A MATLAB Implementation of the Jacchia Atmosphere Model

This document is the user's manual for a MATLAB function named `jatmos70.m` that implements the algorithm described in the classic report *New Static Models of the Thermosphere and Exosphere with Empirical Temperature Profiles*, L. G. Jacchia, SAO Special Report No. 313, May 6, 1970, Smithsonian Institution, Astrophysical Observatory, Cambridge, Massachusetts. This implementation also includes the improved integration or quadrature method described in "An Improvement in the Numerical Integration Procedure Used in the NASA Marshall Engineering Thermosphere Model", by M. P. Hickey, NASA CR 179389, August 1988.

This computer model approximates a *dynamic* Earth atmospheric where density is a function of the calendar date, universal time, the sub-satellite position, and the solar activity. The model is valid for altitudes between 90 and 2500 kilometers and is useful for modeling the effect of atmospheric drag on the long-term evolution of orbits.

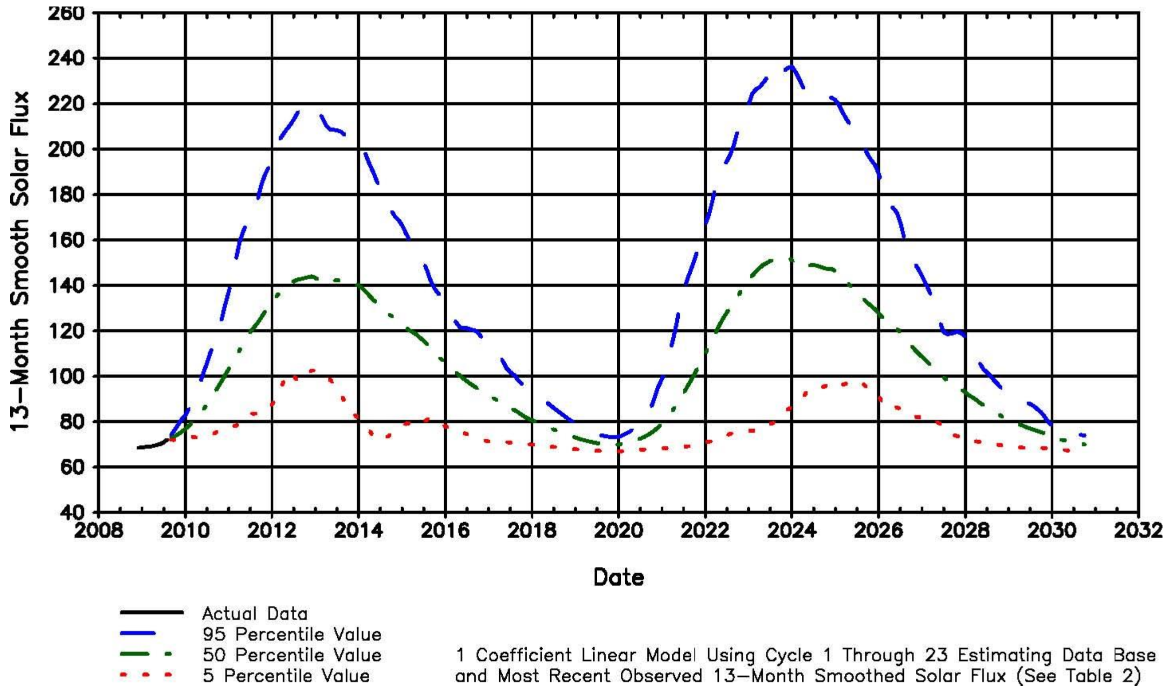
The following is the syntax of the MATLAB function that calculates the atmospheric properties.

```
function outdata = jatmos70(indata)

% Jacchia 1970 atmosphere main driver

% input

%  indata(1)  = geodetic altitude (kilometers)
%  indata(2)  = geodetic latitude (radians)
%  indata(3)  = geographic longitude (radians)
%  indata(4)  = calendar year (all digits)
%  indata(5)  = calendar month
%  indata(6)  = calendar day
%  indata(7)  = utc hours
%  indata(8)  = utc minutes
%  indata(9)  = geomagnetic index type
%               (1 = indata(12) is Kp, 2 = indata(12) is Ap)
%  indata(10) = solar radio noise flux (jansky)
%  indata(11) = 162-day average F10 (jansky)
%  indata(12) = geomagnetic activity index

% output

%  outdata(1)  = exospheric temperature (deg K)
%  outdata(2)  = temperature at altitude (deg K)
%  outdata(3)  = N2 number density (per meter-cubed)
%  outdata(4)  = O2 number density (per meter-cubed)
%  outdata(5)  = O number density (per meter-cubed)
%  outdata(6)  = A number density (per meter-cubed)
%  outdata(7)  = He number density (per meter-cubed)
%  outdata(8)  = H number density (per meter-cubed)
%  outdata(9)  = average molecular weight
%  outdata(10) = total density (kilogram/meter-cubed)
%  outdata(11) = log10(total density)
%  outdata(12) = total pressure (pascals)
```

Prior to calling this function, your main script must call a MATLAB function called `j70iniz`. This function performs initialization required by `jatmos70` and its support functions.

The solar activity data, `indata` array elements 10, 11 and 12, can be extracted from solar activity bulletins called *Future Solar Activity Estimates for Use in Prediction of Space Environmental Effects on Spacecraft*. These bulletins contain tables of 5, 50 and 95 percentile solar activity data. Solar activity bulletins and data files are issued monthly by NASA Marshall Space Flight Center and are available on the Internet at sail.msfc.nasa.gov. This site also contains a PDF version of NASA TM-4759, "Statistical Technique for Intermediate and Long-Range Estimation of 13-Month Smoothed Solar Radio Flux and Geomagnetic Index".

