Guidance and Control of Flying Vehicle by Explicit Model Predictive Controller

Mohammad Mahdi Soori[1], Hosein Sadati[2*]

Abstract–The integrated guidance and control (IGC) system offers a significant advantage by leveraging the synergy between guidance and control subsystems to enhance the overall performance of flying vehicles. In the context of air defense missile systems, where speed is critical for intercepting fast-moving targets, this paper introduces a novel approach for designing and implementing an explicit linear model predictive controller (MPC) specifically tailored for three-dimensional scenarios. The proposed controller is developed to minimize the time to collision and miss distance by fully exploiting the interactions within the IGC framework, significantly enhancing the response time and speed, making it suitable for high-speed air defense applications. A key innovation of this work lies in the adoption of the explicit MPC approach, where the optimization problem is solved offline for all potential state vector values. The optimal control commands are formulated as explicit functions of the state variables and stored in memory. During real-time operation, the controller rapidly evaluates these precomputed functions to generate control commands, eliminating the need for computationally expensive online optimizations. This design significantly reduces the computational load, making it particularly suitable for hardware with limited processing capacity. Simulation results validate the superior performance of the proposed explicit MPC compared to conventional PID and LQR controllers. Specifically, the IGC system employing the proposed controller demonstrated a marked reduction in both miss distance and time to collision. These findings underscore the effectiveness and practicality of the explicit MPC in improving the precision, speed, and efficiency of guidance and control for advanced flying vehicles, particularly in air defense missile systems.

Keywords: Flying vehicle, target, integrated guidance and control, model predictive control, optimization

1. Introduction

Guidance, navigation and control functions are critical to all forms of air and space vehicles, including missiles. In practice, these functions work together in series to maneuver a vehicle. It is now common to develop guidance completely separate from control (autopilot) and vice versa. Almost all textbooks and technical articles on this topic have dealt with it [1].

Some more advanced guidance algorithms not only achieve interception, but also control the interception angle of the missile upon impact. However, all these algorithms are rooted in the collision triangle concept, which minimizes the change of line of sight between the interceptor and the target, and may suffer from instability at the end of the task. In a multi-loop structure, steering is generated using engagement

kinematics while the autopilot stabilizes the body dynamics and follows the acceleration provided by the steering.

1 Department of Mechanical Engineering K. N. Toosi University of Technology Iran. Email: hosein728061@gmail.com

2* Corresponding Author : Department of Mechanical Engineering K. N. Toosi University of Technology, Iran. Email: mmsoorimech@yahoo.com

Received: 2025.01.03; Accepted: 2025.01.27

1.1 Integrated guidance and control

Unlike the conventional three-loop autopilot structure, Integrated Guidance and Control (IGC) is an integrated framework in which guidance and control are considered to be integrated within rather than independent of each other.

The advantage of IGC is their ability to use interactions between command and control subsystems. IGC intends to increase the performance of the missile by taking advantage of the synergy between the guidance and control processes. Depending on the structure of the IGC, some provide additional feedback paths in the flight control system, while others require less. Putting G&C into a single IGC system improves its optimization potential. Because optimization of parameters can be done directly. Cost functions include

key performance parameters such as missile and target relative speed of approach, line-of-sight angle, impact angle, and many parameters not readily available to autopilot are now directly available. In the conventional approach, the guidance law has no knowledge of the amount of spin or acceleration applied to the missile, instead, guidance only knows the relative position and speed of engagement. As the range-target decreases, small changes in geometry result in large acceleration commands that can exceed the performance range of the autopilot. In addition, the autopilot cannot adjust itself based on relative engagement kinematics, as it does not receive this information. As a result, conventional G&C systems rely on making the autopilot time constant as small as possible to improve stability. The autopilot time constant designs the distance from miss to target in conventional G&C systems[2].

