



Düzce University Journal of Science & Technology

Research Article

A Model Reference Adaptive Control Approach to Terrain Following Flight Control System

 Berk İNAN^{a,*},  İbrahim ALIŞKAN^b

^a Department of Avionics Engineering, Institute of Science, Yıldız Technical University, İstanbul, TURKEY

^b Department of Control and Automation Engineering, Faculty of Electrical and Electronics, Yıldız Technical University, İstanbul, TURKEY

* Corresponding author's e-mail address: berk.inan.tr@gmail.com

DOI: 10.29130/dubited.1595224

ABSTRACT

Automatic Flight Control System (AFCS) Terrain Following (TF) mode allows military aircraft to fly at a certain altitude above ground level at a low altitude. TF mode reduces the probability of aircraft detection by enemy airborne radars. TF mode minimizes the effort the pilot spends to control the aircraft and allows the pilot to focus on other tasks or missions. In this study, the F-16 nonlinear model is linearized around a selected equilibrium point. The state variables of the linear model are decomposed into state space matrices on the lateral and longitudinal axes. Three different control methods, namely PID (Proportional-Integral-Derivative), LQR (Linear Quadratic Regulator), and MRAC (Model Reference Adaptive Control), are used. The results show that the designed algorithms can effectively control the aircraft's altitude, speed, pitch angle, angle of attack, and pitch rate on the longitudinal axis and the aircraft flies in accordance with the terrain profile. Finally, it is observed that MRAC outperforms PID and LQR methods due to its adaptive capability.

Keywords: AFCS, F-16, LQR, MRAC, PID, Terrain Following.

Arazi Takipli Uçuş Kontrol Sistemine Model Referans Uyarlamalı Kontrol Yaklaşımı

ÖZET

Otomatik Uçuş Kontrol Sistemi (OUKS) Arazi Takibi modu askeri hava araçlarının alçak irtifada yer seviyesinin üzerinde belirli bir irtifada uçmasını sağlar. Arazi takip modu düşman radarları tarafından hava aracının tespit edilebilme olasılığını azaltır. Pilotun hava aracını kontrol etmesi için sarf ettiği iş gücünü azaltır ve pilotun diğer görevlere veya misyonlara odaklanmasına olanak tanır. Bu çalışmada, F-16 doğrusal olmayan modeli seçilen bir denge noktası etrafında doğrusallaştırılmıştır. Doğrusal modelin durum değişkenleri yatay ve dikey eksenlerde durum uzayı matrislerine ayrıştırılmıştır. PID (Oransal-İntegral-Türevsel), LQR (Doğrusal Kuadratik Regülatör) ve MRAC (Model Referans Uyarlamalı Kontrol) olmak üzere üç farklı kontrol yöntemi kullanılmıştır. Sonuçlar, tasarlanan algoritmaların uçağın boylamasına eksenindeki irtifa, hız, yunuslama açısı, hücum açısı ve yunuslama hızını etkili bir şekilde kontrol edebildiğini ve uçağın arazi profiline uygun şekilde uçuşunu göstermektedir. Son olarak, MRAC'ın adaptasyon kabiliyetinden dolayı PID ve LQR methodlarına üstünlük sağladığı gözlenmiştir.

Anahtar Kelimeler: OUKS, F-16, LQR, MRAC, PID, Arazi Takibi.

I. INTRODUCTION

Fighter jets are exceptional vehicles that contain many important advanced technologies [1]. These jets perform valuable tasks such as conducting air-to-air or air-to-ground attack operations, protecting friendly forces in the air, ensuring the security of the airspace, neutralizing enemy air defence systems, and conducting reconnaissance flights. To perform these tasks safely, advanced flight control modes are required [2]. Terrain Following (TF) mode, which is among the advanced flight control modes, is a life-saving military aviation technology used in fighter aircraft. The main purpose of using TF flight in fighter aircraft is to navigate the vehicle at high speed and minimum altitude which will minimize the possibility of being detected and tracked by enemy radars [3]. Fighter pilots have developed tactics to avoid the enemy by using terrain irregularities, vegetation, or artificial structures. This type of low-altitude flight tactic is called terrain flight by the US military. Terrain flight is divided into low-level, contour, and nap-of-the-earth. Among the three types, maximum protection is provided by nap-of-the-earth, which follows the roughness of the earth [4]. Due to the enemy's defensive capabilities, fighter jets prefer "nap-of-the-earth" flight. Nap-of-the-earth flight is performed by pilots perceiving the contours of the earth via their eyes and adjusting flight controls based on the observed terrain. In an aircraft autopilot equipped with a TF mode, terrain flight can be executed through the flight control computer, which is fed by sensor data [5]. TF flight can be conducted either by the pilot manually controlling the flight commands or by adjusting the altitude and speed through an automatic flight control system. While the pilot focuses on navigation and mission-related tasks, they must also maintain the aircraft at an altitude of approximately 50 feet above the ground. Automatic TF mode reduces the pilot's workload and enhances situational awareness [6]. Especially during night flights, in fighter jets that can be operated by a single pilot, automatic flight control system modes significantly reduce the pilot's workload while planning attacks or evading enemy threats. This provides a more efficient and safer combat capability [7]. The TF mode must operate independently of weather conditions. The TF radar scans the terrain in both altitude and azimuth. The flight control computer calculates climb and descent commands. Flight control computer sends them to the actuators. If the TF radar cannot provide sufficient information to the flight control computer due to low ground reflections, the radar altitude data is provided by the Radar Altimeter. Additionally, the Inertial Navigation System (INS) is used for attitude and velocity data. The Attitude and Heading Reference System (AHRS) provides secondary attitude and velocity information, allowing for comparison between the data sources [8].

A study utilizes optimal control theory, particularly Wiener-Hopt approaches, to develop guidance systems that minimize TF errors while maintaining stability [9]. The TF system can use digital maps instead of traditional forward-looking radar. Fuzzy PID control that integrates normal acceleration signals can be utilized for better control accuracy and stability. The results show that the system provides precise control and safe flight while improving anti-jamming capabilities [10]. F-16 longitudinal and lateral motion characteristics were compared using PID and LQR controllers. In longitudinal and lateral axes, the PID controller provided obvious superiority in terms of settling and rise time, while the LQR controller provided superiority in peak overshoot value [11]. The aircraft gains altitude basically with pitch up manoeuvres and loses altitude with pitch down manoeuvres. The genetic algorithm method is used to optimize PID and LQR parameters for pitch axis controller design to control pitch manoeuvres. As a result of the study, it is shown that the LQR controller tuned with GA responds significantly better with less settling time, and rise time in terms of PID and LQR performance comparison [12]. In addition to PID and LQR studies, the study conducted with fuzzy logic PID revealed that the fuzzy logic controller gave better results than PID and LQR in terms of rise time, settling time and overshoot percentage parameters [13]. LQR can be used as a baseline controller in MRAC control. LQR + MRAC controller design was made using the longitudinal dynamics of an aircraft similar to the F-16 [14]. There may be changes in model parameters in case the aircraft is damaged or the control surfaces malfunction. These changes cause the aircraft's equilibrium point to shift. According to the changing equilibrium point, the actuator and engine commands are adjusted adaptively thanks to MRAC, allowing the aircraft to continue its flight safely. The aircraft architecture that can tolerate faults is called the "fault tolerance flight control system" [15]. In case of actuator malfunction such as stuck or erroneous sensor reading, LQR causes the aircraft to oscillate, while MRAC provides good performance [16]. In cases where there

is no uncertainty, the LQR controller offers exceptionally good tracking performance. In case of uncertainty, it may cause loss of aircraft. As a result of the simulation, it has been proven that the baseline LQR controller and MRAC provide successful results in the presence of unknown parameters [17]. The design was made by executing MRAC with the Baseline LQR controller. It was observed that the controller gave sufficient response to the unmatched disturbance effect [18]. By creating a Lyapunov candidate function, a close loop can be created to ensure stability [19]. The purpose of MRAC is to keep the system stable by adaptively adjusting the controller gains according to the reference model and to try to get closer to the reference model response. Sensor or actuator malfunctions may occur during flight. Redundant systems are used to mitigate sensor and actuator errors. However, there is a possibility of common mode failure of the aircraft. In the case of common mode failure, the redundancy architecture may lose its effect. When a direct MRAC is used, it may have negative results on the close loop stability in sensor measurements. MRAC cannot guarantee signal boundness in the case of sensor bias. The paper recommends using Modified MRAC in the case of sensor bias error scenarios [20]. MRAC flight tests were performed on the F/A-18 jet called Full-scale Advanced Systems Testbed (FAST) by NASA. Baseline nonlinear dynamic inversion controller, basic MRAC, and complexity MRAC were tested in flights by NASA. The handling qualities of the aircraft with complex MRAC gave successful results in the pitch-roll coupling test scenario. It was concluded that complexity leads to better performance but increases the effort in terms of aircraft certification activities [21]. Studies on the verification and validation activities of adaptive controllers by the Civil Aviation Authority are ongoing. For this reason, classical control methods are preferred in civil aircraft [22].

