A Flight Dynamics Model for Exploring the Distributed Electrical eVTOL Cyber Physical Design Space

James D. Walker
Southwest Research Institute
San Antonio, TX USA
james.walker@swri.org

F. Michael Heim
Southwest Research Institute
San Antonio, TX USA
frederick.heim@swri.org

Bapiraju Surampudi
Southwest Research Institute
San Antonio, TX USA
bapiraju.surampudi@swri.org

Pablo Bueno
Southwest Research Institute
San Antonio, TX USA
pablo.bueno@swri.org

Alexander Carpenter
Southwest Research Institute
San Antonio, TX USA
alexander.carpenter@swri.org

Sidney Chocron
Southwest Research Institute
San Antonio, TX USA
sidney.chocron@swri.org

Jon Cutshall
Southwest Research Institute
San Antonio, TX USA
jon.cutshall@swri.org

Richard Lammons
Southwest Research Institute
San Antonio, TX USA
richard.lammons@swri.org

Theodore Bapty
Vanderbilt University
Nashville, TN USA
theodore.a.bapty@vanderbilt.edu

Brian Swenson
Southwest Research Institute
San Antonio, TX USA
brian.swenson@swri.org

Sydney Whittington
Southwest Research Institute
San Antonio, TX USA
sydney.whittington@swri.org

***Abstract*—As part of DARPA's Symbiotic Design for Cyber Physical Systems program, software tools were developed for a UAV and air taxi eVTOL design challenge. These included the development of a corpus of parts, an assembly technique, and then a 6-degree-of-freedom flight dynamics model with an autopilot based on trim states and a linear quadratic regulator to fly the assembled air vehicle. This paper describes the flight dynamics model and autopilot and controls in some detail. It also presents examples of parts, designs and performance.**

Keywords—flight dynamics model, unmanned aerial vehicle (UAV), air taxi, eVTOL, design tools, cyber physical modeling

I. INTRODUCTION (*HEADING 1*)

As part of DARPA's Symbiotic Design for Cyber Physical Systems program, a design challenge was needed for performers who were developing new design tools. The Air Taxi (Hybrid or Electric) aeroNautical Simulation (ATHENS) team put together tools and parts and design challenges for UAVs and air taxis with eVTOL capabilities (electric vertical takeoff and landing). In design, the UAVs are smaller and the air taxis are larger and can take a payload of 1 to 4 passengers on short trips, with optional cargo, in a ground-traffic-congested metropolitan area, implying maneuvering capabilities. The technologies that are enabling the rapid growth in this aeronautical vehicle area are distributed electric propulsion, enabled by improved batteries and electric motors, and improved controls, enabled by advances in sensors and computational processing capability.

Developing the design challenge software environment takes two items: a collection or corpus of parts, such as batteries, motors, propellers, wings, and body parts, that are available from current manufacturers, and a flight simulation environment for determining the performance of the vehicle designed from those parts.

This paper addresses the flight simulation software as developed by the SwRI team, which we refer to as the Flight Dynamics Model (FDM). The 6-degree-of-freedom model works with the state vector of the assembled vehicle, which includes the translational and rotational accelerations in the body frame of the aircraft, the quaternion connecting the body frame from the world (laboratory) frame, the location of the aircraft in the world frame, the revolution rates of the motors, and the currents in various parts of the electrical system. This state vector is updated through an ordinary differential equation based on physics assumptions of flight and electrical behaviors of the electrical system which includes the separate vector of controls. An autopilot is also supplied with the FDM, where various weights are available that influence the behavior of the autopilot and hence the controls input. The flight dynamics model combined with the autopilot make it possible to simulate the flight of the air vehicle on assigned paths and score the performance. The performance depends both on the physical, mechanical design of the air vehicle and on the cyber, controller performance.

II. THE FLIGHT DYNAMICS MODEL

The flight dynamics model is based on the physics of flight with regards to the following components of an electrically-distributed vehicle:

- Propellers

- Electric motors
- Batteries or a auxiliary power unit supplying electrical power
- Wings and horizontal and vertical stabilizers
- Body structural components which may include a fuselage.

