Modeling and Simulating Vectored Thrust Aircraft

Jiangwen Cheng   
*Department of Mechanical Engineering*   
*University of Washington*Seattle, WA, United States  
jw1210@uw.edu

*Abstract*— This paper presents a comprehensive control study on vectored thrust aircraft, specifically focusing on the analysis of system linearity, dynamics, and causal characteristics. The study also delves into the stability of a simplified vertical takeoff and landing (VTOL) aircraft model, employing a dual-pronged approach involving eigenvalues and Lyapunov stability analysis. To enhance the system’s stability and control, state feedback control is implemented. The application of state feedback control is executed with precision, considering various input scenarios such as step input, impulse input, and sinusoidal input. The system’s response to these inputs is thoroughly analyzed, shedding light on the efficacy of the control approach. Finally, the system coupled with a Linear Quadratic Regulator (LQR) controller to achieve optimal performance.

# Introduction

The exploration of motion controls in mechanical systems, particularly in aircraft dynamics, has been integral to various applications. The introduction of vectored thrust technology marks a paradigm shift, unlocking unprecedented maneuverability and vertical takeoff and landing (VTOL) capabilities [1]. This paper addresses the overarching challenge of understanding and simulating the dynamics of hovering vectored thrust aircraft. Our objectives include exploring the equilibrium position, linearizing dynamics around it, and analyzing the resulting system. Additionally, we delve into designing state feedback control, providing a comprehensive insight into the intricate dynamics of these aircraft, encompassing both theoretical foundations and practical control design aspects.

The significance of this problem is highlighted in the dynamic aviation landscape, driven by a continuous quest for improved maneuverability and operational adaptability. Technologies like vectored thrust challenge conventional aerodynamics, offering immense potential to reshape the future of flight. VTOL capabilities bring newfound flexibility, enabling aircraft to operate in confined spaces without relying on traditional runways.

# Problem Formulation

## Model

To better understand the movement of vectored thrust aircraft, such as the Harrier “jump jet”. Firstly, we need to understand the principles on which it operates. The Harrier fighter jet achieves vertical takeoff by altering its thrust direction downwards and employing small maneuvering thrusters positioned on its wings. Per the description, we can make a simplified mode to analyze the dynamic motion of the aircraft. The following Fig.1 shows the simplified model of the aircraft Harrier “jump jet” [1].

图示

描述已自动生成

**Fig.1. Simplified Model of Harrier “Jump Jet”**

## Physics

In a vectored thrust aircraft, the primary dynamic aspect involves the generation and control of thrust to achieve desired motions, particularly vertical takeoff, and landing. The fundamental principle lies in manipulating the direction of the thrust produced by the propulsion system. This is often achieved with thrust vectoring nozzles, which can be swiveled to redirect the exhaust gases. The main equations governing the motion of the aircraft are derived from Newton’s second and third laws of motion. Let’s consider a simplified model for vertical takeoff, where the aircraft generates thrust directed downward. In this scenario, the reactive forces can be decomposed into two components: one perpendicular to the aircraft named it and another parallel to the aircraft named it . The aircraft, subject to these forces, will also experience a rotational motion characterized by an angle, often denoted as . To analyze the system, we establish a Cartesian coordinate system, and Newton’s second law is applied to decompose the forces and express the moments. The equations utilized for system analysis are the following: equation (1) and equation (2).