PID and LQR controller designs were made for the F-16 aircraft with stable eigenvalues; according to the results in pitch and roll axes, it was observed that PID gave faster results in terms of rise and settling time, while LQR gave better results in terms of overshoot [23]. In the study focusing on handling qualities, the controllability of the aircraft even at high angle of attack using the NDI method was mentioned [24]. Adaptive back stepping flight control method for the F-16 aircraft was developed using neural networks; it is aimed to provide stability by adaptively updating the controller parameters in cases of structural damage and actuator malfunction [25]. A control study was performed in the outer loop for F-16 using the Receding Horizon Control technique in the longitudinal axis; with this study, control performance was increased [26]. The sliding mode control method was designed for angle of attack feedback on the F-16 longitudinal axis; better results were obtained compared to traditional control methods [27]. In case of uncertainty or actuator failure, the stability of the aircraft may be lost. The multilayer adaptive neural network method has been used to handle F-16 aircraft configuration changes during flight [28]. In the study where fractional order MRAC design was made for F-16 model pitch rate feedback, it was stated that it was more successful than integer order as it eliminated oscillation during transition [29].

In this study, F-16 aircraft MATLAB/Simulink model is used because it is open source availability. The nonlinear model of F-16 was trimmed to linearized at 5000 ft and 600 knots. Using the F-16 altitude, speed, pitch angle, angle of attack and pitch rate state variables on the longitudinal axis, TF mode design was performed with PID, LQR and MRAC control methods. The behaviour of the control surfaces, altitude, speed, angle and rate of the aircraft navigating in the canyon profile results are analysed. The aim is to provide the performance that allows the aircraft to navigate in the canyon profile without creating catastrophic results. The simulation results show the usability of the TF mode. The contribution of this paper to the literature can be presented as the development of a Model Reference Adaptive Controller under terrain profile constraints. The paper focuses on the design and implementation of TF control algorithm. It explores various control methods, including PID, LQR, and MRAC, to enhance the performance and safety of F-16 fighter jet during low altitude flights.

Remained of the paper is listed as follows: In Section 2, the F-16 model is described, containing input-output configurations, control surface limits, and the trimming and linearization processes in MATLAB. In Section 3, TF mode, PID, LQR and MRAC design stages and simulation result comparison are given. In Section 4, the conclusion includes evaluation of the TF mode controller performance and discussion of potential future work.

II. F-16 MODEL

A. F-16 MODEL OVERVIEW

The MATLAB F-16 model developed by Russell [30], based on the book by Stevens and Lewis [31] and the technical report by NASA Nguyen [32], has been used. The F-16 plant uses thrust, elevator, aileron, rudder, and leading edge flap inputs. The rudder on the lateral axis allows the aircraft to change its heading angle. The aileron on the lateral axis allows the aircraft to change its roll angle. The elevator on the longitudinal axis allows the aircraft to pitch up or pitch down. The leading edge flap on the longitudinal axis increases the aircraft's lift. The Russell model offers both high fidelity and low fidelity options. The high fidelity model has an angle of attack range from -20 to 90 degrees, while the low fidelity model has an angle of attack range from -30 to 30 degrees. In this study, the leading edge flap control input is disregarded, and the low fidelity model is used. The F-16 plant model produces outputs of north/east position, altitude, roll/pitch/yaw angle, velocity, angle of attack, sideslip, and roll/pitch/yaw rate in response to the given control inputs.

The F-16 model has saturation limits for its control inputs. Thrust operates within the range of 1,000 to 19,000 lbs, the elevator ranges from -25 to 25 degrees, the aileron from -21.5 to 21.5 degrees, and the rudder from -30 to 30 degrees. Even if excessive commands are sent to the actuators that physically move the control surfaces, the control surfaces will be able to operate within the designed limits.

B. F-16 NON-LINEAR MODEL

If the fundamental nonlinear time-invariant system model given in (1) is considered, the 6-DOF aircraft nonlinear model and the model states can be presented as follows [33].

$$\dot{x} = f(x(t), u(t)) \quad (1)$$

x consists of 12 state variables and u consist of 4 control inputs that are used in the model as following:

$$x = [n_{pos} \quad e_{pos} \quad h \quad \phi \quad \theta \quad \psi \quad V \quad \alpha \quad \beta \quad p \quad q \quad r]^T \quad (2)$$

$$u = [\delta_{th} \quad \delta_e \quad \delta_a \quad \delta_r]^T \quad (3)$$

In the equations, n_{pos} and e_{pos} are the north and east positions, h is altitude, ϕ , θ , and ψ are the roll, pitch, and yaw angles, V is total velocity, α is the angle of attack, β is the sideslip angle, p , q , and r are angular rates around the roll, pitch, and yaw axes. δ_{th} , δ_e , δ_a and δ_r are the throttle, elevator, aileron and rudder control inputs of the system. u , v , and w are velocity components along the x , y , z body axes in given Figure 1.

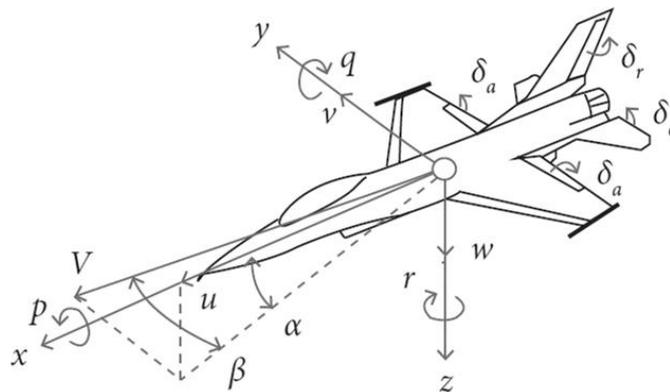


Figure 1. Aircraft variables and control inputs representation [34]

The equations of motion from NASA F-16 wind tunnel tests [32] and the Steven and Lewis book [31] are found in the “c function” of the Russell MATLAB/Simulink model [30] and are presented in (4) through (18).

Forces:

$$\dot{u} = rv - qw - g \sin \theta + \frac{\bar{q}S}{m} C_{X,t} + \frac{T}{m} \quad (4)$$

$$\dot{v} = pw - ru + g \cos \theta \sin \phi + \frac{\bar{q}S}{m} C_{Y,t} \quad (5)$$

$$\dot{w} = qu - pv + g \cos \theta \cos \phi + \frac{\bar{q}S}{m} C_{Z,t} \quad (6)$$

Moments:

$$\dot{p} = \frac{J_z L_{tot} + J_{xz} N_{tot} - (J_z(J_z - J_y) + J_{xz}^2)qr + J_{xz}(J_x - J_y + J_z)pq + J_{xz}qH_{eng}}{J_x J_z - J_{xz}^2} \quad (7)$$

$$\dot{q} = \frac{M_{tot} + (J_z - J_x)pr - J_{xz}(p^2 - r^2) - rH_{eng}}{J_y} \quad (8)$$

$$\dot{r} = \frac{J_x N_{tot} + J_{xz} L_{tot} + (J_x(J_x - J_y) + J_{xz}^2)pq - J_{xz}(J_x - J_y + J_z)qr + J_x q H_{eng}}{J_x J_z - J_{xz}^2} \quad (9)$$

Navigations:

$$\dot{n}_{pos} = u \cos \theta \cos \psi + v(\sin \phi \cos \psi \sin \theta - \cos \phi \sin \psi) + w(\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \quad (10)$$

$$\dot{e}_{pos} = u \cos \theta \sin \psi + v(\sin \phi \sin \psi \sin \theta + \cos \phi \cos \psi) + w(\cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi) \quad (11)$$

$$\dot{h} = u \sin \theta - v \sin \phi \cos \theta - w \cos \phi \cos \theta \quad (12)$$

Kinematics:

$$\dot{\phi} = p + \tan \theta (q \sin \phi + r \cos \phi) \quad (13)$$

$$\dot{\theta} = q \cos \phi - r \sin \phi \quad (14)$$

$$\dot{\psi} = \frac{q \sin \phi + r \cos \phi}{\cos \theta} \quad (15)$$

Velocity and aerodynamic angles:

$$\dot{V} = \frac{u\dot{u} + v\dot{v} + w\dot{w}}{V} \quad (16)$$

$$\dot{\alpha} = \frac{u\dot{w} - w\dot{u}}{u^2 + w^2} \quad (17)$$

$$\dot{\beta} = \frac{\dot{v}V_t - v\dot{V}}{V^2 \cos \beta} \quad (18)$$

where L_{tot} , M_{tot} , and N_{tot} are total aerodynamic moments calculated using the coefficients obtained from the NASA wind tunnel test results, J_x , J_y , J_z , and J_{xz} are moments of inertia taken from NASA wind tunnel test data, H_{eng} is the engine moment along the roll axis, $C_{X,t}$, $C_{Y,t}$ and $C_{Z,t}$ force coefficients of x, y, z axis calculated via NASA wind tunnel test data, \bar{q} is dynamic pressure calculated via air density and aircraft velocity, S is F-16 wing surface area, T is the thrust input and g is gravity component [32].