Each of these items has its involvement with the flight of the vehicle, and we will describe how each is included in the model. Our main references for developing the model were [1] and [2].

The vehicle state is contained in a state vector, which is assumed centered at the center of mass. This vector is comprised of the following

$$\vec{x}^T(t) = (U\ V\ W\ P\ Q\ R\ q_0\ q_1\ q_2\ q_3\ x\ y\ z\ \Omega_i\ Q_j) \qquad (1)$$

This vector represents the current state of the vehicle, where the various items follow standard assumptions in aeronautical modeling

- (U, V, W) is the vehicle velocity in the body frame of the aircraft (i.e., the frame that is rigidly attached to the vehicle); it is assumed that the forward direction is X, to the right Y, and down Z (down is positive). This frame is right handed. In aeronautics literature this frame is referred to as *frd* for forward, right, down.
- $(P, Q, R) = \vec{\Omega}$ is the rotation of the body, where P is about the X axis (roll rate), Q about the Y axis (pitch rate), and R about the Z axis (yaw rate).
- $\vec{q}$ is the four-component quaternion that represents the rotation connecting the world frame to the body frame.
- (x, y, z) is the location in the world frame (or laboratory frame – a fixed frame) – i.e., where the aircraft is positioned in space.
- Ω_i is the spin (or rotation) rate for the i^{th} motor in radians per second, and each motor has an entry in the state vector.
- Q_j is the drawn charge from the j^{th} battery, and each battery has an entry in the state vector.

In addition, there is a vector of controls $\vec{u}$ that provide information to the electrical motors and the servos that operate flaps and ailerons. It is assumed each control channel ranges from 0 to 1.

The physics, contained in a routine called fderiv, allows for updating the state vector in time by providing the time derivative of the state vector. The routine knows about the geometry of the aircraft (which we don't explicitly show in the equation), and with that and the current state vector $\vec{x}$ and set of controls $\vec{u}$, it provides the time derivative of the state vector,

$$\frac{d\vec{x}}{dt} = \vec{f}(\vec{x}, \vec{u}). \qquad (2)$$

When the aircraft is flying, this routine is used to integrated the motion of the aircraft through a 4th order Runge-Kutta scheme.

The details of $\vec{f}$ are in how the forces are computed. In particular, the propellers provide a thrust, the wings provide lift and drag, there is a body drag, and there is gravity. We will go in reverse of the previous sentence's list, which is the order of difficulty.

To determine the gravity, it is necessary to know which way is down in the body frame. The orientation information comes from the quaternions, and the direction cosine (rotation) matrix which connects the velocity in the world frame $\vec{v}$ with the velocity in the body frame $\vec{V}$ is $R(\vec{q})$, where $\vec{V} = R(\vec{q})\vec{v}$. Thus, the gravitation force on the center of mass of the vehicle.

$$\vec{F}_g = R(\vec{q})\hat{z}, \qquad (3)$$

where $\hat{z} = (0, 0, 1)$ is the unit vector pointing down in the world frame. Capital F means force in the body frame (again, the frame that is rigidly attached to the aircraft and moves with it).

Next, the body drag is computed based on flow through the air. Because of this, it is assumed in the rest of this section that the speed is the speed of the vehicle in their air, which is updated from the body frame speed to account for the air being in motion (wind) in the world frame,

$$\vec{U} \ \text{ replaced by } \ \vec{U} - R(\vec{q})\vec{v}_{wind}. \qquad (4)$$

