MEEN 655: Final Project

Utilizing Feedback Linearization for Rocket Yaw Control

Texas A&M University

5/8/2022

Roman Yoder

Abstract:

This project derives a feedback linearization controller to control the yaw of a two-dimensional rocket and achieve maximum vertical acceleration. The dynamic equations are first developed for a rocket with no control input, this was necessary to prove the natural stability of the rocket trajectory. With this stability confirmed it is possible to develop a feedback linearization controller that not only achieved zero yaw, but zero tangential velocity.

Introduction (Development):

Necessary before any controller design was to develop the primary equations for rocket dynamics. Significant simplification was utilized for a multitude of reasons. First, many aerodynamic effects are empirically derived from experiments, because of the nature of rocket science relating to the department of defense specific values or coefficients are confidential. Second, this investigation's primary goal was to drive the rocket's yaw to its desired value. This made it beneficial to assume negligible thermodynamic effects and fluid variations.

Rocket Free Body Diagram:

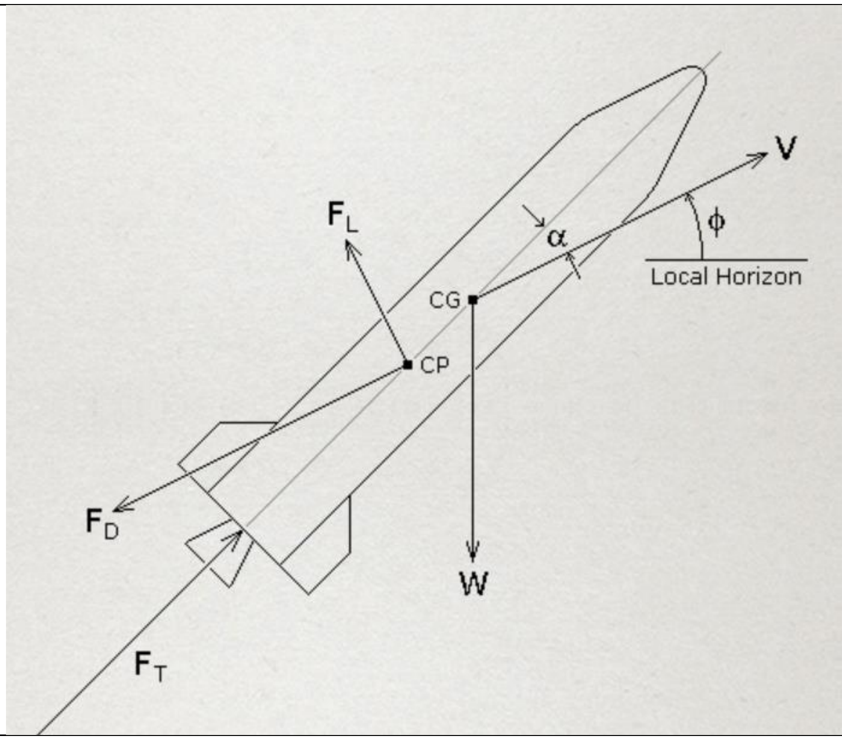


Figure 1: Rocket Diagram [1]

Figure 1 represents the starting point of the derived models used in this project [1].

F_D represents the drag force opposite to the velocity, F_L the lift force perpendicular to velocity, W the weight, and F_T the thrust provided by the rocket. The weight acts about the center of gravity of the rocket and the aerodynamics forces F_D and F_L act about the center of pressure. Φ represents the angle of the velocity vector from the local horizon and α the angle between the rocket's central axis and the velocity- Ψ will refer to the angle from the local axis to the rocket's central axis.

Both F_D and F_L can be resolved into forces parallel and perpendicular to the rocket's central axis, F_A and F_N. Both forces are directly dependent on the dynamic pressure and their respective coefficients shown below:

$$F_N = (1/2)(\rho)(V^2)(A)(C_N) \quad \text{(eq.1)}$$

$$F_A = (1/2)(\rho)(V^2)(A)(C_A) \quad \text{(eq.2)}$$

A represents the frontal area of the rocket, V the absolute velocity, and ρ the density of air. Both C_N and C_A can be resolved into functions dependent on α and the drag coefficient (C_D).

$$C_N = C_{N\alpha} * sin(\alpha) \quad \text{(eq.3)}$$

$$C_A = \frac{C_D * cos(\alpha) - (1/2) * C_N * sin(2*\alpha)}{1 - (sin(\alpha))^2} \quad \text{(eq.4)}$$

It should be noted that to avoid a singularity in MATLAB the value of 1 in the denominator of equation 4 is slightly perturbed. $C_{N\alpha}$ and C_D are both inherent to the design of a rocket and dependent on the Mach number and various fluid properties.

The dynamic equations are developed using two perspectives. The rotational equations are taken from the perspective of the rocket about its center of gravity, whereas the translational frame is taken globally. First the rotational equation later utilized in the control law derivation is developed.

$$J * \frac{d^2\Psi}{dt^2} = -(R_{cg-cp})F_N \quad \text{(eq.5)}$$

$R_{cg\text{-}cp}$ represents the moment arm from the center of gravity to the center of pressure. J represents the total moment of inertia of the rocket about the center of gravity.

$$m * \frac{d^2X}{dt^2} = -F_N * sin(\Psi) - cos(\Psi) * F_A + F_T * cos(\Psi) \quad \text{(eq.6)}$$

$$m * \frac{d^2Y}{dt^2} = F_N * cos(\Psi) - sin(\Psi) * F_A + F_T * sin(\Psi) - m * g \quad \text{(eq.7)}$$

F_T represents the force due to thrust, m the total mass of the rocket and g the acceleration due to gravity.

