

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/333457343>

State-Space Modelling of a Rocket for Optimal Control System Design

Article · May 2019

DOI: 10.3844/jastsp.2019.■■■■■

CITATIONS

5

READS

1,346

4 authors, including:



Bhar Kisabo Aliyu

National Space Research and Development Agency

44 PUBLICATIONS 84 CITATIONS

[SEE PROFILE](#)



Olusegun Samuel Sholiyi

11 PUBLICATIONS 34 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Internal Combustion Engine [View project](#)



Data Analytics [View project](#)

State-Space Modelling of a Rocket for Optimal Control System Design

Aliyu Bhar Kisabo, Aliyu Funmilayo Adebimpe,
Odooh Clement Okwo and Sholiyi Olusegun Samuel

Centre for Space Transport and Propulsion (CSTP) Epe Lagos-State Nigeria

Article history

Received: 08-02-2019

Revised: 19-02-2019

Accepted: 23-05-2019

Corresponding Author:

Aliyu Bhar Kisabo

Centre for Space Transport

and Propulsion (CSTP) Epe

Lagos-State Nigeria

Email: aliyukisabo@yahoo.com

Abstract: In this study, we present the derivation of the mathematical model for a rocket's autopilot in state-space. The basic equations defining the airframe dynamics of a typical six-Degree-of-Freedom (6DoF) are non-linear and coupled. Separation of these nonlinear coupled dynamics is presented as lateral and longitudinal dynamic equations. The need to determine aerodynamic coefficients and their derivative components are brought to light here, which is the crux of the equations. Methods of obtaining such coefficients and their derivatives in a sequential form are also put forward. After the aerodynamic coefficients and their derivatives are obtained, the next step is to *trim* and *linearize* the decoupled non-linear 6DoFs. In a novel way, we presented the linearization of the decoupled 6DoF equations in a generalized form. This provides a lucid and easy way to implement trim and linearization using any computer program.

Keywords: Rocket, Six-Degree-of-Freedom (6DoF), State-Space, Trimming, Linearization

Introduction

Over the years a number of authors (Papp, 2017; Faruqi, 2007; Chelaru and Barbu, 2009; Belega, 2012) have considered modelling, rocket/missile dynamics for atmospheric flights. In the majority of the published work on these mathematical models, trim and locally linearisation are done without detail explanation to the variables in the decoupled airframe dynamics. As such, the easy to write computer programs to facilitate this process numerically has been impeded.

With the advent of fast processors and numerical software like MATLAB®, Maple®, Python®, etc., it is now possible to take a complex non-linear 6DoF equation like that of a rocket and run a program that can trim and linearize it with easy. This has made the field of control system design to grow at an exponential rate (Siouris, 2004).

It is a known fact the mathematical models are developed with their use in mind. This means before delving in to the realisation of a model one should be well informed of the purpose for which that mathematical model will serve (Blakelock, 1965). Especially, in the field of control system design, a mathematical model in transfer function might not be the

ideal for *optimal control* design. Though, Problem Solving Environments (PSEs) like MATLAB/Simulink® come with built-in functions capable of transforming, say state-spacemodel to transfer functions. One should bear in mind that this is not without a 'cost'.

Mathematical Model

The inevitable part of any control problem is modeling the process or plant. The aim is to obtain a mathematical equation that sufficiently predicts the response of the plant to all inputs. For a rigid dynamic body, its motion can be described in translational, rotational and angular inclination at all times.

Translational Motion

An accelerometer is often used to measure force on dynamic body. For a rocket in motion, these forces (Handbook, 1995) are represented as given in (1). Actually, this is a measure of the specific *force*, i.e., the nongravitational force per unit mass in x , y , z -directions respectively. The specific force (also called the *g-force* or *mass-specific force*) has units of acceleration or ms^{-1} . So it is not actually a force at all, but a type of acceleration:

$$\begin{aligned}\dot{u} &= \frac{F_{A_{xb}} + F_{P_{xb}} + F_{g_{xb}}}{m} - (qw - rv), m/s^2 \\ \dot{v} &= \frac{F_{A_{yb}} + F_{P_{yb}} + F_{g_{yb}}}{m} - (ru - pw), m/s^2 \\ \dot{w} &= \frac{F_{A_{zb}} + F_{P_{zb}} + F_{g_{zb}}}{m} - (pv - qu), m/s^2\end{aligned}\quad (1)$$

where:

- $F_{A_{xb}}, F_{A_{yb}}, F_{A_{zb}}$ = Aerodynamic force vectors expressed in the body coordinate system, N
 $F_{g_{xb}}, F_{g_{yb}}, F_{g_{zb}}$ = Gravitational force vector expressed in body coordinate system, N
 $F_{P_{xb}}, F_{P_{yb}}, F_{P_{zb}}$ = Thrust vector expressed in the body coordinate system, N
 m = Instantaneous rocket mass, kg
 p, q, r = angular rates of roll, pitch and yaw, respectively, rad/s
 u, v, w = Linear velocity, m/s
 $\dot{u}, \dot{v}, \dot{w}$ = linear acceleration, m/s^2

The aerodynamic forces have the following components:

$$\begin{aligned}F_{A_{xb}} &= -0.5\rho V_M^2 C_A S \\ F_{A_{yb}} &= 0.5\rho V_M^2 C_{N_y} S \\ F_{A_{zb}} &= 0.5\rho V_M^2 C_{N_z} S\end{aligned}\quad (2)$$

where:

- C_A = Axial force coefficient, *dimensionless*
 C_N = Normal force coefficient, *dimensionless*
 C_{N_y} = Normal force coefficient on y b -axis
 $C_{N_y} = C_N \frac{-v}{\sqrt{v^2 + w^2}}$, *dimensionless*
 C_{N_z} = Normal force coefficient on z b -axis
 $C_{N_z} = C_N \frac{1}{\sqrt{v^2 + w^2}}$, *dimensionless*
 S = Reference area, m^2
 V_M = Magnitude of velocity vector of the center of mass of the rocket, m/s
 ρ = Atmospheric density, kg/m^3 .