The body drag is then based on the approximation

$$\vec{F}_b = -\frac{1}{2}\rho \begin{pmatrix} X_{fuse,uu}|U + 2v_i|(U + 2U_{in}) \\ Y_{fuse,vv}|V + 2V_{in}|(V + 2V_{in}) \\ Z_{fuse,ww}|W + 2W_{in}|(W + 2W_{in}) \end{pmatrix} \qquad (5)$$

where $X_{fuse,uu}$, $Y_{fuse,vv}$, $Z_{fuse,ww}$ are constants (thinking of the fuselage) and for UAVs we set it equal to a constant times the presented area in each body coordinate axis. This approximation is based on plate or brick geometries and it is readily apparent that it does not provide the correct body drag force for a sphere (because in this formulation the drag need not point in the direction of travel – for that to occur it would need to be a constant times $\vec{U} + 2\vec{U}_{in}$, which it is not in general). $\vec{U}_{in}$ is wind speed induced by the propellers, and in theory should always be taken into account, but often is not included since we are not computing detailed flows around the aircraft. The body drag brings up another complication since it need not be the case that the center of the drag force on the body is the same as the center of mass. Thus, the body drag can give rise to a moment $\vec{M}_b$ where

$$\vec{M}_b = \left(\vec{X}_{fuse} - \vec{X}_{cm}\right) \times \vec{F}_b, \qquad (6)$$

and where $\vec{X}_{fuse}$ is the aircraft's acting center of the body force, $\vec{X}_{cm}$ is the center of mass, and $\times$ is the cross product.

For the wings, we use data from wind tunnels for the various NACA airfoils. Since we are interested in motion in many directions and replicating flight for the range of quadcopters to fixed-wing aircraft and many innovative designs, we view wings as half wings (or wing segments) and treat them with a given center and include the flow over the wing due to rotation of the aircraft as well as induced flow from the propellers. For a wing segment with center $\vec{X}_w$, the local wing velocity is given by

$$\vec{U}_w = \vec{U} + \vec{\Omega} \times \left(\vec{X}_w - \vec{X}_{cm}\right) + \eta_w \vec{U}_{in} \quad (7)$$

It is assumed the wing leading edge lies in the Y-Z plane and that the normal to the surface of the wing, $\vec{n}_w$, also lies in the Y-Z plane. The wing or wing segment can be at any angle (hence also representing vertical tails and V-tails) and we define a speed $W' = (0, V, W) \cdot \vec{n}_w$. A first step in determining the behavior of the wings is computing the angle of attack. For aircraft, the angle of attack α and the sideslip angle β are given by where the U_w is used for U,

$$\alpha = \tan^{-1}(W'/U), \quad \beta = \sin^{-1}\left(V'/\sqrt{U^2 + V^2 + W^2}\right) \quad (8)$$

The control surfaces (ailerons and/or flaps) are represented by adjusting the angle of attack of the wing segment, so that

$$\alpha \quad \text{replaced by} \quad \alpha + \tau\delta \quad (9)$$

where τ is an effectiveness and the deflection angle is δ. Typically the deflection will be given by

$$\delta = \delta_{min} + (\delta_{max} - \delta_{min})u_c \quad (10)$$

where $\delta_{min} < 0 < \delta_{max}$ and $0 \le u_c \le 1$, so that $u_c = 1$ represents lowering the flap to their full extent (hence increasing the effective angle of attack which leads to increased lift). The NACA tables then given lift and drag as a function of the angle of attack. To provide a lift and drag for all directions, the approach outlined in [1] pp. 649-650 is followed, but in the end leads to a lift force and a drag force for each wing segment, and these are summed,

$$\vec{F}_L = \sum \vec{F}_{Lj}(\vec{U}_{wi}, \alpha_i), \ \vec{F}_D = \sum \vec{F}_{Dj}(\vec{U}_{wi}, \alpha_i), \quad (11)$$

There are further adjustments due to three dimensional effects of finite wing length and taper. Similarly, there are moments due to the application of these forces to a center of the wing as compared to the center of mass.

