration. The data items to print are:

- takeoff distance (S_{to}),
- landing distance (S_{ldg}),
- cruise lift coefficient (C_{lift}),
- cruise drag coefficient (C_{drag}),
- rotational velocity ($V_{rot_{knots}}$),
- weight of fuel in cruise segment (W_{fuel}),
- weight of fuel in climb segment (W_{fclm}),
- wing weight (W_{wing}),
- engine weight (W_{eng_n}),
- fixed weight overhead (W_{fixed}),

- fixed cargo weight (W_{cargo}), and
- gross takeoff weight (W_{to}).

A simple header indicating the iteration number shall precede the data set outputs. Again, it is up to the developer to design the header and document this no later than the low-level design document. At this point it shall be up to the customer to approve the design. If the print flag is set to zero (0) output shall not be produced during the iterative computations.

A final data set shall be printed when the iterative computations are completed. This final data set includes the same twelve (12) parameters as listed in the above paragraph. A final header shall precede this output to indicate the final data set.

3.8.2 Handling Errors

Since the possibility for error exists output shall be produced indicating the kind of error incurred. Two errors are possible, (a) the maximum number of iterations has been reached without convergence, or (b) the gross takeoff weight computed in an iteration exceeded the maximum gross takeoff weight allowed. In both cases output shall be produced that indicates the type of error occurred followed by a final printed of the data set (the final data set described above) which should be indicative of the error.

3.8.3 Output for each Mach Iteration

The above description of outputs indicates the data sets to be printed for one mach value iteration. Every mach value iteration requires the aircraft sizing data sets to be printed. A header indicating the current mach value that the data set is computed and printed for shall precede every initial data set for that mach value. It is left to the developer(s) to specify the format of the header and report this format no later than with the high-level design document. The customer shall have the option to reject or accept the header format at that point.

3.8.4 Output Format

As mentioned above, the format of the data set output is left to the application designers. The output shall be easily readable which means that white space should be used as needed to indicate breaks in data set iterations as well as mach value iterations.

4. Coding Environment

This application shall be designed such that it can be implemented and executed on computers running the operating system OSF/1 version of Unix on a DEC Alpha platform or the MS-DOS or PC-DOS versions 6.0 or high higher on an IBM-compatible PC. Coding shall be done using the C programming language. To allow for portability between the two operating systems and machines running those, only standard C library calls shall be used in the implementation. The choice of program editor is left open to the coding team since this may vary across machines.

4.1 Executable File

The executable file of this application shall be named “mdo.exe”.

4.2 Library Interface

Two function calls have been mentioned in the computation of the weight characteristics, *DynamicPressure* and $V_{cruise_{mph}}$. Due to their complexity this two functions have been taken from an earlier implementation of this application, compiled and archived in a library. The original implementation of this application was done in FORTRAN 77 and therefore the functions were implemented as subroutines and not function calls. In coding this application the programmers need to know the library name where the procedures are located, the correct procedure names, and the parameters to pass to the procedures.

The FORTRAN library in which these two procedures are located is called “libmdo.a” and will be provided to the programmers. It shall be stored in the same directory in which the source and object code of the application will be stored. During compilation the compiler needs to be told which library to link with and where the library can be found. A sample command line for compilation is:

```
cc -o mytest mytest.c -Lsubdirectory -lmdo -lm
```

where

mytest is the executable to compile to,
mytest.c is the C source code file,
subdirectory is the subdirectory where the library is located
-lmdo tells the compiler the name of the library, and
-lm tells the compiler to also use the standard mathematical library.

The standard mathematical library is need since a call to calculate an exponent is included in the FORTRAN library which otherwise would not be resolved.

The C procedure call to the DynamicPressure subroutine in the library shall be:

```
dynamic_pressure (&Height, &Mach, &Sw, &DynamicPressure)
```

where

&Height is the address to Height,
&Mach is the address to Mach,
&S_w is the address to S_w, and
&DynamicPressure is the address to DynamicPressure.