The propulsive forces in (1) are computed as follows (Tewari, 2011):

$$\begin{aligned}F_{P_{xb}} &= F_p \cos \gamma_2 \cos \gamma_1, N \\ F_{P_{yb}} &= F_p \cos \gamma_2 \sin \gamma_1, N \\ F_{P_{zb}} &= -F_p \sin \gamma_2, N\end{aligned}\quad (3)$$

where:

- F_p = Thrust force vector, N
 $F_{P_{xb}}, F_{P_{yb}}, F_{P_{zb}}$ = Components of thrust vector F_p , N
 γ_1 = Angle from x b -axis projecting thrust vector F_p on x b , y b -plane, rad (deg)
 γ_2 = Angle projecting thrust vector F_p on x b , y b -plane to the thrust vector F_p , rad .
 l_p = Distance from aerodynamic centre to centre of mass, m
 l = Distance from centre of mass to nozzle, m

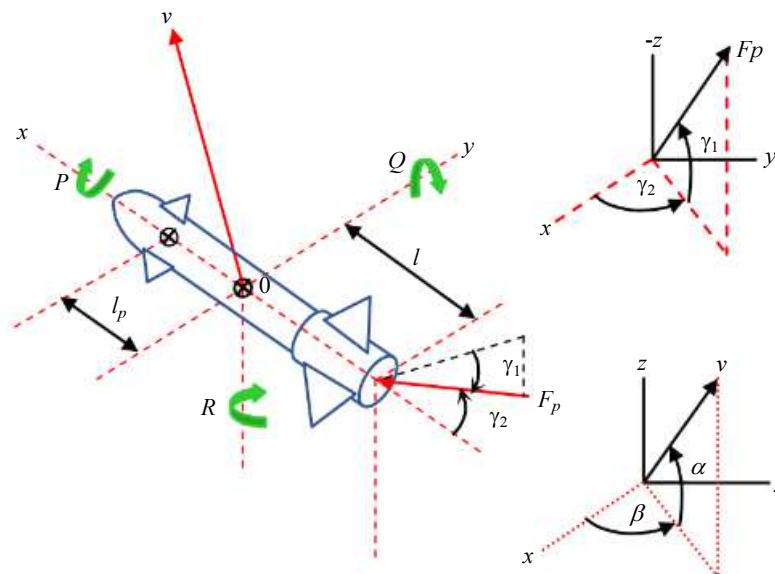


Fig. 1: Propulsion force from the nozzle of a rocket

11

L_p, M_p, N_p = Propulsion in duced moments in roll, pitch and yaw, respectively, Nm
 I_x, I_y, I_z = Inertia, $kg-m^2$
 p, q, r (P, Q, R) = Angular rates in roll, pitch and yaw, respectively, rad/s (deg/s)
 $\dot{p}, \dot{q}, \dot{r}$ = Angular acceleration in roll, pitch and yaw, respectively, rad/s^2 (deg/s^2):

$$\begin{aligned} L_A &= 0.5 \rho V_M^2 C_l S d, Nm \\ M_A &= 0.5 \rho V_M^2 C_m S d, Nm \\ N_A &= 0.5 \rho V_M^2 C_n S d, Nm \end{aligned} \quad (12)$$

where:

C_l = Roll moment coefficient
 C_m = Pitch moment coefficient
 C_n = Yaw moment coefficient
 d = Aerodynamic reference length of body, m

The aerodynamic moment coefficients are all about the center of mass and dimensionless and are expressed mathematically as:

$$\begin{aligned} C_l &= C_{l_\delta} \delta_r + \frac{d}{2V_M} (C_{l_p} p) \\ C_m &= C_{m_{ref}} - C_{N_z} \frac{x_{cm} - x_{ref}}{d} + \frac{d}{2V_M} (C_{m_q} + C_{m_\alpha}) q \text{ dimensionless} \quad (13) \\ C_n &= C_{n_{ref}} + C_{N_y} \frac{x_{cm} - x_{ref}}{d} + \frac{d}{2V_M} (C_{n_r} + C_{n_\beta}) r \end{aligned}$$

where:

C_{l_p} = Roll damping derivative relative to roll rate $p, rad^{-1} (deg^{-1})$
 C_{l_δ} = Slope of curve formed by roll moment coefficient C_l versus control-surface deflection, $rad^{-1} (deg^{-1})$
 $C_{m_{ref}}$ = pitching moment coefficient about reference moment station, *dimensionless*
 C_{m_q} = Pitch damping derivatives relative to pitch rate $q, rad^{-1} (deg^{-1})$
 C_{m_α} = Pitch damping derivative relative to angle of attack rate $\dot{\alpha}$ (slope of curve formed by C_a verses α), $rad^{-1} (deg^{-1})$
 C_{N_y} = Normal force on y -axis, *dimensionless*
 C_{N_z} = Normal force on z -axis, *dimensionless*
 C_{n_r} = Yaw damping derivative relative to yaw rate \dot{r} , $rad^{-1} (deg^{-1})$
 $C_{n_{ref}}$ = Yawing moment coefficient about reference moment station, *dimensionless*
 C_{n_β} = Yaw damping derivative relative to angle of sideslip rate $\dot{\beta}$, $rad^{-1} (deg^{-1})$
 x_{cm} = Distance from rocket nose to center of mass, m

x_{ref} = Distance from rocket nose to reference moment station, m
 δ_r = Effective control-surface deflection causing rolling moment, $rad (deg)$.

Considering that the rocket we intend to control is via fin deflection, fins on the rocket will designated as shown in Fig. 2.

Hence, the following moment coefficients are also given as:

$$\begin{aligned} C_{m_{ref}} &= C_{m_\alpha} \alpha + C_{m_\delta} \delta_\eta \text{ dimensionless} \\ C_{n_{ref}} &= C_{n_\beta} \beta + C_{n_\delta} \delta_\zeta, \end{aligned} \quad (14)$$

where:

$C_{m_{ref}}$ = Pitching moment coefficient about reference moment station (This is the static value normally measured in the wind tunnel.), *dimensionless*
 C_{m_α} = Slope of curve formed by pitch moment coefficient. C_m versus angle of attack, $\alpha \text{ rad}^{-1} (deg^{-1})$ slope of tune formed
 C_{m_δ} = Slope of tune formed by pitch moment coefficient C_m versus control-surface deflection for pitch, $\delta_p \text{ rad}^{-1} (deg^{-1})$
 C_{n_β} = Slope of curve formed by yaw moment coefficient C_n versus angle of sideslip β , $rad^{-1} (deg^{-1})$
 C_{n_δ} = Slope of curve formed by yaw moment coefficient C_n versus effective control-surface deflection for yaw $\delta_y \text{ rad}^{-1} (deg^{-1})$
 α = Angle of attack, $rad (deg)$
 β = Angle of sideslip, $rad (deg)$
 $\delta_\eta = \delta_1 = \delta_3$ = Effective control-surface deflection causing pitch moment, $rad (deg)$
 $\delta_\zeta = \delta_4 = \delta_2$ = Effective control-surface deflection causing yaw moment, $rad (deg)$.

The angle of attack, angle of sideslip and roll angle required for the realization of the aerodynamic coefficients are:

$$\begin{aligned} \alpha &= \tan^{-1} \left(\frac{w}{u} \right) \text{ or } \alpha_t = \tan^{-1} \left[\sqrt{\frac{v^2 + w^2}{u}} \right], \\ \beta &= \sin^{-1} \left(\frac{v}{V_M} \right) \text{ rad (deg)} \quad (15) \\ \phi &= \tan^{-1} \left(\frac{v}{w} \right), \text{ rad (deg)} \end{aligned}$$

where, ϕ is aerodynamic roll angle, $rad (deg)$ and α_t is the total angle of attack.

Table 1: Aerodynamic coefficients as a function of angle-of-attack

S/N	Coefficient	Description
1	C_N	Normal force (body axis)
2	C_L	Lift (wind axis)
3	C_M	Pitch moment (body axis)
4	X_{cp}	Center of pressure in calibers from moment reference center
5	C_A	Axial Force (body axis)
6	C_D	Drag (wind axis)
7	C_Y	Side Force (body axis)
8	C_n	Yaw Moment (body axis)
9	C_l	Roll Moment (body axis)
10	$C_{N\alpha}$	Normal force derivative with angle of attack
11	$C_{M\alpha}$	Pitch moment derivative with angle of attack
12	$C_{Y\beta}$	Side force derivative with side slip angle
13	$C_{n\beta}$	Yaw moment derivative with sideslip angle (body axis)
14	$C_{l\beta}$	Roll moment derivative with sideslip angle (body axis)
15	C_{Mq}	Pitch moment derivative with pitch rate
16	C_{Nq}	Normal force derivative with pitch rate
17	C_{Aq}	Axial force derivative with pitch rate
18	$C_{M\dot{\alpha}}$	Pitch moment derivative with rate of change of angle of attack
19	$C_{N\dot{\alpha}}$	Normal force derivative with rate of change of angle of attack
20	C_{lp}	Roll moment derivative with roll rate
21	C_{np}	Yaw moment derivative with roll rate
22	C_{yp}	Side force derivative with roll rate
23	C_{lr}	Roll moment derivative with yaw rate
24	C_{nr}	Yaw moment derivative with yaw rate
25	C_{Yr}	Side force derivative with yaw rate