We now discuss the propellers and electric motors, the most complex part of the model. The body-frame geometric center of a propeller is $\vec{X}_p$ and the unit normal to the drive shaft is $\vec{n}_p$. We need to know the speed of the air flow past the propeller supposing it was not spinning. That speed is given by

$$\vec{V}_p = \left(\vec{V} + \vec{\Omega} \times \left(\vec{X}_p - \vec{X}_{cm}\right)\right) \cdot \vec{n}_p, \quad (12)$$

where the dot denotes the dot product. For propeller performance, we use tables that are computed for each propeller design that provide its performance in terms of the spin (or rotation) rate and what is called the advance ratio J. In traditional usage and tables, the spin rate n is given in rotations per second, and is thus determined from the state vector quantity from $n = \Omega/2\pi$. The advance ratio is given by $J = V_p/(nD)$, where D is the dimeter of the propeller. The tables then provide the thrust of the propeller and the power of the propeller as

$$F_p = C_T(n, J)\rho n^2 D^4, \ P = C_p(n, J)\rho n^3 D^5 \quad (13)$$

Once we have the spin rate n for each propeller, we sum over all propellers to get a final, total force from the propellers

$$\vec{F}_p = \sum_j (F_p \vec{n}_p)_j. \quad (14)$$

In addition to the force, moments are applied to the flight body due to the torque applied to the propeller and the torque from the propeller forces being offset from the center of mass.

Determining the spin is fairly complex. We do that by matching the mechanical torque from the table and the torque supplied by the electric motor. To do this requires solving a cubic equation which is done with Newton's method. To be explicit, the mechanical torque is given by

$$\tau_{prop}(\Omega) = \frac{P(\Omega)}{\Omega} = \frac{1}{2\pi} A C_p(n(\Omega), J(n(\Omega)))\rho n(\Omega)^2 D^5 \quad (15)$$

The torque from the electric motor is

$$\tau_{motor} = k_T(I - I_o) \quad (16)$$

where I is the current and I_0 is the idle current – current that always goes through the motor, even when it is not spinning. We assume that I_0 is not a function of the applied voltage. We assume that the control channel driving the motor has $0 \le u_c \le 1$. The relation between the current and the voltage is

$$V_{motor} = V_{battery} u_c = \frac{\Omega}{k_V} + I \cdot R_w \quad (17)$$

The equation matching torques is

$$\tau_{motor} = k_T \left(\frac{V_{motor}}{R_w} - \frac{\Omega}{k_V R_w} - I_o\right) = \tau_{prop} = \frac{P(\Omega)}{\Omega} \quad (18)$$

For each motor this equation is iteratively solved for Ω. Once determined, the current draw on the battery can be found. It can be that the air velocity is so high that the propeller can provide no thrust, given the voltage. This can be recognized if the solution has $I < I_0$. When this occurs, the current is set to the idle current and the thrust to zero. The presence of speed controllers are included through additional electrical losses.

III. Rigid Body Dynamics

After the forces and moments are calculated, the rigid body dynamics of the aircraft is determined. The equation for the translational acceleration in the body frame is

$$\frac{d\vec{U}}{dt} = -\vec{\Omega} \times \vec{U} + \frac{1}{m}(\vec{F}_g + \vec{F}_b + \vec{F}_L + \vec{F}_D + \vec{F}_p) \quad (19)$$

where m is the mass of the aircraft. The equation for the rotational acceleration in the body frame is

$$\frac{d\vec{\Omega}}{dt} = I_m^{-1}(\vec{M}_b + \vec{M}_L + \vec{M}_D + \vec{M}_p - \vec{\Omega} \times (I_m \vec{\Omega})) \quad (20)$$

where I_m is the moment of inertia tensor about the center of mass.

The quaternions are updated in standard fashion, though it was important to minimize drift to include a weighted constraint equation in the updating scheme [2] (pg. 47 and 48).

The updated position in the world frame is determined by

$$\frac{d\vec{x}_{pos}}{dt} = R(\vec{q})^T \vec{U} \quad (21)$$

The changes in the propeller rotation rates are determined through the matching-torque calculation described above. It is assumed that there is a delay in the requested spin rate and that actual spin rate. In the model this delay is represented by a delay

time τ which is typically set to 0.05 seconds. The ordinary differential equation for the actual motor spin rate Ω_i is

$$\frac{d\Omega_i}{dt} = -(\Omega_i - \Omega)/\tau. \quad (22)$$

Finally, the charge used from each battery is updated through

$$\frac{dQ_j}{dt} = I_j \quad (23)$$

In usage, when this charge reaches 80% of the charge on the fully charged battery, we say the battery is fully used. To go beyond this charge usage in a realistic fashion would require us to adjust battery voltage as the battery charge was depleted.

