

Virtual Flight Testing of High Performance Fighter Aircraft Using High-Resolution CFD

C. Justin Ratcliff¹, David J. Bodkin², James D. Clifton³, Marcelious Willis, III⁴
United States Air Force SEEK EAGLE Office, Eglin AFB, FL 32542

This paper discusses the current phase of a multi-year development effort to provide a computational method for determining static and dynamic stability and control characteristics of USAF high-performance fighter aircraft. The work presented herein builds on previous efforts utilizing an incremental approach to add simulation capabilities to the current spectrum of computational modeling of aircraft. Static simulations and prescribed motion flight test maneuvers conducted in Computational Fluid Dynamics (CFD) have been accomplished and show good predictive capabilities when compared against wind tunnel data and Lockheed Martin's (LM) performance data. Prescribed motion with aircraft control surface articulation has also been accomplished in CFD and shows an improvement in the resulting aircraft moment coefficient estimations. This paper focuses on the virtual flight test capability achieved by incorporating a pilot model, the F-16 flight control system (FLCS), and six degree-of-freedom (6-DoF) motion computation into the CFD maneuver simulation. Flight test maneuvers were performed in a virtual environment by using CFD to determine the forces and moments acting upon the aircraft and allowing the aircraft to respond as governed by the pilot model, FLCS, and 6-DoF. Virtual flight test simulations were accomplished with a full-scale F-16C aircraft using unstructured, viscous, overset grids and the *Cobalt* solver with *MATLAB* interface. Virtual flight test maneuver response is shown to compare well to validated, flight-test corrected data.

1 Engineer, Stability and Control/Handling Qualities, USAF SEEK EAGLE Office, Eglin AFB, FL

2 Engineer, Stability and Control/Handling Qualities, USAF SEEK EAGLE Office, Eglin AFB, FL

3 Team Lead, Stability and Control/Handling Qualities, USAF SEEK EAGLE Office, Eglin AFB, FL

4 Engineer, Stability and Control/Handling Qualities, USAF SEEK EAGLE Office, Eglin AFB, FL

Definitions

$\dot{\mathbf{v}}_B$	=	body-axis linear acceleration vector
p	=	body-axis roll rate
q	=	body-axis pitch rate
r	=	body-axis yaw rate
Ω_B	=	$\begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix}$
\mathbf{v}_B	=	body-axis linear velocity vector
B_B	=	yaw-pitch-roll inertial-to-body axes rotation matrix
\mathbf{g}_0	=	gravity vector
\mathbf{F}_B	=	body-axis external applied forces vector
m	=	mass
$\dot{\boldsymbol{\omega}}_B$	=	body-axis angular acceleration vector
J	=	inertia tensor
$\boldsymbol{\omega}_B$	=	body-axis angular velocity vector
\mathbf{T}_B	=	body-axis external applied torque vector
ϕ	=	Tait-Bryan roll angle
θ	=	Tait-Bryan pitch angle
$\dot{\Phi}$	=	Tait-Bryan angular acceleration vector
$\xi(\Phi)$	=	$\begin{bmatrix} 1 & \tan(\theta)\sin(\phi) & \tan(\theta)\cos(\phi) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \frac{\sin(\phi)}{\cos(\theta)} & \frac{\cos(\phi)}{\cos(\theta)} \end{bmatrix}$
$\dot{\mathbf{p}}_{NED}$	=	inertial-axis linear velocity vector
$RMSE$	=	$\sqrt{\frac{\sum_{i=1}^n (x_i - x_{ref,i})^2}{n}}$
$NRMSE$	=	$1 - \frac{RMSE}{\max(x_{ref}) - \min(x_{ref})} \quad \dots \quad (1 = \text{perfect fit})$
R	=	$\frac{\sum_{i=1}^n (x_i - \bar{x}) \cdot (x_{ref,i} - \bar{x}_{ref})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \cdot \sum_{i=1}^n (x_{ref,i} - \bar{x}_{ref})^2}}$
$\% \text{ err}$	=	$\frac{ x - x_{ref} }{\max(x_{ref})} \cdot 100$

1. Introduction

Aircraft acquisition programs, especially fighter programs, are costly enterprises. Complicating the already daunting task of designing a high-performance aircraft is the fact that nearly all fighters ever developed have had costly nonlinear aerodynamic or fluid-structure interaction issues not discovered until flight test (FT). The main reason for these “failures” is that the predictive methods used were not able to reveal the onset and nature of the problems early in the design phase. Unfortunately, in future aircraft designs, the problems will only become more complex as thrust vectoring, active aeroelastic structures, and other related technologies are implemented for stability and control (S&C) augmentation. In order to reduce the risk to aircrews during testing and reduce the costs incurred by extensive wind tunnel and flight tests, the effort described herein and in previous publications [1,2,3,4,5] attempts to enable engineers and designers to analyze and predict the nonlinear, flight-dynamic behavior of the aircraft and/or associated armament both early in the design phase and throughout sustainment. Previous papers discuss more fully the nature of the current multi-year effort to develop a comprehensive simulation capability for analyzing and predicting these nonlinear phenomena, and the reader is encouraged to review those works. The present paper focuses on the current phase of the effort wherein computational fluid dynamics (CFD) flow solutions are married with flight control system (FLCS) calculation, six degree-of-freedom (6-DoF) computation, and engine flow simulation to create a virtual flight test maneuver capability for analyzing and predicting these nonlinear phenomena before they negatively impact aircraft acquisition programs. This marriage of capabilities is rolled into a software suite known to the developers as COMSAC, which stands for *COMputational Stability And Control*.

COMSAC is a multi-year effort to develop a software suite for analyzing stability and control characteristics of high-performance fighter aircraft using high-fidelity computational resources such as CFD on DoD HPC servers. COMSAC also serves as the name of the software suite itself. Beginning with static simulations in CFD, COMSAC has progressed through both single-grid, rigid-body prescribed motion and overset-grid, rigid-body prescribed motion. Current efforts are focused on engine modeling, FLCS control, 6-DoF motion, and pilot modeling to create a virtual flight test maneuver in a computational environment. Due to the abundance of validation data and validation tools compared to other aircraft, most of the development work has focused on the F-16. However, COMSAC analysis is easily expanded, and has been expanded to varying degrees, to other aircraft including the A-10, F-15, F-22, and F-35 in multiple configurations at multiple flight conditions.

2. Method

Although prescribed motion maneuvers, as presented in previous works, are a great step toward imitating real-world flight test maneuvers in a computational environment, no simulation or virtualization of flight test maneuvers could be complete without the capability of the simulation to react to and act upon the computational environment. Thus, the current work focuses on the tools necessary to allow the simulated aircraft and virtual environment to interact and react as they would in real flight, producing a “virtual” flight test.

Creating an accurate representation of a real-world flight test maneuver in a virtual environment involves the cooperation of several components, each performing a specific task related to the execution of the maneuver. A flow solver providing forces and moments communicates with a pilot model, FLCS, 6-DoF motion solver, and an engine model to accurately simulate the aircraft in flight. Figure 1 illustrates the basic flow of the simulation.

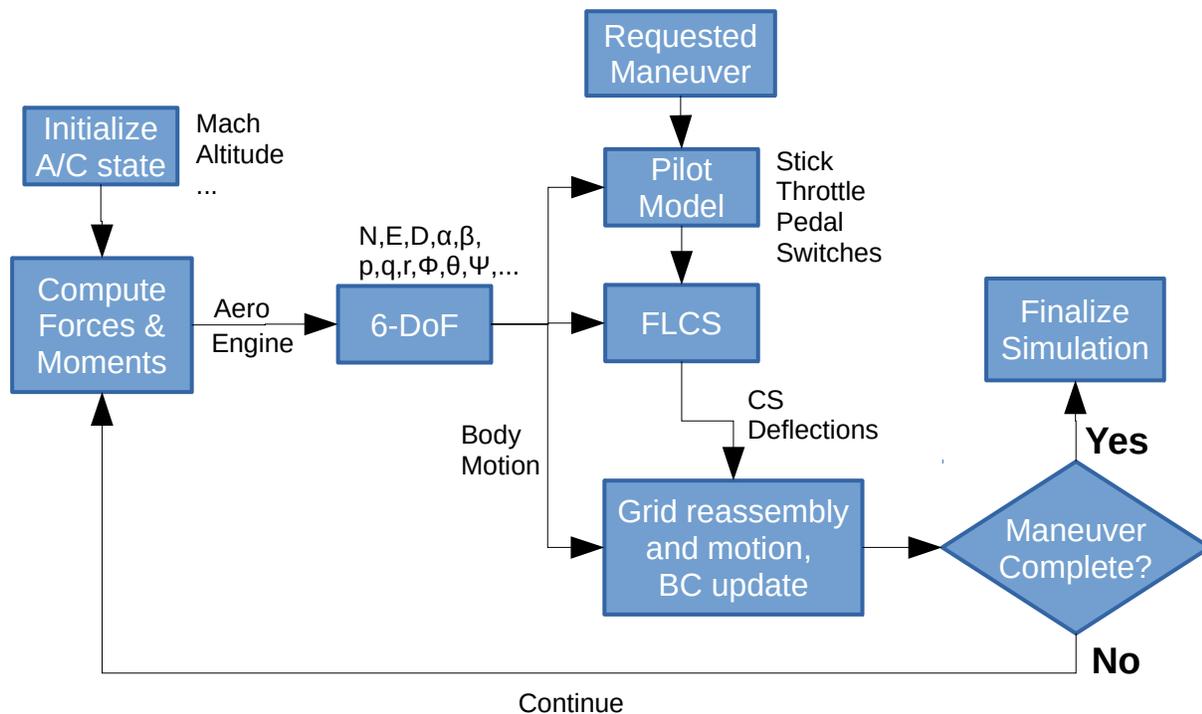


Figure 1: Top-level virtual flight test simulation flow

The flow solver of choice in the current work is *Cobalt*. *Cobalt* has built within it the capability to accept external control to manipulate both the grid motions and the surface boundary conditions (BC) *during* computation of the CFD solution. This ability allows the user to alter both the grid positions for articulating control surfaces using motion control and, for instance, the thrust setting from the engine exit using BC control. Also, 6-DoF code can be coupled with the external control code to have the aircraft react realistically to the changes in forces and moments caused by control surface articulation and boundary condition inputs. External control is handled via MATLAB using a MATLAB interface built into *Cobalt*.