Both the moment of inertia and total mass are subject to change due to the change in mass propelling the rocket. Each of these dependencies are outlined below.

$$F_T = \dot{m}_f V_e \quad \text{(eq.8)}$$

Where m_f represents the mass of fuel and V_e the escape velocity from the rocket nozzle. Because the mass of the fuel tank varies both the moment of inertia and total mass are time dependent.

$$J_{total} = J_f + J_r \quad \text{(eq.9)}$$

$$m_{total} = m_f + m_r \quad \text{(eq.10)}$$

The angle of the velocity vector can be defined as follows:

$$\emptyset = \tan^{-1}(\frac{V_y}{V_x}) \quad \text{(eq.11)}$$

Implemented with 4-quadrant tangent in MATLAB.

α can then be defined additionally as:

$$\alpha = \Psi - \emptyset \quad \text{(eq.12)}$$

Overview of assumptions and parameter values:

- The density of air is constant
- The flow rate of m_f is constant
- $C_{N\alpha}$ and C_D are constant and do not vary with Mach number
- The center of gravity of the rocket and fuel tank coincide

Clearly, these assumptions present significant inaccuracies in the simulation. How the controller would adjust for these parameters in a physical application will be addressed later.

Necessary for the stability of the rocket is that the center of pressure lies behind the center of gravity, this results in a *restoring torque* that aligns the rockets yaw or Ψ with its current velocity. This is the primary purpose of fins on a rocket or missile- to move the center of pressure rearward. Further the moment of inertia for the rocket is reduced to that of the rod about its centroid and the fuel tank of that of a sphere. Resulting in:

$$J_R = (1/12) * m_r * {L_r}^2 \qquad \text{(eq.13)}$$

$$J_f = (2/3) * m_f * {R_f}^2 \qquad \text{(eq.14)}$$

The parameters listed in the table 1 below have been accumulated from various sources being average industry values and dimensions of retired air to surface missiles or estimated.

Table 1: Rocket Parameters

C_D	0.5
$C_{N\alpha}$	6.3
F_T	130 * 10^4 (Newtons)
M_r	810 kg
M_f (initial)	560 kg
Change of fuel mass	-28 (kg/s)
L_r (length of rocket)	4 m
L_{cg} (distance to the center of gravity)	2 m
L_{cp} (distance to center of pressure)	2.46 m
A (frontal area)	0.166 m^2
P (density)	1.225 kg/m^3

<u>Uncontrolled Simulation Results:</u>

The rocket is assumed to already be in flight with undesired initial conditions.

Table 2: Simulation Initial Conditions

X_o	0
V_{xo}	200 (m/s)
Yo	0
V_{yo}	200 (m/s)
Ψ_o	0.5 (rad)
d Ψ_o/dt	0.5 (rad/s)

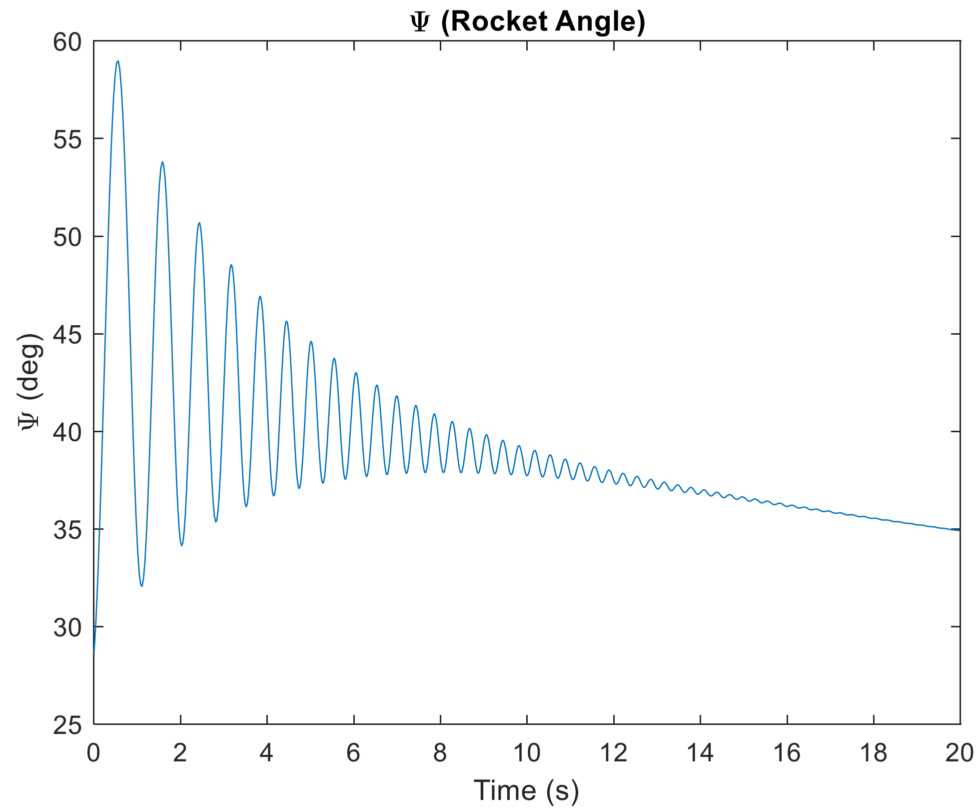

Ψ (Rocket Angle)
Ψ (deg)
Time (s)
60
55
50
45
40
35
30
25
0
2
4
6
8
10
12
14
16
18
20