IV. Controls

The intent of the DARPA Symbiotic Design for Cyber Physical Systems program is to explore the interaction of the physical design and the cyber design. To that end, the FDM also has an autopilot that works as follows.

A basic background to the autopilot is the determination of trim states. These are found through (2) by adjusting the orientation of the vehicle (through the quaternions) and the controls until a specified translational speed for a straight line or a curved path is determined. Numerically these are found using the MINPACK package [3] and the nonlinear simplex algorithm [4]. The objective is to drive the translational and rotational accelerations to zero. To the magnitude of those accelerations is added weighting of the total power usage. The reason for including electrical power in the objective function is to identify a best trim state, as typically there are many.

Given a trim state, we approximately linearize (2) about that trim state in terms of the state vector and control vector,

$$\vec{f}(\vec{x},\vec{u}) \approx A(\vec{x} - \vec{x}_{trim}) + B(\vec{u} - \vec{u}_{trim}) + \vec{G} \quad (24)$$

The linearized trim state is then used to find a set of controls by means of the linear quadratic regulator problem, by minimizing the functional

$$\int_0^{+\infty} (\vec{x}^T Q \vec{x} + \vec{u}^T R \vec{u}) dt \quad (25)$$

where Q and R and weighting matrices. A solution to this equation is given by the algebraic Riccati equation [5]

$$Q + PA + A^T P - PBR^{-1}BP = 0 \quad (26)$$

Where P is a nonnegative definite matrix P. Within our code this is done with the package RICPACK [6]. Once P is found, the control law is

$$\vec{u} = \vec{u}_{trim} + K(\vec{x} - \vec{x}_{trim}) \quad \text{where} \quad K = -R^{-1}B^T P \quad (27)$$

where the state vector has been adjusted to move it from the current flight orientation to the trim state orientation. The cyber control of the autopilot comes through the weights in Q and R. We currently have Q and R as diagonal matrices, defined in pieces, where Qv weights the body frame translational velocity, Qr weights the body frame angular velocity, Qq weights the reduced quaternions, and Qp weights the world frame positions from a zero reference. Similarly, R is a diagonal matrix of weights for the controls (which in some sense relates to minimizing power). Adjusting these weights has a large influence on the controls

V. Parts Corpus

To design an air vehicle, a corpus of parts is supplied. The corpus consists of both parametrically defined components and commercial-off-the-shelf (COTS) parts. For ease of virtual construction, each corpus class was modeled by one representative CAD part, which was automatically dimensioned by the performer's selection of discrete COTS part or continuous parameters (Fig. 1). The majority of structural parts were continuously defined via select parameters to enable vast spatial customization; these include wings, tubes, connectors, plates, and flanges. Batteries, motors, propellers, electronic speed controllers (ESC), and servos were defined via discrete choices from a list of COTS parts.

UAV seed designs were assembled from the corpus parts, typically using multiples from each class. These construction recipes were given to the performers as a baseline design from which they could choose to build from; they also had the choice to create completely novel configurations (Fig. 2). Each part in an assembly was dimensioned and weighed in CAD, enabling the calculation of realistic moments of inertia for use within the FDM. Furthermore, most parts in the corpus contained performance parameters, such as voltage or capacity for batteries. This information was tracked, along with critical part orientation information from the CAD model to the FDM input file.