One of the main approaches to IGC using SMC was presented in 2010 by Harrell and Balakrishnan using terminal second-order sliding mode control [3]. by Harrell and Balakrishnan using terminal second-order sliding mode (TSM) control. In 2019, Wang et al proposed an integrated guidance and control method with limited impact angle for the missile to achieve unidirectional attack capability[4]. To improve the ability to damage the target, He et al.[5] designed an integrated guidance and control law with impact angle constraint to deal with the problem of tracking unknown maneuvering targets. To deal with the limitations of stimulus saturation in real systems, Ma et al in [6] investigated an integrated control law using dynamic level control, feedforward control and adaptive neural network. And Michel and Stechel thoroughly investigated the sliding mode control for the integrated plane model[7]. In 2020, Tian et al. presented a unified model to avoid practical problems such as the field of view limitation, and solved the field limitation by converting output to input saturation[8]. In 2021, Sinha et al presented an integrated guidance and control model with limited time. In this research, due to the simplicity of the design, sliding mode control is used, while a non-linear finite time disturbance observer is used to estimate the target maneuver [9]. In 2022, Lee proposed a unified model for hypersonic homing missiles. In this design, high-speed targets are hit with proper accuracy by using the sliding model controller, and by using the Monte Carlo method, the non-hit distance was reduced to the minimum[10]. In 2023 Xiaohui Liang et al investigated the nonlinear integrated missile guidance and control system with external uncertainties and disturbances and proposed a new adaptive neural network (NN) control scheme with the help of estimates obtained by NN and disturbance observer (DOB). In this paper, the weight learning rule NN and DOB are updated according to the tracking and estimation errors. Under the operation of the proposed adaptive NN rules, a good tracking characteristic and guidance effects can be obtained for the integrated missile guidance and control (IGC) system. Finally, the simulation results of two different scenarios show the correctness of the designs. It is worth mentioning that the missile tracking process shows a smoother trajectory and a shorter distance can be achieved with the proposed NN adaptive control approach[11]. Xiangyu et al. 2024 investigates the integrated guidance and control (IGC) law design problem with impact angle and general field of view (FOV) constraints. First, the IGC model for non-maneuvering moving target tracking is parameterized by state-dependent coefficient matrices. The nominal IGC law for target interception with the desired impact angle is obtained by solving the state-dependent Riccati equation. Second, since the relative degrees of general FOV constraints exceed one according to the IGC model, high-order control barrier functions are constructed. Satisfying the FOV constraints is equivalent to ensuring that the hypersurface sets defined by the barrier functions are constant, which translate into dependent constraints on the control input. The nominal IGC law is modified in a minimally invasive way by quadratic programming. Then, the proposed method is extended to the case of maneuvering target tracking using a relative coordinate framework. Finally, numerical simulations are performed to confirm the effectiveness of the proposed method. [12]

In this article, the development of an explicit linear model predictive controller (MPC) for an integrated guidance and control (IGC) model is introduced and explored in depth. The proposed approach leverages the explicit formulation of the controller to enhance real-time performance in control systems. Unlike traditional MPCs, where optimization is performed online at every time step, the explicit controller precomputes the solution to the optimization problem offline for all feasible values of the state vector. This process involves determining the optimal control input as an explicit function of the state variables, which is subsequently stored in the controller's memory. During real-time operation, the controller simply evaluates this precomputed function at each time step, using the current values of the state vector to generate the control commands. This method significantly reduces the computational burden during runtime, as the need for solving optimization problems repeatedly in real time is eliminated. Consequently, this approach is particularly advantageous for applications where computational

resources are limited, such as hardware with constrained processing capabilities or systems requiring high-speed control actions. The benefits of this explicit MPC design were validated through simulations, which demonstrated its superior performance compared to traditional controllers like proportional-integral-derivative (PID) and linear quadratic regulator (LQR) controllers. Specifically, the results highlighted that the proposed controller, when implemented alongside the integrated guidance and control model for air defense missile systems, achieved a notable reduction in both the miss distance and the time to collision. These improvements underscore the effectiveness of the proposed method in achieving more precise, efficient, and high-speed control outcomes, making it a promising solution for advanced missile defense systems, particularly in time-critical and resource-constrained environments. The results from these simulations demonstrate that the proposed approach yields the best performance in terms of speed and accuracy for air defense missile systems.

2. Mathematical modeling

A missile-target engagement scenario involves a missile trying to intercept a target by changing course. All through homing guidance, sensors onboard the missile are used to guide the missile to impact. The initial conditions of this scenario include three main assumptions. (a) Midcourse guidance is successful, (b) The speed of the missile and the target are close to each other in the collision course, (c) In order to intercept and completely destroy the target, the impact angle of the missile and the target is limited. The engagement geometry of this conflict scenario is shown in Fig. 1.