The solar activity data file contains 5, 50 and 95 percentile data for the 10.7 cm solar flux and Ap geomagnetic index. The following is a typical plot of actual and predicted solar activity.



The following illustrates the header and first few data lines of a typical solar activity data file.

TABLE 3 ESTIMATES OF 13-MONTH SMOOTH SOLAR ACTIVITY FOR
BALANCE OF CYCLE 24 CYCLE 25 AND BEGINNING OF CYCLE 26

TIME		10.7 CM SOLAR FLUX (F10.7) PERCENTILE			GEOMAGNETIC INDEX (Ap) PERCENTILE		
		95.0%	50%	5.0%	95.0%	50%	5.0%
2009.4170	JUN	71.0	70.0	69.4	4.7	4.3	3.9
2009.5003	JUL	72.1	70.5	69.2	4.9	4.2	3.6
2009.5837	AUG	73.1	71.0	69.0	5.2	4.0	3.3
2009.6670	SEP	74.6	71.5	68.8	5.5	3.9	3.0
2009.7503	OCT	76.6	72.1	68.6	5.4	3.9	2.7
2009.8337	NOV	80.0	72.9	68.6	5.2	3.8	2.5
2009.9170	DEC	82.9	73.8	68.8	5.0	3.4	2.2
2010.0003	JAN	84.9	74.8	68.2	4.4	2.6	1.1
2010.0837	FEB	88.2	75.9	67.6	4.3	2.2	0.4

This software suite includes a MATLAB script named `demo_jatmos70` that demonstrates how to interact with this function. Typical solar activity data are "hardwired" in this script.

The following is the output created by this script.

```
demo_jatmos70

calendar date       03-Jun-1979

universal time       00:00:00.000

geodetic altitude                303.0417 kilometers

geodetic latitude                -21.0000 degrees

east longitude                    36.0000 degrees

solar radio noise flux            60.0700 jansky

162-day average F10              109.5600 jansky

geomagnetic activity index         9.3000

exospheric temperature           698.9464 degrees K

temperature                      690.8524 degrees K

total density             6.47341420e-012 kg/m**3

total pressure            2.24837188e-006 pascals

average molecular weight  1.65377812e+001
```

The following is the MATLAB source code for this script.

```
% demo_jatmos70        May 15, 2013

% this script demonstrates how to interact with
% the j70iniz and jatmos70 functions which compute
% the properties of the atmosphere using the Jacchia
% 1970 model with MET modifications

% Orbital Mechanics with MATLAB

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all;

rtd = 180.0 / pi;

dtr = pi / 180.0;

% initialize the algorithm
```

```
j70iniz;

% algorithm inputs

%  indata(1)  = geodetic altitude (kilometers)
%  indata(2)  = geodetic latitude (radians)
%  indata(3)  = geographic longitude (radians)
%  indata(4)  = calendar year (all digits)
%  indata(5)  = calendar month
%  indata(6)  = calendar day
%  indata(7)  = utc hours
%  indata(8)  = utc minutes
%  indata(9)  = geomagnetic index type
%                (1 = indata(12) is Kp, 2 = indata(12) is Ap)
%  indata(10) = solar radio noise flux (jansky)
%  indata(11) = 162-day average F10 (jansky)
%  indata(12) = geomagnetic activity index

indata(1) = 303.04166;
indata(2) = -21.0 * dtr;
indata(3) = 36.0 * dtr;
indata(4) = 1979;
indata(5) = 6;
indata(6) = 3;
indata(7) = 0;
indata(8) = 0;
indata(9) = 2;
indata(10) = 60.07;
indata(11) = 109.56;
indata(12) = 9.3;

day = indata(6) + indata(7) / 24.0 + indata(8) / 1440.0;

jdate = julian(indata(5), day, indata(4));

[cdstr, utstr] = jd2str(jdate);

% compute atmospheric properties

outdata = jatmos70(indata);

% algorithm outputs

%  outdata(1)  = exospheric temperature (deg K)
%  outdata(2)  = temperature at altitude (deg K)
%  outdata(3)  = N2 number density (per meter-cubed)
%  outdata(4)  = O2 number density (per meter-cubed)
%  outdata(5)  = O number density (per meter-cubed)
%  outdata(6)  = A number density (per meter-cubed)
%  outdata(7)  = He number density (per meter-cubed)
%  outdata(8)  = H number density (per meter-cubed)
%  outdata(9)  = average molecular weight
%  outdata(10) = total density (kilogram/meter-cubed)
%  outdata(11) = log10(total density)
%  outdata(12) = total pressure (pascals)

% print results

clc; home;
```

```
fprintf('\ndemo_jatmos70\n\n');

fprintf('\ncalendar date       ');

disp(cdstr);

fprintf('\nuniversal time       ');

disp(utstr);

fprintf('\n\ngeodetic altitude            %12.4f kilometers \n', indata(1));

fprintf('\ngeodetic latitude            %12.4f degrees \n', rtd * indata(2));

fprintf('\neast longitude               %12.4f degrees \n', rtd * indata(3));

fprintf('\nsolar radio noise flux       %12.4f jansky \n', indata(10));

fprintf('\n162-day average F10          %12.4f jansky \n', indata(11));

fprintf('\ngeomagnetic activity index   %12.4f \n', indata(12));

fprintf('\n\nexospheric temperature       %12.4f degrees K\n', outdata(1));

fprintf('\ntemperature                  %12.4f degrees K\n', outdata(2));

fprintf('\ntotal density             %12.8e kg/m**3\n', outdata(10));

fprintf('\ntotal pressure            %12.8e pascals\n', outdata(12));

fprintf('\naverage molecular weight  %12.8e \n\n', outdata(9));
```