Before a maneuver can be simulated in COMSAC, an aircraft trim must be realized at the desired conditions. This is accomplished by first executing a “pre-trim” wherein the aircraft is translated and oriented into the expected trim condition and orientation (velocity, angles, and control surface deflections). The pre-trim condition is obtained by running a trim simulation utilizing the local aerodynamic database within COMSAC using data for the configuration most similar to the one of interest. The expected trim is then run in CFD, and the flow solution is allowed to stabilize at this pre-trim condition. At this point the FLCS and 6-DoF modules are brought into the simulation for the “real” trim. For the “real” trim, the stick trim input values to the FLCS are determined by varying the input values until rates and accelerations about the aircraft body axes from the 6-DoF are near zero. Control surface deflections are also adjusted during this time to their final trimmed positions. Once a final trimmed state is achieved, the simulation parameters, including the flow solution, are saved in the form of a “restart” for reuse later as the starting point for maneuver simulations starting from that flight condition.

Due to the “real” trim routine being incomplete at the time of this writing, all simulations presented herein were restarted directly from a pre-trim solution. However, the configuration run in all the simulations presented herein is included in the ATLAS aerodynamic database built into COMSAC for computing pre-trim conditions. Thus, the expected trim condition computed during the pre-trim is the actual, real-life trim condition for the configuration according to the validated, flight-test corrected LM aerodynamic data. The pre-trimmed CFD solution was verified by running the CFD simulation with no stick input for one second to determine if translational or rotational rates would build. Results showed that no significant rates developed and the pre-trim was sufficient to begin the maneuvers.

Flow Solver: Cobalt

Cobalt is a cell-centered, finite volume CFD code. It solves the unsteady, three-dimensional, compressible Reynolds Averaged Navier-Stokes (RANS) equations on hybrid unstructured grids. Its foundation is based on Godunov’s first-order accurate, exact Riemann solver. Second-order spatial accuracy is obtained through a Least-Squares Reconstruction. A Newton sub-iteration method is used in the solution of the system of equations to improve time accuracy of the point-implicit method. Strang *et al* [6] validated the numerical method on a number of problems, including the Spalart-Allmaras model, which forms the core for the Detached Eddy Simulation model available in *Cobalt*. Tomaro *et al* [7] converted the code from explicit to implicit, enabling CFL numbers as high as 10^6 . Grismer *et al* [8] parallelized the code, yielding linear speed-up on as many as 2,800 processors. The parallel METIS (PARMETIS) domain decomposition library of Karypis *et al* [9] is also incorporated into *Cobalt*. *Cobalt* has the ability to solve rigid-body static, prescribed motion, and 6-DoF motion simulations. Control surface actuation and weapon separation simulation are also possible using overset grids. Other capabilities include equilibrium air physics and Delayed DES [10].

Cobalt: External Control Interface

A nice feature available within *Cobalt* since V4.0 is an external control interface. The external control interface employed within *Cobalt* allows scripts written by the user to be executed during the CFD simulation. The user’s scripts, written in MATLAB/Simulink, may adjust boundary conditions and/or move grids, overset or not. This capability allows the user to incorporate such custom enhancements as FLCs computation, 6-DoF motion, engine models, pilot models, or any other influencing factor the user desires into the simulation. The scripts are executed by *Cobalt* within a MATLAB/Simulink session instantiated during *Cobalt* initialization. Three “wrapper” scripts called by *Cobalt* control the execution flow of the external controller during initialization, iteration, and finalization stages of the simulation. Within these wrappers, the user may, for instance, direct the simulation to compute control surface deflections, update engine output, move the grids based on control surface deflections and 6-DoF motion, etc. Figure 2 diagrams the basic flow of a *Cobalt* simulation utilizing external control.

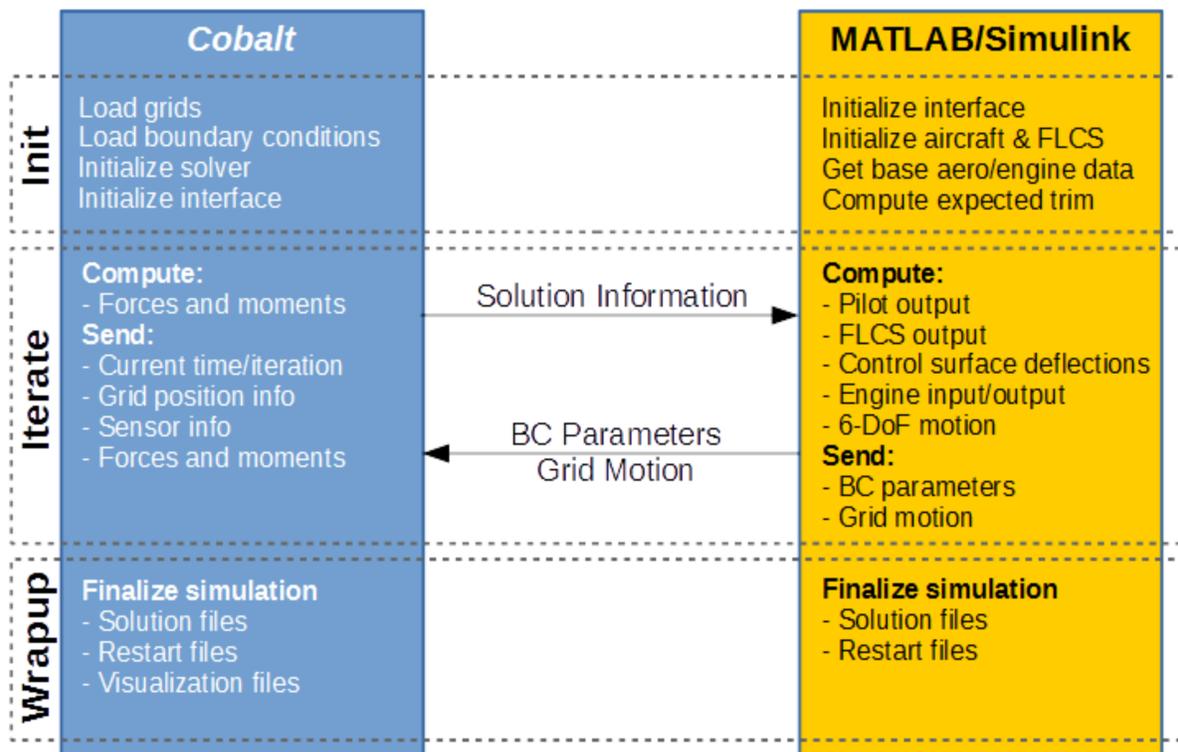


Figure 2: *Cobalt* external control interface details

Simulation Timing

During execution of the simulation, which typically must execute at 1250 Hz or even 2500 Hz to obtain a convergent flow solution, the various components of the simulation (flow solution, 6-DoF, FLCS, pilot, engine, etc.) must operate at appropriate rates to properly simulate the systems and dynamics of the real-world aircraft. For instance, a human's reaction time to external stimuli is on the order of tenths of a second, significantly slower than the simulation. Furthermore, most aircraft FLCS operate on the order of tens or hundreds of Hertz. Thus, timing must be carefully tracked and considered in the execution of the simulation components.

Pilot Model

Aircraft control for performing flight-test-representative maneuvers is effected through setting control stick forces. For simple maneuvers that do not require specific conditions be reached such as a doublet, a time history of stick forces is prescribed. For more complex maneuvers, such as rolling to a specific angle or pulling to a desired G value, a control scheme that uses aircraft states is required. Initially a Proportional, Integral, Derivative (PID) approach is being used. Setting of the gains is performed before running the CFD maneuver through iterative trials on the local desktop simulator. Currently in development is a LQ tracker approach that will remove the requirement for tuning the gains before simulation. This method will use a linear system derived from CFD that represents the aircraft, at the current condition, to calculate required stick forces during the simulation to achieve the desired aircraft state.

Flight Control System

For the work discussed in this paper, flight control system (FLCS) is the F-16 Block 40 Digital Flight Control System model. Work has begun to incorporate other aircraft flight control systems into the COMSAC suite, but as of the date this paper, that work is not complete.