The C procedure call to the $V_{cruise_{mph}}$ subroutine in the library shall be:

```
cruise_velocity (&Height, &Mach, &Vcruisemph)
```

where

&Height is the address to Height,
&Mach is the address to Mach,
&$V_{cruise_{mph}}$ is the address to $V_{cruise_{mph}}$.

Note: It is important that all parameters shall be call by reference, providing a pointer to the actual parameter memory location. The variables DynamicPressure and $V_{cruise_{mph}}$ are of data type **double precision real**.

Since C and FORTRAN differ in how the actual procedure names are resolved during compilation, mainly in adding an underscore (_) either at the end or the beginning of the procedure name, the following definitions shall be included in the C program:

```
#define dynamic_pressure dynamic_pressure_
#define cruise_velocity cruise_velocity_
```

4.3 Programming Style

Note: This section is to be regarded as a guideline only, not a requirement.

It is left to the designers and implementors of this application how modular the C code shall be. Good programming standards suggest the use of procedure and function calls to allow for a modular design and aid in testing and maintaining. Also, the use of global variables is discouraged since they only lend themselves to confusion and easy introduction of program errors. Parameter passing should be used whenever necessary and possible. An overall good program design shall allow for a quick, efficient and correct application implementation with good programming standards encouraged and supported by the design.

Appendix A - List of Symbols

AR - Aspect Ratio
C_{drag} - Cruise Drag Coefficient
C_{D_0} - Zero Lift Drag
$C_{D_{Wave}}$ - Transonic Wave Drag Coefficient
$C_{drag_{to}}$ - Drag Coefficient at Takeoff
C_{fix} - Fixed Component Weight Multiplier Factor
C_{lift} - Cruise Lift Coefficient
$C_{L_{max}}$ - Maximum Cruise Lift Coefficient
E - Oswald Efficiency Factor
F_{clm} - Fuel Used in Climb to Cruise Altitude Multiplier
ft - feet
H - Cruise Altitude
KA - Wing Structural Technology Factor
lbs - pounds
LD - Lift to Drag Ratio
Mach - Mach Number (Speed)
$Mach_{crit}$ - Critical Mach Number
N - Ultimate Load Factor
N_{eng} - Number of Engines
nm - Nautical Miles
Range - Flight Distance
RE - Reynolds Number
S_{air} - Air Distance in Landing Calculation
S_c - Clibmout Distance for Takeoff
sfc - Specific Fuel Consumption (lb/lb/hr)
S_{fuse} - Wetted Area for Fuselage
S_g - Ground Takeoff Distance
S_{ldg} - Total Landing Distance
S_{lg} - Landing Ground Roll
S_{pod} - Pod Wetted Area
S_r - Rotation Distance for Takeoff
S_t - Transition Distance for Takeoff
S_{tail} - Horizontal Tail Wetted Area
S_{to} - Takeoff Distance

S_{tot} - Total Wetted Area

S_{vtail} - Vertical Tail Wetted Area

S_W - Wing Area

Sweep - Mid-chord Wing Sweep

S_{wing} - Wetted Area for Wing

TC - Wing Thickness Ratio

T_{max} - Maximum Thrust per Engine

V_{cruise} - Cruise Velocity

V_a - Landing Approach Velocity

V_{rot} - Rotation Velocity at Takeoff

V_{stall} - Stall velocity

V_{Td} - Landing Touchdown Velocity

W_{cargo} - Fixed Cargo Weight

W_{cruise} - Cruise Weight

W_{eng} - Engine Weight

W_{fclm} - Fuel Weight used in Climb Segment

W_{fix} - Fixed Weight (computed as a fraction of W_{to_i})