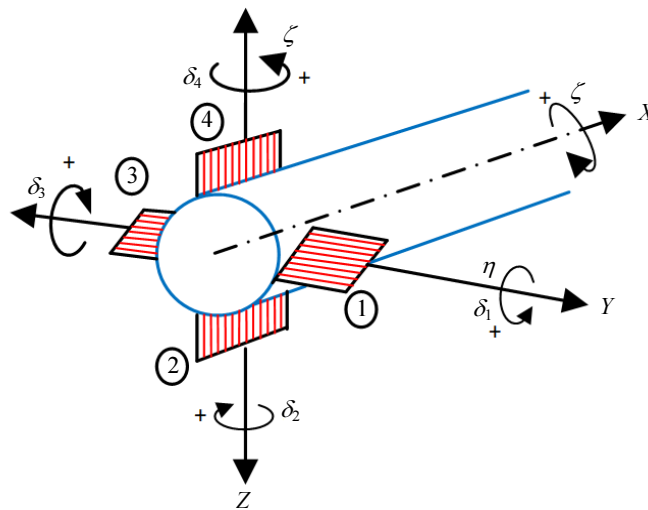


Fig. 2: Fin control and designation for control

Table 1 give a list of the aerodynamic coefficients that must be obtained for every rocket design before a model can be realized. There exist numerical and semi-numerical means of obtaining such coefficients. Examples of software that can do semi-empirical computation of such coefficients and their derivatives are MISSILE DIGITAL DATCOM® (Rosema *et al.*, 2008) and Flexible Structures Simulator (FSS) (PSUAI, 2013). Finally, a wind tunnel test is expected to validate and update all coefficients and their derivatives before the system engineer delves in the control design.

The third and final component to fully describe the motion of a rocket is its angular inclination or attitude. We chose the Euler angles to describe the attitude of the rocket.

Euler Angles

Missile attitude is required for a number of simulation functions including the calculation of angle of attack, seeker gimbal angles, fuze look-angles and warhead spray pattern. In simulations with six degrees of freedom, the missile attitude is calculated directly

by integrating the set of equations that define Euler angle rates, i.e.:

$$\begin{aligned}\dot{\phi} &= p + (q \sin \phi + r \cos \phi) \tan \theta, \text{rad} / \text{s} (\text{deg} / \text{s}) \\ \dot{\theta} &= q \cos \phi - r \sin \phi, \text{rad} / \text{s} (\text{deg} / \text{s}) \\ \dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta}, \text{rad} / \text{s} (\text{deg} / \text{s})\end{aligned}\quad (16)$$

where:

θ = Pitch angle, *rad (deg)*
 ϕ = Roll angle, *rad (deg)*
 $\dot{\psi}, \dot{\theta}, \dot{\phi}$ = Rates of change of Euler angles in heading, pitch and roll, respectively, *rad/s (deg/s)*
 p, q, r = Angular of roll, pitch and yaw, respectively, *rad/s*

The three heading angle of $\dot{\psi}, \dot{\theta}, \dot{\phi}$ as shown in Fig. 3 can be measure directly using a magnetometer. Note also, that the rulers angles in (16) can all be derived by integrating gyroscopic measurements. As such we might not need a instrument that will measure it directly. Nevertheless, a magnetor meter can be added to the instrumentation on board to measure heading.

Combining (1), (11) and (16) gives a complet6 degree of freedom equation of motion for a rocket in flight. This could be written together as given in (17). In today's modern aerospace industry, a single device like the MPU6050- a MEMs based integrated chip can be used

to give numerical values for state variables of (17) on any dynamic body. For control system design, the rocket system as described in (17) needs to be separated into the two planes (decoupled), these are called the lateral (*la*) and longitudinal (*lo*) dynamic equations of motion:

$$\begin{aligned}\dot{u} &= \frac{F_{A_{x_b}} + F_{P_{x_b}} + F_{g_{x_b}}}{m} - (qw - rv), m / s^2 \\ \dot{v} &= \frac{F_{A_{y_b}} + F_{P_{y_b}} + F_{g_{y_b}}}{m} - (ru - pw), m / s^2 \\ \dot{w} &= \frac{F_{A_{z_b}} + F_{P_{z_b}} + F_{g_{z_b}}}{m} - (pv - qu), m / s^2 \\ \dot{p} &= \frac{L_A + L_p - qr(I_z - I_y)}{I_x}, \text{rad} / s^2 (\text{deg} / s^2) \\ \dot{q} &= \frac{M_A + M_p - rp(I_x - I_z)}{I_y}, \text{rad} / s^2 (\text{deg} / s^2) \\ \dot{r} &= \frac{N_A + N_p - pq(I_y - I_x)}{I_z}, \text{rad} / s^2 (\text{deg} / s^2) \\ \dot{\phi} &= p + (q \sin \phi + r \cos \phi) \tan \theta, \text{rad} / \text{s} (\text{deg} / \text{s}) \\ \dot{\theta} &= q \cos \phi - r \sin \phi, \text{rad} / \text{s} (\text{deg} / \text{s}) \\ \dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta}, \text{rad} / \text{s} (\text{deg} / \text{s})\end{aligned}\quad (17)$$