C. TRIM APPROACH AND LINEARIZATION

The nonlinear structure of the model can be easily seen from the equations. Non-linear system is hard to analyse in terms of controller design. The linear system is used in most of the controller algorithm designs in the literature [35], [36], [37] and [38]. In controller design, the aircraft response is examined by giving control input or external disturbance while the aircraft is at the equilibrium point. If the aircraft is not in equilibrium condition, deviation from the initial conditions occurs that are unrelated to the control inputs making the analysis more difficult. Aircraft equilibrium condition is known as trimmed flight conditions [31]. The aircraft may or may not be in equilibrium. In the equilibrium state, there are no forces or moments [39].

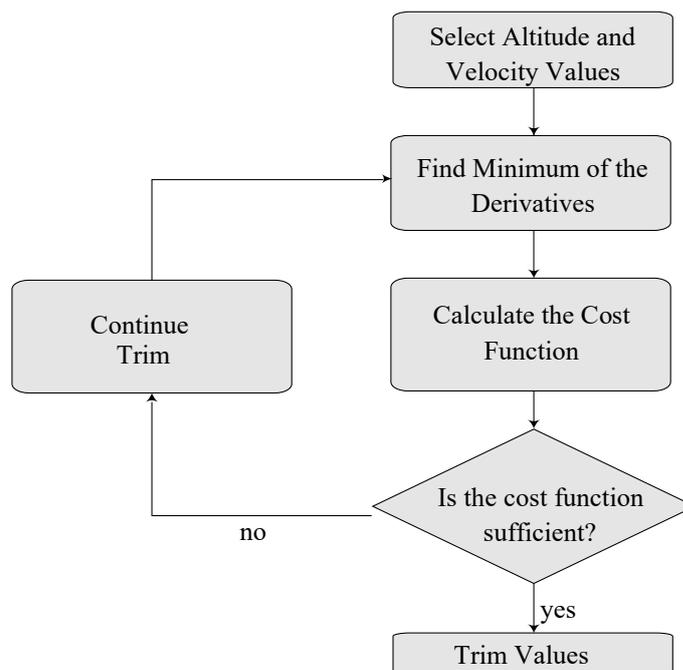


Figure 2. F-16 trim algorithm flowchart

To analyse the behaviour of the F-16 for the flight condition of 5000ft altitude and 600ft/s velocity trimming and linearization are made by using MATLAB/Simulink. Control positions at the trim point are determined with iterations. Figure 2 shows the iterative trim algorithm flowchart.

Trim point is obtained by considering the aircraft is on a steady wings level flight. To start the iteration, initial guess values are necessary given in Table 1.

Table 1. Initial guess for trim [30]

Straight Level Flight	Low Fidelity
Trust (<i>lb.</i>)	5000
Elevator (<i>deg.</i>)	-0.09
Aileron (<i>deg.</i>)	0.01
Rudder (<i>deg.</i>)	-0.01
Alpha (<i>deg.</i>)	8.49

The initial guess values in Table 1 are good values for finding the F-16 trim point [30]. The MATLAB function “fminsearch” is a Nelder and Mead Simplex algorithm that performs the minimization [40], [31]. The cost function (21) is minimized with MATLAB “fminsearch”. It tries to find the minimum values in equation (19) by iterating.

$$\dot{x} = [n_{pos} \quad e_{pos} \quad \dot{h} \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi} \quad \dot{V} \quad \dot{\alpha} \quad \dot{\beta} \quad \dot{p} \quad \dot{q} \quad \dot{r}]^T \quad (19)$$

W (20) is the created weight vector:

$$W = [0 \quad 0 \quad 5 \quad 10 \quad 10 \quad 10 \quad 2 \quad 10 \quad 10 \quad 10 \quad 10 \quad 10] \quad (20)$$

The cost function is given below:

$$Cost = \sum_{i=1}^{12} W_i x_i^2 \quad (21)$$

The near-zero convergence of the cost function with iterations is shown in Figure 3.

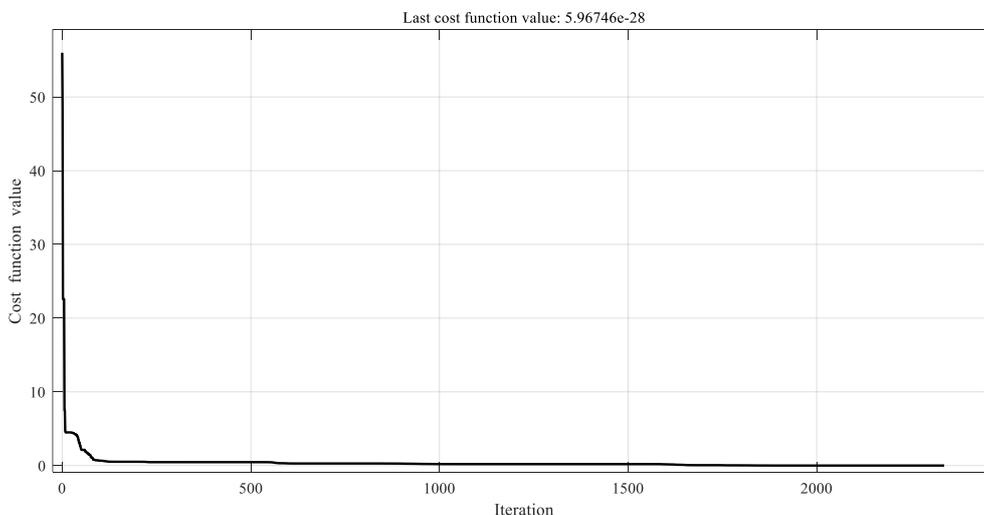


Figure 3. Cost function iterations

The cost function can be reduced to less than 1E-30 with iterations. Since cost functions below 1E-12 will have negligible tolerance in state and controls, results below 1E-12 can be used [31]. The trimming result can be found in Table 2:

Table 2. *Trimmed values*

Straight Level Flight	Low Fidelity
Trust (<i>lb.</i>)	2598.929
Elevator (<i>deg.</i>)	-1.762
Aileron (<i>deg.</i>)	-3.0554x10 ⁻¹⁵
Rudder (<i>deg.</i>)	-3.9758x10 ⁻¹⁴
Alpha (<i>deg.</i>)	1.5508
Cost	5.9675x10 ⁻²⁸

Aircraft trimmed at an equilibrium point. The MATLAB "linmod" function [41] linearizes the F-16 nonlinear model using the thrust, elevator, aileron, rudder, and alpha trim values in Table 2 [42], [43]. The obtained matrices are high dimensional since they include all the dynamics of the aircraft. The longitudinal and lateral planes are coupled. It is quite difficult to analyse. To design the controller, the movement of the aircraft is separated to two axes as longitudinal and lateral. The selected state variables (22) and control inputs (23) on the longitudinal axis are given below:

$$x_{long} = [h \quad \theta \quad V \quad \alpha \quad q]^T \quad (22)$$

$$u_{long} = [\delta_{th} \quad \delta_e]^T \quad (23)$$

The reduced model is obtained by first reducing to a longitudinal low fidelity model into a 7x7 state space with actuator dynamics. Then, this model was reduced to a 5x5 state space flight dynamics model. Actuator dynamics will be added to the Simulink model as engine and elevator dynamics. The matrix dimensions are given in Table 3.