The F-16 FLCS used currently is a Simulink model of the basic roll, pitch, and yaw inner loop control laws. Figure 3 shows a top level view of the FLCS model.

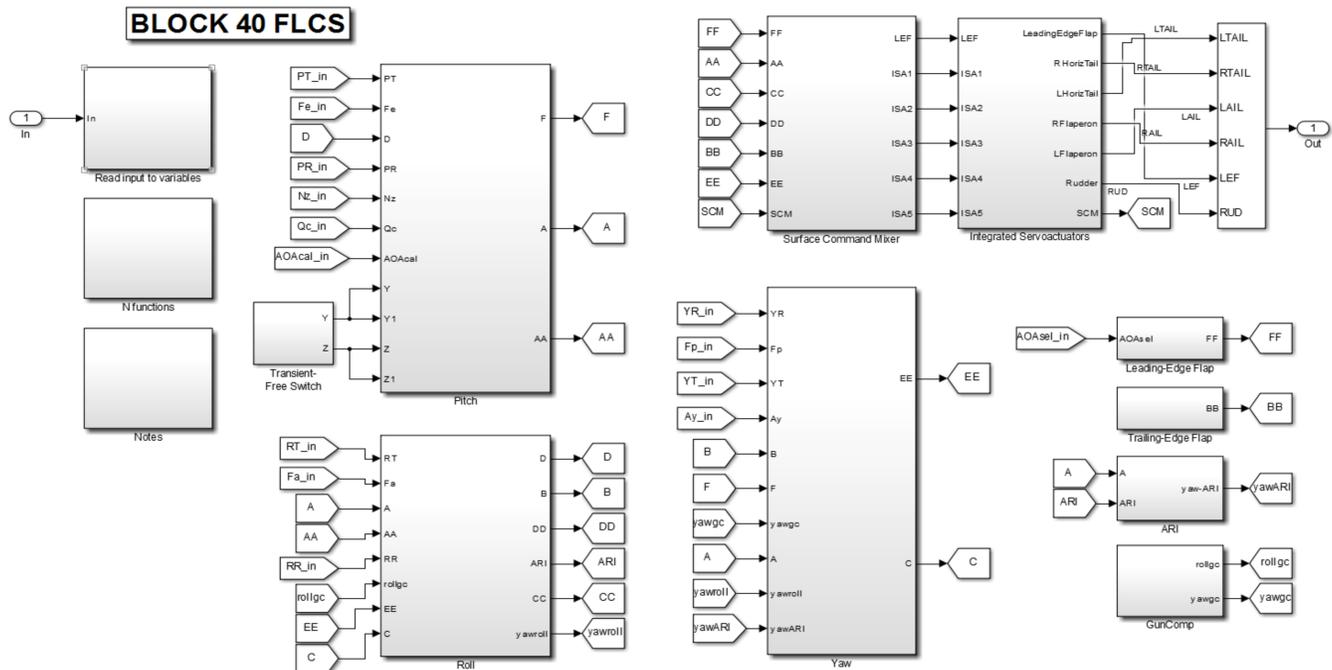


Figure 3: F-16 Block 40 flight control system Simulink model

The FLCS requires 20 inputs including pilot stick and trim inputs, throttle setting, aircraft angles and rates, accelerometer readings, and air data information. The FLCS outputs 6 control surface deflection commands corresponding to the F-16 control surfaces. F-16 leading edge flaps (LEFs) move in unison, so one FLCS output controls both LEFs. Control surface commands are filtered through a transfer function model of the actuator dynamics to get the final control surface deflection each iteration of the simulation. Since the FLCS operates at a much slower frequency than the simulation frequency, the control surface deflection output is appropriately interpolated to smoothly transition the control surfaces according to the FLCS command and actuator transfer function outputs.

6-DoF Simulation

For motion calculation, the Flat-Earth equations are employed. Since the maneuvers of interest last on the order of seconds, the flat earth assumption is acceptable. Equation 1 lists the 6-DoF equation set used.

$$\begin{aligned}
 \dot{\mathbf{v}}_B &= -\Omega_B \mathbf{v}_B + B_B \mathbf{g}'_0 + \frac{\mathbf{F}_B}{m} \\
 \dot{\boldsymbol{\omega}}_B &= -J^{-1} \Omega_B J \boldsymbol{\omega}_B + J^{-1} \mathbf{T}_B \\
 \dot{\boldsymbol{\Phi}} &= \boldsymbol{\xi}(\boldsymbol{\Phi}) \boldsymbol{\omega}_B \\
 \dot{\mathbf{p}}_{NED} &= B_B^T \mathbf{v}_B
 \end{aligned}$$

Equation 1: Flat-Earth 6-DoF equations of motion

The 6-DoF calculation requires force and moment inputs, aircraft mass properties, and previous aircraft state information each iteration. The 6-DoF calculations are performed every simulation iteration, so there is no interpolation of the aircraft motion.

Engine Model

The engine model employed describes the installed flow characteristics of the General Electric F110-GE-100 low-bypass turbofan engine. Thrust, drag, and entrance and exit flow characteristics are tabulated based on Mach, altitude, and throttle setting. A lookup algorithm is employed to extract the necessary data while running COMSAC. During the simulation, thrust is simulated in one of two ways: (1) engine thrust is retrieved from the lookup table and forces and moments due to engine effects are added to the aerodynamic forces and moments computed by the flow solver to provide the total forces and moments, or (2) engine entrance and exit flow characteristics are retrieved from the lookup table and boundary condition parameters are updated within the flow solver, and the total forces and moments acting on the aircraft due to both the aircraft aerodynamics and engine intake and thrust are provided directly by the flow solver. For method 1, engine entrance and exit face boundary conditions are set to farfield. Forces and moments due to other engine effects, such as the gyroscopic moment due to engine shaft rotation, are accounted for within the engine subroutine of the MATLAB interface. Due to the boundary condition control module being incomplete at the time of this writing, only method 1 is employed throughout this paper.

Cobalt: Boundary Condition Control

Modeling the variation of engine input and output during a simulated maneuver with *Cobalt* involves manipulating the boundary condition of the engine entrance and exit patches within the CFD grid. For the engine entrance, a simple mass flow sink is utilized, and the entrance mass flow is adjusted based on the values obtained from the lookup tables. For the engine exit, a source is used, and static pressure, static temperature, and exit flow Mach number are adjusted within *Cobalt* to obtain the appropriate thrust. During each “boundary condition” call to the external control interface, *Cobalt* reads from the MATLAB workspace the updated parameter values for the various externally-controlled boundaries and applies them.

Cobalt: Overset Grid Motion

Manipulating grid motion within *Cobalt* involves the computation of a 3x3 rotation matrix, the X, Y, Z position of the center of rotation, and the dX, dY, dZ position increment from the previous center of rotation for each overset grid. For the F-16C employed here, there are 8 grids: 1 fuselage and 7 control surfaces. During each “motion” call to the external control interface, *Cobalt* reads from the MATLAB workspace an 8x15 matrix containing all the grid motion information and moves and reassembles the overset grids appropriately.

3. Validation Data

This section describes the data used for validating results from dynamic CFD simulations of the USAF F-16C Falcon.

F-16C Lockheed Martin Performance Data

F-16C Block 40 performance data came from two sources. Data that includes scheduled leading edge flaps is based on flight test (FT) results. Data with fixed LEFs is based on 1/9th scale model WT results. Both sets of data have had engine effects removed and are corrected to full scale conditions at their corresponding Mach and altitude. The data is also corrected with Block 40 increments from earlier F-16 variants.

F-16C ATLAS Program

Lockheed Martin's Aircraft Trim, Linearization and Simulation (ATLAS) program is a generalized, 6-DoF, nonlinear, non-real-time simulation. It is a non-real-time version of Lockheed Martin's flying qualities simulator. The aerodynamic database for ATLAS is based on WT test data and includes flight test corrections.

4. Results

The discussion below encompasses studies of the USAF F-16C Falcon without stores using CFD. Virtual flight test results shown start from a pre-trim solution at Mach 0.6 and 10,000 ft. Previous works [4,5] have illustrated the difficulty of realizing acceptable predictions for aerodynamic moments utilizing only static and/or single-grid, rigid-body prescribed motion simulations. Enhanced results were obtained once control-surface articulation was added. The aircraft response of the virtual flight test maneuvers presented herein compares well to Lockheed Martin's ATLAS program. Furthermore, coefficients and derivatives computed from the virtual flight test maneuver aircraft response compare well with those computed from LM validated aerodynamic data.

Eight sets of results are presented: moving control surface grid details, 6-DoF module, FLCS module, 4 virtual flight test maneuvers, and parameter estimation results. The first three are included to illustrate the “step-up” process undertaken to arrive at the current state of virtual flight testing, and to show that the individual pieces comprising the COMSAC suite function nearly identically to their LM ATLAS counterpart when examined in isolation. Two types of virtual flight test maneuvers are presented: classical doublets and modern Schroeder-phased [11], multi-sine [12] input maneuvers. For the doublets, a typical pilot stick or pedal input is applied. For the multi-sine maneuvers, the pilot input is held at trim values (zero input, except throttle, for the cases presented herein), and the multi-sine input signals are injected directly to the control surface after the FLCS computation in addition to the FLCS command. Parameter estimation results are presented to illustrate the ability to extract aircraft model parameters from virtual flight test responses. Goodness-of-fit statistics (NRMSE, R², and % error) are presented in tables for the virtual flight test maneuvers and the parameter estimation models to compare *Cobalt* results to those of ATLAS.