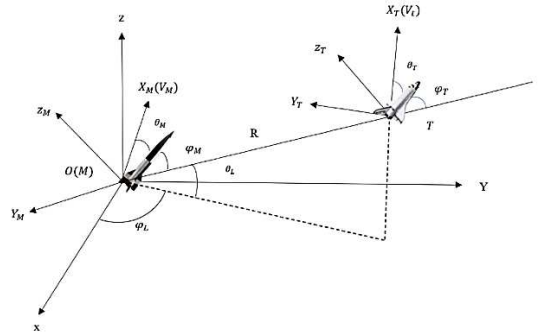

Fig. 1 -Kinematics of conflict

The general purpose of this article will be to design a proper controller for accurate target tracking. Therefore, the problem of missile-target engagement is discussed, which includes all the topics required for accurate modeling, including engagement kinematics, missile dynamics, and integrated guidance and control model. The o-xyz in Fig. 1 represents the inertial coordinate system. The coordinate system in which the missile and target velocities are described is shown in Fig. 1 as ($M-x_My_Mz_M$، $T-x_Ty_Tz_T$) , and R is the relative missile/target range. In respective orders, vectors V_M and V_T are the velocities of the missile and target, ($\theta_M\varphi_M$) and ($\theta_T\varphi_T$) are the angles of the missile and the target relative to the velocity coordinates and line of sight coordinate system, ($\theta_L\varphi_L$) is the elevation angle and the side angle[13]. The kinematics of the engagement is described by the equations

$$\ddot{R}-R\dot{\theta}_L^2-R\dot{\phi}_L^2\cos^2\dot{\theta}_L=a_{tR}-a_{mR} \quad (1)$$

$$R\ddot{\theta}_L+2\dot{R}\dot{\theta}_L+R\dot{\phi}_L^2\sin\dot{\theta}_L\cos\theta_L=a_{t\theta}-a_{m\theta} \quad (2)$$

$$-R\ddot{\phi}_L^2\cos\theta_L-2\dot{R}\dot{\phi}_L\cos\theta_L+2R\dot{\theta}_L\dot{\phi}_L\sin\theta_L=a_{t\phi} \quad (3)$$

In equations 1 to 3, ($a_{mR}, a_{m\theta}, a_{m\phi}$) and , ($a_{tR}, a_{t\theta}, a_{t\phi}$) represent the acceleration components for the missile and target, respectively. Also, θ is the angle of the flight path. Since the acceleration for the missile is usually provided by the aerodynamic force, in particular in the final phase of the guidance, closer attention must be given to the relationship between the acceleration of the missile and the aerodynamic force. The aerodynamic force on the missile is calculated in the inertial coordinate system as equations 4 to 7.

$$a_{mz}=\frac{Z}{m} \quad (4)$$

$$a_{my}=\frac{Y}{m} \quad (5)$$

$$Y=qS\left(c_y^{\alpha}\alpha+c_y^{\beta}\beta+c_y^{\delta_z}\delta_z\right) \quad (6)$$

$$Z=qS\left(c_z^{\alpha}\alpha+c_z^{\beta}\beta+c_z^{\delta_y}\delta_y\right) \quad (7)$$

In equations 4 to 7, (a_{mz}, a_{my}) are the acceleration components of the missile along the inertial coordinate system, m indicates the mass of the missile, ρ is the air density, (Y، Z) indicate the upward and lateral forces, ($q=\frac{1}{2}\rho V_m^2$) is the dynamic pressure, S represents the aerodynamic reference area of the missile, α is the angle of attack , β is the lateral slip angle, ($\delta_x\delta_y\delta_z$) denotes the deflection angles of the missile wings, ($c_z^{\beta}c_z^{\alpha}c_z^{\delta_y}$) are partial derivatives of the lift force coefficient, and ($c_z^{\beta}c_z^{\alpha}c_z^{\delta_y}$) partial derivatives are the lateral force coefficients.

By combining equations 2, 3, 6, and 7, the equations for the line of sight angle appear as equations 8 and 9, as follows.

$$\ddot{\theta}_L = -\dot{\phi}_L^2 \sin\theta_L \cos\theta_L - \frac{2\dot{R}\dot{\theta}_L}{R} - M_1 Y \cos(\gamma_V) - \frac{\sin\theta_L \sin(\phi_L - \varphi_V)}{mR}(Y\cos(\gamma_V) + Z\sin(\gamma_V)) + M_1(mg\cos\theta + Z\sin(\gamma_V)) + d_{\theta L} \quad (8)$$

$$\ddot{\phi}_L = -\frac{2\dot{R}\dot{\phi}_L}{R} + 2\dot{\theta}_L\dot{\phi}_L \tan\theta_L - M_2 Y\cos(\gamma_V) + \frac{\cos(\phi_L - \varphi_V)}{mR\cos\theta_L}(Y\cos(\gamma_V) + Z\sin(\gamma_V)) + M_2(mg\cos\theta + Z\sin(\gamma_V)) + d_{\phi L} \quad (9)$$

In equations 8 and 9, parameters $M_1$and M_2 are defined as relations 10 and 11.