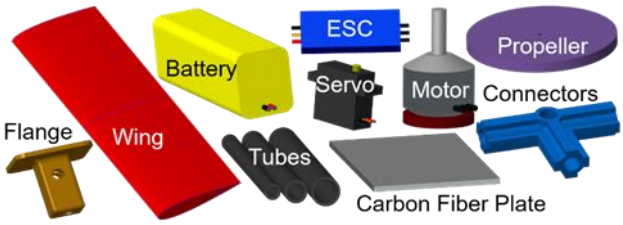

Fig. 1. CAD model of each design corpus class

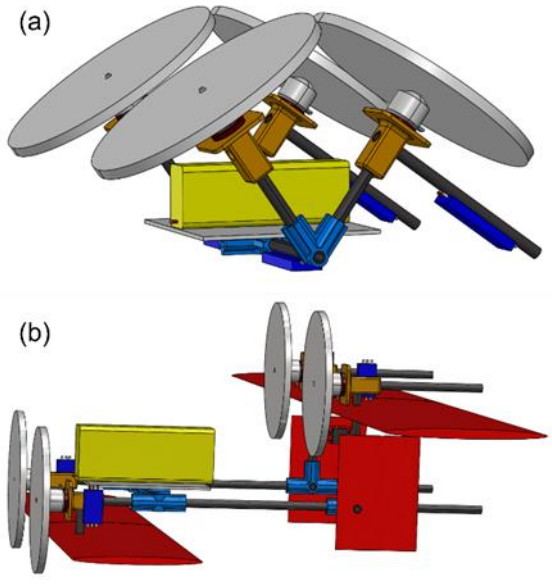

Fig. 2. CAD rendering of novel UAV designs. (a) Angled thrust quadcopter. (b) Tandem wing tail-sitter.

The design is assembled through the OpenMETA tools using CREO for the final CAD assembly (such as was done for DARPAs Adaptive Vehicle Make program [7]). After a vehicle is put together, the specific geometry is assembled and detailed geometric information is supplied as an input file to the FDM model.

VI. Example

Flight performance of novel eVTOL UAV designs (Fig. 2) was calculated over a series of trim states and flight paths; for demonstration purposes the tandem wing tail-sitter design, named "the Axe" (Fig. 2b) will be discussed further. As required of all designs, it can ascend vertically and fly horizontally.

The Axe was an unconventional design that performed well over all available paths and could successfully complete the most challenging flight path, the oval (Fig. 3), at the upper most limit flight speed of 50 m/s. Therefore, it is interesting to note its flight characteristics in those conditions. In Table 1 the trim state output can be seen, which details that the aircraft is very slightly pitched up and is close to the limits of its motors.

The FDM computes and outputs many of the state variables (Fig. 4). For the oval, the Axe begins in a straight section and continues for about 15 seconds; it then pitches and rolls slightly to make it through the first turn after around 35 seconds. It then does this again for a total flight time of nearly 70 seconds. Table II shows estimated costs of materials.

VII. Conclusions

A 6-degree-of-freedom flight simulator and corresponding autopilot was developed for use in the DARPA Symbiotic Design for Cyber Physical Systems program. It has been used to design UAVs and will be used to design Air Taxi's in the near future. The integration of the flight dynamics model and the controls algorithms allows straightforward exploration of the interplay between physical and cyber aspects of designs, furthering the efforts to find unique designs given the enable technologies of distributed electrical propulsion systems and advanced diagnostics and controls.

Acknowledgements

This work was supported by DARPA under the Symbiotic Design for Cyber-Physical Systems with contract FA8750-20-C-0541. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. We also thank the feedback of multiple participants of the TA1 and TA2 teams from SRI, Siemens, Peraton, Vanderbilt, MIT, Berkeley, CRA, and STR, for feedback on the tools that helped (and continue) to guide the capabilities of the system. Finally, we thank MetaMorph Inc., for their support of the OpenMETA framework.

References

[1] B. L. Stevens, F. L. Lewis, and E. N. Johnson, Aircraft Simulation and Control, 3rd ed., Hoboken, New Jersey, Wiley, 2016.

[2] M. E. Dreier, Introduction to Helicopter and Tiltrotor Flight Simulation, 2nd ed.Reston, Virginia, AIAA, 2018.

[3] J. J. More, D. C. Sorensen, and K. E. Hillstrom, User Guide for MINPACK-1, Argonne National Laboratory Report ANL-80-74, Argonnne, IL, 1980

[4] J. A. Nelder and R. Mead, "A simplex method for function minimization," Computer J., vol. 7, pp. 308-313, 1965.