(1)

(2)

All of parameters involved in the system and their corresponding values are summarized as Table I below. m is mass of aircraft, J is the inertia of aircraft, r is the distance between center of mass of aircraft and force . c is the damping coefficient and g is gravitational constant. Those specific values will be utilized to compute matrices in state space.

|  |  |  |
| --- | --- | --- |
| **Symbols** | **Description** | **Value** |
| m | Aircraft mass | 4. 0 kg |
| J | Aircraft Inertia | 0.00475 kg m^2 |
| r | Force moment distance | 0.25 m |
| c | Damping coefficient | 0.05 kg m/s |
| g | Gravitational constant | 9.8 m/s^2 |

**Table I: Summary of the Aircraft Model Parameters**

# Analysis

## Non-linear Differential Equations and Model Characteristics

By using Equation (1) and Equation (2), the motion of the aircraft can be expressed as following three coupled second order differential equations. The forces are decomposed into the x-axis and y-axis directions, and under the influence of damper, their sum equals the product of the mass and acceleration of the aircraft in the horizontal and vertical directions, respectively. Similarly, the generated torque equals the product of inertia and angular acceleration. The aircraft can maintain balance only if it satisfies Newton’s second law.

(3)

(4)

(5)

From above equations, state variables of the system can be defined as = ). Therefore, the origin (0, 0, 0, 0, 0, 0) can be selected as equilibrium point of interest. And inputs can be redefined with zero input, then , , which composed another equilibrium condition . Differential equations describing the motion of the aircraft above can be rewrite as below:

(6)

(7)

(8)

Based on the equations, we can distinguish that the inputs are forces and and the outputs are the displacement of aircraft in x and y direction and the rotation of the aircraft. Equations (6) (7) (8) are the representation of the model plant. Since the system is a multiple inputs and multiple outputs system, a decoupled system can be realized to make it easy to analyze.

## Jocobian Linearization

Since the system is still a nonlinear system, and for simplicity of analysis, we need to linearize system equations above. When dealing with nonlinear systems, linearization using the Jacobian matrix is a common approach. The Jacobian matrix provides a local linear approximation of a nonlinear system around a given operating point or equilibrium. By calculating the partial derivatives of the system’s equations with respect to its variables, the Jacobian matrix captures the system’s sensitivity to changes in those variables at a specific point. This allows for the creation of a linear model that approximates the system’s behavior in the vicinity of the chosen point. Equation used to compute the state space matrix of linearized system is shown as below.

(9)

## State Space and Output Equations of MIMO System

To make the dynamic system easy to analyze, we can express the original differential equations in state space form. The system state variables are defined as = ), and inputs are u = (). Then equations (6) (7) (8) can be written in an un-linearized state space form as below, which can be linearized further by applying Jacobian linearization.

To derive the linearized model of the system, we need to apply equation (9) - Jacobian linearization on the state space form around the equilibrium point (, ). Then, we can get below state space matrix A, B, C and D for equation (10) and (11) - linearized state-space and output equations.

(10)

(11)

A =

B =

C =

D =

The linearized equation of motion can be obtained by deriving it from above linear state-space model, and it is expressed as following equations:

(13)

(14)

Through the state space matrix, A, B, C and D, transfer function of each output can be computed by equation (15). Since we have already considered the angle at the equilibrium point to be 0, here we only consider the transfer function representation of the vectored thrust aircraft in the horizontal and vertical directions. Note that zero initial conditions are considered for convivence of analysis.

(15)

Following equation (16) and (17) show the transfer function in horizontal and vertical direction, and their corresponding pole and zeros are calculated as well.

Transfer function for x:

(16)

Zeros: 14.3710, -14.3710, -0.0125 and 0

Poles: -0.0125, -0.0125, 0, 0, 0 and 0

Transfer function for y:

(17)

Zeros: -0.0125, 0, 0 and 0

Poles: -0.0125, -0.0125, 0, 0, 0 and 0

## Characteristics of the Model

The output transfer functions show that the dynamic system described by equations (3), (4), and (5) exhibits proper transfer functions, where the highest power of s in the denominator (m) surpasses the highest power of s in the numerator (n). Specifically, the relative degree of the transfer function is greater than zero (m − n ≥ 0). From another aspect, system acceleration depends on current and past values, not future values. The above two facts indicate that the system is causal. Given the presence of second-order time derivatives and forces that depend on time (as indicated by terms involving θ), the system described by the equations is dynamic. Because it models the motion and behavior of the system over time, not only depends on input but input values at other times. The system is continuous because it involves continuous-time derivatives. The system is nonlinear due to the presence of trigonometric functions. The system equations do not contain explicit dependence on time (t), and there are no time-varying parameters. The coefficients and parameters in the equations remain constant over time. Therefore, it’s also a time- invariant system.