$$M_1 = \frac{\cos\theta\cos\theta_L + \sin\theta\sin\theta_L\cos(\phi_L - \varphi_V)}{mR} \quad (10)$$

$$M_2 = \frac{\sin\theta\cos(\phi_L - \varphi_V)}{mR\cos\theta_L} \quad (11)$$

In equations 8 to 11, φ_V is the ballistic angle, γ_V the rotation angle, ($d_{\theta L}d_{\phi L}$) are the approximate error for the $\theta_L$and ϕ_L. Kinematic equations of flight path angle and ballistic angle are shown as equations 12 and 13, as follows.

$$\dot{\theta} = \frac{Y\cos\gamma_V - Z\sin\gamma_V - mg\cos\theta}{mV_m} \quad (12)$$

$$\dot{\varphi}_V = \frac{-Y\sin\gamma_V - Z\cos\gamma_V}{mV_m\cos\theta} \quad (13)$$

The kinematic and dynamic equations of the missile rotating around the center of mass in three-dimensional space are given as equations 14 through 16 as follows.

$$\dot{\alpha} = -\omega_x \tan\beta\cos\alpha + \omega_y \tan\beta\sin\alpha + \omega_z - \frac{Y}{mV_m\cos\theta} + \frac{g\cos\theta\cos\gamma_V}{V_m\cos\theta} \quad (14)$$

$$\dot{\beta} = \omega_x \sin\alpha + \omega_y\cos\alpha + \frac{Z + mg\cos\theta\sin\gamma_V}{mV_m} \quad (15)$$

$$\dot{\gamma_V} = \cos\alpha\sec\beta\omega_x - \sin\alpha\sec\beta\omega_y + \frac{Y(\tan\theta\sin\gamma_V + \tan\beta) + Z\tan\theta\cos\gamma_V}{mV_m} - \frac{\cos\theta\cos\gamma_V\tan\beta}{V_m}g \quad (16)$$

In equations 14 to 16, parameters ω_x , ω_y , and ω_z are the angular velocities in the roll, yaw, and pitch channels, respectively.

The dynamic equations of the state of the missile are obtained as equation 17.

$$\begin{cases} \dot{\omega_x} = \frac{J_y - J_z}{J_x}\omega_z\omega_y + \frac{M_x}{J_x} \\ \dot{\omega_y} = \frac{J_z - J_x}{J_y}\omega_x\omega_z + \frac{M_y}{J_y} \\ \dot{\omega_z} = \frac{J_x - J_y}{J_z}\omega_y\omega_x + \frac{M_z}{J_z} \end{cases} \quad (17)$$

In equation 17, $(J_x\ ,\ J_y\ ,\ J_z)$denote the moments of inertia for the roll, yaw, and pitch channels, respectively. Also, the three moments $(M_x,, M_y,\ M_z)$ are defined as in equation 18.

$$\begin{cases} M_x = qSL\left(m_x^{\alpha}\alpha + m_x^{\beta}\beta + m_x^{\delta_x}\delta_x\right) \\ M_y = qSL\left(m_y^{\beta}\beta + m_y^{\delta_y}\delta_y\right) \\ M_z = qSL\left(m_z^{\alpha}\alpha + m_z^{\delta_z}\delta_z\right) \end{cases} \quad (18)$$

In equation 18, L is the reference length, (m_x^{α},$m_x^{\beta}m_x^{\delta_x}$) are the moment coefficient of the partial derivatives related to the roll channel, ($m_y^{\beta}, m_y^{\delta_y}$) are those related to the yaw channel, and ($m_z^{\alpha}, m_z^{\delta_z}$) are those related to the pitch channel.

By combining relations 8, 9, and 14 with 18, the three-dimensional equation of integrated missile guidance and control can be written as equation 19 [13].In these equations, x the state variables of the system are

$$\begin{cases} \dot{x}_0 = x_1 \\ \dot{x}_1 = f_1 + b_1\overline{x_2} + d_1 \\ \dot{x}_2 = f_2 + b_2 x_3 + d_2 \\ \dot{x}_3 = f_3 + b_3 u + d_3 \end{cases} \quad (19)$$

, where

$x_0 = \begin{bmatrix} \theta_L - \theta_{Lf} \\ \phi_L - \phi_{Lf} \end{bmatrix}$ ، $x_1 = \begin{bmatrix} \dot{\theta}_L \\ \dot{\phi}_L \end{bmatrix}$ ، $\overline{x_2} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$, $x_2 = \begin{bmatrix} \alpha \\ \beta \\ \gamma_v \end{bmatrix}$ ،

$x_3 = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$ ، $u = \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \end{bmatrix}$,

and $d_i (i = 1, 2, 3)$ shows the approximate system error. Moreover, the matrices b_i and $f_i$are shown in relations (20) to (25).