[5] J. Jabczyk, Mathematical Control Theory: An Introduction, Boston, Birkauser, 1992.

[6] W. F. Arnold and A. J. Laub, "Generalized eigenproblem algorithms and software for algebraic Riccati equations," in Proceedings of the IEEE, vol. 72, no. 12, pp. 1746-1754, Dec. 1984, doi: 10.1109/PROC.1984.13083.

[7] J. D. Walker, S. Chocron, M. S. Moore, and G. C. Willden, "Blast and ballistic survivability analysis tools for design optimization developed in DARPA's Adaptive Vehicle Make (AVM)," 2015 NDIA Ground Vehicle Systems Engineering and Technology Symposium, Novi, Michigan, August 4-6, 2015.

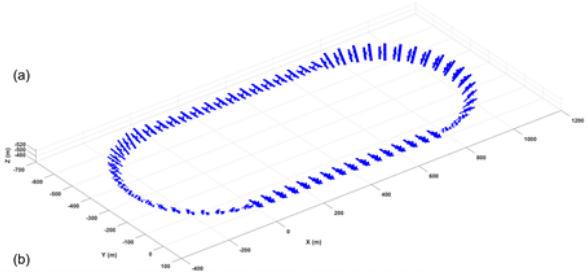

Fig. 3. 3D view of Axe flying the oval path.

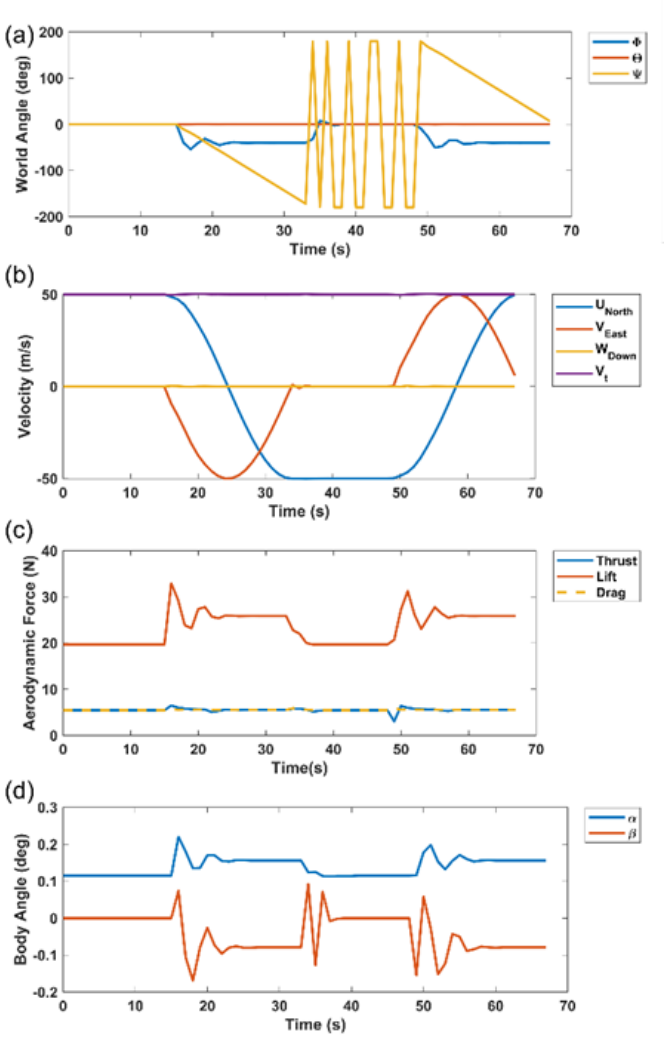

Fig. 4. Plots of state variables. (a) World frame angles. (b) World frame velocities. (c) Body frame aerodynamic forces. (d) Body frame angles

