USER GUIDE: F-16 FLIGHT SIMULATION PACKAGE

Duc H. Nguyen and Mark H. Lowenberg
University of Bristol

OVERVIEW

This package contains 4 x 3 = 12 simulations of the F-16 aircraft, written in 4 environments (MATLAB, MATLAB with C code acceleration, Simulink, and Julia) at 3 levels of complexity (4^{th}, 8^{th}, and 12^{th}-order). Emphasis was placed on simplicity and performance instead of functionality, while still preserving the nonlinear coupling of the rigid-body equations of motion. The aerodynamic data is based on the 1979 NASA report by L. Nguyen et al.

The 4^{th}-order simulation contains 3 degrees of freedom (DoF) with only longitudinal motion. This model can be used as an entry point for users new to aircraft flight dynamics. A full description of the equations of motion and aerodynamic data can be found in the appendix of DOI: 10.1007/s11071-022-07283-z (also included in this package). The 8^{th} and 12^{th}-order simulations have six degrees of freedom, with the former containing only the 8 uncoupled states in the equations of motion.

Some common tools for aircraft flight dynamics analysis (trimming, linearisation, trajectory visualisation, and parallel simulation) are also included:

	MATLAB	**MATLAB (ACCELERATED)**	**SIMULINK**	**JULIA**
4^{TH} ORDER (3 DoF)	Trim: level flight	Trim: level flight	Linear analysis Parallel simulation	Live plotting Parallel simulation
8^{TH} ORDER (6 DoF)	Trim: turning flight	Trim: turning flight	-	-
12^{TH} ORDER (6 DoF)	3D trajectory	3D trajectory	-	Live plotting 3D trajectory

In the 3 DoF trim routine, the code finds the necessary stabilator deflection and engine thrust to achieve a specified flight path angle and airspeed. In the 6 DoF trim routine, the code finds the necessary aileron, stabilator, and rudder deflections, along with engine thrust, to achieve a coordinated and level turn at a specified load factor and airspeed. Both trim routines are based on an optimisation search, which minimises the sum of the squared state derivatives (i.e., trimmed when all state derivatives are zero).

The Julia simulations contain a live plotting routine that displays the state and aircraft trajectories on the go (while the differential equation solver is still running). This plotting routine is controlled by four switches, which can be adjusted to create some basic flight animations.

```
t_refresh     = 2   # Refresh the figure every __ second of simulated time. Set to t_sim to refresh only once after the simulation finishes running
draw_aircraft = 1   # For each time the figure is refreshed, draw an aircraft on the top-right subplot? (1: yes, 0: no)
keep_aircraft = 1   # Retain the previously drawn aircraft in the top-right subplot after each refresh event? (1: yes, 0: no)
t_wait        = 0.1 # Wait __ seconds in real time each time the figure is refreshed (0 for fastest response)
```

C CODE ACCELERATION

The MATLAB-based simulations can be sped up by running the .MEX file, which is a C-code accelerated copy of the original function called by the ODE solver. All MATLAB simulations in this package come with

.MEX file included. For example, to run the accelerated code in the 4th-order simulation, disable line 17 and enable line 18.

```
14    u0 = [de; CG; T];
15    x0 = [alpha0; V0; q0; theta0];
16    options = odeset('RelTol',1e-4); % default: 1e-3
17    tic; [t,x] = ode45(@eom_4,[0 500],x0,options,u0); toc % MATLAB
18    % tic; [t,x] = ode45(@eom_4_mex,[0 500],x0,options,u0); toc % Accelerated
```

The .MEX file is separated from its original copy and should be re-generated whenever the MATLAB model is changed. To do this, please refer to MATLAB's help centre:
uk.mathworks.com/help/coder/ug/generate-cc-code.html

STATES AND CONTROL INPUTS

	STATES	CONTROL INPUTS
4TH ORDER (3 DoF)	α, V, q, θ	δ_e T CG
8TH ORDER (6 DoF)	α, β, V p, q, r ϕ, θ	$\delta_a, \delta_e, \delta_r$ T CG
12TH ORDER (6 DoF)	α, β, V p, q, r ϕ, θ, ψ X, Y, Z	$\delta_a, \delta_e, \delta_r$ T CG