$$b_1 = \begin{bmatrix} -M_1 q S c_y^{\alpha} & -\dfrac{q S c_z^{\beta} \sin\theta_L \sin(\phi_L - \phi_V)}{mR} \\ -M_2 q S c_y^{\alpha} & \dfrac{q S c_z^{\beta} \cos(\phi_L - \phi_V)}{mR \cos\theta_L} \end{bmatrix} \quad (20)$$

$$b_2 = \begin{bmatrix} -\tan\beta\cos\alpha & \tan\beta\sin\alpha & 1 \\ \sin\alpha & \cos\alpha & 0 \\ \cos\alpha\sec\beta & -\sin\alpha\sec\beta & 0 \end{bmatrix} \quad (21)$$

$$b_3 = \begin{bmatrix} \dfrac{qSLm_x^{\delta_x}}{J_x} & 0 & 0 \\ 0 & \dfrac{qSLm_y^{\delta_y}}{J_y} & 0 \\ 0 & 0 & \dfrac{qSLm_z^{\delta_z}}{J_z} \end{bmatrix} \quad (22)$$

$$f_1 = \begin{bmatrix} -\dfrac{2\dot{R}}{R}\dot{\theta}_L - \dot{\varphi}_L^2 \sin\theta_L \cos\theta_L + M_1 mg\cos\theta \\ -\dfrac{2\dot{R}}{R}\dot{\phi}_L + 2\dot{\theta}_L\dot{\phi}_L \tan\theta_L + M_2 mg\cos\theta \end{bmatrix} \quad (23)$$

$$f_2 = \begin{bmatrix} -\dfrac{qSY_y}{mV\cos\beta} + \dfrac{g}{V\cos\beta}\cos\gamma \\ -\dfrac{qSZ_z}{mV} + \dfrac{g}{V}\cos\theta\sin\gamma \\ -\dfrac{F_{YZ}}{mV} - \dfrac{g}{V}\cos\theta\cos\gamma\tan\beta \end{bmatrix} \quad (24)$$

$$f_3 = \begin{bmatrix} \dfrac{J_y - J_z}{J_x}\omega_z\omega_y + \dfrac{qSL\left(m_x^{\alpha}\alpha + m_x^{\beta}\beta\right)}{J_x} \\ \dfrac{J_z - J_x}{J_y}\omega_x\omega_z + \dfrac{qSLm_x^{\beta}\beta}{J_y} \\ \dfrac{J_x - J_y}{J_z}\omega_y\omega_x + \dfrac{qSLm_z^{\alpha}\alpha}{J_z} \end{bmatrix} \quad (25)$$

3 -Controller design

The missile hits the target when R (the distance between the missile and the target) approaches R_{hit} (the minimum distance between the missile and the target). The goal of the controller design in this article is to the make the angles θ_L and ϕ_L converge to their reference values θ_{Lf} and ϕ_{Lf}, respectively, and also make the distance between the missile and approach R_{hit}.

Backtracking method is used to solve the problem. And the control input (u) will be calculated by the Higher order continuous sliding mode control. The backsteping approach states that a virtual control signal named x_{2d} is designed at first. This signal is the desired behavior of the state variable x_2 and its design is such that the state variable x_0 tends to zero. So, in fact, the desired behavior of x_2 is obtained in such a way that the first goal of the control problem is established. In the next step, we design the virtual control signal x_{3d}. This signal is the reference signal of state variable x_3 and it is designed in such a way that if x_3 follows it, x_2 also follows x_{2d} and as a result x_0 will tend to zero. In the last stage, the real control input u is designed in such a way that it converges x_3 to x_{3d}, and as a result, the control loop is completed. By doing this, the second goal of the control problem (zeroing the distance between the missile and the target (R) will be established.

Two S virtual control signals are obtained by the backtracking method, but the real control signal, which is the main goal of this article, will be calculated from the explicit linear model predictive controller. To calculate virtual signals, we will have relations (26) and (27).

$$S_1 = x_0 + cx_1 \quad (26)$$

$$x_{2d} = -(cb_1)^{-1}(x_1 + cf_1 + K_1 \tanh(S_1))$$

$$S_2 = x_2 - x_{2d} \quad (27)$$

$$x_{3d} = -(b_2)^{-1}(f_2 + K_2 \tanh(S_2))$$

In equations (26) and (27), the coefficients c,K_1 , and K_2 are fixed and will be obtained with the help of optimization methods of PSO along with the variables related to the real controller.