TABLE I. TRIM STATE FOR THE AXE DESIGN AT 50 M/S FORWARD FLIGHT SPEED, SHOWING INFORMATION SUCH AS CONTROL SETTINGS FOR THE SEVEN CONTROL CHANNELS (FOUR MOTORS DRIVING PROPELLERS AND THREE CONTROL SURFACES ON WING SEGMENTS) AS WELL AS THE ORIENTATION AND VARIOUS PHYSICAL PROPERTIES OF THE STATE. ALSO SHOWN IS THE MATRIX OF COMPUTED CONTROLS.

```
Objective steady state speed is UVW world =    50.00     0.00     0.00 m/s =   111.85     0.00     0.00 miles per hour
 Finished lmdif1 call: info =  1 (should be 1, 2 or 3: see MINPACK documentation)
 UVW world, body   (m/s)     50.00000000     0.00000000    -0.00000000    49.99990016     0.00000000     0.09992239
 UVWdot,PQRdot (m/s^2)      -0.00000073     0.00000000     0.00000000     0.00000000    -0.00000000     0.00000000
 Pitch angle theta (deg)     0.11450269
 Thrust world, body  (N)     5.45076208     0.00000000    -0.01069308     5.45077297     0.00000000     0.00000000
 Lift world, body    (N)    -0.00000000     0.00000000   -19.68064137     0.03933073     0.00000000   -19.68060207
 Drag world, body    (N)    -5.45076354     0.00000000     0.00355431    -5.45075976     0.00000000    -0.00733876
 Controls    1 -  6          0.99569503     0.99580438     0.99112835     0.99100555     0.51795967     0.51812533
 Controls    7 -  7          0.50000000
 Motor RPM   1 -  4      18889.49169413 18891.00125852 18826.18810242 18824.47888500
 Motor Amps  1 -  4          6.30455639     6.31048031     6.05954274     6.05301928
 Battery # 1 Current =   24.728 amps,  Time to 20% charge remaining =  698.8 s,  Flight distance =  34940.716 m
Total power from batteries  545.371 watts
 RICPACK AXE solver returns Ind(1), Ind(2), CPERM(1):           0           0   7348416.2216869323       (Success = 0, 0, <1.e19)
 Controls K where uc = uc_trim + K(x - x_trim)
   -0.36848     0.01005    -0.03090    -0.00160    -0.01101    -0.08257    -0.17237     1.12570   -12.59879    -0.50773    -0.40519    -0.03098
   -0.36876    -0.00993    -0.03312     0.00158    -0.01170     0.08225     0.17194     1.01326    12.57080    -0.50804     0.40415    -0.02551
   -0.35773    -0.00972     0.05053     0.00158     0.01909     0.08068     0.16922    -0.97585    12.32129    -0.49126     0.39605     0.07450
   -0.35743     0.00982     0.05266    -0.00153     0.01974    -0.08051    -0.16749    -0.86480   -12.29350    -0.49091    -0.39545     0.06909
    0.01730     0.10017    -0.24094    -0.01362    -0.20964    -0.09253    -0.61999   -14.95821   -11.25027     0.02920    -0.32479     0.69508
    0.01800    -0.10023    -0.24769     0.01339    -0.21253     0.08846     0.60267   -15.30344    10.78049     0.03031     0.31018     0.71058
    0.00020     0.69419    -0.00081     0.01309    -0.00108    -0.28070     0.72800    -0.05836    13.08833     0.00030     0.39693     0.00233
```

TABLE II. LIST OF PARTS USED IN THE AXE. (~) DENOTED ESTIMATED COST OF NON-COTS PARTS

Class	*Identifier*	*#*	*Cost ($)*
Battery	6000 mAh 6S 75C	1	130
Motor	2315XF-885	4	256
Propeller	7x5E	4	11
ESC	XF-UAS35	4	244
Wing	NACA 0006	6	~200
Servo	HS-53	6	72
Tube	Pultruded Carbon Fiber Tube	16	~34
Connector	3 Ports at 90 Degrees	5	~50
Flange	Standard Flange	4	~40
Plate	Carbon Fiber Plate	1	~24
Total		51	1061