## Stability Analysis

Based on the A matric and numerical values in Table I, we can find the eigenvalues of the linearized dynamic system, which are (0, 0, 0, , , 0). Also, eigenvalues can be calculated through characteristic equation of transfer function. Despite the absence of eigenvalues with positive real parts, the presence of zero eigenvalues renders the system Bounded Input Bounded Output (BIBO) unstable. Therefore, it’s evident that the system is unstable. Referring to Fig.2 pole-zero plot as below.

图表

描述已自动生成

**Fig.2. Pole – Zero Map of the Original System**

To employ the Lyapunov stability theorem for the linearized system, we begin by choosing the identity matrix (I) for the matrix Q. Because this selection is motivated by the simplicity of identity matrix, which facilitates the analysis process. Applying Lyapunov analysis in Python can easily find the solution matrix P which is shown as below Fig. 3. The figure unmistakably illustrates the symmetry of matrix P.

表格

描述已自动生成

**Fig.3. Solution Matrix P after Applying Lyapunov**

In accordance with the Lyapunov theorem, the assurance of asymptotic stability at the origin is provided by the existence of a unique positive definite symmetric solution denoted as matrix P. As per the positive definite matrix theorem, it is established that all eigenvalues of matrix P are positive. Hence, the computation of eigenvalues is carried out as follows [-4.59551e+14, -7.55878e+79, -2.98266e+49, -9.80375e+17, -2.94159e+18, -2.49953e-02]. All the eigenvalues are negative real numbers. Therefore, it can be inferred that the linearized dynamics system is unstable from Lyapunov theorem.

## Similarity Transformation

To obtain the similarity transformation of matrix A, two matrices need to be introduced. T is a non-singular matrix that forcing on new state matrix. Z is the new state after T transform applied on. The relationship between the two matrices and their corresponding derivatives are shown as follows:

(18)

(19)

Applying the equation (19) in state equation (10) and output equation (11), transformed equations can be easily found, referring to equation (20) and (21)

(20)

(21)

It’s clear that new state space matrix can be expressed as following transformed form.

From above, a transformed state equation and output equation can be calculated using matrix T. Then similar realization of the system can be achieved as below:

To write the A matrix of original system into a diagonal or Jordan form, eigenvalues of A can compute as above, and a diagonal form matrix can easily be written as below. Then equation (22) is a good way to compute the transformation matrix T, which is also computed and shown as follows:

A =  (22)

To enhance the stability of matrix inversion in Python, particularly when encountering singular errors, consider implementing the pseudo-inverse approach. Utilize the pseudo-inverse method to calculate the inverse of matrix T, and subsequently apply it to compute the matrix for updated new states. Then, and matrix can be computed as follows:

From above new state space matrix , , and D, transfer functions for motion in x direction and y direction can be found easily. Basically, the transfer functions are same as previous transfer functions we got, referring to equation (16) and (17). This is because the similarity transformation transfers original state-space system to new state-space system via nothing more than a change of variables.

## Controllable and Observable Canonical Form

To obtain controllable canonical form and observable canonical form, we need take care of each coefficient of above two transfer functions. For the transfer function in x direction , controllable canonical form of , , can be found as follows:

At the time, observable canonical form of are found as below:

For the transfer function in y direction , controllable canonical form of , , can be found as follows:

Observable canonical form of are found as below:

From above controllable and observable canonical form summary, it’s evident that the controllable canonical form of the two transfer functions and is no difference. That is because both share the same denominator.