Time-accurate simulations were run on 512 - 2,048 cores at a time step of 0.0008 seconds and with 7 Newton sub-iterations. All grids are unstructured and were created with SolidMesh [13], a solid modeling and unstructured grid generation system, and the AFLR3 grid generator [14,15] (Mississippi State) or Pointwise. Full-span grid size was approximately 39 million cells for a clean F-16C with control surfaces modeled and LAU-129 tip launchers. All grids are unstructured mixed element grids containing tetrahedral and five and six sided

pentahedral elements. An initial boundary layer spacing corresponding to $y^+ = 1$ was specified for all grids. Additionally, transparent surfaces were used in areas of interest to capture vortical flow and shedding.

F-16 Overset Grid: Moving Control Surfaces

An unstructured overset method was used to simulate moving control surfaces. Gaps were introduced between the wing and the control surface to allow the surfaces to rotate about the hinge line without intersecting. Figure 4 below left, is a snapshot of the F-16 grid with control surfaces cutout. It can be seen that much of the increase in grid size, an increase of 31.10 million cells, is due to the dense point spacing around the surface gaps – the darkened areas around the control surfaces. To create gaps between the fuselage and the LEFs, flaperons, and rudder, part of the control surface was removed. On the long edges (hinge line) of these control surfaces, material was removed from the non-moving aircraft surface. To allow rotation, the inner edge of the control surface was rounded and a matching offset surface created in the wings and vertical stabilizer. As the horizontal tails are all moving, a gap was created by translating them away from the body. A gap distance of 0.25” was utilized to allow enough space to avoid problems with grid reconstruction during simulated movement. A downside to gap cutting is that material is removed from the model and air is free to flow through the gaps. This is not representative of actual aircraft design, and the effects have yet to be determined. Figure 4 below right, shows the inboard intersection of the wing (green), LEF (magenta), and body (orange) on the F-16 in the non-deflected position.

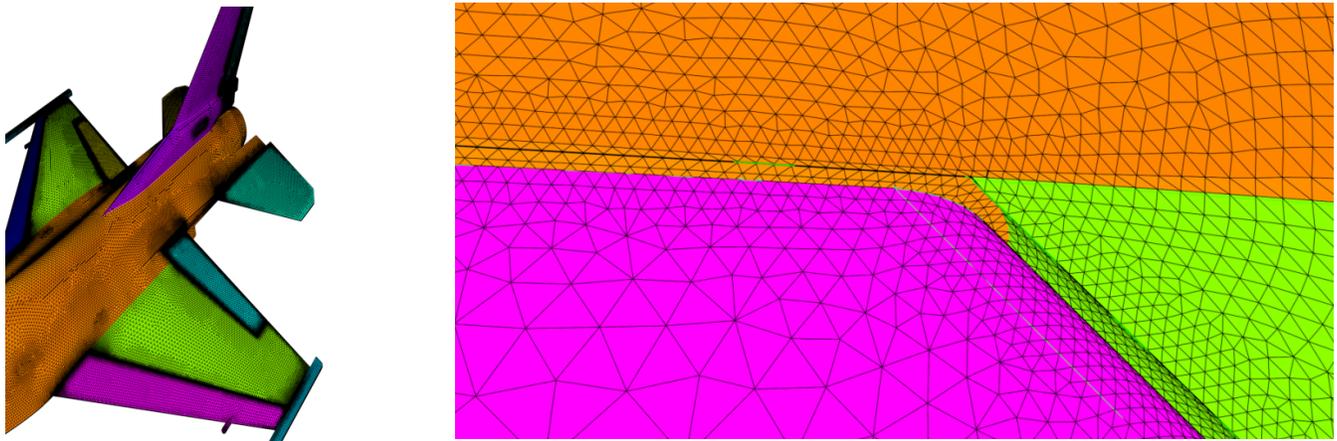


Figure 4. F-16 grid with control surfaces cutout (left) and the inboard intersection of the wing (green), LEF (magenta), and body (orange) (right).

Figure 5 depicts the F-16 horizontal tail and LEFs at different instances during a pitch doublet simulation. The horizontal tail is shown at center, maximum leading edge down, and maximum leading edge up deflections, and the LEF is shown at the starting in-flight condition and maximum leading edge down deflection. Color contours depict pressure variations over the surface during the maneuver.

In early testing, the amount of time per iteration utilizing 512 processors on the DoD HPC machine Garnet was approximately 6 minutes per iteration for the grid with all moving control surfaces modeled. Recent versions of *Cobalt* have employed improvements aimed at increasing the overset grid reassembly and flow solution computation efficiency. Recent tests with *Cobalt* V6+ and V7+ have resulted in a time per iteration of roughly 53 seconds on 512 processors on the DoD HPC machine Spirit.

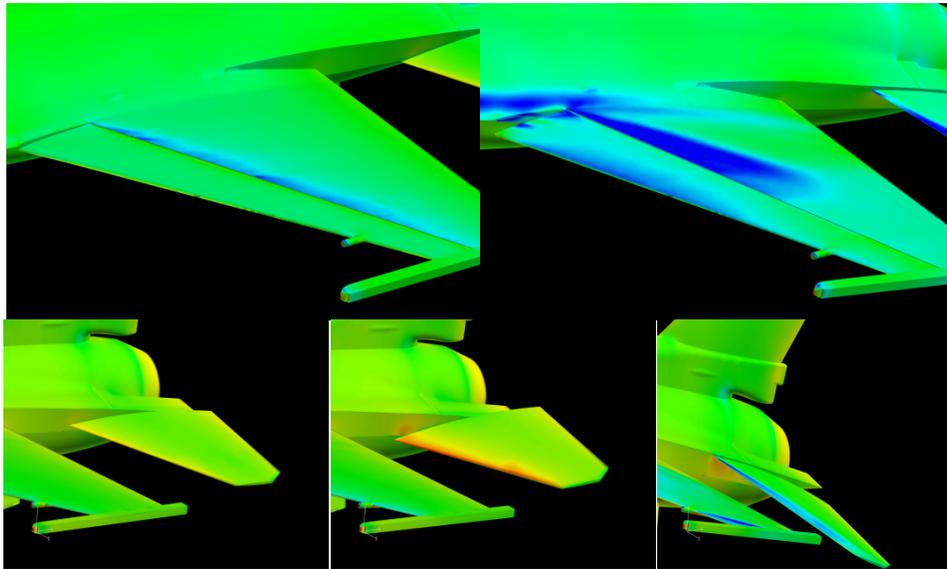


Figure 5: Overset Grid of Clean F-16 with moving horizontal tails and LEFs; Prescribed motion pitch doublet at Mach 0.6 and 10,000 feet.

Six-Degree of Freedom Module: F-16 Wind-up Turn

A 6-DoF module was created to simulate aircraft flight due to changes in applied forces and moments. For the purposes of COMSAC, CFD is used to obtain aerodynamic force and moment data for new configurations while a local aerodynamic database is employed once data is obtained from wind-tunnel testing, flight testing, or CFD. The 6-DoF module utilizes MATLAB's ODE45 routine and the flat earth approximation. To ensure the model was producing accurate results, an F-16 ATLAS simulation of a wind-up turn (WUT) was conducted at Mach 0.6 and 10,000 feet. The time history of the maneuver is shown in Figure 6. The 6-DoF output trace from COMSAC is practically indistinguishable from that of ATLAS for the same inputs.

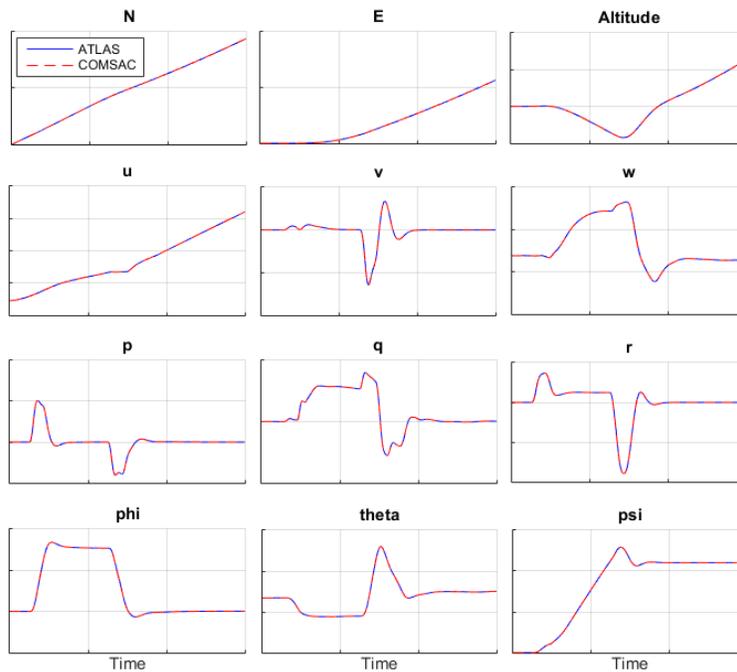


Figure 6. 6-DoF output comparison between F-16 ATLAS and COMSAC; WUT at Mach 0.6 and 10,000 feet.