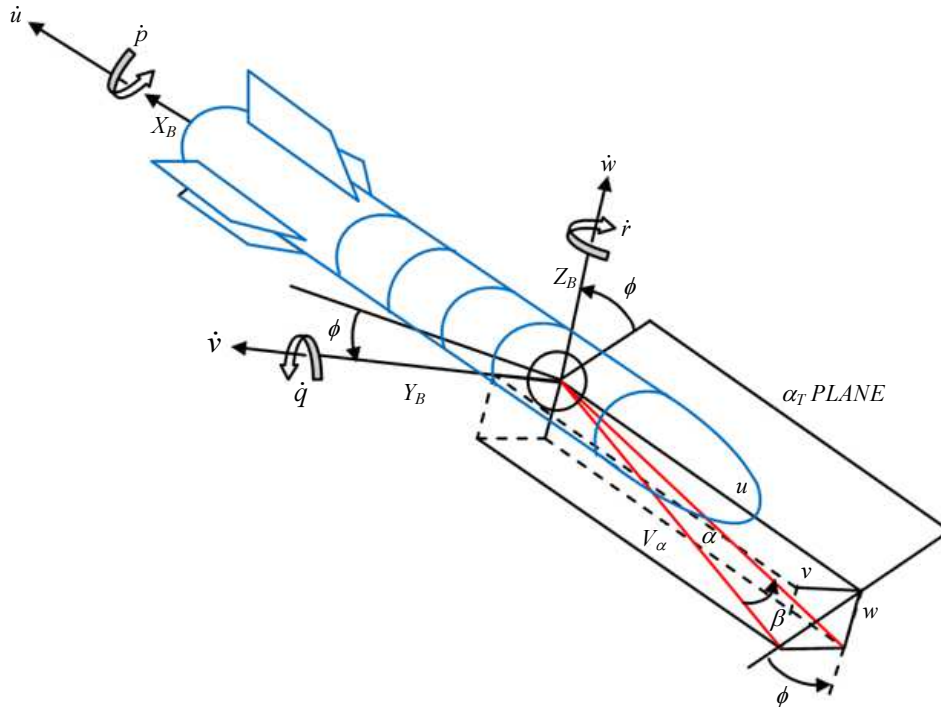


Fig. 3: 6-degree-of freedom motion of a rocket

Since the control system we intend to design is in a family of linear controllers and (17) is a non-linear system of differential equations, trimming and linearization must be done.

Trimming

To trim or find equilibrium values requires a good knowledge of advance computational techniques. A trim point, is also known as an equilibrium point. At equilibrium, all inputs to a system will attenuate and cause the system to return to a non-disturbance state or a nominal one. For a rocket, hence, a trim point involves setting controls(input) that causes the rocket to fly straight and level in the longitudinal plane. The suitable input values are the control surface deflections, the thrust and the rocket attitude (Yang *et al.*, 2005). The set of conditions are the rocket's accelerations. The variables associated with the trim problems can be divided into three categories:

- Objective variables
- Control variables
- Flight condition variables

The objective variables need to be driven towards the specified values, often zero (i.e., steady flight with zero sideslip). The objective parameters are combined in the objective vectors o_{la} and o_{lo} , for the lateral and longitudinal missile dynamics respectively as:

$$o_{la} = [\dot{v} \quad \dot{p} \quad \dot{r} \quad \beta]^T. \quad (18)$$

$$o_{lo} = [\dot{u} \quad \dot{w} \quad \dot{q} \quad \alpha]^T. \quad (19)$$

For a typical solution, the sideslip angle in (18), should be zero. In that case, the drag is at a minimum. The control parameters are tuned to drive the objective parameters to desired values. Together, they form the control vector c_{la} and c_{lo} , described in (22):

$$c_{la} = [\delta_a \quad \delta_r \quad \phi \quad \psi]^T, \quad (20)$$

$$c_{lo} = [\delta_e \quad \tau \quad \theta]^T, \quad (21)$$

Finally, the 12 states of the 6DOF equation of motion must be initialized; with the initial state conditions. In MATLAB®, the *trim* command is used to find equilibrium points. The object of trimming is to bring the forces and moments acting on the rocket into a state of equilibrium. That is the condition when the axial, normal and side forces and the roll, pitch and yaw moments are all zero. Mathematically, trimming combined implicit and explicit Jacobian approaches. For rigid rocket,

Jacobian trim approach is the preferred due to the fact that with rough estimates of the Jacobian and a first guess, convergence is likely to occur. Though, any robust optimization routine could be used to solve the trim problem. Mathematically, the Jacobian approach can be described as given in (22):

$$o_{i+1} - o_i = J_i [c_{i+1} - c_i]. \quad (22)$$

In (22), J_i is the Jacobian matrix evaluated near control input c_i . It's elements are first order partial derivatives and depict the effect of changes in each control input on acceleration.

Note, changes in lateral plane induces changes in the longitudinal plane and vice versa thus, we can write (23) a the Jacobian for the lateral dynamics and (24) for the longitudinal or pitch dynamics:

$$J_i = \left. \frac{\partial o}{\partial c} \right|_{c_i} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \delta_a} & \frac{\partial \dot{u}}{\partial \delta_r} & \frac{\partial \dot{u}}{\partial \phi} & \frac{\partial \dot{u}}{\partial \psi} \\ \frac{\partial \dot{v}}{\partial \delta_a} & \frac{\partial \dot{v}}{\partial \delta_r} & \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{w}}{\partial \delta_a} & \frac{\partial \dot{w}}{\partial \delta_r} & \frac{\partial \dot{w}}{\partial \phi} & \frac{\partial \dot{w}}{\partial \psi} \\ \frac{\partial \dot{p}}{\partial \delta_a} & \frac{\partial \dot{p}}{\partial \delta_r} & \frac{\partial \dot{p}}{\partial \phi} & \frac{\partial \dot{p}}{\partial \psi} \\ \frac{\partial \dot{q}}{\partial \delta_a} & \frac{\partial \dot{q}}{\partial \delta_r} & \frac{\partial \dot{q}}{\partial \phi} & \frac{\partial \dot{q}}{\partial \psi} \\ \frac{\partial \dot{r}}{\partial \delta_a} & \frac{\partial \dot{r}}{\partial \delta_r} & \frac{\partial \dot{r}}{\partial \phi} & \frac{\partial \dot{r}}{\partial \psi} \\ \frac{\partial \beta}{\partial \delta_a} & \frac{\partial \beta}{\partial \delta_r} & \frac{\partial \beta}{\partial \phi} & \frac{\partial \beta}{\partial \psi} \end{bmatrix}_{c_i}. \quad (23)$$