Figure 2

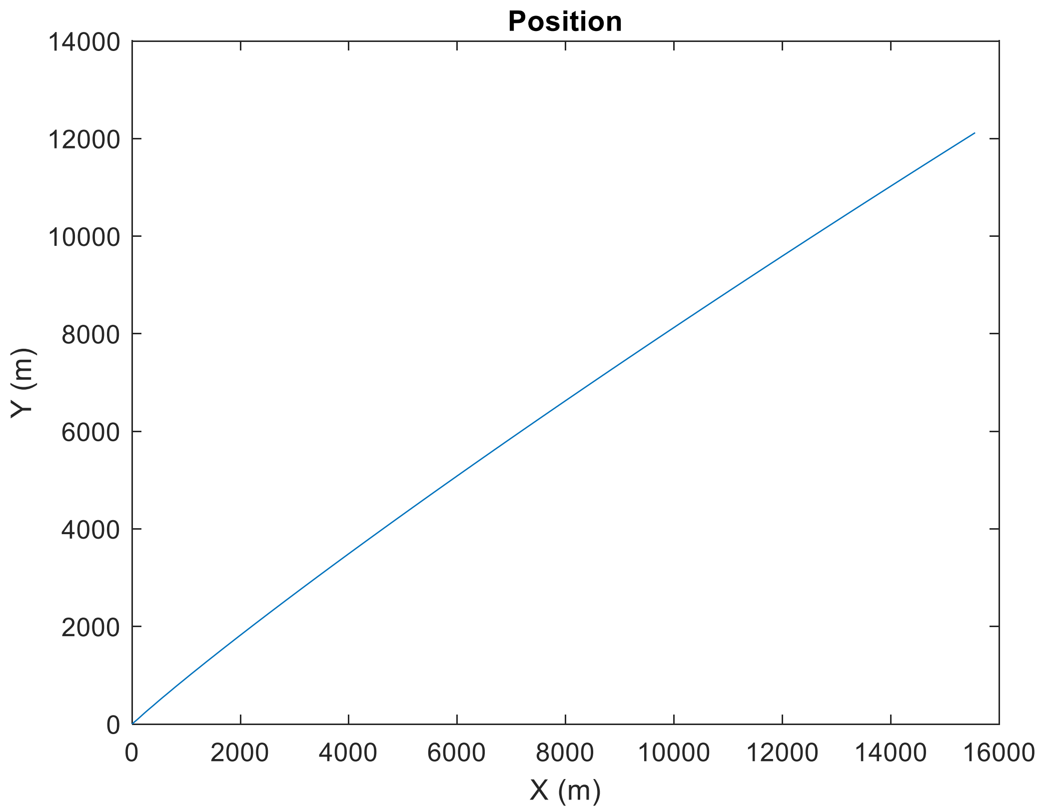

Position
14000
12000
10000
8000
6000
4000
2000
0
Y (m)
0
2000
4000
6000
8000
10000
12000
14000
16000
X (m)

Figure 3

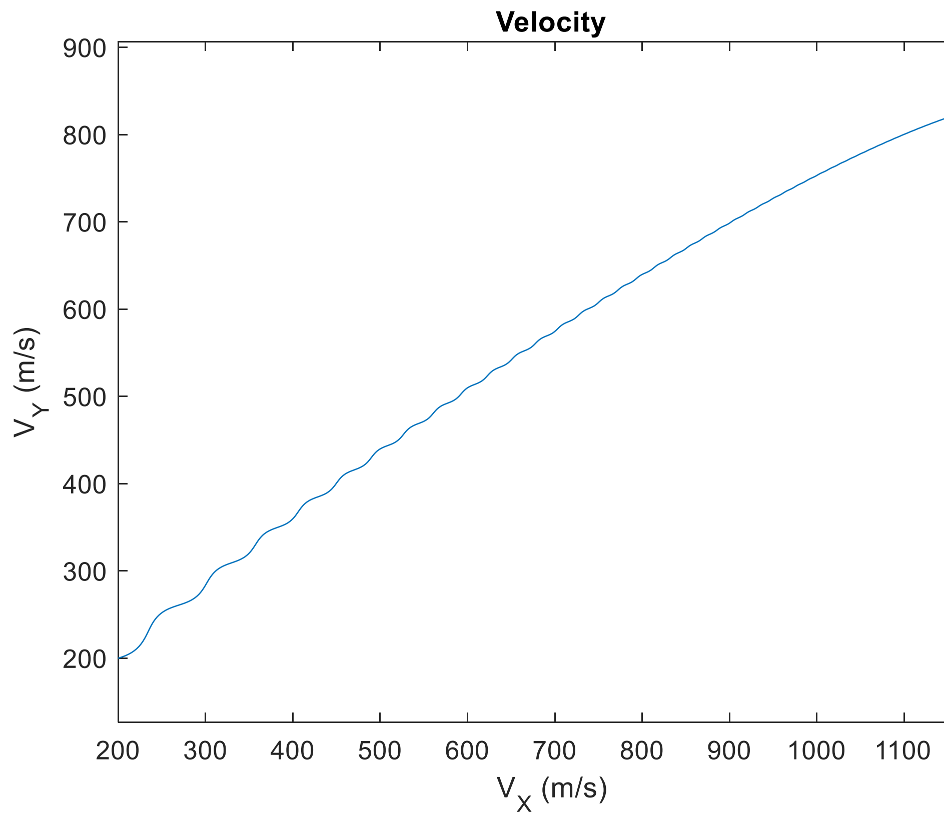


Figure 4

The rocket achieves stability, however, has an infinite amount of equilibrium trajectories that are highly dependent upon the initial conditions of the rocket and any additional disturbances.