Table 3. *F-16 matrix dimensions*

	A	B	C	D
Full Matrices	18x18	18x4	18x18	18x4
Reduced Matrices for Longitudinal Axis	7x7	7x2	5x7	5x2
Reduced Matrices for Longitudinal Axis without Actuator Dynamics	5x5	5x2	5x5	5x2

In LQR and PID designs, actuator dynamics were extracted from matrices and used as Simulink blocks. Matrix A is the system matrix that defines the F-16 dynamic behaviour in trimmed point. Matrix B is the control matrix that defines control inputs effects on system. Matrix C is the output matrix that convert the system outputs to meaningful measurements such as degree to radian. The A, B, C and D matrices to be used in the design of the F-16 TF controller, which was created as a result of the linearization process via the Russel F-16 model [30] are given equations (24) and (25).

$$\begin{bmatrix} \dot{h} \\ \dot{V}_t \\ \dot{\alpha} \\ \dot{\theta} \\ \dot{q} \\ \delta_{th} \\ \delta_e \end{bmatrix} = \begin{bmatrix} 0 & 0 & -600.0 & 600.0 & 0 & 0 & 0 & 0 \\ 1.23e-4 & -0.0136 & 17.3 & -32.2 & -0.291 & 0.00157 & 0.23 & 0 \\ 1.61e-6 & -1.78e-4 & -1.05 & 1.29e-13 & 0.919 & -7.11e-8 & -0.00221 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ -5.76e-20 & 6.36e-18 & -3.07 & 0 & -1.42 & 0 & -0.225 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -20.2 & 0 \end{bmatrix} \begin{bmatrix} h \\ V_t \\ \alpha \\ \theta \\ q \\ \delta_{th} \\ \delta_{el} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1.0 & 0 \\ 0 & 20.2 \end{bmatrix} \begin{bmatrix} u_{th} \\ u_e \end{bmatrix} \quad (24)$$

$$\begin{bmatrix} h \\ V_t \\ \alpha \\ \theta \\ q \end{bmatrix} = \begin{bmatrix} 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 57.3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 57.3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 57.3 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} h \\ V_t \\ \alpha \\ \theta \\ q \\ \delta_{th} \\ \delta_{el} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} u_{th} \\ u_e \end{bmatrix} \quad (25)$$

III. TERRAIN FOLLOWING CONTROL LAWS

A. TERRAIN FOLLOWING CONTROL LAWS WITH PID

The TF PID controller block diagram is shown in Figure 4. TF PID controller algorithm can be divided into two sub controllers: the Airspeed Controller, which sends the engine command, and the Altitude Controller, which sends the elevator actuator command.

Airspeed hold controller is used to control the engine. Airspeed varies depending on the pitch angle (θ). In the pitch down situations, the airspeed (V) of the aircraft increases. It is adjusted by changing the thrust with the airspeed hold controller. In the pitch up situations, the airspeed of the aircraft decreases. To prevent stall, thrust is increased within the limits. The difference between the selected airspeed reference (V_{ref}) and the speed (V) feedback of aircraft generates the speed error (V_{er}). The speed error (V_{er}) is processed by the velocity controller to adjust the thrust. The engine command (δ_{th}) is generated.

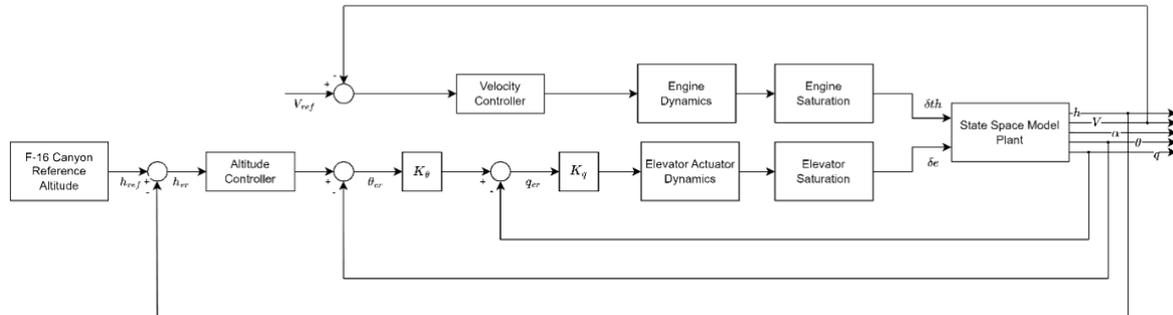


Figure 4. TF PID controller

The altitude controller is used to control the elevator actuator. TF reference altitude (h_{ref}) is generated by the F-16 Canyon Reference Altitude block. The altitude error signal (h_{er}) is generated by taking the difference between the aircraft's reference altitude (h_{ref}) and its current altitude (h). The altitude error (h_{er}) signal is processed by the altitude controller to produce the pitch angle reference (θ_{ref}) signal. The pitch angle reference (θ_{ref}) signal represents the reference needed to reduce the altitude error (h_{er}) to zero. Multiplying the pitch angle error (θ_{er}) signal with the pitch angle gain (K_θ), the pitch rate reference (q_{ref}) signal is generated. The elevator actuator command (δ_e) is generated by multiplying the difference between the pitch rate reference (q_{ref}) and the pitch rate (q) feedback by the pitch rate gain (K_q).

The velocity controller was implemented as a PID controller tuned using the step response method of Ziegler–Nichols. A step function is applied to the system as the control input. L (delay time) and T is (time constant) parameters are determined according to the step response [44]. K is selected as 0.9, L is selected as 0.07 and T is selected as 0.89.

Table 4. Ziegler Nichols PID step response method [44]

K_P	T_i	T_d	K_i	K_d
1.2/a	2L	L/2	K_P/T_i	K_P/T_d

The altitude attitude controller is implemented as P controllers in Simulink and these were tuned using the Simulink auto-tuning tool [45]. Controller performance improvement was made by changing the gains using the trial and error method in Simulink. The gains are given in Table 5.

Table 5. PID controller gains

Controller	Feedback	P	I	D
Velocity Controller	Airspeed	16.9524	484.3537	0.5933
	Pitch Rate	-0.75	0	0
Altitude Controller	Pitch Angle	350	0	0
	Altitude	33	7	6.7

B. TERRAIN FOLLOWING CONTROL LAWS WITH LQR

The TF LQR controller block diagram is shown in Figure 5. The TF LQR controller algorithm can be divided into two loops: the Inner Loop, which increases stability, and the Outer Loop, which reduces altitude error.

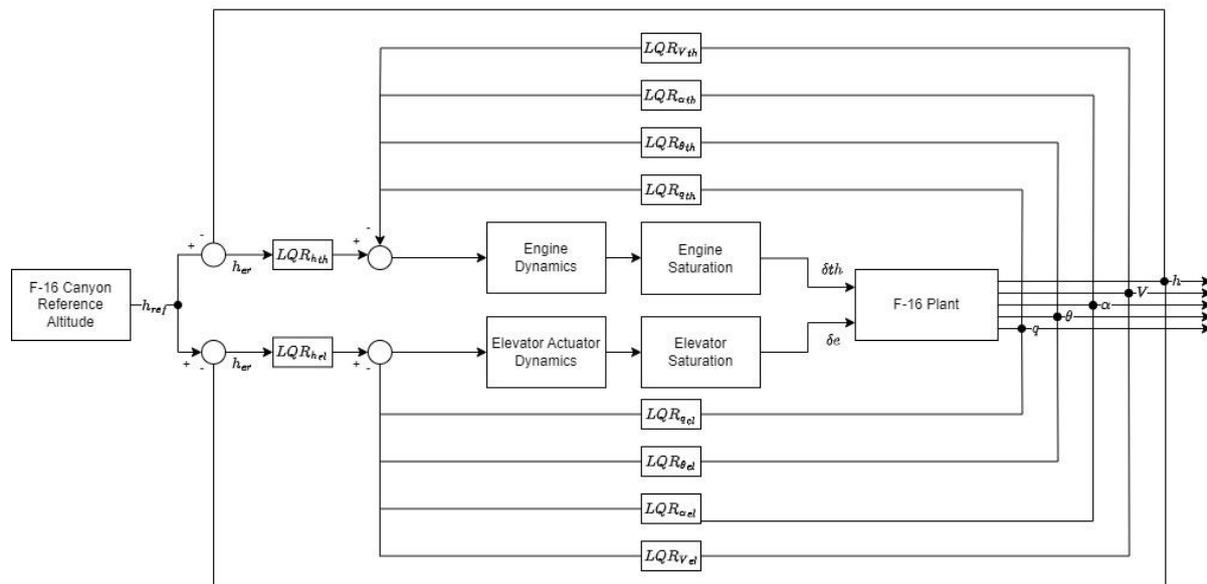


Figure 5. TF LQR Controller

Each LQR gain is adjusted to control the motor and elevator actuator. The experimentally selected Q and R matrices affect the controller performance. The Q matrix ensures that the state errors approach zero. The R matrix affects the magnitude and response speed of the controller output. The selected Q – cost weighted matrix and R – cost weighted matrix are given below:

$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0.1 & 0 & 0 & 0 \\ 0 & 0 & 0.0001 & 0 & 0 \\ 0 & 0 & 0 & 0.00001 & 0 \\ 0 & 0 & 0 & 0 & 0.01 \end{bmatrix} \quad (26)$$

$$R = \begin{bmatrix} 0.005 & 0 \\ 0 & 0.5 \end{bmatrix} \quad (27)$$

The MATLAB “lqr” function calculates the LQR gains using the algebraic Riccati equation. TF LQR gains are given in Table 6:

Table 6. LQR controller gains

Gain	Engine	Elevator
Altitude	0.07	-4.47
Airspeed	1.27	-0.04
Angle of Attack	-15.2	1010
Pitch Angle	16	-1300
Pitch Rate	1.07	-58.5

C. TERRAIN FOLLOWING CONTROL LAWS WITH MRAC

The TF MRAC controller block diagram is shown in Figure 6.