3-1 PID controller design

PID controller is introduced as a standard control structure in classical control theory. The performance of the system is improved by accurately adjusting the values of the three proportional gains K_p, integral gain T_i, and derivative gain T_d. By adjusting these parameters, the steady state error and output fluctuations can be controlled in response to the step input. In order to evaluate the proposed controller of this article, first a simple PID controller is designed. According to the system equations, the output of the missile system is its angular velocities. These speeds are to tend to zero. As a result, tracking error is defined as equation 28.
The general form of the PID controller is of the form given in Eq. (29) as:

$$e = 0 - [\omega_x, \omega_y, \omega_z]^T = -[\omega_x, \omega_y, \omega_z]^T \quad (28)$$

$$u = K_p e + T_i \int e dt + T_d \left(\frac{de}{dt}\right) \quad (29)$$

ZieglerNichols method is used to obtain the K_p, T_i , and T_d gains in relation (29). For this purpose, the system should be linearized first. By a frequency analysis then, the PID controller gains are obtained from Eq. (30) [14].

$$\begin{aligned} K_p &= 0.6\, K_{cr} \\ T_d &= 0.125\, P_{cr} \\ T_i &= 0.5\, P_{cr} \end{aligned} \quad (30)$$

The values of K_{cr} and P_{cr} are calculated from Eq. (31).

$$K_{cr} = GmP_{cr} = \frac{2\pi}{w_{cg}} \quad (31)$$

In relation (31), the parameter Gm is the phase margin and w_{cg} is the frequency where the phase margin is measured and the system phase will be -180.

3-2 LQR controller design

The 2nd-order linear regulator or LQR controller attempts to minimize the objective function (32).

$$J = \int_0^{t_f} \frac{1}{2}[x^T Q(t)x(t) + u^T R(t)u(t)]dt \quad (32)$$

In Eq. (32), the parameter t_f denotes the final time. Also, Q and R are two weight matrices that are obtained through optimization. The equation for the control input associated with this objective function is obtained as given in Eq. (33).

$$u(t) = -R^{-1}B^T K(t)x(t) \quad (33)$$

where B is the matrix multiplier for the control input in the linearized system, and $K(t)$ is obtained from Eq. (34) below.

$$0 = \dot{K}(t) + Q - K(t)BR^{-1}B^T K(t) + K(t)A + A^T K(t) \quad (34)$$

where A is the matrix multiplier for the state vector in the linearized system.

3-3 Design of predictive model controller

In most control projects where the use of predictive model control is considered a necessity, the linear type predictive model controller option is preferred. In particular, in industrial projects, due to the need for high speed and reliability along with cost considerations, this type of controller has been almost the only option for designers [15]. In general, the pre-linear model controller can be implemented in two ways: online (implicit) and offline (explicit). In the implementation of the online type, the optimization problem of the control cost function is solved online in each time step, and the answer to this problem will determine the optimal control order. But in the offline method, the optimization problem is solved once for all possible values of the state vector and the optimal answer is calculated as an explicit function of the variables of the state vector and loaded into the controller memory. At each time step then, the controller will determine the value of this function based on the values of the state variables to issue the control commands.
As such, the time required to perform control calculations is reduced, hence making it possible to implement the devised controller on the control hardware with limited processing volume.

3-3-1 Designing an explicit linear predictive model controller

In the design of the predictive model controller, the calculation of the optimal control command vector requires solving a quadratic optimization problem at each time step. Although efficient quadratic programming solvers based on active set and interior point methods are available today, it is not possible to use these solvers in control hardware with limited processing capacity.
To reduce the need for online processing power to increase the speed of calculating the control command in the constrained linear predictive model controller used for real applications, the explicit predictive model controller is usually used. This controller has all the stability and performance features of the equivalent online predictive

model controller, and only the need for calculations related to solving the online optimization problem has been eliminated. For this purpose, multi-parameter programming is used to solve the optimization problem of the predictive model controller.

Parametric programming is a term used to solve the optimization problem for a range of values of a parameter. The difference between parametric programming and multi-parameter programming is that in the latter we deal with a vector of parameters. We show that in this method, the control law can be calculated offline in the form of a continuous piecewise affine function of the state variables and the reference signal vector such as $a_h(k) = F_i x + g_i$. This control law can be set in the form of a reference table and loaded in the memory of the control hardware, thus reducing the online processing effort to finding the i related to the state vectors and the reference signal.

The main idea of the multi-parameter algorithm is to use the necessary and sufficient conditions of optimality to construct the critical region in the neighborhood of a given parameter and search the space outside this region in a recursive method.