W_{fuel} - Fuel Weight used in Cruise Segment

W_{Ref} - Initial Cruise Weight

W_{to} - Takeoff Gross Weight

W_{Wing} - Wing Weight

TPR - Taper Ratio

Appendix B - Logical Input File Structure View

The following is a logical breakdown of the subsections included in the input data file. For simplicity reasons an actual input file has all subsection headers removed as can be seen in Appendix C.

```
-----------------------------------------
*            CONTROL FLAGS              *
-----------------------------------------
1         -> IPTDET, Detailed Weight Convergence Print Flag
-----------------------------------------
*     PARAMETRIC MACH NUMBER SETUP      *
-----------------------------------------
0         -> NJMAC, Number of Mach Iterations
0.6500    -> MACH, Initial Mach Value
0.0500    -> MSTEP, Mach Value Increment per Iteration
-----------------------------------------
*            INITIAL POINTS             *
-----------------------------------------
9.0       -> AR, Aspect Ratio
3800.0    -> SW, Wing Area
32000.0   -> H, Flight Altitude
1.3       -> SWEEP, Mid-Chord Wing Sweep
0.10      -> TC, Wing Thickness to Chord Ratio
0.30      -> TPR, Wing Taper Ratio
5000.0    -> Range, Flight Distance
-----------------------------------------
* BASELINE WEIGHTS (FIXED/FRACT/COMPUT) *
-----------------------------------------
580000.0  -> WTOREF, Reference Total Take-Off Weight
100000.0  -> WFUELRF, Reference Fuel Weight
150000.0  -> WCARGO, Fixed Cargo Weight
7500.0    -> WENG, Engine Weight Per Engine
0.0200    -> FCLM, Climb Fuel Fraction
0.200     -> CFIX, Aircraft Fixed Weight Fraction
4.5       -> N, Structural Load Factor for Wing Weight
-----------------------------------------
*          PROPULSION VARIABLES         *
-----------------------------------------
45000.00  -> TMAX, Maximum Thrust per Engine
0.640     -> SFC, Fuel Consumption Fraction per Engine
4         -> Number of Engines
-----------------------------------------
*    AERODYNAMIC/GEOMETRIC VARIABLES    *
-----------------------------------------
2.5       -> CLMAX, Maximum Cruise Lift Coefficient
0.85      -> E, Oswald e constant
10367.0   -> SFUSE, Fuselage Wetted Area
1428.0    -> STAIL, Horizontal Tail Wetted Area
```

```
800.0      -> SVTAIL, Vertical Tail Wetted Area
2412.0     -> Pod Wetted Area
```

Appendix C - Physical Input File Structure View

The filename of the input data file shall be “mdo.inp”. The following is a sample of an actual input data file that the application should be able to process.

```
1          -> IPTDET, Detailed Weight Convergence Print Flag
0          -> NJMAC, Number of Mach Iterations
0.6500     -> MACH, Initial Mach Value
0.0500     -> MSTEP, Mach Value Increment per Iteration
9.0        -> AR, Aspect Ratio
3800.0     -> SW, Wing Area
32000.0    -> H, Flight Altitude
1.3        -> SWEEP, Mid-Chord Wing Sweep
0.10       -> TC, Wing Thickness to Chord Ratio
0.30       -> TPR, Wing Taper Ratio
5000.0     -> Range, Flight Distance
580000.0   -> WTOREF, Reference Total Take-Off Weight
100000.0   -> WFUELRF, Reference Fuel Weight
150000.0   -> WCARGO, Fixed Cargo Weight
7500.0     -> WENG, Engine Weight Per Engine
0.0200     -> FCLM, Climb Fuel Fraction
0.200      -> CFIX, Aircraft Fixed Weight Fraction
4.5        -> N, Structural Load Factor for Wing Weight
45000.00   -> TMAX, Maximum Thrust per Engine
0.640      -> SFC, Fuel Consumption Fraction per Engine
4          -> Number of Engines
2.5        -> CLMAX, Maximum Cruise Lift Coefficient
0.85       -> E, Oswald e constant
10367.0    -> SFUSE, Fuselage Wetted Area
1428.0     -> STAIL, Horizontal Tail Wetted Area
800.0      -> SVTAIL, Vertical Tail Wetted Area
2412.0     -> Pod Wetted Area
```