Flight Control System Module: F-16 360 degree Right Roll

The F-16 Block 40 inner loop control laws have been programmed in MATLAB/Simulink. F-16 ATLAS is being used to validate and verify the Simulink control law models within COMSAC. To do this, a maneuver is simulated in ATLAS and COMSAC using the same inputs. A time history of the control surface deflections during the maneuver is shown in Figure 7. Notice that the Simulink model prediction for control surface deflections follows ATLAS's deflections.

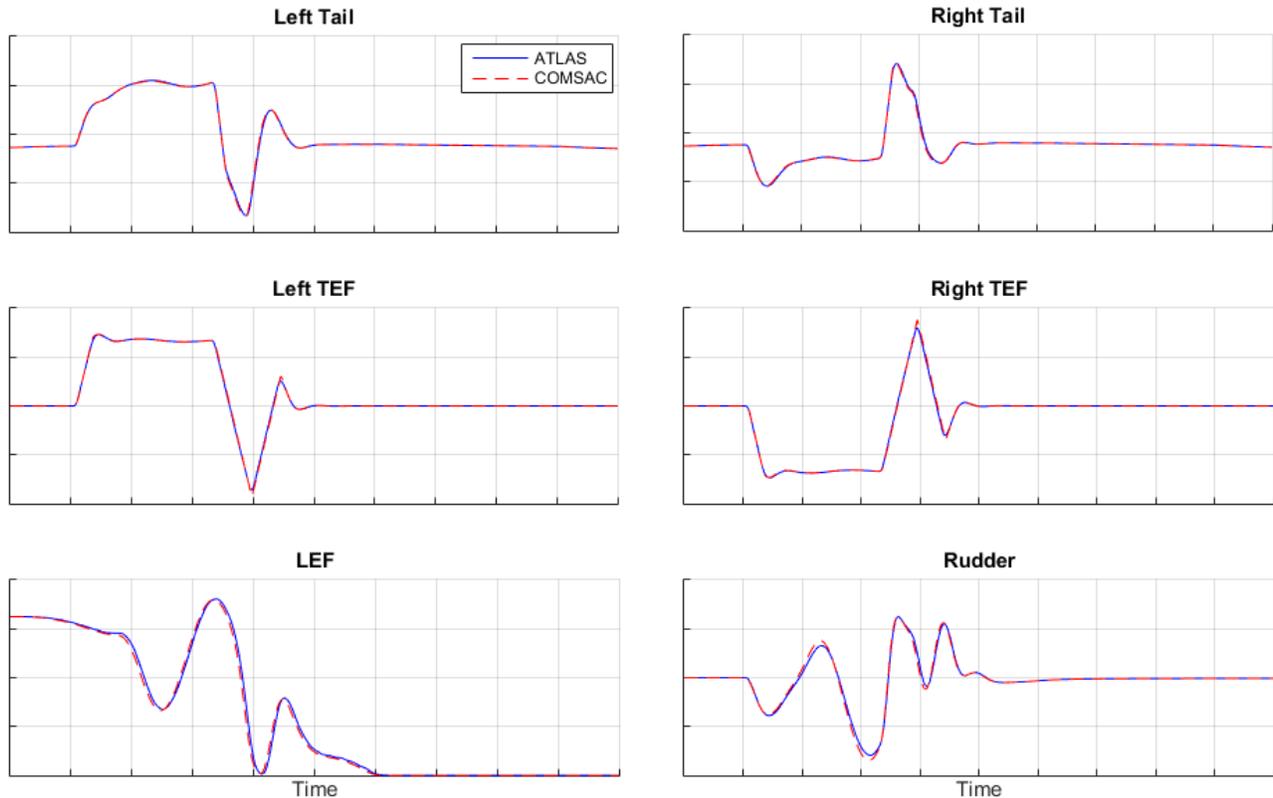


Figure 7. Control surface deflection output comparison between F-16 ATLAS and COMSAC; 360 deg Right Roll at Mach 0.6 and 10,000 feet.

F-16 Virtual Flight Test: Pitch Doublet

A virtual pitch doublet was performed in COMSAC using *Cobalt* on DoD HPC system Spirit. The maneuver was executed on 512 processors, and the average computation time per iteration was approximately 53 seconds. The maneuver was restarted from a pre-trim solution at Mach 0.6 and 10,000 ft. Figure 8 shows the pitch stick input for the maneuver. The same maneuver performed in LM F-16 ATLAS served as validation data. Pitch stick input was supplied as a time history taken from the ATLAS maneuver.

Figure 9 shows the lift, drag, and pitching moment coefficient traces throughout the maneuver. The virtual flight test maneuver shows good agreement with validation data for both force and moment coefficients. Trend and magnitude are seen to match well. Table 1 shows the goodness-of-fit statistics between the ATLAS and COMSAC results. Of note is the fact that there appears to be a slight time offset in the aircraft response of the virtual flight test maneuver as compared to the ATLAS maneuver. This is currently under investigation.

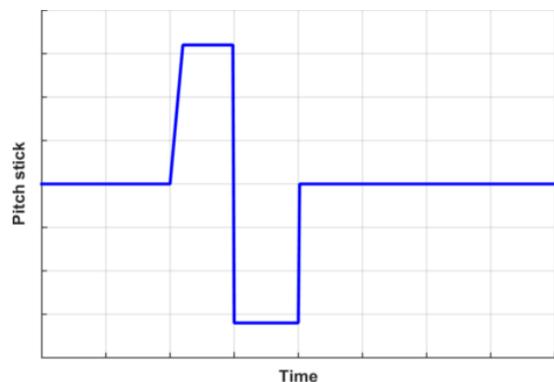


Figure 8: Pitch doublet pitch stick input

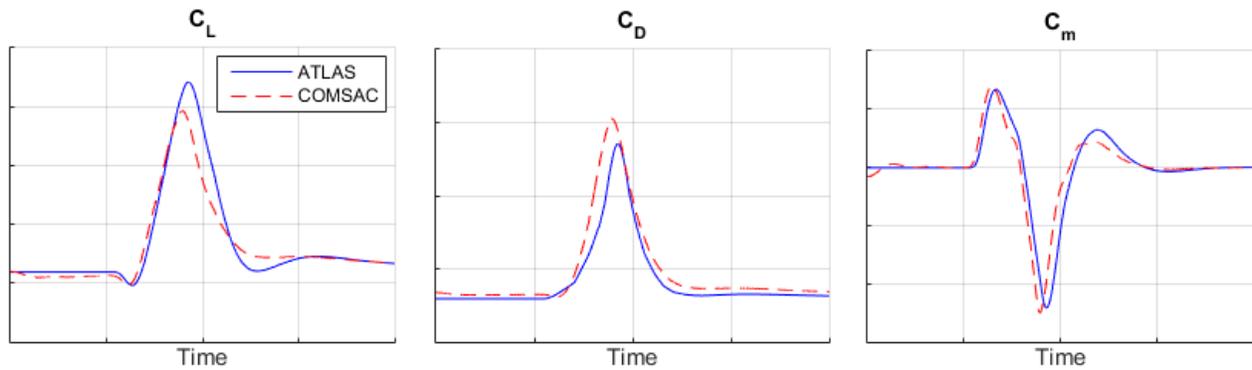


Figure 9: C_L , C_D , and C_m comparison between F-16 ATLAS and COMSAC; Virtual pitch doublet at Mach 0.6 and 10,000 ft.

Coeff	NRMSE	R ²	Max % err	Mean % err	Median % err
C_L	0.92	0.93	22.2	3.4	1.8
C_D	0.90	0.95	32.8	4.8	2.7
C_m	0.93	0.85	40.7	5.9	1.5

Table 1: C_L , C_D , and C_m comparison statistics between F-16 ATLAS and COMSAC; Virtual pitch doublet at Mach 0.6 and 10,000 ft.

F-16 Virtual Flight Test: Yaw Doublet

A virtual yaw doublet was performed in COMSAC using *Cobalt* on DoD HPC system Spirit. The maneuver was executed on 512 processors, and the average computation time per iteration was approximately 53 seconds. The maneuver was restarted from a pre-trim solution at Mach 0.6 and 10,000 ft. Figure 10 shows the yaw pedal input for the maneuver. The same maneuver performed in LM F-16 ATLAS served as validation data. Yaw pedal input was supplied as a time history taken from the ATLAS maneuver.

Figure 11 shows the side force, rolling moment, and yawing moment coefficient traces throughout the maneuver. The virtual flight test maneuver shows good agreement with validation data for both force and moment coefficients. Trend and magnitude are seen to match well. Table 2 shows the goodness-of-fit statistics between the ATLAS and COMSAC results. As with the pitch doublet, there appears to be a slight time offset in the aircraft response of the virtual flight test maneuver as compared to the ATLAS maneuver. This is currently under investigation.