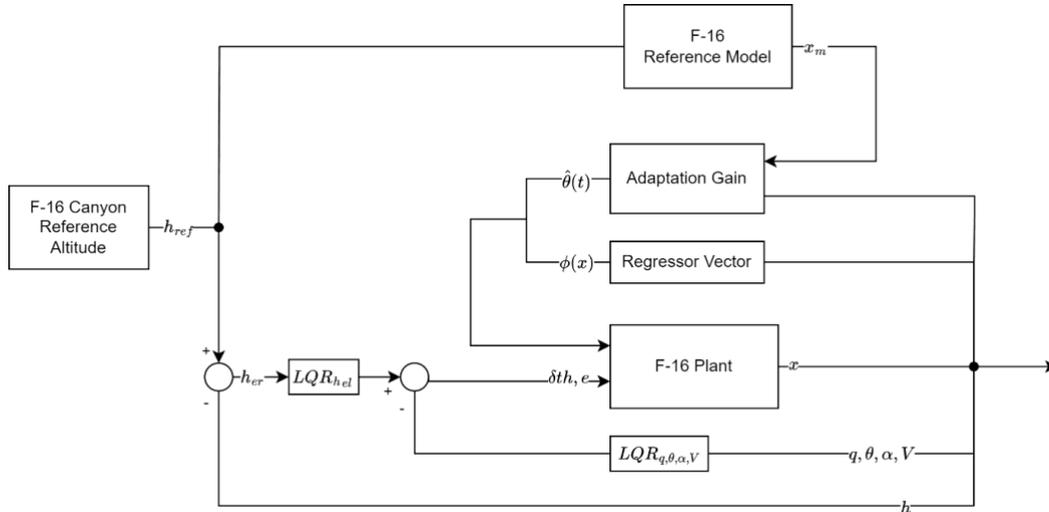


Figure 6. TF MRAC Controller

Plant model can be expressed as (28):

$$\dot{x} = Ax + B(u + f(x)) \quad (28)$$

$f(x)$ is matched uncertainty. θ^T is a constant matrix. $\phi(x)$ is known regressor vector.

$$f(x) = \theta^T \phi(x) \quad (29)$$

Reference model can be expressed as (30):

$$\dot{x}_m = A_m x_m + B_m r \quad (30)$$

In the absence of any disturbance input, the difference between the plant model and reference model state variables becomes zero.

$$\dot{e} = \dot{x} - \dot{x}_m \quad (31)$$

LQR is used as the baseline controller. When the difference between the plant model and the reference model is not zero, MRAC starts generating commands. Control inputs are given in equation (32).

$$\begin{aligned} u &= u_{mrac} + u_{lqr} \\ &= K_x^T x + K_r^T r - \theta^T \phi(x) \end{aligned} \quad (32)$$

The matching condition between the plant model and the reference model is given in equation (33).

$$\begin{aligned} \dot{e} &= Ax + B(u + f(x)) - A_m x_m + B_m r \\ &= Ax + B(K_x^T x + K_r^T r - \theta^T \phi(x) + \theta^T \phi(x)) - A_m x_m + B_m r \\ &= Ax + B(K_x^T x + K_r^T r) - A_m x_m + B_m r \\ &= (A + BK_x^T)x + BK_r^T r - A_m x_m - B_m r \end{aligned} \quad (33)$$

The reference model matrices A_m and B_m are defined by the LQR gains at Table 6. A_m is Hurwitz. LQR is used as the MRAC baseline controller [46]. A_m is created using the LQR gains in Table 6.

$$A_m = A + BK_x^T \quad (34)$$

$$B_m = BK_r^T \quad (35)$$

$\tilde{\theta}$ parameter estimation error is given in equation (36).

$$\tilde{\theta} = \theta - \hat{\theta}(t) \quad (36)$$

When the derivative of $\tilde{\theta}$ is taken, the constant parameter becomes zero as in equation (37).

$$\dot{\tilde{\theta}} = -\dot{\hat{\theta}}(t) \quad (37)$$

The calculated adaptation gain is added to the control input in equation (38).

$$u = K_x^T x + K_r^T r - \hat{\theta}^T(t) \phi(x) \quad (38)$$

Equality (38) is written into equality (28) to obtain x .

$$\begin{aligned}
\dot{x} &= Ax + Bu + Bf(x) \\
&= Ax + B \left(K_x^T x + K_r^T r - \hat{\theta}^T(t) \phi(x) \right) + B\theta^T \phi(x) \\
&= Ax + BK_x^T x + BK_r^T r - B\hat{\theta}^T(t) \phi(x) + B\theta^T \phi(x) \\
&= (A + BK_x^T)x + BK_r^T r + B(-\hat{\theta}^T(t) + \theta^T) \phi(x) \\
&= (A + BK_x^T)x + BK_r^T r + B\tilde{\theta} \phi(x)
\end{aligned} \tag{39}$$

For use in Lyapunov analysis, the error equation is found as in equation (40).

$$\begin{aligned}
\dot{e} &= (A + BK_x^T)x + BK_r^T r + B\tilde{\theta} \phi(x) - A_m x_m - B_m r \\
&= A_m e + B\tilde{\theta} \phi(x)
\end{aligned} \tag{40}$$

The Lyapunov candidate function is chosen as in (41).

$$V(e, \theta) = e^T P e + \text{tr}[\tilde{\theta}^T \Gamma_\theta^{-1} \tilde{\theta}] \tag{41}$$

The Lyapunov function satisfies the stability condition when $\dot{V} < 0$. The adaptation gain is adjusted to satisfy this condition. The goal is to find the adaptation gain that makes the derivative of the Lyapunov function less than zero. The term $-e^T Q e$ is multiplied by the positive semi-definite matrix Q . Since the result of this multiplication is always negative, it does not affect the equation (43). By using equation (42) in conjunction with Barbalat's Lemma, the tracking error is asymptotically stable with $e(t) \rightarrow 0$ as $t \rightarrow \infty$ [47].

$$\begin{aligned}
\dot{V} &= \dot{e}^T P e + e^T P \dot{e} + \text{tr}[\dot{\tilde{\theta}}^T \Gamma_\theta^{-1} \tilde{\theta}] + \text{tr}[\tilde{\theta}^T \Gamma_\theta^{-1} \dot{\tilde{\theta}}] \\
&= \dot{e}^T P e + e^T P \dot{e} + \text{tr}\left[(-\dot{\hat{\theta}}(t)^T) \Gamma_\theta^{-1} \tilde{\theta}\right] + \text{tr}\left[\tilde{\theta}^T \Gamma_\theta^{-1} (-\dot{\hat{\theta}}(t))\right] \\
&= (e^T A_m^T + \phi(x)^T \tilde{\theta}^T B^T) P e + e^T P (A_m e + B\tilde{\theta} \phi(x)) \\
&\quad + \text{tr}\left[(-\dot{\hat{\theta}}(t)^T) \Gamma_\theta^{-1} \tilde{\theta}\right] + \text{tr}\left[\tilde{\theta}^T \Gamma_\theta^{-1} (-\dot{\hat{\theta}}(t))\right] \\
&= e^T A_m^T P e + \phi(x)^T \tilde{\theta}^T B^T P e + e^T P A_m e + e^T P B \tilde{\theta} \phi(x) \\
&\quad + \text{tr}\left[(-\dot{\hat{\theta}}(t)^T) \Gamma_\theta^{-1} \tilde{\theta}\right] + \text{tr}\left[\tilde{\theta}^T \Gamma_\theta^{-1} (-\dot{\hat{\theta}}(t))\right] \\
&= -e^T Q e + \phi(x)^T \tilde{\theta}^T B^T P e + e^T P B \tilde{\theta} \phi(x) \\
&\quad + \text{tr}\left[(-\dot{\hat{\theta}}(t)^T) \Gamma_\theta^{-1} \tilde{\theta}\right] + \text{tr}\left[\tilde{\theta}^T \Gamma_\theta^{-1} (-\dot{\hat{\theta}}(t))\right]
\end{aligned} \tag{42}$$

