The quarterly newsletter for JSBSim, an open source flight dynamics model in C++

Volume 1, Issue 2

15 July 2004

Back of the Envelope

Creating Flight Models

1) Gather Data
   - TCDS
   - Textbooks
   - Tech Reports
   - Etc.
2) Use Aeromatic for first cut.
   - www.jsbsim.org
3) Adjust data file based on found data.

Inside this issue:

A User Perspective: Building an F-16 Flight Model

Constructing a detailed Flight Model for an advanced aircraft such as the F-16 is not a simple or quick task. One has to find the correct data, know the correct dimensions and body locations, and — for most modern aircraft — one will also need to know how the flight computer operates. Constructing the flight model for the F-16 Fighting Falcon (called the *Viper* by its pilots) is tedious, especially for those who don't have a degree in aeronautics and just want to practice a hobby.

<u>Pitfalls</u>

My first attempt at building a flight model for my favorite high speed aircraft was based on the FORTRAN code written by David Murray. Even without any knowledge of FORTRAN it was easy to extract the data from the source. But now I was faced with another challenge: converting this data to the proper format. JSBSim uses the Lift and Drag coefficients while Murray's data uses the body axis forces and moments which required all moments and forces to be recalculated. As it turned out later on, it is also extremely important to know the direction of all axes because they can turn out to be defined slightly different.

<u>Success</u>

After finally managing to get all the tables in the form required by JSBSim, the result was an F-16 as it would have been without the help of a flight computer. It was very sensitive on the elevators (as would be expected for a neutrally stable aircraft design like the F-16) that could easily make 20g turns and was a little tricky to keep airborne. But, the aircraft responded quite nicely to non-pitch related input and was still nice to operate.

<u>Flight Computer</u>

To get the model to operate properly it was necessary to add Flight Control components to the configuration file. The Falcon isn't controlled by a direct coupling between user input and surface deflections. Instead, the flight computer decides the surface deflections based on current aircraft state (monitored using sensor input) and pilot in-

(Continued on page 4)

JSBSim Data Output and Analysis

There are several ways to get data out of JSBSim (running either in standalone mode or while integrated with another simulator such as FlightGear).

Data can be output to a file, to the console (where it can be redirected), or through a socket. Data output is set up through the OUTPUT section of the aircraft configuration file, at present. This is perhaps not an ideal place to specify output formatting, and the specification may change in the future. The format for OUTPUT specification is as follows:

```
<OUTPUT NAME="{n}"TYPE="{t}">
  RATE_IN_HZ        rate
  SIMULATION        ON|OFF
```

(Continued on page 2)

News Items

JSBSim Project Coordinator Jon Berndt has written a paper for the 2004 AIAA Modeling and Simulation Technology Conference held in August in Providence, Rhode Island. The paper will be hosted at a later time on the JSBSim web site.

- *New config version (1.65)*
- *Thrusters now part of engine definition*
- *Aerial onload of fuel now possible*
- *Fuel temperature now changes with atmospheric qualities*

Bill Galbraith has altered the source code to the well-known USAF application called Digital DATCOM. It will now output aerodynamic characteristics for a specified aircraft configuration directly in JSBSim configuration file format. Digital DATCOM is a program that calculates the aerodynamic characteristics for an aircraft based on the description of the aircraft stored in an input file. This tool is in beta release at this time. It is available at Bill's web site:

`http://www.holycows.net/datcom`

(Continued from page 1)

```
  ATMOSPHERE        ON|OFF
  MASSPROPS         ON|OFF
  AEROSURFACES      ON|OFF
  RATES             ON|OFF
  VELOCITIES        ON|OFF
  FORCES            ON|OFF
  MOMENTS           ON|OFF
  POSITION          ON|OFF
  COEFFICIENTS      ON|OFF
  GROUND_REACTIONS  ON|OFF
  FCS               ON|OFF
  PROPULSION        ON|OFF
</OUTPUT>
```

Specific properties can also be logged if desired by including a PROPERTY definition in the OUTPUT section. For example:

```
PROPERTY fcs/elevator-pos-deg
```

There is a facility in both JSBSim and FlightGear for piping data through a socket. For JSBSim, you can specify a PORT and an IP address. Here is an example:

```
<OUTPUT NAME="192.168.0.2"
TYPE="SOCKET" PORT="5156">
```

You can use an application such as the netcat tools (nc) to see data coming across the socket. For instance, if you send data over the socket via the IP address and port as specified in the previous example, you would use this command line argument on the receiving computer:

```
nc -l -p 5156
```

The above command sets up a listening process on port 5156.