The predictive model controller is usually implemented as discrete time in control systems. Therefore, we convert the state equations into the discrete time state space equation form using the zeroth order retainer and the sampling time T_s.

$$x(k+1) = A_d x(k) + B_d u(k) \tag{35}$$

Therefore, the output equations of the discrete-time system will be in the following form.

$$y(k) = C_d x(k) \tag{36}$$

Relations (35) and (36) express the model based on which the predictive model controller is developed. Therefore, this model is called control-oriented model. With the aim of including the defined constraint on the rate of acceleration changes in the control equations and also creating the integration behavior in the controller to reduce the steady state error, the control input by relating its increase value at each sampling time as $u(k) = u(k-1) + \Delta u(k)$ in the form of new state variables, we add to the equations of the previous control-oriented model[30] .

$$\begin{aligned} x_a(k+1) &= A_a x_a(k) + B_a \Delta u(k) \\ \begin{bmatrix} x(k+1) \\ u(k) \end{bmatrix} &= \begin{bmatrix} A_d & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} \\ &\quad + \begin{bmatrix} B_d \\ I \end{bmatrix} \Delta u(k) \end{aligned} \tag{37}$$

This new control-oriented model is called the augmented model. The output equations of the augmented model can be expressed as follows.

$$\begin{aligned} y(k) &= C_a x_a(k) \\ y(k) &= [C_d \quad 0] \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} \end{aligned} \tag{38}$$

By defining N_p and N_c as prediction and control horizons, the input and output equations of the prediction model can be calculated using the following equation.

$$Y = F \begin{bmatrix} x(k) \\ u(k-1) \end{bmatrix} + \phi \Delta U \tag{39}$$

In the equation of the above forecasting model, Y vector can be taken from equation (40),

$$Y = \begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ \vdots \\ y(k+N_p|k) \end{bmatrix} \tag{40}$$

ΔU vector from equation (41),

$$\begin{aligned} \Delta U &= \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix} \\ &= \begin{bmatrix} \Delta a_h(k|k) \\ \Delta a_h(k+1|k) \\ \vdots \\ \Delta a_h(k+N_c-1|k) \end{bmatrix} \end{aligned} \tag{41}$$

matrix A from equation (42),

$$F = \begin{bmatrix} C_a A_a \\ C_a A_a^2 \\ \vdots \\ C_a A_a^{N_p} \end{bmatrix} \tag{42}$$

and finally obtained the matrix ϕ from equation (43).

$$\phi = \begin{bmatrix} C_a B_a & \dots & 0 \\ C_a A_a B_a & \dots & 0 \\ \vdots & \ddots & \vdots \\ C_a A_a^{N_p} B_a & \dots & C_a A_a^{N_p - N_c} B_a \end{bmatrix} \tag{43}$$

3-3-2 Cost function

The basis of the predictive model control method is associated with the use of a function of inputs and outputs (or state variables) known as the performance function, which should be optimized in the prediction horizon while satisfying the constraints in the problem. The state variables in the future time are calculated using the predictive model and the system conditions in the present time $(x_a(k|k) = x_a(k))$as the initial conditions of the relationship (37). Solving this optimization problem gives the sequence of control inputs obtained as $\Delta U = [\Delta a_h(k|k), \dots, \Delta a_h(k + N_{c-1}|k)]^T$. Based on the descending horizon control rule then, the first input of this sequence is applied to the control system and the same process is repeated again in the next time steps.

The cost function in predictive model control is commonly

chosen as a linear or quadratic function. Although solving the first-order equation requires less computational volume than solving the second-order equation, the existence of the overall minimum value is not guaranteed in this method. Therefore, the cost function is chosen in the form of a quadratic function so that due to the convexity of this function, the existence of a local minimum is equivalent to the existence of a global minimum (a necessary and sufficient condition for optimality). The cost function for the pursuit problem is defined as follows.

$$J = \sum_{i=1}^{N_p} [y(k+i|k) - r(k+i|k)]^T Q [y(k+i|k) - r(k+i|k)] + \sum_{j=0}^{N_c-1} \Delta u^T(k+j|k) R \Delta u(k+j|k) \quad (44)$$

In the cost function of the investigated problem, Q is a positive definite symmetric matrix and R is a positive scalar.Also,
$r(k+i|k)[x_{rd}(k+i|k) \quad v_{rd}(k+i|k) \quad v_{hd}(k+i|k)]^T$
is the reference vector during the prediction window and expresses the controller's expectation of the behavior of the reference vector in future times. The values of the elements of this vector during the prediction window are usually considered constant and equal to the values of the reference signal at the sampling moment k. It is possible at times to predict the changes of the controller's reference signals in advance, a capability known as "previewing" the reference signal in the control of the predictive model. In reality, previewing the reference signal actually means a more accurate definition of the control expectations in the future, significantly improving the system performance, particularly in systems with fast dynamics.
The cost function in matrix form can be expressed as follows.