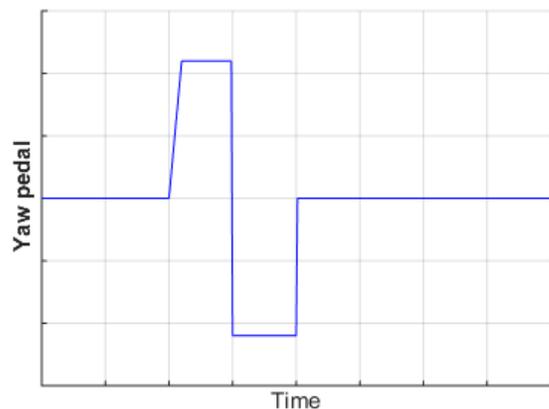


Figure 10: Yaw doublet yaw pedal input

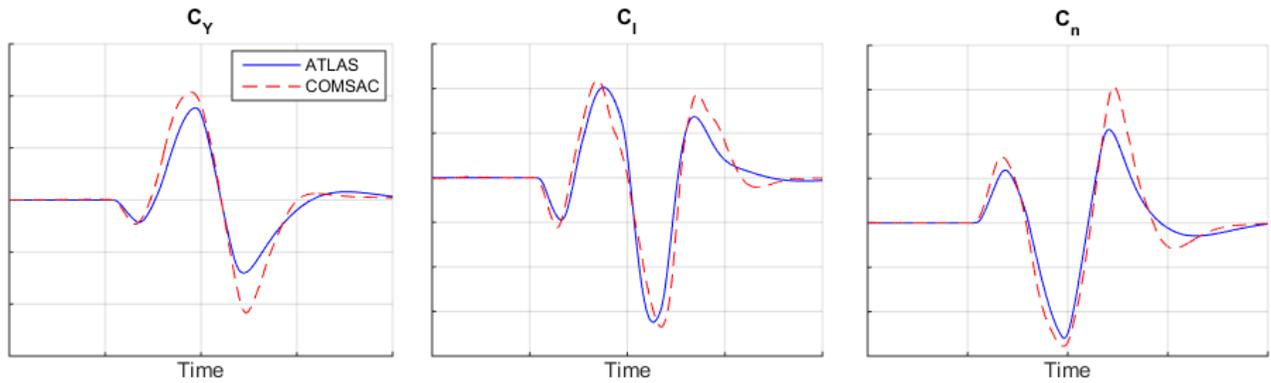


Figure 11: C_Y , C_l , and C_n comparison between F-16 ATLAS and COMSAC; Virtual yaw doublet at Mach 0.6 and 10,000 ft.

Coeff	NRMSE	R^2	Max % err	Mean % err	Median % err
C_Y	0.92	0.97	44.0	8.7	4.3
C_l	0.93	0.91	34.9	7.4	4.9
C_n	0.93	0.96	43.5	8.1	4.2

Table 2: C_Y , C_l , and C_n comparison statistics between F-16 ATLAS and COMSAC; Virtual yaw doublet at Mach 0.6 and 10,000 ft.

F-16 Virtual Flight Test: Multi-sine Signal Injection – Longitudinal

A virtual maneuver was performed in COMSAC using *Cobalt* on DoD HPC system *Spirit*. The maneuver was executed on 2048 processors, and the average computation time per iteration was approximately 31 seconds. The maneuver was restarted from a pre-trim solution at Mach 0.6 and 10,000 ft. The control surface deflections were commanded as direct signal injections to the control surfaces computed as Schroeder-phased, multi-sine input signals [11,12,16,17,18] while the pilot stick, pedal, and throttle inputs were held at trim values. Figure 12 shows the signal inputs and associated control surface deflections for the maneuver. The FLCS distortion of the input signals was not accounted for in the design of the input signals. The same maneuver performed locally in COMSAC using the ATLAS database served as validation data.

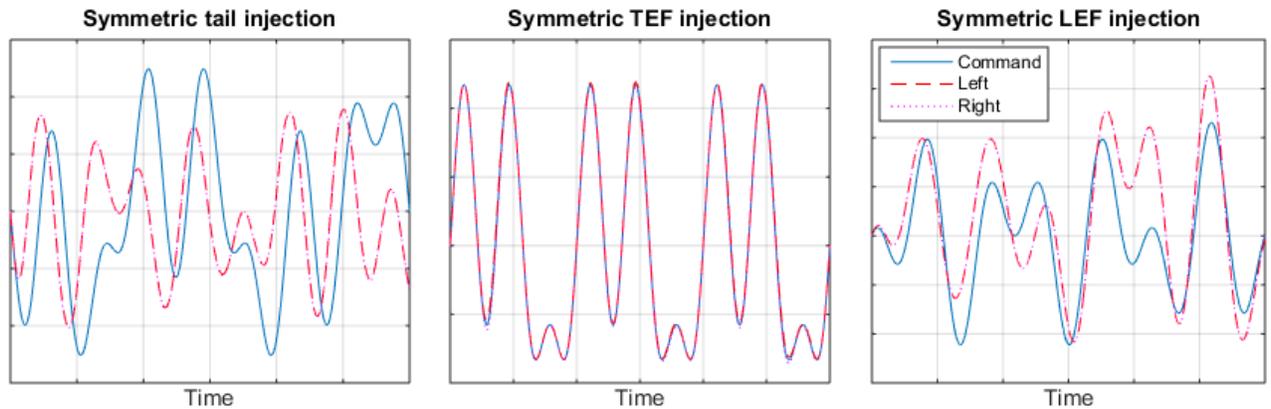


Figure 12: Signal injection commands and control surface deflections; Virtual longitudinal maneuver at Mach 0.6 and 10,000 ft.

Figure 13 shows the lift, drag, and pitching moment coefficient traces throughout the maneuver. The virtual flight

test maneuver shows good agreement against the same maneuver computed with validation data for both force and moment coefficients. Trend and magnitude are seen to match well. Table 3 shows the goodness-of-fit statistics between the ATLAS data and *Cobalt* data. Drag shows some discrepancy, though the scale is less significant than it appears as shown in the figure.

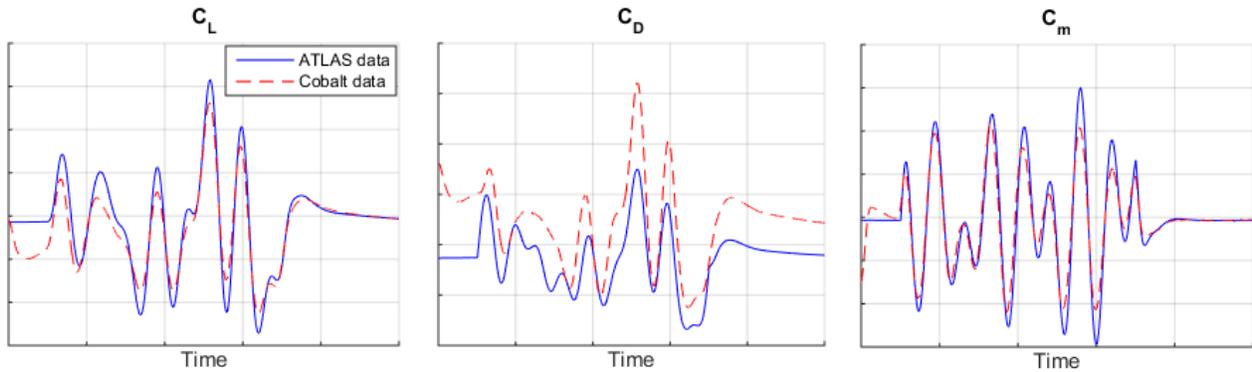


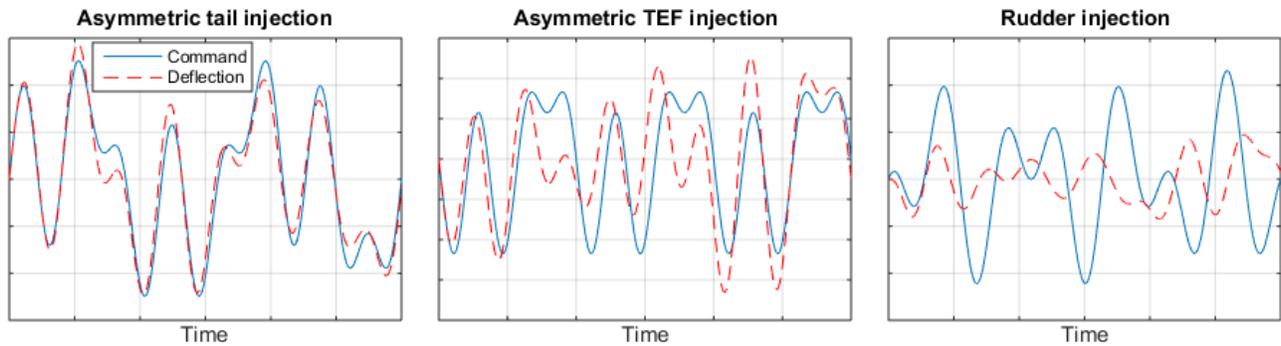
Figure 13: C_L , C_D , and C_m comparison between ATLAS data and *Cobalt* data; Virtual longitudinal maneuver at Mach 0.6 and 10,000 ft.