Currently, the set of data sent over a socket is predetermined.

Once data has been saved as a .csv file it can be plotted by any program that can read that format, such as Excel. There is a special purpose plotting program called SimPlot (available at the JSBSim web site in source code format or as a Windows binary). SimPlot is useful for rapid plotting of parameters. It can be used in batch runs to automatically create plots and a companion web page that references the plots, including a thumbnail image of the plot.

The XML format file that tells the plotting program what to plot is simple and intuitive. Here is an excerpt:

```
<?xml version="1.0"?>
<plotset>
 <plot>
 <title>Ground Track</title>
 <label axis="x">Long</label>
 <label axis="y">Lat</label>
 <scale>
  <min axis="x">121.5</min>
  <min axis="y">46.5</min>
  <max axis="x">122.5</max>
  <max axis="y">47.5</max>
 </scale>
 <parameter axis="x">Long</parameter>
 <parameter axis="y">Lat</parameter>
</plot>
```

Recent JSBSim Aircraft Models:

- Boeing 777
- Boeing 707
- Pilatus PC-7

See "Dave's Hangar" for these:

http://home.comcast.net/~davidculp2/hangar/hangar.html

SimPlot can be used in batch runs to automatically create plots and a companion web page that references the plots, including a thumbnail image of the plot.

In Depth: Creating a Flight Model for JSBSim, Aerodynamics

JSBSim models the aerodynamics portion of flight dynamics for a vehicle based on the *coefficient buildup* method. That is, the total contributions to lift, drag, side force, and the pitching, rolling, and yawing moments are calculated by adding up all the constituent parts that come from the wing, the stabilizers, flaps, ailerons, etc. Take lift, for example: for an aircraft, lift mostly comes from the wing. However, when the yoke is pushed forward to a point, a nose-down pitching moment occurs, which happens because the horizontal tail surface is providing a lifting force — far aft of the CG — and this also contributes to the total lift of the aircraft. When the flaps are extended, that also provides an extra increment of lift. The lift on the wing is often given as a function of alpha (angle of attack) - and of course that adds in to the total lift. In the form of an equation, this is how the lift calculation would look:

$$L = qS[\alpha C_{L\alpha} + \delta e C_{L\delta e} + \delta f C_{L\delta f}]$$

where,

L = total lift
q = dynamic pressure
S = wing area
α = angle of attack
δe = elevator deflection
δf = flap deflection
$C_{L\alpha}$ = C_L due to alpha
$C_{L\delta e}$ = C_L due to elevator
$C_{L\delta f}$ = C_L due to flaps

The last three parameters in the list above denote the individual contributions due to lift in the form of coefficients. These items are often defined in the form of tables. The lookup table for lift due to elevator deflection, for instance, might be a function of mach. The table is part of the coefficient definition in the JSBSim aircraft specification file. In this example, mach is also specified in the definition as the lookup index. The table and lookup index part of the coefficient definition describe a coefficient, itself. However, the coefficient is also turned into a specific force quantity. Observe the product qS in the equation for lift shown previously. When the coefficient found by the table lookup is multiplied by the q and S values, the result is a contribution of lift. See Fig. 1 for an example of a complete coefficient definition.

```
<COEFFICIENT NAME="CLDe" TYPE="VECTOR">
  Lift_due_to_Elevator_Deflection
  11
  velocities/mach-norm
  aero/qbar-psf|metrics/Sw-sqft|fcs/elevator-pos-rad
  0.0 1.00
  0.6 1.05
  1.0 1.15
  1.2 1.00
  1.6 0.66
  2.0 0.50
  2.4 0.40
  3.0 0.31
  5.6 0.21
  6.0 0.20
  9.0 0.20
</COEFFICIENT>
```

Figure 1. JSBSim Coefficient Definition

Let's look at the coefficient definition in Fig. 1 line-by-line. The first line simply defines the configuration file entry as being a coefficient definition, names the coefficient, and defines the type of the coefficient. Several types of coefficient are available, a simple value, or a one, two, or three dimensional table. A description for the coefficient is given in line two. Line three (in this example) refers to the number of rows of data

(Continued on page 4)

JSBSim models the aerodynamics portion of flight dynamics for a vehicle based on the coefficient buildup method.