Controller Derivation:

The controller is derived from the feedback linearization technique. However, before the control law is developed it is important to select the correct variable it utilizes as the output to achieve the desired goal- being to maximize vertical acceleration. If Ψ is maintained at 90° then a maximum amount of the throttle force F_T can influence the vertical acceleration over time, there are further additional benefits to selecting Ψ as the output that will be discussed. The best approach to manipulate Ψ through physical means comes down to design and implementation of the rocket itself. Some approaches use articulating air foils or thrust vectoring. As it was simple modification to the original dynamic equations thrust vectoring was employed, for which a crude drawing may be found in the appendix (A.1). The primary equations of motion are then modified as follows with the additional parameter of Θ as the control input.

$$J * \frac{d^2\Psi}{dt^2} = -(R_{cg-cp})F_N + (R_{R-cg})F_T * sin(\Theta) \quad \text{(eq.15)}$$

$$m * \frac{d^2X}{dt^2} = -F_N * sin(\Psi) - cos(\Psi) * F_A + F_T * cos(\Psi + \Theta) \quad \text{(eq.16)}$$

$$m * \frac{d^2Y}{dt^2} = F_N * cos(\Psi) - sin(\Psi) * F_A + F_T * sin(\Psi - \Theta) - m * g \quad \text{(eq.17)}$$

Where $R_{R\text{-}cg}$ represents the length from the center of gravity to the thrust force. To achieve Feedback-Linearization the equations are put into the following standard form:

$$\dot{\zeta} = A_c\xi + B_c(b(\zeta,\eta) + a(\zeta,\eta)u) \quad \text{(eq.18)}$$

$$\dot{\eta} = f(\eta, \zeta) \qquad \text{(eq.19)}$$

$$y = C_c \zeta \qquad \text{(eq.20)}$$

Given the output:

$$y = \psi - 90° \qquad \text{(eq.22)}$$

Then,

$$\zeta_1 = \psi - 90° \qquad \text{(eq.23)}$$

$$\dot{\zeta}_1 = \zeta_2 = \dot{\psi} \qquad \text{(eq.24)}$$

$$\dot{\zeta}_2 = \zeta_3 = \ddot{\psi} \qquad \text{(eq.25)}$$

$\ddot{\psi}$ is directly affected from the control input Θ, resulting in a relative degree of 2, and would be advantageous if:

$$\ddot{\psi} = -k_1\zeta_1 - k_2\zeta_2 = \omega \qquad \text{(eq.26)}$$

With equation 15 back solving for Θ results in:

$$\Theta = \sin^{-1}\left(\frac{J*\omega + (R_{cg-cp})F_N}{F_T(R_{R-cg})}\right) \qquad \text{(eq.27)}$$

k_1 and k_2 are selected as 100 and 200 to ensure the poles lie in the left half plane. Because the relative degree of the system does not match the degree of the original system, the stability of the system is determined by equation 19. This is also known as input-output linearization which is restricted by the *zero dynamics* of equation 19. The transformed system then takes the form of:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix} = \begin{pmatrix} 0 & 1 \\ -k_1 & -k_2 \end{pmatrix} \begin{bmatrix} \zeta_1 \\ \zeta_2 \end{bmatrix} \qquad \text{(eq.28)}$$

With:

$$\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \\ \eta_5 \end{bmatrix} = \begin{bmatrix} x \\ V_x \\ y \\ V_y \\ m_f \end{bmatrix} \qquad \text{(eq.29)}$$

Then:

$$\begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \\ \dot{\eta}_3 \\ \dot{\eta}_4 \\ \dot{\eta}_5 \end{bmatrix} = \begin{bmatrix} V_x \\ a_x \\ V_y \\ a_y \\ \dot{m}_f \end{bmatrix} \qquad \text{(eq.29)}$$

However, to achieve the desired goal the stability of V_y, a_y and constant the constant mass flow rate is of no concern, rather the concern is that of V_x and a_x.

Elementary Stability Analysis:

First, given the relative magnitude of the thrust force F_T and assuming reasonable bounds on the magnitude of the velocity. It can be concluded that the vectored force F_T will be able to overcome any moments generated on the rockets body. Then it is assumed that Ψ will be driven to 90°, the question being how this impacts the tangential acceleration given a non-zero V_x and how the controller responds given an instantaneous Ψ of 90° but a nonzero rotational velocity.

First, assuming a Ψ of 90°, a rotational velocity of zero and substituting for Θ equation 16 reduces to:

$$m * \frac{d^2X}{dt^2} = -F_N + F_T * cos(90° + \sin^{-1}(\frac{(R_{cg-cp})F_N}{F_T(R_{R-cg})})) \quad \text{(eq.30)}$$

Utilizing the following identity:

$$cos(90° + \sin^{-1}(a)) = -a \quad \text{(eq.31)}$$

Equation 30 becomes:

$$m * \frac{d^2X}{dt^2} = -F_N - (\frac{(R_{cg-cp})F_N}{(R_{R-cg})}) \quad \text{(eq.32)}$$

$$m * \frac{d^2X}{dt^2} = -(1 + \frac{(R_{cg-cp})}{(R_{R-cg})})F_N \quad \text{(eq.33)}$$

With:

$$\beta := (1 + \frac{(R_{cg-cp})}{(R_{R-cg})}) \quad \text{(eq.34)}$$

And β>0, substituting for F_N results in:

$$m * \frac{d^2X}{dt^2} = -(\beta)(1/2)(\rho)(V^2)(A)(C_{N\alpha} * sin(\alpha)) \quad \text{(eq.33)}$$

Which with equation 11 and 12 is proportional to:

$$m * \frac{d^2X}{dt^2} \propto -sin(90° - \tan^{-1}(\frac{V_y}{V_x})) \quad \text{(eq.34)}$$

Given that Ψ will be forced to 90° repeatedly, the magnitude of Vy will be much greater than Vx over time, with the velocity vector approaching 90°. Therefore, it is the sign of V_x that influences the sign of the tangential acceleration vector. This results in two cases:

For a negative V_x:

$$m * \frac{d^2X}{dt^2} \propto -sin(-\epsilon) \quad \text{(eq.35)}$$

Where $\epsilon > 0$

Therefore:

$$m * \frac{d^2X}{dt^2} \propto \epsilon \quad \text{(eq.36)}$$

Resulting in positive acceleration which reduces the negative V_x component. For a positive V_x the inverse holds, therefore given an instantaneous position of 90° and no rotational velocity the controller will work to eliminated tangential accelerations. Further if we assume that Ψ is instantaneously 90° and the rotational velocity is not, the control law reduces to:

$$\Theta = \sin^{-1}\left(\frac{J*(-k_1*\dot{\psi})+(R_{cg-cp})F_N}{F_T(R_{R-cg})}\right) \qquad \text{(eq.37)}$$

Which implies:

$$sign(\Theta) = sign(J*(-k_2*\dot{\psi}) + (R_{cg-cp})F_N) \qquad \text{(eq.38)}$$

Or with a non-zero Ψ:

$$sign(\Theta) = sign(J*(-k_1*(90°\psi) - k_2*\dot{\psi}) + (R_{cg-cp})F_N) \qquad \text{(eq.38)}$$

In which case the value of Θ works to not only control an angular position of 90° and zero angular velocity with the feedback term but also counteract any rotational acceleration due to the normal force at the center of pressure. All these results imply that driving the output to the desired values will not only be maintained and physically possible, but this act will in turn drive the tangential velocity to zero-naturally. This results in an elegant single input controller that drives multiple inputs of a non-linear system to desired equilibrium values. Further, these characteristics are only possible because the natural stability of the uncontrolled rocket, because of this restoring torque and the ability to accelerate the rocket in the desired direction the velocity is constantly realigned with the desired angle of 90°. This effect of restoring torque, as mentioned previously, comes from the fact that the center of pressure lies behind that of the center of gravity.

Controlled Simulation Results:

Table 3: Gain

K_1	5
K_2	4

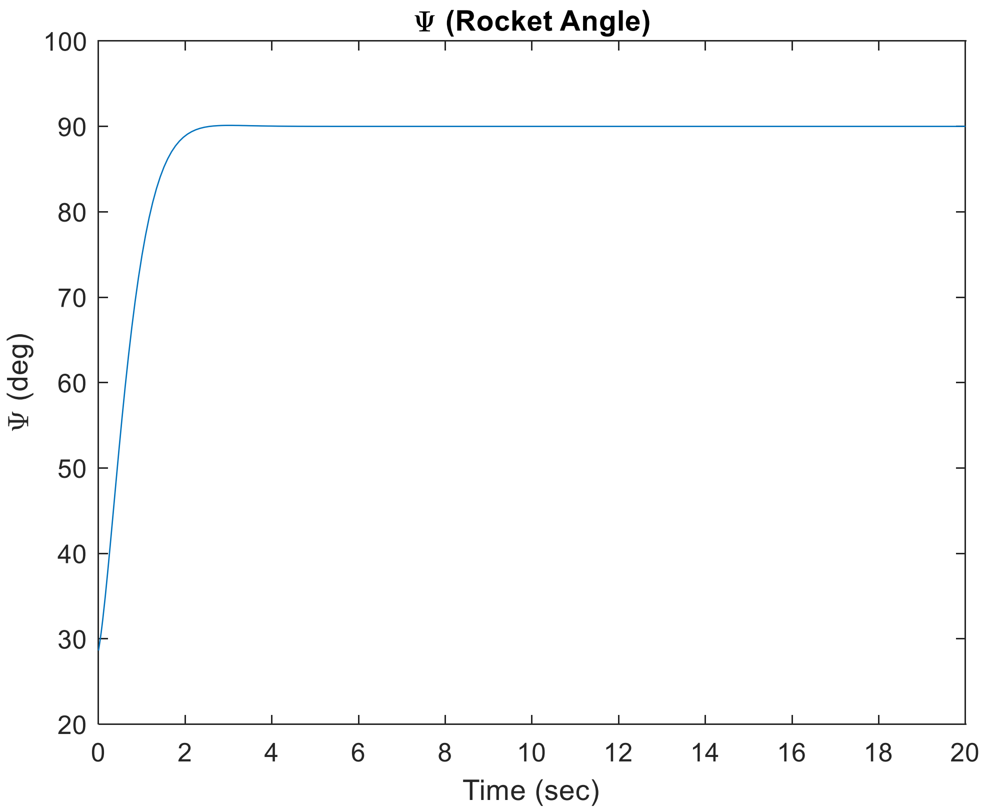


Figure 5

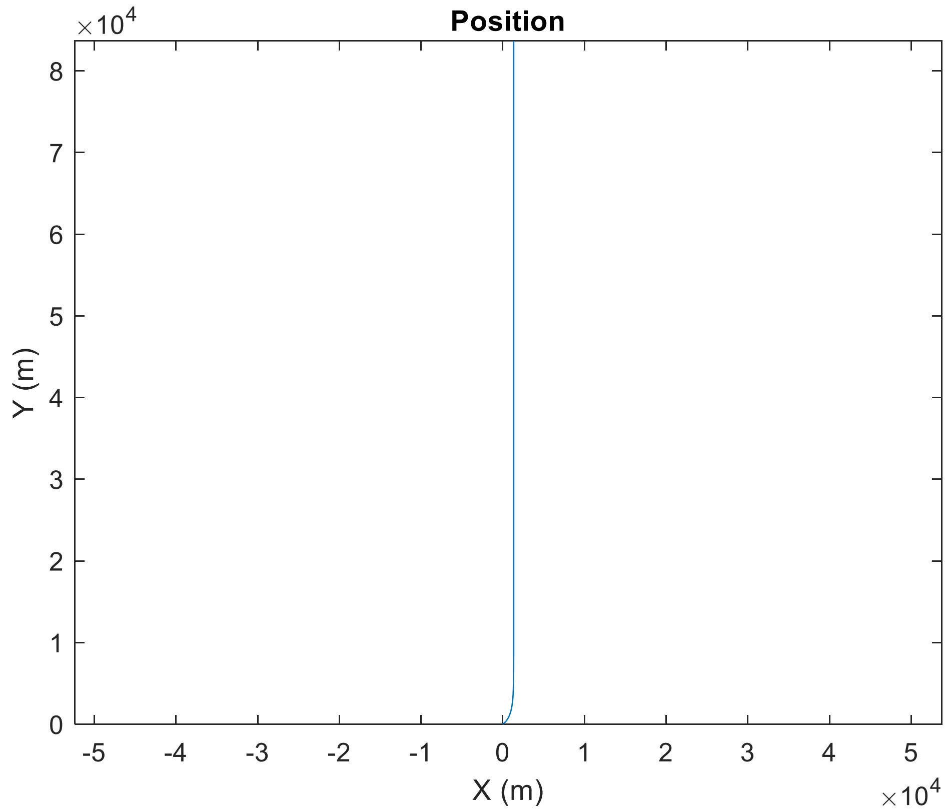


Figure 6

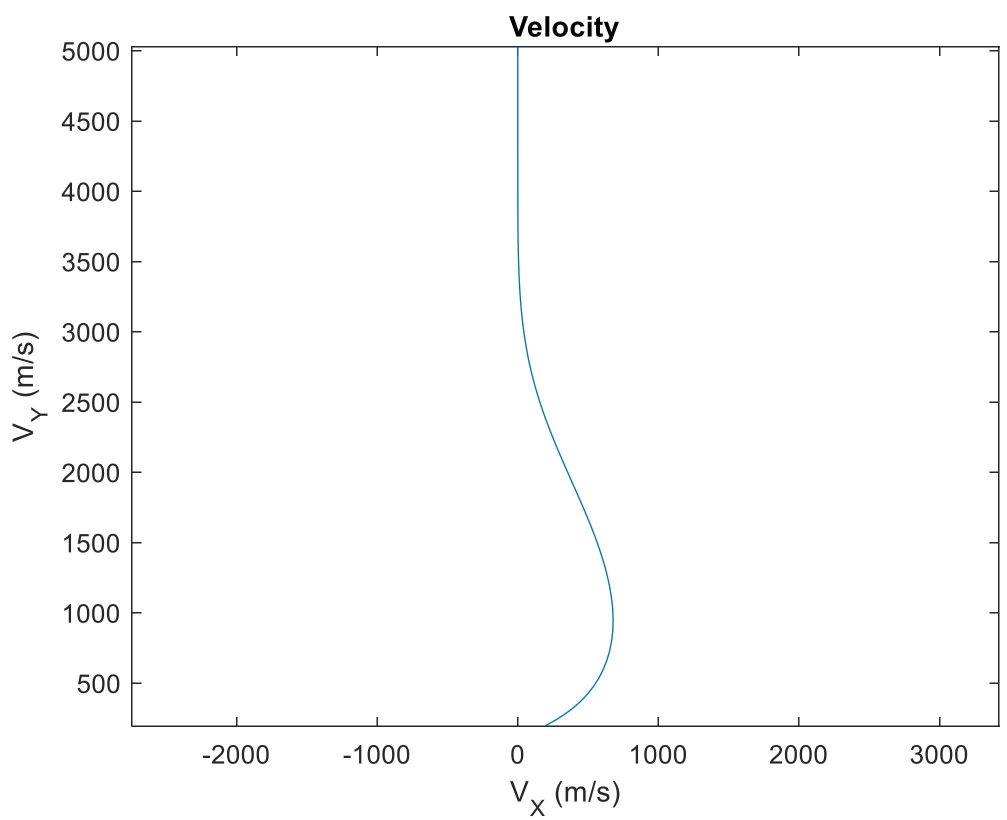


Figure 7

Discussion:

The presented results in figures 5, 6 and 7 demonstrate an impressive control method over the trajectory of the vertical accelerating rocket. Additional results showed that by varying gain constants performance was impacted. First setting the k_2 gain greater than the k_1 represented an improvement, this may be because the control variable only must pass through one relative integrator to affect this parameter whereas the position requires two representing more delay in the system. Further, these parameters where initially set to grandiose magnitudes, which once lowered demonstrated an improvement in performance. Initially with high gain the response represented that of an overdamped system lowering them pushed the response to critically damped up until overshoot occurred-corresponding to under damped. It would be desirable to be able to analytically solve for the desired gain values via a transfer function. This would be done by looking directly at the transfer function corresponding to the input-output relationship and using one of several synthesis techniques (root locus, pole placement, etc.). Also, one trial utilized 0 gains to evaluate if the feedforward term alone could stabilize the rocket. This resulted in a limit cycle in which the rocket perpetually flew in a circle. This is because the rocket kept trying to counteract the given moments, there by generating a new velocity and torque, which resulted in a positive feedback loop where the controller could not account for the error in rotation.