$$J_i = \left. \frac{\partial o}{\partial c} \right|_{c_i} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \delta_e} & \frac{\partial \dot{u}}{\partial \tau} & \frac{\partial \dot{u}}{\partial \theta} \\ \frac{\partial \dot{v}}{\partial \delta_e} & \frac{\partial \dot{v}}{\partial \tau} & \frac{\partial \dot{v}}{\partial \theta} \\ \frac{\partial \dot{w}}{\partial \delta_e} & \frac{\partial \dot{w}}{\partial \tau} & \frac{\partial \dot{w}}{\partial \theta} \\ \frac{\partial \dot{p}}{\partial \delta_e} & \frac{\partial \dot{p}}{\partial \tau} & \frac{\partial \dot{p}}{\partial \theta} \\ \frac{\partial \dot{q}}{\partial \delta_e} & \frac{\partial \dot{q}}{\partial \tau} & \frac{\partial \dot{q}}{\partial \theta} \\ \frac{\partial \dot{r}}{\partial \delta_e} & \frac{\partial \dot{r}}{\partial \tau} & \frac{\partial \dot{r}}{\partial \theta} \\ \frac{\partial \beta}{\partial \delta_e} & \frac{\partial \beta}{\partial \tau} & \frac{\partial \beta}{\partial \theta} \end{bmatrix}_{c_i}. \quad (24)$$

If the rocket is assumed to be at equilibrium, or trim condition then the equations of motion can be linearized

and the 6DOF equation of motion can be resolved into their lateral and longitudinal states.

Linearization

The system of first order non-linear differential equations of the rocket as presented by (17) is said to be in state-variable form if its mathematical model is described by a system of n first-order differential equations and an algebraic output equation as (White, 2003):

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, \dots, x_n, u) \\ \dot{x}_2 &= f_2(x_1, \dots, x_n, u) \\ &\dots \\ \dot{x}_n &= f_n(x_1, \dots, x_n, u) \\ y &= h(x_1, \dots, x_n, u)\end{aligned}\quad (25)$$

The column vector $x = [x_1, \dots, x_n]^T$ is called the *state* of the system. The scalars u and y are called the control input and the system output, respectively. Denoting:

$$f(x, u) = \begin{bmatrix} f_1(x_1, \dots, x_n, u) \\ f_2(x_1, \dots, x_n, u) \\ \vdots \\ f_n(x_1, \dots, x_n, u) \end{bmatrix}, \quad (26)$$

Concisely, (26) is written as:

$$\begin{aligned}\dot{x} &= f(x, u), \\ y &= h(x, u).\end{aligned}\quad (27)$$

where, f and h are nonlinear functions of x and u , then we say that the system is nonlinear. To linearize (26), we desire it to become:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du.\end{aligned}\quad (28)$$

where, A is $n \times n$, B is $n \times 1$, C is $1 \times n$ and D are all scalar. In MATLAB/Simulink®.

One reason for approximating the nonlinear system (26) by a linear model of the form (28) is that, by so doing, one can apply rather simple and systematic linear control design techniques. Given the nonlinear system (26) and an equilibrium or trimmed points $x^* [x_1^* \dots x_n^*]^T$ obtained when $u = u^*$, noting that $\Delta x = x - x^*$ we define a coordinate transformation as follows:

$$\Delta x = \begin{bmatrix} \Delta x_1 \\ \vdots \\ \Delta x_n \end{bmatrix} = \begin{bmatrix} x_1 - x_1^* \\ \vdots \\ x_n - x_n^* \end{bmatrix}.$$

Further, denoting $\Delta u = u - u^*$ and $\Delta y = y - h(x^*, u^*)$. The new coordinates Δx , Δu and Δy represent the variations of x , u and y from their equilibrium values. You have to think of these as a new state, new control input and new output respectively. The linearization of (26) at x^* , which the equilibrium or trim (White, 2003) state is given by:

$$\begin{aligned}\Delta \dot{x} &= A \Delta x + B \Delta u \\ \Delta y &= C \Delta x + D \Delta u,\end{aligned}\quad (29)$$

where:

$$\begin{aligned}A &= \left[\frac{\partial f}{\partial x} \right]_{x^*, u^*} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(x_1^*, \dots, x_n^*, u^*) & \dots & \frac{\partial f_1}{\partial x_n}(x_1^*, \dots, x_n^*, u^*) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1}(x_1^*, \dots, x_n^*, u^*) & \dots & \frac{\partial f_n}{\partial x_n}(x_1^*, \dots, x_n^*, u^*) \end{bmatrix}, \\ B &= \left[\frac{\partial f}{\partial u} \right]_{x^*, u^*} = \begin{bmatrix} \frac{\partial f_1}{\partial u}(x_1^*, \dots, x_n^*, u^*) \\ \vdots \\ \frac{\partial f_n}{\partial u}(x_1^*, \dots, x_n^*, u^*) \end{bmatrix}, \\ C &= \left[\frac{\partial h}{\partial x} \right]_{x^*, u^*} = \begin{bmatrix} \frac{\partial h}{\partial x_1}(x_1^*, \dots, x_n^*, u^*) & \dots & \frac{\partial h}{\partial x_n}(x_1^*, \dots, x_n^*, u^*) \end{bmatrix}, \\ D &= \left[\frac{\partial h}{\partial u} \right]_{x^*, u^*}.\end{aligned}$$