The quarterly newsletter for JSBSim, an open source flight dynamics model in C++

Highlighted References

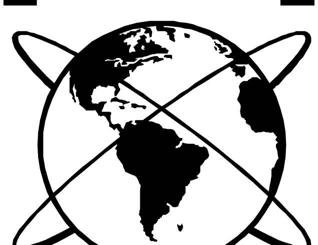
Online:

http://www.aoe.vt.edu/~durham/AOE5214/
(A flight dynamics and control textbook online)

Published Books:

"Aircraft Control and Simulation", Stevens and Lewis
http://www.amazon.com/exec/obidos/ASIN/0471613975/103-8884689-9020660

Visit us on the web at: www.jsbsim.org



(Continued from page 1)

put. This means that a pilot doesn't command elevator deflections, but a g-load. Neither does the pilot command aileron or rudder deflections, but rather a roll-rate or yaw-rate. The advantage of this is that the flight computer keeps the aircraft in its current state if there is no pilot input, basically meaning that the aircraft will fly a straight line, no matter what, it will automatically compensate for unwanted pitch, roll and yaw effects (at least within it's capabilities).

Control Laws

The flight computer compares the requested (pilot) input and the current state and computes the error (the difference between the two) and calculates the actions needed to minimize this error and commands the surfaces to take that action. This (in rare circumstance) may even result in an opposite surface deflection than commanded by the pilot. By letting the computers command the aero surfaces there is a risk of getting in an unwanted (stall) situation too easily because there is no feedback to the pilot (like stick input that is too easy for normal conditions, which could mean a stalled surface). To solve that problem the designers of the F-16 decided to let the flight computer prevent stall conditions in every situation. If (for instance) the angle of attack exceeds 25 degrees, full pitch down is commanded to alleviate that; wind tunnel data shows that the aircraft gets unstable beyond that.

Current State

The current configuration file of the F-16 has almost everything implemented but stall prevention. Some quick attempts to get that feature implemented shows that it may be the hardest task of all. When comparing the current model to the experience I have with military F-16 simulators shows that it feels like it behaves almost identically. Roll rate, g-load, turn radius, (ground) acceleration and deceleration (in-air) don't seem to be too far off compared to the simulators. Also, maximum speed (Mach 2.2 at 40,000 ft) looks to be on par with the accepted specifications. But there is still a lot to do. — *EH*

Online

www.f16viper.org

www.f-16.net

(Continued from page 3)

in the table. Line 4 is the property that is used as a lookup index into the table. Line 5 is a list of items (separated by a vertical bar) that, when multiplied all together with the coefficient value determined by lookup into the accompanying table, produces (in this example) the lift force due to elevator deflection. Lines beginning at 6 define the lookup table. The left column is a list of mach numbers, the right column defines coefficient values associated with each mach.

How do we know what these coefficients are? How do we know the values in the table? There are many places to look for good information about the aerodynamic qualities for a specific aircraft. They include textbooks, technical report servers, aircraft manufacturer web sites, etc. There is another tool that is helpful in determining aerodynamic coefficients. It is called DATCOM. This tool started out as a series of books that one could use in determining aerodynamic characteristics for a hypothetical aircraft, but McDonnell Douglas turned those books into a program that became known as Digital DATCOM. More recently, Bill Galbraith has modified the source code for Digital DATCOM to make it put out aerodynamic coefficients in a format that is directly usable in the AERODYNAMICS section of the JSBSim aircraft configuration file. *-JB*

Next Issue:

"Back of the Envelope" is a new communication tool written for a wider audience than core JSBSim developers, including instructors, students, and other users. The articles featured will likely tend to address questions and comments raised in the mailing lists and via email. If you would like to suggest (or even author) an article for a future issue, please email the editor at: jsb@hal-pc.org.

Some possible topics for future issues includes:

- The Property System
- JSBSim Configuration Files in XML
- Integrating the Equations of Motion in JSBSim
- Scripting JSBSim runs