Zachary Christopher Rowland
26732157

Aircraft Generation Code

REMOTELY DEPLOYED AND RECOVERED UAV

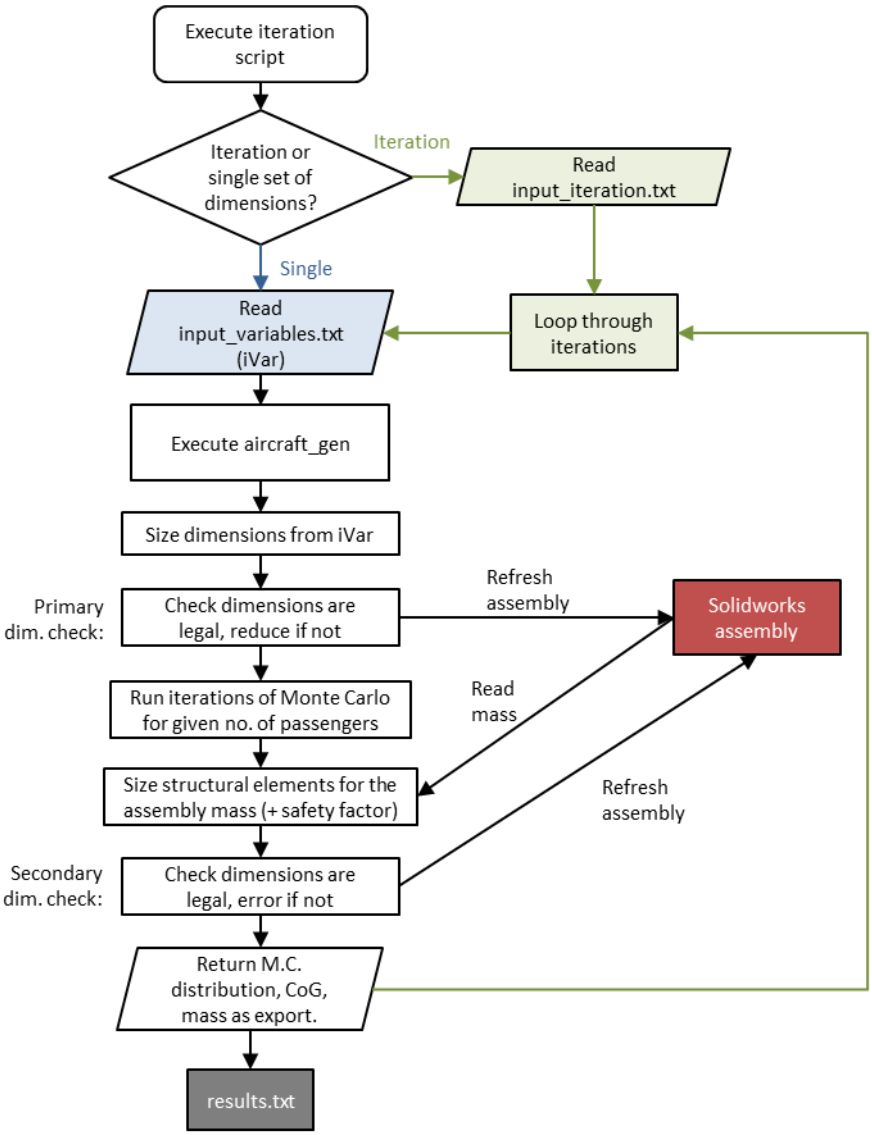
GDP #33

Zachary Christopher Rowland
26732157

1. Introduction

Explain purpose.

Operation flowchart.



2. Features

Features.

2.1. Solidworks DirectX link

The Matlab code links directly with the open

2.2. Dimension Checking

Explain how this works.

2.3. Passenger Allocation through Monte Carlo

Explain how this works.

2.4. Aerodynamic Analysis

Explain how this works.

2.5. Structural Analysis

Explain how this works. Includes secondary dimension checking.

3. Output and Data Plotting

Explain how this works.

Zachary Christopher Rowland
26732157

Appendix

A. Best Practice for Coding

All variables and functions should be named using ‘upper CamelCase’ (or ‘PascalCase’), with all compound words are capitalised, unless specified otherwise. For example:

For functions:	ReadDimensions(filepath, filename, format)
For variables:	TotalLift

Exceptions to this rule are as below.

A.1. Dimension and unit variables, strings

All dimensions and unit variables (including when referred to in strings) are to be written in the following format.

[subassembly][DimensionName]_[DimensionType]_[unit]

For example,

wingSpan_Length_mm

wingMainSpar_InnerDiameter_mm

A.2. Other exceptions

Some variables break the rules set out above due to the frequency of their use in the code, to allow them to stand out, or because their names are so short as to make the standard practice redundant. These variables should be left named as they are unless it is practical to fit them to the standard.

B. Best Practice and Structure for Folders and Files

B.1. Folders

The project folders are organised as follows. All folder names are given in total lowercase without spaces or underscores, regardless of the number of words in the folder title. This is to encourage single-word folder titles.

- **Mk1a:** The aircraft version, given by a number, with its corresponding sub-version, given as a letter.
    - **cad:** All CAD files and folders associated with the given version of the aircraft assembly.
        - **fuselage**
        - **motor**
        - **tail**
        - **wing**
        - **old**

- **matlab:** All Matlab files and folders, the run script and the ‘input_variables.txt’ and ‘input_iteration.txt’ text files.
  - **fileio:** All functions associated with file input and output.
  - **interface:** All functions associated with user and Solidworks interface (except where involving file I/O).
  - **ops:** Functions integral to the operation of the software that call all other functions aside from the run script.
  - **results:** All results and data generated by the software should be output here.
  - **tools:** Tools used to alter the Solidworks assembly, or run any relevant calculations.
  - **unused:** Functions and scripts not currently used in the software that may be useful in future.

The CAD folders each contain their respective subassembly and parts, with a Solidworks equation file titled as ‘[assembly]_equations.txt’. Each folder also contains an ‘old’ folder, for storing previous versions of parts. The top level CAD folder contains the master assembly, with its own equation file. The only parts included at the top level should be those that connect multiple subassemblies (such as the spars and booms).

B.2. Files

Matlab functions are not versioned as this is handled through Git, however if a working function is removed from the software it should be moved to the ‘unused’ folder. All functions should be named as they are in the code (e.g. ReadDimensions.m for the function ReadDimensions).

CAD files should be titled in the following ‘lower CamelCase’ format.

[subassembly][PartName]_v[x.y]

With x corresponding to the aircraft version and y corresponding to the part version. The aircraft version does not include the letter sub-version. For example:

wingAirfoilRib_v1.1

See Appendix A for how to title dimensions corresponding to each part.

Assemblies should be simply titled as follows.

[subassembly]Assem_v[x]

With x defined as before, but with the aircraft sub-version. For example:

masterAssem_v1a