Note that the matrices A , B , C , D have constant coefficients in that all partial derivatives are evaluated at the numerical values $(x_1^*, \dots, x_n^*, u^*)$.

Lateral Dynamics of a Rocket

Equations of motion in the lateral plane are described by (30). Notice that (30) comprises of one of the force equations (F_y), one of the momentum equations (M_y) and two of the Euler angles from the 6DoF equations (decoupled from (17)):

$$\begin{aligned}\dot{v} &= \frac{F_{A_{y_b}} + F_{P_{y_b}} + F_{g_{y_b}}}{m} - (ru - pw), m/s^2 \\ \dot{p} &= \frac{L_A + L_P - qr(I_z - I_y)}{I_x}, rad/s^2 \left(\text{deg}/s^2 \right) \\ \dot{r} &= \frac{N_A + N_P - pq(I_y - I_x)}{I_z}, rad/s^2 \left(\text{deg}/s^2 \right) \\ \dot{\phi} &= p + (q \sin \phi + r \cos \phi) \tan \theta, rad/s \left(\text{deg}/s \right) \\ \dot{\psi} &= \frac{q \sin \phi + r \cos \phi}{\cos \theta}, rad/s \left(\text{deg}/s \right)\end{aligned}\quad (30)$$

For a completely computed aerodynamic coefficients and their derivatives, (30) can be expressed in state-space form (Cook, 2007) as:

$$\begin{bmatrix} \dot{v} \\ \dot{p} \\ \dot{r} \\ \dot{\phi} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} y_v & y_p & y_r & y_\phi & y_\psi \\ l_v & l_p & l_r & l_\phi & l_\psi \\ n_v & n_p & n_r & n_\phi & n_\psi \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ p \\ r \\ \phi \\ \psi \end{bmatrix} + \begin{bmatrix} y_\xi & y_\zeta \\ l_\xi & l_\zeta \\ n_\xi & n_\zeta \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \xi \\ \zeta \end{bmatrix}. \quad (31)$$

All the variables in the A matrix of (31) are the lateral dimensionless aerodynamic stability derivatives with respect to the system state vectors. The variables in the B matrix of (31) are the lateral dimensionless aerodynamic control derivatives with respect to the designated control surfaces.

Longitudinal Dynamics of a Rocket

The longitudinal dynamics is motion is also called the pitch plane. Equations describing the motion of a rocket in this plane can be describes as given in (32). Notice that (32) comprises of two of the force equations (F_x and F_z), two of the momentum equations (M_x and M_z) and two of the Euler angles from the 6DoF equations as given in (17):

$$\begin{aligned} \dot{u} &= \frac{F_{A_{xb}} + F_{P_{xb}} + F_{G_{xb}}}{m} - (qw - rv), m/s^2 \\ \dot{w} &= \frac{F_{A_{zb}} + F_{P_{zb}} + F_{G_{zb}}}{m} - (pv - qu), m/s^2 \\ \dot{q} &= \frac{M_A + M_P - rp(I_x - I_z)}{I_y}, rad/s^2 \left(\text{deg}/s^2 \right) \\ \dot{\theta} &= q \cos \phi - r \sin \phi, rad/s \left(\text{deg}/s \right) \end{aligned} \quad (32)$$

Just as with (31), (32) can also be re-expressed in state-space as:

$$\begin{bmatrix} \dot{u} \\ \dot{w} \\ \dot{q} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} x_u & x_w & x_q & x_\theta \\ z_u & z_w & z_q & z_\theta \\ m_u & m_w & m_q & m_\theta \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} u \\ w \\ q \\ \theta \end{bmatrix} + \begin{bmatrix} x_\eta & x_\tau \\ z_\eta & z_\tau \\ m_\eta & m_\tau \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \eta \\ \tau \end{bmatrix} \quad (33)$$

In MATLAB® the *linmod* (MathWorks, 2014) command is used to invoke linearization. The assumption made for decoupling the linear model is that the cross coupling effects between the two modes is negligible. These assumptions are:

- The rocket is designed with conventional control surfaces that do not give significant cross-coupling control between the lateral and longitudinal modes

- The rocket is symmetrical about the xz plane in which the inertia cross coupling in (xy and xz planes) result to cross-coupling between the lateral and longitudinal modes are minimum