Notes on implementation and approximations:

Clearly, implementing this controller on a real rocket neglects a sleuth of both physical effects and parameters. But this controller could be improved upon to prove tangible viability. First, it would be necessary to model the aerodynamics of the rocket itself. Deriving empirical relations for the aerodynamic constants and their dependance on fluid properties and Mach numbers. Given estimations of these parameters it would be possible to use a form of Gain Scheduling to vary the controller parameters. Further, if the dynamic equations of the rocket alter fundamentally at some speed, then an entirely different control method could be utilized. Necessary in the implementation of this controller would be cutting edge accelerometers- the output from which could be utilized to create state estimates that feed directly into the control law. Additionally, parameters such as the force of thrust and mass burn rate may vary. These must be estimated from the given air density at the corresponding altitude and additional thermodynamic parameters such as combustion chamber temperature and pressure.

One folly of this controller is the means of articulation rely on the presence of a thrust. Given the fact that the rocket or rather missile may still be in flight after the fuel has been expended, it would be advantageous to have an additional means of control- for example rotating fins. It may even be advantageous to control the rate of combustion directly- typically rocket nozzles are optimized for one altitude and this results in inconsistences.

Further the assumptions of this simulation assumed a 2-D space. The equations of a real system would be of an extensively higher dimension (from additional directions and unknown parameters). This would also require more means of control such as thrust rotation and tilt.

Notes on position and altitude control:

Given the results, the position in the tangential direction is not driven to zero once the rocket has fully stabilized. Is it possible to eliminate this position difference given the current controller? Yes, this would be done by first detecting this position deviation and modulating the set point of alpha. This would require a predictive model of how to vary α to first approach the desired position and then stabilize at said position. This same strategy could be employed to keep the rocket flying tangentially at a constant altitude. This could be utilized in an air to surface missile like those launched from fighter jets. First, the controller would determine the necessary α such that the vertical forces have a net value of zero with a control input of zero. Then the controller would correspond by utilizing the control input to achieve this angle.

Conclusion:

Overall, the results of this report demonstrate the utility of the feedback linearization technique. By strategically pick the right output parameter and designing the system to be inherently stable control of complex systems, like that of rockets, is possible.

Future iterations on this project could involve building parameter estimation scheme or improving the accuracy of the simulation. Eventually this could include a redesign of the system for 3 dimensions. Further work could also be put into developing a model to predict the required stabilization angle for various velocity vectors. This could then be used to implement a tracking-based controller creating a guided rocket system.

Citations:

[1] http://www.braeunig.us/space/aerodyn_wip.htm#drag

[2] Pérez-Roca, S., Marzat, J., Piet-Lahanier, H., Langlois, N., Farago, F., Galeotta, M., and Le Gonidec, S., 2019, "A Survey of Automatic Control Methods for Liquid-Propellant Rocket Engines," Progress in Aerospace Sciences, **107**, pp. 63–84.

[3] Guerrero, V. A., Barranco, A., and Conde, D., "Active Control Stabilization of High Power Rocket," p. 87.

[4] "Active Control Stabilization of High Power Rocket.Pdf."

[5] Cronvich, L. L., 1983, "MISSILE AERODYNAMICS," N umber, **4**, p. 12.

[6] "Npc-Chapter4-Scan.Pdf."

[7] Kiehn, D., 2021, "Stability Analysis and Flight Control Design of the Winged Reusable Launch Vehicle ReFEx," CEAS Space J, **13**(1), pp. 51–64.

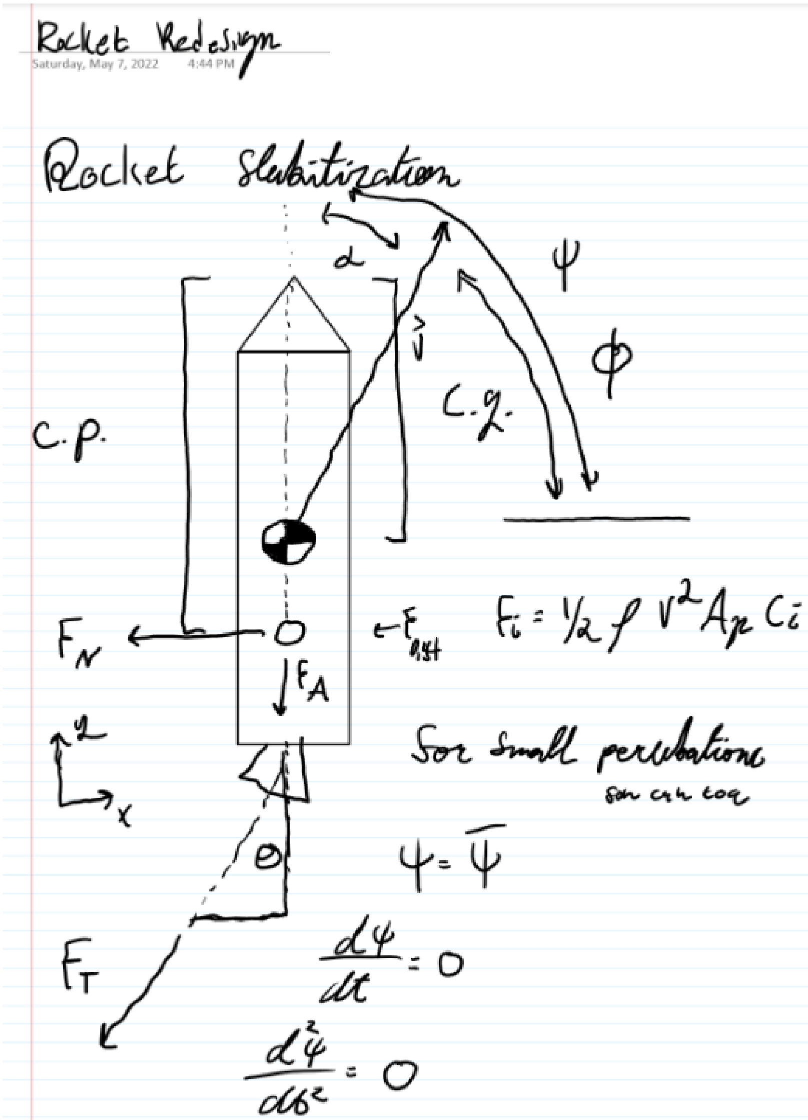
[8] "Kiehn2021_Article_StabilityAnalysisAndFlightCont.Pdf."