## Discrete the State Space Model

When implementing controls digitally (e.g., on a microcontroller), a continuous time system must be represented in the discrete time domain [4]. Fig. 4. shown as following is used to illustrate converting time function to discrete sequence.

图片包含 游戏机, 物体, 桌子

描述已自动生成

**Fig.4. Converting Continuous Signal to Discrete Signal**

Before computing the transfer function of a discrete system, it is essential to apply a zero-order hold on the discrete-time signal. This process transforms the signal into a “staircase” function, effectively restoring it to a continuous state. To derive the solution for the original continuous state space, starting from time and progressing to, it is crucial to select an appropriate sampling time. In this context, let’s consider a sampling time T of 0.1 seconds. This ensures a reliable and accurate representation of the system dynamics, allowing for meaningful analysis and computations. Applying the sampling time in the discrete time interval, solution of original system at discrete time can be calculated by following equations:

(23)

If A matrix is a non-singular matrix, there is an explicit form of . Applying Taylor expansion on and taking the expansion result back to equation of above, an equivalent expression of can be derived as below:

(24)

However, the A matrix is a singular matrix. Explicit form cannot be utilized to find the matrix. Using zero order hold command in Python, it’ quick and convenient to calculate the state space matrices in discrete time. Results are shown as following:

Therefore, the state space matrices in discrete time can be expressed as below:

## State Feedback Control

To design a state-feedback control for the dynamic system of vectored thrust aircraft, pole-placement technique must be applied on it. The idea is to place the closed-loop eigenvalues at desired locations to achieve the desired system response. Prior to delving into the specifics, it is essential to highlight the importance of the general feedback structure. This step is crucial in fostering a comprehensive grasp of the functioning of a state feedback controller. Referring to Fig.5 below. Also, the closed-loop system dynamics are derived and given by equation (25) and (26):

墙上的钟表

中度可信度描述已自动生成

**Fig. 5. State Feedback Structure Illustration**

(25)

(26)

In above figure, K is the state feedback gain matrix; is the state variables; u is the input vector of the system, v is new input; T is the output vector of the system; A, B, C, D are matrices defined the dynamic system. And in this system, zero initial condition is considered.

A fact is that if Σ = (A, B, C, D) is in controllable canonical form, we can completely change all the eigenvalues of (A−BK) by choice of state-feedback gain matrix K [4]. Prior to select proper eigenvalues for the system, desired performance specifications need to be considered at first. For vectored thrust aircraft designed to achieve seamless vertical take-off and landing, rapid stabilization is imperative for the vehicle to promptly settle down. Therefore, selecting eigenvalues with a negative real part (located in the left-half complex plane) can contribute to a fast-settling time. And damping ratio affects the level of oscillations in the response. A moderate damping ratio is usually preferred to balance stability with a fast response. Complex conjugate pairs with a negative real part and moderate imaginary parts can achieve this balance. Avoiding eigenvalues too far from the origin can help prevent excessive control effort, especially for systems with physical constraints. Finally, stability is paramount in aircraft control. Placing eigenvalues in the left-half complex plane ensures a stable closed-loop system. Based on previous discussion, two pairs of complex conjugates and , and two negative real numbers -1 and -5 are selected for the system poles.

Applying these eigenvalues in MATLAB, the state feedback gain matrix K can be computed.

## Observer State Feedback Control

When state feedback control is not implementable, then problem may be solved by observer state feedback control design. Based on equation (10), the state vector can be estimated by following:

(27)

(28)

is named observer state vector.

More specifically, the observer has a structure that uses the actual output measurement as following:

(29)

L is named observer gain.