The MRAC adaptation gain that makes equation (42) negative is chosen as (43) or (44). $\hat{\theta}$ and its transpose is considered equal.

$$\hat{\theta} = \Gamma_{\theta} e^T P B \phi(x) \quad (43)$$

$$\hat{\theta}^T = \phi(x)^T B^T P^T e \Gamma_{\theta}^T \quad (44)$$

The result of the Lyapunov function that provides stability is shown in equation (45).

$$\begin{aligned} \dot{V} = & -e^T Q e + \phi(x)^T \tilde{\theta}^T B^T P e + e^T P B \tilde{\theta} \phi(x) \\ & + tr[(-\phi(x)^T B^T P^T e \Gamma_{\theta}^T) \Gamma_{\theta}^{-1} \tilde{\theta}] + tr[\tilde{\theta}^T \Gamma_{\theta}^{-1} (-\Gamma_{\theta} e^T P B \phi(x))] \end{aligned} \quad (45)$$

The selected semi-positive definite P is shown in matrix (46).

$$P = \text{diag}(0 \quad 0 \quad 0.0025 \quad 0 \quad 0.0002375 \quad 0 \quad 0) \quad (46)$$

(47) uses angle of attack and pitch rate data. Altitude, airspeed and pitch angle feedback are not used.

$$\phi(x) = [0 \quad 0 \quad \alpha \quad 0 \quad p]^T \quad (47)$$

Adaptation rate directly affects MRAC performance. A higher adaptation rate provides faster tracking but causes the aircraft to oscillate. The optimum adaptation matrix selected is given in (48).

$$\Gamma = \text{diag}(0 \quad 0 \quad 135 \quad 0 \quad 0 \quad 0 \quad 0) \quad (48)$$

The main purpose of MRAC is to generate commands according to the reference model when there is a disturbance (e.g. actuator failure, structural deterioration, sensor bias). In the MRAC study, equation (49) is injected [48] into the plant model as matched uncertainty.

$$\theta^T = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.4 & 0 & 0 \end{bmatrix} \quad (49)$$

D. COMPARISON OF PID, LQR AND MRAC TERRAIN FOLLOWING MODE

In this section, at first, an artificial flight path was created to test the developed control algorithms. A canyon model was selected to demonstrate the different maneuvers. The canyon starts at 150ft AGL (above ground level). After flying at a steady level, the F-16 begins a climb maneuver when it encounters the hill. After the climb, it transitions into a steady level flight by performing a pitch-down. Subsequently, it begins a descent maneuver with pitch-down action. When the altitude reaches 150 ft above the ground, the aircraft performs a pitch-up maneuver to transition back to steady flight. After two climb maneuver, F-16 completes the terrain following flight. Figure 7 shows the outputs of the PID, LQR and MRAC controllers according to the sensor readings. Sensor delays are ignored. The ± 140 ft altitude is selected as the success criterion of the controllers. All three controllers operate within the defined band.

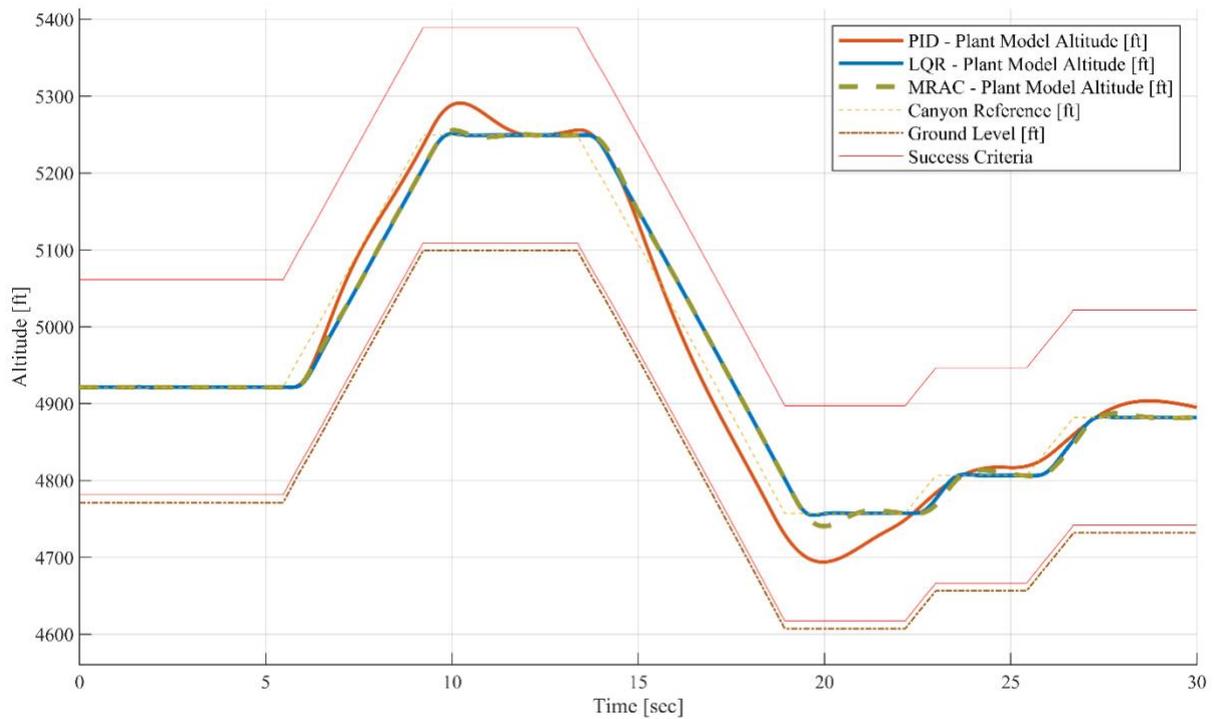


Figure 7. PID, LQR and MRAC Terrain Following

Figure 8 shows elevator commands generated by the controllers. While the PID controller produces an aggressive elevator command, LQR and MRAC produce smooth commands. While following the terrain following path, saturation is observed especially in the elevator commands in MRAC.

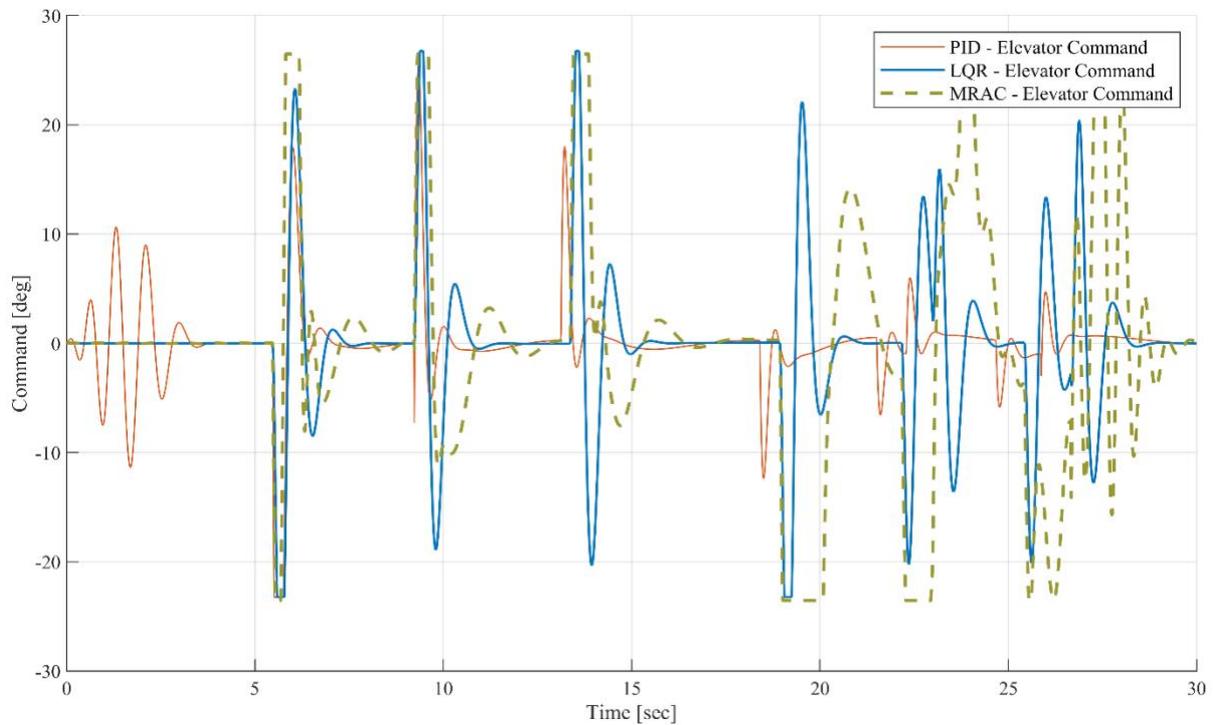


Figure 8. PID, LQR and MRAC elevator commands

When the values of PID, LQR, and MRAC TF designs are examined with the mean absolute error method, MRAC has the lowest error value with 13.3. The LQR method gave the best result after MRAC

with 15.7. The PID method provides low precision control compared to other methods with a value of 17.1.