It can be shown that a typical trimmed and linearized model of the pitch plane (longitudinal dynamics) for a rocket (BharKisabo, 2011) is given as presented in (34). Notice that compared to (33), the velocity in x -direction is not included in (34). This basically due to the fact that in this pitch plane, translational motion for a rocket is predominantly in the z -direction (velocity w):

$$\begin{aligned} \begin{bmatrix} \dot{\theta} \\ \dot{q} \\ \dot{w} \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 \\ 14.7805 & 0 & 0.01958 \\ -100.858 & 1 & -0.1256 \end{bmatrix} \begin{bmatrix} \theta \\ q \\ w \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ 3.4858 \\ 20.42 \end{bmatrix} \delta_\eta + \begin{bmatrix} 0 \\ 14.7805 \\ -94.8557 \end{bmatrix} \alpha_w \quad (34) \\ y &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ q \\ w \end{bmatrix} \end{aligned}$$

where:

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 & 0 \\ 14.7805 & 0 & 0.01958 \\ -100.858 & 1 & -0.1256 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 3.4858 \\ 20.42 \end{bmatrix}, \\ G &= \begin{bmatrix} 0 \\ 14.7805 \\ -94.8557 \end{bmatrix} C = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}, D = 0. \end{aligned}$$

Discussion of Result

From (34), it can be seen that a three state-variable model, has been realized in state-space. This implies that modern observer like the kalman filter can be designed to estimate and predict the trajectory of such rocket dynamics. This mathematical model also can be use to design all the control algorithms that fall in the class of modern (Optimal theory) control. Particularly, this model is important in the realization of the longitudinal autopilot system of a rocket.

Conclusion

Mathematical models are the bedrock of most scientific activities. Here we were able to define the non-linear airframe dynamics of a rocket in 6DoF. We went further to decouple the 6DoF equations of motion for the rocket and presented forms in which the decouple 6DoF equations could be easily trimmed and

linearized with a computer program like MATLAB®. The process presented here can be repeated for any size of rocket, aircrafts/Unmanned Aerial Vehicle (UAV). Note, if the aerospace vehicle being model is not a rocket and a state-space model is needed, all the procedures outlined in this study will remain the same. The only major changes that will be accommodated will come from the numerical values of the aerodynamic coefficients and their derivatives.

Acknowledgment

Authors wish to acknowledge the relentless effort of staff members at CSTP, especially those working at the system dynamics laboratory.

Author's Contributions

Aliyu Bhar Kisabo: Designed the study and drafted the first manuscript.

Udooh Clement okwo and Funmilayo Adebimpe Aliyu: Verified all result and computer simulations and assisted with literature search.

Sholiyi Olusegun Samuel: Supervised the entire study and edited the final manuscript.

Ethics

Authors declare that there are no ethical issues that may arise after the publication of this manuscript. This article is original and contains unpublished material.

References

- Belega, B.A., 2012. A mathematical model for computing the trajectory of rockets in a resistant medium taking into account the earth's rotation. The system differential equations of the rocket trajectory. Proceedings of the International Conference of Scientific Paper, (CSP' 12).
- BharKisabo, A., 2011. Expendable Launch Vehicle Flight Control: Design and Simulation with Matlab/Simulink. 1st Edn., LAP LAMBERT Academic Publishing, ISBN-10: 3844327290, pp: 96.
- Blakelock, J.H., 1965. Automatic Control of Aircraft and Missiles. 1st Edn., John Wiley and Sons, Inc., pp: 348.
- Chelaru, T.V. and C. Barbu, 2009. Mathematical model for sounding rockets, using attitude and rotation angles. Int. J. Applied Math. Inform.
- Cook, M.V., 2007. Flight Dynamics Principles: A Linear Approach to Aircraft Stability and Control. 2nd Edn., ISBN-13: 978-0-7506-6927-6.
- Faruqi, F.A., 2007. State space model for autopilot design of aerospace vehicles. Weapons Systems Division Defence Science and Technology Organisation, DSTO-TR.
- Handbook, M., 1995. Missile Flight Simulation Part One Surface-to-Air Missiles, MIL-HDBK-1211(MI).
- MathWorks, 2014. Control system Toolbox™. Revised for Version 9.7. The MathWorks, Inc.
- Papp, I., 2017. Missile mathematical model and system design. AARMS, 16: 29-35.
- PSUAI, 2013. Software package for Flexible Structures Simulation (FSS). International Institute for Advanced Aerospace Technologies of St. Petersburg State University of Aerospace Instrumentation, Performed according to the Contract No. I-011-2012.
- Rosema, C., J. Doyle, L. Auman and M. Underwood, 2008. MISSILE DATCOM User's Manual - 2011 U.S. Army Aviation and Missile Research, Development and Engineering Centre. William B. Blake-Control Design and Analysis Branch Control Sciences Division. AFRL-RB-WP-TR-2011-3071 (Supersedes 2008 Revision, AFRL-RB-WP-TR-2009-3015)
- Siouris, G.M., 2004. Missile Guidance and Control Systems. 1st Edn., Springer, New York, ISBN-10: 0387007261, pp: 666.
- Tewari, A., 2011. Advanced Control of Aircraft, Spacecraft and Rockets. 1 Edn., John Wiley and Sons, Ltd., ISBN-10: 9780470745632, pp: 454.
- White, R.E., 2003. Computational Mathematics: Models, Methods and Analysis with MATLAB and MPI. CRC Press, New York, ISBN-10: 0203494474, pp: 408.
- Yang, W.Y., W. Cao, T.S. Chung and J. Morris, 2005. Applied Numerical Methods Using MATLAB. 1 Edn., John Wiley and Sons, Inc., Hoboken, New Jersey, ISBN-10: 0471698334, pp: 528.