Per theorem, if the continuous, time-invariant system in equation (10) and (11) is observable, then all eigenvalues of (A - LC) can be arbitrary assigned. Same as state feedback control, eigenvalues are selected with no difference. Two pairs of complex conjugates and , and two negative real numbers -1 and -5. Applying this theorem and selected eigenvalues, L matrix can be calculated in MATLAB as below:

L =

Eigenvalues of (A - LC) are also calculated for system stability confirmation. Poles are , , -3 and -9. All real part of the poles is negative. Since observer gain already found, an augmented matrix could be composed if integrating system dynamic equation (10) and (11) with observer dynamic equation (29). Augmented system is shown as follows:

(30)

(31)

For the convenience of simulation of the augmented system, zero initial condition will be considered.

## Simulation of State Feedback Control

Initially, simulations are performed to implement basic state feedback control. The ensuing plots illustrate the responses of the two state variables, x and y, a unit step input, to a unit impulse input, and a sinusoidal input with 10 times the magnitude at frequencies of 2πHz and 20πHz.图表, 折线图

描述已自动生成

**Fig.6. Step Response of State Feedback Control**

图表, 折线图

描述已自动生成

**Fig.7. Impulse Response of State Feedback Control**

**图表, 直方图

描述已自动生成**

**Fig.8. Sinusoidal Response (10sin ()) of State Feedback Control**

**图表, 折线图

描述已自动生成**

**Fig.9. Sinusoidal Response (10sin ()) of State Feedback Control**

In state feedback control, the step response indicates stability as time approaches infinity. Specifically in the context of an aircraft, it initially exhibits horizontal motion before reaching a settled state. The aircraft achieves horizontal movement and stabilization within a second. Additionally, with an impulse input, the system attains stability. Introducing a sinusoidal input also ensures stability, the system exhibits periodic oscillations that persist indefinitely. Therefore, the system can define as a stable system under the state feedback control with eigenvalues were selected.

## Simulation of Observer State Feedback Control

Subsequently, simulations were carried out for observer state feedback control. It’s noteworthy that the responses in both x and y are unified, given that only two pertinent transfer functions exist in the augmented system, while the other two transfer functions yield zeros. Same as previous input signal, response in x and y were plotted under unit step input, unit impulse input, and sinusoidal input with 10 times the magnitude at frequencies of 2πHz, 5πHz and 9πHz.

图表, 折线图

描述已自动生成

**Fig.10. Step Response of Observer State Feedback Control**

图表, 折线图

描述已自动生成

**Fig.11. Impulse Response of Observer State Feedback Control**

图形用户界面

描述已自动生成

**Fig.12. Sinusoidal Response of Observer State Feedback Control**

In observer state feedback control, it’s evident that system stability is unattainable for various inputs. In the x-direction, the response consistently trends in a negative direction for all input scenarios. The behavior in the y-direction varies slightly between step and impulse inputs. With an impulse input, the aircraft takes several minutes to reach a stable state—a notably lengthy settling time. However, for step and sinusoidal inputs, the aircraft perpetually fails to settle and instead moves towards an infinitely distant position. These observations collectively indicate that the system, under observer state feedback control, is inherently incapable of achieving stability.

## Linear Quadratic Regulators (LQR)

Due to arbitrary selected eigenvalues may not 100 percent promise a stable control on the VOTL dynamic system. Linear quadratic regulator is considered to apply on the system to make it stable. Here details about the formulas related to LQR are not going to be described, but cost function of equation (32) is shown as below to help illustrate the controller better. We can obtain the K, S, E matrices through MATLAB or Python command “lqr” directly. Q and R matrix are positive definite weighting matrices that define the cost. E matrix is composed of eigenvalues of the system. Beginning with diagonal matrix for Q and R as below.

(32)

Where , u = represent the local coordinates around the desired equilibrium point (, ).

u = , then u = u + = ;

where ,

Then, transformed from equation (11) to (12)

(33)

(34)

After applying the “lqr” command to the original state space matrix, we can easily get the K, S and E values. Eigenvalues of the closed loop system are computed as following:

[-5.3853, -2.05632.3125j, -1.0002, 0.37510.3307].