PID, LQR and MRAC performances compared in Table 7. Despite parametric uncertainty, MRAC demonstrated good performances in maneuvers. PID method showed slow settling time and high overshoot responses. The following data show the usability of MRAC.

Table 7. Controller performances

Flight Phase	PID		LQR		MRAC	
	Settling Time [sec]	Overshoot [ft]	Settling Time [sec]	Overshoot [ft]	Settling Time [sec]	Overshoot [ft]
1. Climb	2,20	41,0	1,76	44,70	1,48	32,8
2. Transition to Steady Flight	2,28	41,7	0,55	1,77	0,69	6,23
3. Descent	2,32	46,2	1,71	43,47	2,50	39,7
4. Transition to Steady Flight	2,80	63,3	0,56	2,35	1,33	16,72
5. Climb	0,66	18,8	0,67	30,34	0,67	29,25
6. Transition to Steady Flight	1,83	10,8	0,57	1,35	1,64	6,8
7. Climb	1,08	18,7	0,92	30,52	1,02	25,9
8. Transition to Steady Flight	2,71	21,7	0,60	1,25	1,41	5,5

Figure 9 shows the behavior of MRAC and LQR against parametric uncertainty. It is seen that LQR cannot provide stable operation of the system in case of parametric uncertainty (49). In the simulation, it is seen that the aircraft controlled by LQR collided with the ground and crashed.

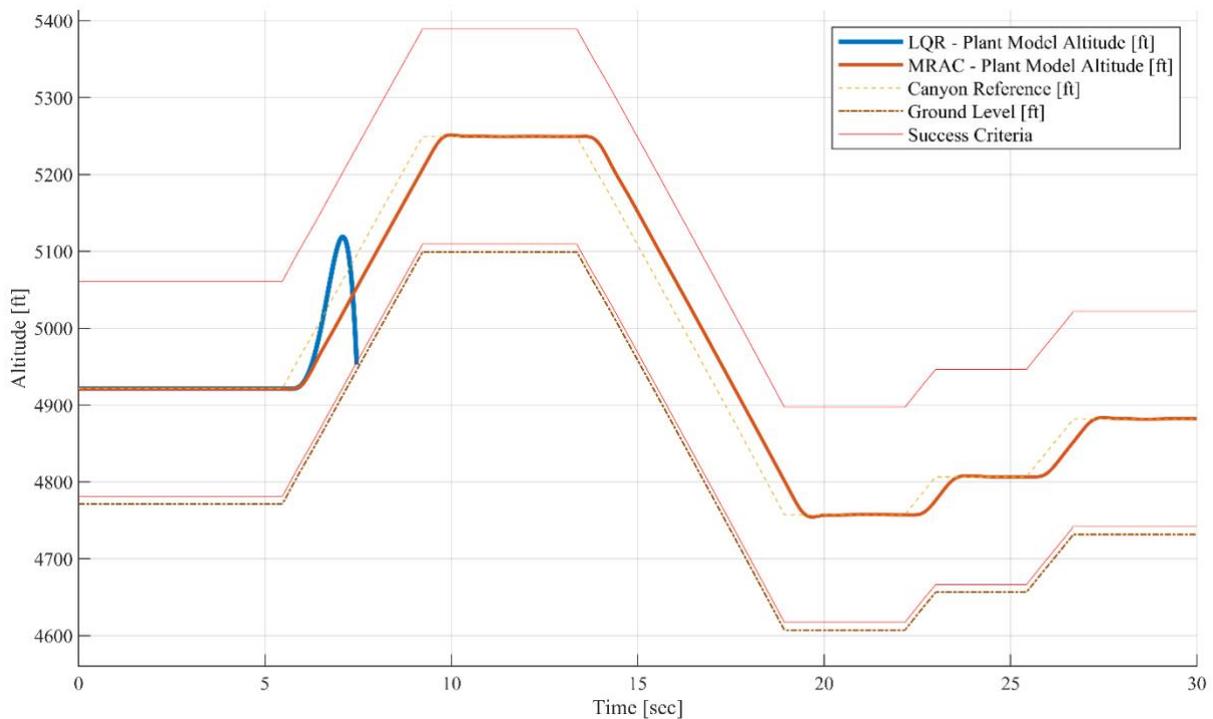


Figure 9. LQR and MRAC parametric uncertainty results

IV. CONCLUSION

In the study, F-16 was modelled on the longitudinal axis. Matched uncertainty was given to the model and a suitable MRAC was designed. MRAC was tested for TF mode. Concept designs of modern and classic controllers were shown for the TF mode. The results show that LQR and MRAC controllers give better results than PID. The MRAC controller stands out compared to PID and LQR controllers due to its adaptive gain adjustment ability. The study shows that the MRAC method can be used as an alternative to PID and LQR methods.

Additionally, the elevator actuator, which is the basic component of attitude tracking performed on the longitudinal axis, has technical features that directly affect controller design. Future studies will focus on evaluating controller performance using different actuators. Furthermore, the actuator leaning on the mechanical stop point, called saturation, affects the autopilot performance of aircraft. AFCS MRAC algorithms and AFCS system architectures that operate to keep the actuator away from saturation limits will be studied. Moreover, TF mode logical design studies will be conducted on sensor selection, sensor redundancy, mode engagement, or disengagement conditions.

ACKNOWLEDGEMENTS: This work is supported by the 2210-D Domestic Industry-Oriented Master's Scholarship Program of the TÜBİTAK Scientist Support Programs Directorate and is carried out within the scope of Turkish Aerospace's 'Problem and Thesis Pool 2021-1'.

V. REFERENCES

- [1] A. Bongers and J. L. Torres, "Technological change in U.S. jet fighter aircraft," *Res Policy*, vol. 43, no. 9, pp. 1570–1581, Nov. 2014.
- [2] I. Khademi, B. Maleki, and A. Nasserri Mood, "Optimal three dimensional Terrain Following/Terrain Avoidance for aircraft using direct transcription method," in *2011 19th Mediterranean Conference on Control & Automation (MED)*, Corfu, Greece, Jun. 2011, pp. 254–258.
- [3] P. Lu and B. L. Pierson, "Optimal Aircraft Terrain Following Analysis and Trajectory Generation," *Journal of Guidance, Control and Dynamics*, vol. 18, no. 3, pp. 555–560, 1995.
- [4] V. H. Cheng and B. Sridhar, "Technologies for Automating Rotorcraft Nap-of-the-Earth Flight," *Journal of the American Helicopter Society*, vol. 38, no. 2, pp. 78–87, 1993.
- [5] E. N. Johnson and J. G. Mooney, "A Comparison of Automatic Nap-of-the-earth Guidance Strategies for Helicopters," *J Field Robot*, vol. 31, no. 4, pp. 637–653, 2014.
- [6] R. Strenzke, J. Uhrmann, A. Benzler, F. Maiwald, A. Rauschert, and A. Schulte, "Managing Cockpit Crew Excess Task Load in Military Manned-Unmanned Teaming Missions by Dual-Mode Cognitive Automation Approaches," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, USA, Aug. 2011, p. 6237.
- [7] F. Barfield, J. Probert, and D. Browning, "All Terrain Ground Collision Avoidance and Maneuvering Terrain Following for Automated Low Level Night Attack," *IEEE Aerospace and Electronic Systems Magazine*, vol. 8, no. 3, pp. 40–47, 1993.