[9] Nielsen, J. N., "The Present Status and the Future of Missile Aerodynamics," p. 37.

[10] Rubio Hervas, J., and Reyhanoglu, M., 2014, "Thrust-Vector Control of a Three-Axis Stabilized Upper-Stage Rocket with Fuel Slosh Dynamics," Acta Astronautica, **98**, pp. 120–127.

Appendix:

A.1: Rocket Control Mechanism



$$F_i = \frac{1}{2} \rho V^2 A_p C_i$$

For small perturbations

sin cos tan

$$\psi = \overline{\psi}$$

$$\frac{d\psi}{dt} = 0$$

$$\frac{d^2\psi}{dt^2} = 0$$

1) we look at the response with no thrust vectoring

$$\vec{V} = \sqrt{V_x^2 + V_y^2} \qquad \phi = \tan^{-1}\left(\frac{V_y}{V_x}\right)$$

$$\vec{a} = \sqrt{a_x^2 + a_y^2}$$

I can write my moment of inertia equation

A.2: No Controller Code


```
function dxdt = nocon(t,x)
dxdt = zeros(7,1);
%Known States
xm=x(1);
xdot=x(2);
y=x(3);
ydot=x(4);
psi=x(5);
psidot=x(6);
mf=x(7);
%constants
Lr=4;
Lcgr=0.5*Lr;
Lcgf=Lcgr;
Lcp=Lcgr+0.46;
mr=810;
% rhoc=;
rhoa=1.225;
% gam=;
% R=;
% Tc=;
% At=;
A=0.46^2*(pi/4);
m_dot=-28;
% Isp=;
Ft=130*1000;
g=9.81;
Cd=0.5;
Cnb=6.3;
phi=atan2(ydot,xdot);
alpha=psi-phi;
vt=sqrt(xdot^2+ydot^2);
mt=mf+mr;
Jf=(2/3)*mf*(0.2)^2;
Jr=(1/12)*(mr)*Lr^2;
```

```
J=Jf+Jr;
%Additional Parameters
Cn=Cnb*sin(alpha);
Ca=(Cd*cos(alpha)-(0.5)*Cn*sin(2*alpha))/(1.00001-(sin(alpha)↙
^2));
%Forces
Fn=(0.5)*(rhoa)*(vt^2)*(A)*Cn;
Fa=(0.5)*(rhoa)*(vt^2)*(A)*Ca;
%Accelerations
xddot=(1/mt)*(-Fn*sin(psi)-cos(psi)*Fa+Ft*cos(psi));
yddot=(1/mt)*(Fn*cos(psi)-sin(psi)*Fa+Ft*sin(psi)-mt*g);
psiddot=(1/J)*(-(Lcp-Lcgr)*Fn);
%Code Output
  dxdt(1) =xdot;
  dxdt(2) =xddot;
  dxdt(3) =ydot;
  dxdt(4) =yddot;
  dxdt(5) =psidot;
  dxdt(6) =psiddot;
  dxdt(7) =m_dot;
end
```

A.3: Code Yaw Controller


```
function dxdt = yawcon(t,x)
dxdt = zeros(7,1);
%Known States
xm=x(1);
xdot=x(2);
y=x(3);
ydot=x(4);
psi=x(5);
psidot=x(6);
mf=x(7);
%constants
Lr=4;
Lcgr=0.5*Lr;
Lcgf=Lcgr;
Lcp=Lcgr+0.46;
mr=810;
% rhoc=;
rhoa=1.225;
% gam=;
% R=;
% Tc=;
% At=;
A=0.46^2*(pi/4);
m_dot=-28;
% Isp=;
Ft=130*10000;
g=9.81;
Cd=0.5;
Cnb=6.3;
phi=atan2(ydot,xdot);
alpha=psi-phi;
vt=sqrt(xdot^2+ydot^2);
mt=mf+mr;
Jf=(2/3)*mf*(0.2)^2;
Jr=(1/12)*(mr)*Lr^2;
```

```
J=Jf+Jr;
%Additional Parameters
Cn=Cnb*sin(alpha);
Ca=(Cd*cos(alpha)-(0.5)*Cn*sin(2*alpha))/(1.00001-(sin(alpha)↙
^2));
%Forces
Fn=(0.5)*(rhoa)*(vt^2)*(A)*Cn;
Fa=(0.5)*(rhoa)*(vt^2)*(A)*Ca;
%Controller
k2=4;
k1=5;
w=-k1*(psi-(pi/2))-k2*psidot;
theta=asin((J*w+(Lcp-Lcgr)*Fn)/(Ft*(Lr-Lcgr)));

%Accelerations
xddot=(1/mt)*(-Fn*sin(psi)-cos(psi)*Fa+Ft*cos(theta+psi));
yddot=(1/mt)*(Fn*cos(psi)-sin(psi)*Fa+Ft*sin(psi-theta)-mt*g);
psiddot=(1/J)*(-(Lcp-Lcgr)*Fn+Ft*(Lr-Lcgr)*sin(theta));
%Code Output
  dxdt(1) =xdot;
  dxdt(2) =xddot;
  dxdt(3) =ydot;
  dxdt(4) =yddot;
  dxdt(5) =psidot;
  dxdt(6) =psiddot;
  dxdt(7) =m_dot;
end
```