A more intuitive way to judge the stability of the system is through the pole-zero plot that is shown as Fig.13. From the pole-zero map, it shows that all poles in the left side of the map, that means the system is in a stable state. It also an evident that our LQR controller efficiently successfully make the system stable.

图表, 散点图

描述已自动生成

**Fig.13. Pole-Zero Map of Controlled Closed Loop System**

The subsequent illustrations demonstrate that employing a linear quadratic regulator enables the vectored thrust aircraft’s dynamic system to attain a stable state efficiently. The ensuing analysis depicts the system’s response to step input, impulse input, and sinusoidal input at a frequency of 2πHz.图表, 折线图

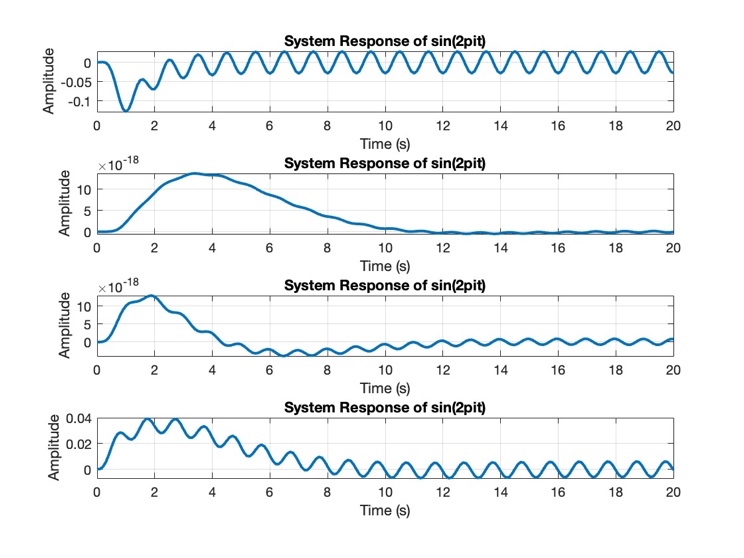
描述已自动生成

**Fig.14. Step Response of System with LQR**

图表, 折线图

描述已自动生成

**Fig.15. Impulse Response of System with LQR**



**Fig.16. Sinusoidal Response of System with LQR**

Based on above figures, the system response converges to a stable state no matter in what kind of input signal. The aircraft will move to the negative direction of x a little bit in a second and move to 1 meter height in y direction in a second under step input. Both illustrate the stability of the system. For impulse input, the system finally reaches to origin point because of the equilibrium state setting, which also shows the stability. The last sinusoid input also achieves a stable state because it’s in a stable oscillation. In summary, LQR can optimize system efficiently.

# Conclusion

In conclusion, the integration of state feedback control proves to be a pivotal strategy in enhancing the stability of dynamic systems. By actively influencing the system dynamics based on the internal state information, this approach enables effective stabilization and robust performance. Furthermore, the application of Linear Quadratic Regulator (LQR) in the control scheme emerges as a powerful tool for optimizing system stability. The LQR method, with its ability to balance control effort and system performance through optimal state feedback gains, contributes significantly to achieving a well-tailored and efficient control strategy. Together, state feedback control and LQR present a formidable combination, offering a comprehensive solution to not only stabilize but also optimize the dynamic behavior of complex systems.

##### References

1. Karl Johan Astrom, Richard M. Murray, Feedback Systems, An Introduction for Scientists and Engineers, 2nd ed, July 2020, pp.73-116, pp 191-224, pp229-261 *(references)*
2. Laura Sopegno, Patrizia Livreri, “Linear Quadratic Regulator: A Simple Thrust Vector Control System for Rockets”, Publisher: IEEE, August 2022
3. “LQR control design for vertical takeoff and landing aircraft”, Python Control System Library, 2022
4. Xu Chen, Masayoshi Tomizuka, Introduction to Modern Controls, with illustrations in MATLAB and Python, Nov 12, 2023