States
α: angle of attack
β: sideslip angle
V: total velocity
p, q, r: body-axis roll, pitch, yaw rates
ϕ, θ, ψ: Euler roll, pitch, heading angles
X, Y, Z: Cartesian north, east, down coordinates

Control inputs
$\delta_a, \delta_e, \delta_r$: aileron, stabilator, rudder deflections
T: engine thrust
CG: centre of gravity position along body x-axis

PERFORMANCE

The following run times in seconds were recorded on a laptop with a 24-core Intel i9-13950HX processor and 32 GB of RAM.

Singe-thread: 500-second simulation, relative tolerance 1e-4 with variable steps solvers (ODE45 in MATLAB/Simulink and Tsit5 in Julia). The 8th-order runs correspond to the fully-developed spin case.

ENVIRONMENT	4TH-ORDER	8TH-ORDER
MATLAB	1.14	18.3
Simulink	0.776	0.702
MATLAB (accelerated)	0.0616	0.108
Julia	0.00209	0.0194

Parameter sweep using parallel simulation: 7826 runs, 500-second simulation, same solver settings as above, 23 workers. Overheads and post-processing times are excluded.

ENVIRONMENT	4TH-ORDER
Simulink	81.8
Julia	18.3

OMMITTED FEATURES

Some elements of the F-16 data have been omitted to ensure the simplicity of the code. This includes: leading edge extensions, flaps, speed brake, differential horizontal tail deflection, engine model, engine angular momentum, actuator, aerodynamic asymmetry, inertia asymmetry (non-zero I_{xy}), coupling between air density and height.

ACKNOLWEDGEMENTS

This work was supported by the UK Engineering and Physical Sciences Research Council, grant number EP/Y014545/1.

To cite this code, please use the following reference:

Another F-16 simulation? A user-friendly Julia, MATLAB, and Simulink flight simulation package suitable for undergraduate teaching
10th CEAS Aerospace Europe Conference
1-4 December 2025
Turin, Italy
DOI: [to be added]

REFERENCES

Websites

The trajectory visualisation code for MATLAB is a reduced version of the code by Valerio Scordamaglia
uk.mathworks.com/matlabcentral/fileexchange/4572-trajectory-and-attitude-plot-version-2

The 3D model of the F-16 airframe was down-sampled from the model provided in the flight visualisation code by Rodney Rodríguez Robles
uk.mathworks.com/matlabcentral/fileexchange/86453-aircraft-3d-animation

ChatGPT Plus was used to assist with writing Julia code
chatgpt.com

Papers and reports

Aerodynamic data: based on the 1979 NASA report by L. Nguyen et al [1].

Trim routine: the cost function framework is based on appendix B1 in the textbook by Stevens, Lewis, and Johnson [2]. The constrain equations for coordinated turns were taken from the 2008 paper by Marco et al [3].

A description of the 4th-order model can be found in the appendix of [4].

[1] Nguyen, L. T., Ogburn, M. E., Gilbert, W. P., Kibler, K. S., Brown, P. W., and Deal, P. L. "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability," *NASA Technical Paper 1538*. NASA, Langley Research Center, Hampton, Virginia, 1979.

[2] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft Control and Simulation: Dynamics, Controls Design, and Autonomous Systems*, Wiley, 2015. doi: 10.1002/9781119174882

[3] Marco, A. D., Duke, E., and Berndt, J. "A General Solution to the Aircraft Trim Problem", *AIAA Modeling and Simulation Technologies Conference and Exhibit*, AIAA Paper AIAA 2007-6703, 2007. doi: 10.2514/6.2007-6703

[4] Nguyen, D. H., Lowenberg, M. H., and Neild, S. A. "Analysing Dynamic Deep Stall Recovery Using a Nonlinear Frequency Approach", *Nonlinear Dynamics,* Vol. 108, No. 2, 2022, pp. 1179–1196. doi: 10.1007/s11071-022-07283-z