Coeff	NRMSE	R^2	Max % err	Mean % err	Median % err
C_L	0.94	0.90	5.6	1.9	1.3
C_D	0.74	0.81	11.5	4.5	4.4
C_m	0.95	0.94	46.4	7.6	5.6

Table 3: C_L , C_D , and C_m comparison statistics between ATLAS data and *Cobalt* data; Virtual longitudinal maneuver at Mach 0.6 and 10,000 ft.

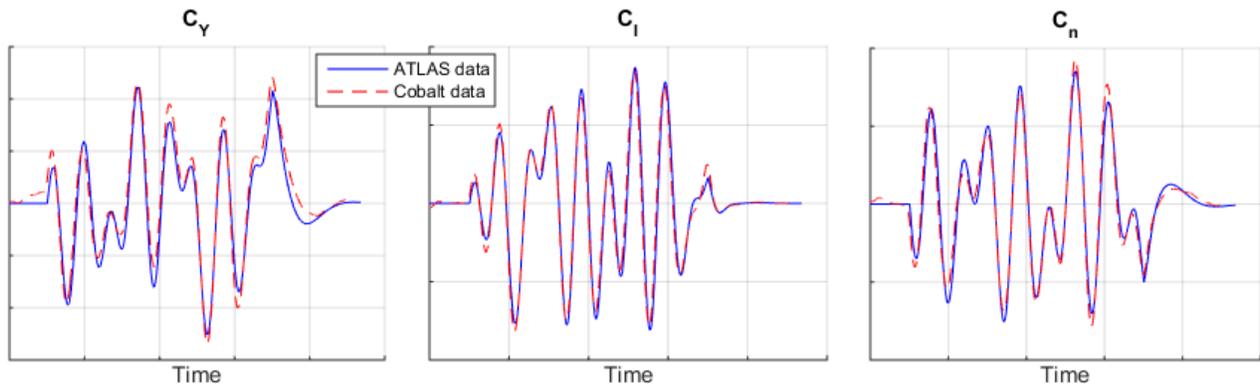
F-16 Virtual Flight Test: Multi-sine Signal Injection – Lateral-Directional

A virtual maneuver was performed in COMSAC using *Cobalt* on DoD HPC system Spirit. The maneuver was executed on 1024 processors, and the average computation time per iteration was approximately 40 seconds. The maneuver was restarted from a pre-trim solution at Mach 0.6 and 10,000 ft. The control surface deflections were commanded as direct signal injections to the control surfaces computed as Schroeder-phased, multi-sine input signals [11,12,16,17,18] while the pilot stick, pedal, and throttle inputs were held at trim values. Figure 14 shows the signal inputs and associated control surface deflections for the maneuver. The FLCS distortion of the input signals was not accounted for in the design of the input signals. The same maneuver performed locally in COMSAC using the ATLAS database served as validation data.



**Figure 14: Signal injection commands and control surface deflections;
Virtual lateral-directional maneuver at Mach 0.6 and 10,000 ft.**

Figure 15 shows the side force, rolling moment, and yawing moment coefficient traces throughout the maneuver. The virtual flight test maneuver shows good agreement against the same maneuver computed with validation data for both force and moment coefficients. Trend and magnitude are seen to match well. Table 4 shows the goodness-of-fit statistics between the ATLAS data and *Cobalt* data.



**Figure 15: C_Y , C_l , and C_n comparison between ATLAS data and *Cobalt* data;
Virtual lateral-directional maneuver at Mach 0.6 and 10,000 ft.**

Coeff	NRMSE	R ²	Max % err	Mean % err	Median % err
C_Y	0.95	0.97	19.3	7.8	7.1
C_l	0.97	0.97	19.8	4.2	2.2
C_n	0.97	0.97	19.6	4.9	3.5

**Table 4: C_Y , C_l , and C_n comparison statistics between ATLAS data and *Cobalt* data;
Virtual lateral-directional maneuver at Mach 0.6 and 10,000 ft.**

F-16 Virtual Flight Test: Parameter Estimation

Linear models of the six primary stability and control coefficients were generated using parameter estimation techniques [19,20] on the data obtained from the virtual flight tests. New longitudinal and lateral-directional maneuvers on which the models were not based were run in LM ATLAS and served as validation data. Two sets of models were generated. The first model set was obtained using data from the aforementioned virtual flight tests conducted locally in COMSAC using ATLAS data. The second was obtained using data from the same flight tests conducted on HPC resources in COMSAC using *Cobalt*. Figure 16 shows the lift, drag, and pitching moment model predictions against the new longitudinal maneuver. As seen in the figure, for lift and pitching moment, the

models from both data sets capture trend and magnitude quite well throughout the maneuver, with some slight discrepancies in magnitude for some of the peaks. For drag, trend is captured well, however the peak magnitude computed by the models shows some considerable discrepancy from the maneuver. Of note, however, is the fact that the models from both data sets predict drag very similarly. Table 5 shows the goodness-of-fit statistics between the ATLAS maneuver data and the *Cobalt* model.

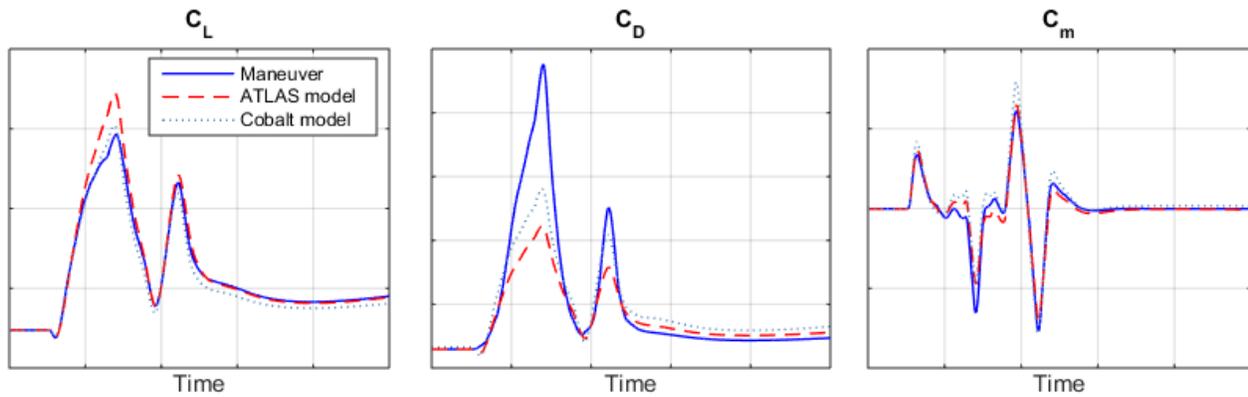


Figure 16: C_L , C_D , and C_m parameter estimation model predictions; Longitudinal maneuver at Mach 0.6 and 10,000 ft.

Coeff	NRMSE	R ²	Max % err	Mean % err	Median % err
C_L	0.96	0.99	8.3	2.8	2.8
C_D	0.90	0.94	41.3	5.6	3.5
C_m	0.96	0.94	25.0	4.7	2.4

Table 5: C_L , C_D , and C_m parameter estimation model comparison statistics; Longitudinal maneuver at Mach 0.6 and 10,000 ft.

Figure 17 shows the side force, rolling moment, and yawing moment model predictions against the new lateral-directional maneuver. For all coefficients, trend is captured quite well throughout the maneuver while magnitude is predicted marginally well. It is unclear how the inaccuracies in the computed coefficient magnitudes actually effect aircraft performance predictions, and studies to investigate the effects are planned. Of note again is the similarity between the predictions of the ATLAS-based and *Cobalt*-based models. Table 6 shows the goodness-of-fit statistics between the ATLAS maneuver data and the *Cobalt* model.

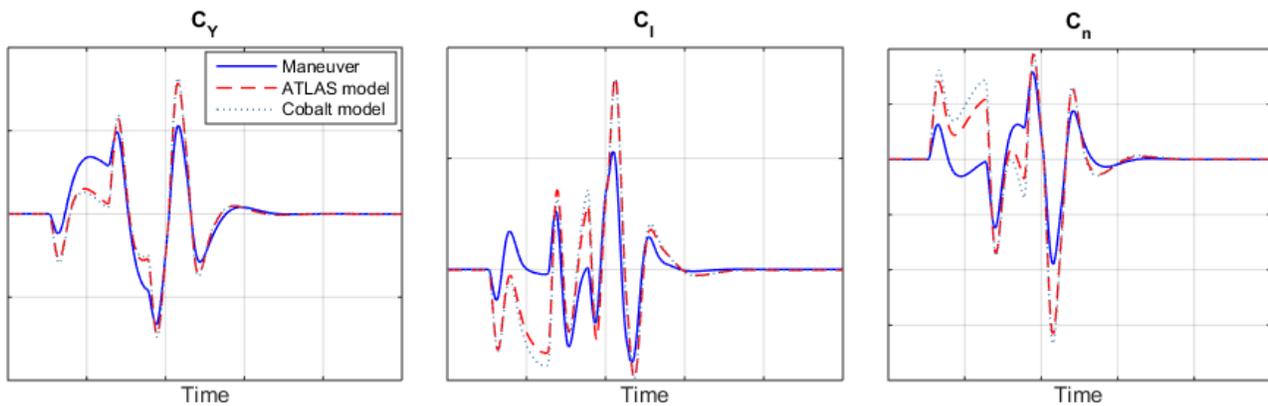


Figure 17: C_Y , C_l , and C_n parameter estimation model predictions; Lateral-directional maneuver at Mach 0.6 and 10,000 ft.