- [8] T. Fleck, "Flight Testing of the Autopilot and Terrain Following Radar System in the Tornado Aircraft," in *AIAA 2nd Flight Testing Conference*, Las Vegas, NV, USA., Nov. 1983, p. 2770.
- [9] R. F. Whitbeck and J. Wolkovitch, "Optimal Terrain Following Feedback Control for Advanced Cruise Missiles," 1982.
- [10] D. Qu, D. Yu, J. Cheng, and B. Lu, "On Terrain Following System with Fuzzy-PID Controller," in *IEEE Chinese Guidance, Navigation and Control Conference*, Yantai, China, Aug. 2014, pp. 497–501.
- [11] W. Ahmed, Z. Li, H. Maqsood, and B. Anwar, "System Modeling and Controller Design for Lateral and Longitudinal Motion of F-16," *Automation, Control and Intelligent Systems*, vol. 4, no. 1, pp. 39–45, 2019.
- [12] Vishal and J. Ohri, "GA tuned LQR and PID controller for aircraft pitch control," in *IEEE 6th India International Conference on Power Electronics (IICPE)*, Kurukshetra, India, Dec. 2014, pp. 1–6.
- [13] M. R. Rahimi, S. Hajighasemi, and D. Sanaei, "Designing and Simulation for Vertical Moving Control of UAV System using PID, LQR and Fuzzy Logic," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 3, no. 5, pp. 651–659, 2013.
- [14] E. Lavretsky, R. Gadiant, and I. M. Gregory, "Predictor-Based Model Reference Adaptive Control," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 4, pp. 1195–1201, Jun. 2010.
- [15] S. Zhang, Y. Feng, and D. Zhang, "Application Research of MRAC in Fault-tolerant Flight Controller," *Procedia Eng*, vol. 99, pp. 1089–1098, 2015.
- [16] S. Syed, Z. Khan, M. Salman, and U. Ali, "Adaptive Flight Control of an Aircraft with Actuator Faults," in *IEEE International Conference on Robotics & Emerging Allied Technologies (ICREATE)*, Islamabad, Pakistan, 2014, pp. 249–254.
- [17] Z. Lachini, A. Khosravi, and P. Sarhadi, "Model Reference Adaptive Control Based on Linear Quadratic Regulator for a Vehicle Lateral Dynamics," *International Journal of Mechatronics, Electrical and Computer Technology*, vol. 4, no. 13, pp. 1726–1745, 2014.
- [18] X. Wang, X. Chen, and L. Wen, "The LQR baseline with adaptive augmentation rejection of unmatched input disturbance," *Int J Control Autom Syst*, vol. 15, no. 3, pp. 1302–1313, 2017.
- [19] J. Roshanian and E. Rahimzadeh, "Novel Model Reference Adaptive Control with Application to Wing Rock Example," *Proc Inst Mech Eng G J Aerosp Eng*, vol. 135, no. 13, pp. 1911–1929, 2021.
- [20] P. Patre and S. M. Joshi, "Accommodating Sensor Bias in MRAC for State Tracking," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, USA, Aug. 2011, p. 6605.
- [21] J. Schaefer, C. Hanson, M. A. Johnson, and N. Nguyen, "Handling Qualities of Model Reference Adaptive Controllers with Varying Complexity for Pitch-Roll Coupled Failures," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, USA, Aug. 2011, p. 6453.
- [22] S. A. Jacklin, "Closing the Certification Gaps in Adaptive Flight Control Software," in *AIAA Guidance, Navigation, and Control Conference*, Honolulu, HI, USA, Aug. 2008, p. 6988.

- [23] M. Norouzi and E. Caferov, "Investigating Dynamic Behavior and Control Systems of the F-16 Aircraft: Mathematical Modelling and Autopilot Design," *International Journal of Aviation Science and Technology*, vol. 4, no. 2, pp. 75–86, 2023.
- [24] C. Kim, C. Ji, G. Koh, and N. Choi, "Review on Flight Control Law Technologies of Fighter Jets for Flying Qualities," *International Journal of Aeronautical and Space Sciences*, vol. 24, no. 1, pp. 209–236, 2023.
- [25] L. Sonneveldt, Q. P. Chu, and J. A. Mulder, "Constrained Adaptive Backstepping Flight Control: Application to a Nonlinear F-16/MATV Model," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2006, p. 6413.
- [26] T. Keviczky and G. J. Balas, "Receding horizon control of an F-16 aircraft: A comparative study," *Control Eng Pract*, vol. 14, no. 9, pp. 1023–1033, 2006.
- [27] S. Seshagiri and E. Promptun, "Sliding Mode Control of F-16 Longitudinal Dynamics," in *American Control Conference*, Seattle, WA, USA, Jun. 2008, pp. 1770–1775.
- [28] D. T. Ocaña, H.-S. Shin, and A. Tsourdos, "Development of a nonlinear reconfigurable f-16 model and flight control systems using multilayer adaptive neural networks," *IFAC-PapersOnLine*, vol. 48, no. 9, pp. 138–143, 2015.
- [29] G. E. Ceballos Benavides, M. A. Duarte-Mermoud, M. E. Orchard, and A. Ehijo, "Enhancing the Pitch-Rate Control Performance of an F-16 Aircraft Using Fractional-Order Direct-MRAC Adaptive Control," *Fractal and Fractional*, vol. 8, no. 6, p. 338, 2024.
- [30] R. Russell, "Non-linear F-16 Simulation using Simulink and Matlab," Version 1.0, University of Minnesota, Minneapolis, MN, USA, Tech. Rep., Jun. 2003.
- [31] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft Control and Simulation*, 3rd ed. Hoboken, NJ, USA: John Wiley & Sons, 2015.
- [32] L. T. Nguyen, M. E. Ogburn, W. P. Gilbert, K. S. Kibler, P. W. Brown, and P. L. Deal, "Simulator Study of Stall/Post-Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability," *NASA Technical Paper 1538*, vol. 12854, 1979.
- [33] Y. Wei, H. Xu, and Y. Xue, "Adaptive Neural Networks-Based Dynamic Inversion Applied to Reconfigurable Flight Control and Envelope Protection under Icing Conditions," *IEEE Access*, vol. 8, pp. 11577–11594, 2020.
- [34] M. Sadeghi, A. Abaspour, and S. H. Sadati, "A novel integrated guidance and control system design in formation flight," *Journal of Aerospace Technology and Management*, vol. 7, no. 4, pp. 432–442, 2015.
- [35] I. Necoara, L. Ferranti, and T. Keviczky, "An adaptive constraint tightening approach to linear model predictive control based on approximation algorithms for optimization," *Optim Control Appl Methods*, vol. 36, no. 5, pp. 648–666, Sep. 2015.
- [36] M. R. Mortazavi and A. Naghash, "Pitch and flight path controller design for F-16 aircraft using combination of LQR and EA techniques," *Proc Inst Mech Eng G J Aerosp Eng*, vol. 232, no. 10, pp. 1831–1843, Apr. 2018.

- [37] Y. Zhu, J. Chen, B. Zhu, and K. Y. Qin, "Synchronised trajectory tracking for a network of MIMO non-minimum phase systems with application to aircraft control," *IET Control Theory & Applications*, vol. 12, no. 11, pp. 1543–1552, Jul. 2018.
- [38] Z. Oreg, H. S. Shin, and A. Tsourdos, "Model identification adaptive control - Implementation case studies for a high manoeuvrability aircraft," in *27th Mediterranean Conference on Control and Automation, MED*, Akko, Israel, Jul. 2019, pp. 559–564.
- [39] O. Albostan, "Flight Control System Design of F-16 Aircraft using Robust Eigenstructure Assignment," Graduate School of Science, Engineering and Technology, İstanbul Technical University, İstanbul, Türkiye, 2017.
- [40] J. C. Lagarias, J. Reeds, M. H. Wright, and P. E. Wright, "Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions," *SIAM Journal on Optimization*, vol. 9, no. 1, pp. 112–147, 1998.
- [41] H. Aktan and H. Demircioğlu, "Speed and Altitude Hold Autopilot Design and Simulation for F-16 Fighter Aircraft by Using CGPC Method," in *20th National Conference on Automatic Control (TOK 2018)*, Kayseri, Türkiye, Sep. 2018, pp. 58–64.
- [42] "linmod." Accessed: Nov. 16, 2024. [Online]. Available: <https://www.mathworks.com/help/simulink/slref/linmod.html>
- [43] A. Halder, K. Lee, and R. Bhattacharya, "Probabilistic Robustness Analysis of F-16 Controller Performance: An Optimal Transport Approach," in *2013 American Control Conference*, Washington, DC, USA: IEEE, 2013, pp. 5562–5567.
- [44] D. Xue, Y. Chen, and D. P. Atherton, *Linear Feedback Control: Analysis and Design with MATLAB*, 1st ed. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2007.
- [45] "PID Tuner." Accessed: Nov. 16, 2024. [Online]. Available: <https://www.mathworks.com/help/control/ref/pidtuner-app.html>
- [46] A. F. Ajami, "Adaptive Flight Control in the Presence of Input Constraints," M.S. thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, 2005.
- [47] N. T. Nguyen, *Model-reference adaptive control*, 1st ed. London, UK: Springer International Publishing, 2018.
- [48] N. T. Nguyen and S. N. Balakrishnan, "Bi-objective optimal control modification adaptive control for systems with input uncertainty," *IEEE/CAA Journal of Automatica Sinica*, vol. 1, no. 4, pp. 423–434, 2014.