Coeff	NRMSE	R ²	Max % err	Mean % err	Median % err
C _Y	0.92	0.80	43.4	8.7	1.6
C _l	0.84	0.48	77.7	16.8	3.3
C _n	0.84	0.55	78.1	17.3	3.3

Table 6: C_Y, C_l, and C_n parameter estimation model comparison statistics; Lateral-directional maneuver at Mach 0.6 and 10,000 ft.

5. Conclusions

A method for performing virtual flight tests of high-performance fighter aircraft using high-resolution CFD has been presented. The approach presented herein utilizes the high-resolution CFD solver *Cobalt* run on HPC resources coupled with full 6-DoF computation, FLCS simulation, an engine model, and a pilot model. The method utilizes a full-span, full-size overset aircraft grid with articulating control surfaces to simulate the response of an aircraft to pilot stick inputs and/or directly-injected control surface deflection signals. Results compare favorably with Lockheed Martin validation data.

The results presented herein indicate the presented approach is a viable means of accurately determining the stability and control characteristics of an aircraft in a computational environment. However, some discrepancies remain to be resolved. Engine entrance and exit boundary conditions and the subsequent flow solution computation at these faces likely contribute significantly to the discrepancies noted in the lift and drag coefficient results. Pitching moment is much more accurate compared to previous results obtained using prescribed motion maneuvers without control surface articulation [4,5]. The apparent time offset seen in some results requires further investigation and will need to be resolved to ensure the accuracy of future results. Furthermore, the injection signal manipulation by the FLCS affects the properties of the input signals and subsequently affects the ability to accurately execute parameter estimation on the aircraft response.

Considering the similarity in the drag prediction between the models of both data sets, the discrepancy in predicting drag is likely due to inaccurate model structure and/or attempting to predict a maneuver outside the acceptable prediction space of the model. The drag model structure will likely require expansion to include other parameters and/or nonlinear terms to more accurately predict drag. Furthermore, the model source data will need to be expanded to include a larger region of the flight envelope around the selected flight condition. The same could be said for all the lateral-directional coefficients as well.

Also contributing to inaccuracies in computing and predicting drag is the current modeling of engine flow using a farfield scheme. Engine flow accuracy is expected to improve as the boundary condition manipulation code is completed and employed and engine entrance and exit plane boundary conditions within CFD are converted from farfield to sink and source respectively.

The anticipation is that, through several CFD runs, models can be generated that span the aircraft's flight envelope. Once generated, these models can be used to perform a complete matrix of flight test maneuvers in seconds (rather than days or weeks in CFD) as part of a pre-flight check in support of flight test planning and risk reduction. However, highly nonlinear regimes and envelope expansion will still require the performance of complete virtual maneuvers in CFD to refine test plans and reduce test risk and cost.

As confidence in CFD analysis grows, CFD can be used to optimize available test resources and aid clearance of new stores when either test resources or a suitable analogy to previously cleared stores is unavailable. However, the reliability of the CFD solver to produce acceptable results in a timely fashion is paramount to complete integration of CFD into the engineering workflow that supports augmentation of warfighter capability. The criticality of DoD HPC resources is self-evident to perform such analyses in a timely fashion, and highlights the need for continued development and expansion of HPC resources. The capabilities outlined here and those under

development represent another step towards the end goal impacting the design phase of the acquisition process and rapidly delivering new capabilities to the warfighter at reduced risk and cost.

Acknowledgments

The computational resources were generously provided by the DoD HPCMP, U.S. Army Engineer Research and Development Center, and Air Force Research Laboratory Department of Defense Shared Resource Center. The authors gratefully acknowledge Dr. Eugene A. Morelli from NASA Langley, who provided the System Identification Programs for Aircraft (SIDPAC), and *COBALT* Solutions for their support. Their work is greatly appreciated.

- 1 Dean, J.P., Morton, S.A., McDaniel, D.R., Clifton, J.D., and Bodkin, D.J. "Aircraft Stability and Control Characteristics Determined by System Identification of CFD Simulations." *AIAA Paper 2008-6378*. AIAA Atmospheric Flight Mechanics Conference. Honolulu, HI. August 2008.
- 2 Dean, J.P., Clifton, J.D., Bodkin, D.J., Morton, S.A., and McDaniel, D.R., "Determining the Applicability and Effectiveness of Current CFD Methods in Store Certification Activities." *AIAA Paper 2010-1231*. AIAA Aerospace Sciences Meeting. Orlando, FL. January 2010.
- 3 Dean, J.P., Clifton, J.D., Bodkin, D.J., and Ratcliff, C.J., "High Resolution CFD Simulations of Maneuvering Aircraft Using the CREATE-AV/Kestrel Solver." *AIAA Paper 2011-1109*. 49th AIAA Aerospace Sciences Meeting. Orlando, FL. January 2011.
- 4 Clifton, J.D., Ratcliff, C.J., Bodkin, D.J., and Dean, J.P. "Determining the Stability and Control Characteristics of High-Performance Maneuvering Aircraft Using High-Resolution CFD Simulation with and without Moving Control Surfaces." *AIAA Paper 2012-4656*. AIAA Atmospheric Flight Mechanics Conference. Minneapolis, MN. August 2012.
- 5 Clifton, J.D., Ratcliff, C.J., Bodkin, D.J., and Dean, J.P. "Determining the Stability and Control Characteristics of High-Performance Maneuvering Aircraft Using High-Resolution CFD Simulation with and without Moving Control Surfaces." *AIAA Paper 2013-0972*. AIAA Aerospace Sciences Meeting. Grapevine, TX. January 2013.
- 6 Strang, W. Z., Tomaro, R. F., and Grismer, M. J. "The Defining Methods of Cobalt60: a Parallel, Implicit, Unstructured Euler/Navier-Stokes Flow Solver." *AIAA Paper 1999-0786*. 1999.
- 7 Tomaro, R. F., Strang, W. Z., and Sankar, L. N. "An Implicit Algorithm for Solving Time Dependent Flows on Unstructured Grids." *AIAA Paper 1997-0333*. 1997.
- 8 Grismer, M. J., Strang, W. Z., Tomaro, R. F., and Witzeman, F. C. "Cobalt: A Parallel, Implicit, Unstructured Euler/Navier-Stokes Solver." *Adv. Eng. Software*, Vol. 29, No. 3-6, pp. 365-373. 1998.
- 9 Karypis, G., Schloegel, K., and Kumar, V. "*Parmetis: Parallel Graph Partitioning and Sparse Matrix Ordering Library, Version 3.1*." Technical Report. Dept. of Computer Science, University of Minnesota. 2003.
- 10 Spalart, P.R., Deck, S., Shur, M.L., Squires, K.D., Strelets, M.Kh., and Travin, A. "A New Version of Detached-Eddy Simulation, Resistant to Ambiguous Grid Densities." *Theoretical Computational Fluid Dynamics*, Vol. 20, pp. 181-195. 2006.
- 11 Schroeder, Manfred. "Synthesis of Low-Peak-Factor Signals and Binary Sequences with Low Autocorrelation (Corresp.)." *Information Theory, IEEE transactions on* 16.1 (1970): 85-89.
- 12 Morelli, Eugene A. "Multiple Input Design for Real-Time Parameter Estimation in the Frequency Domain." Paper REG-360, *13th IFAC Conference on System Identification*, August 2003, Rotterdam, Netherlands.
- 13 Gaither, J.A., Marcum, D.L., and Mitchell, B. "SolidMesh: A Solid Modeling Approach to Unstructured Grid Generation." *7th International Conference on Numerical Grid Generation in Computational Field Simulations*. 2000.
- 14 Marcum, D.L. and Weatherill, N.P. "Unstructured Grid Generation Using Iterative Point Insertion and Local Reconnection." *AIAA Journal*, Vol. 33, No. 9, pp 1619-1625. September 1995.

- 15 Marcum, D.L. "Unstructured Grid Generation Using Automatic Point Insertion and Local Reconnection." *The Handbook of Grid Generation*, edited by J.F. Thompson, B. Soni, and N.P. Weatherill, CRC Press, p. 18-1. 1998.
- 16 Hartmann, William Morris, and Jon Pumplin. "Periodic Signals with Minimal Power Fluctuations." *The Journal of the Acoustical Society of America* 90.4 (1991): 1986-1999.
- 17 Morelli, Eugene A. "Flight-test Experiment Design for Characterizing Stability and Control of Hypersonic Vehicles." *Journal of Guidance, Control, and Dynamics* 32.3 (2009): 949-959.
- 18 Morelli, Eugene A. "Flight Test Maneuvers for Efficient Aerodynamic Modeling." *Journal of Aircraft* 49.6 (2012): 1857-1867.
- 19 Klein, V. and Morelli, E.A., *Aircraft System Identification – Theory and Practice*, AIAA Education Series. AIAA. Reston, VA. 2006.
- 20 Morelli, Eugene A. "System Identification Programs for Aircraft (SIDPAC)." *AIAA Atmospheric Flight Mechanics Conference*. 2002.