$$\begin{aligned} J &= \underbrace{\begin{bmatrix} y(k+1|k) - r(k+1|k) \\ y(k+2|k) - r(k+2|k) \\ \vdots \\ y(k+N_p|k) - r(k+N_p|k) \end{bmatrix}^T}_{Y^T - Ref^T} \underbrace{\begin{bmatrix} Q & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & Q \end{bmatrix}}_{\hat{Q}} \underbrace{\begin{bmatrix} y(k+1|k) - r(k+1 \\ y(k+2|k) - r(k+2 \\ \vdots \\ y(k+N_p|k) - r(k+N \end{bmatrix}}_{Y - Ref} \\ &+ \underbrace{\begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix}^T}_{\Delta U^T} \underbrace{\begin{bmatrix} R & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & R \end{bmatrix}}_{\hat{R}} \underbrace{\begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \vdots \\ \Delta u(k+N_c-1|k) \end{bmatrix}}_{\Delta U} \\ &= (Y - Ref)^T \hat{Q} (Y - Ref) + \Delta U^T \hat{R} \Delta U \end{aligned} \quad (45)$$

$$\begin{aligned} J(\Delta U, x_a, Ref) &= \Delta U^T [\hat{R} + \phi^T \hat{Q} \phi] \Delta U + 2 \left([F \quad -I] \begin{bmatrix} x_a \\ Ref \end{bmatrix} \right)^T \hat{Q} \phi \, \Delta U \\ &+ Ref^T \hat{Q} Ref + x_a^T F^T \hat{Q} (F x_a - 2 Ref) \\ &= \frac{1}{2} \Delta U^T \underbrace{2[\hat{R} + \phi^T \hat{Q} \phi]}_{\hat{H}} \Delta U \\ &+ \underbrace{\begin{bmatrix} x_a \\ Ref \end{bmatrix}^T}_{x^T} \underbrace{\begin{bmatrix} F^T \\ -I \end{bmatrix} 2 \, \hat{Q} \phi}_{\hat{\mathcal{F}}} \Delta U \\ &+ \underbrace{Ref^T \hat{Q} Ref + x_a^T F^T \hat{Q} (F x_a - 2 \, Ref)}_{\hat{N}} \end{aligned} \quad (46)$$

By definition $x = [x_a^T(k) \quad Ref^T]^T$ ، $\hat{H} = 2[\hat{R} + \phi^T \hat{Q} \phi]$ ، $\mathcal{F} = 2 \, [F^T \quad -I]^T \hat{Q} \phi$ ، $\hat{N} = Ref^T \hat{Q} Ref + x_a^T F^T \hat{Q} (F x_a - 2 \, Ref)$ The cost function can be written in a simpler way.

$$J(\Delta U, x) = \frac{1}{2} \Delta U^T \hat{H} \, \Delta U + x^T \mathcal{F} \, \Delta U + \hat{N} \quad (47)$$

Note that the optimization variable, similar to the real-time linear predictive model controller, is the vector of changes in the control input (changes in the angle of the missile actuator) in the control horizon and the vector x (including the vector of state variables at the current moment, the control input in the previous time step, and the reference signal during prediction horizon) are simply the vectors of optimization parameters. To make calculations easier in the next steps, first we express the above quadratic matrix function in the complete square form as [16],

$$J = \frac{1}{2} [\Delta U + \hat{H}^{-1} \mathcal{F}^T x]^T \hat{H} [\Delta U + \hat{H}^{-1} \mathcal{F}^T x] - \frac{1}{2} x^T \mathcal{F} \, \hat{H}^{-1} \mathcal{F}^T x + \hat{N} \quad (48)$$

Considering that the only optimization variable in this problem is the ΔU vector and in the problem considered here we are only looking to find the minimizer function in terms of the problem parameters (the vector of generalized state variables and the reference signal during the forecast horizon), the terms in which the ΔU vector is absent have no effect on the calculation of the gradient of the cost function and, therefore, these terms can be omitted in the definition of the cost function. Finally, by a change in the variable $z = \Delta U + \hat{H}^{-1} \mathcal{F}^T x$, the cost function can be rewritten in the following simple form.

$$\bar{J} = \frac{1}{2} z^T \hat{H} z \quad (49)$$

It is obvious that this function is a definite biconvex function due to the positive definite $\hat{H}$ matrix.

3-3-3 Constraints of the explicit predictive model controller optimization problem

Naturally, the control constraints of the problem will be similar to that as the previous section. To solve the problem

in the multi-parameter optimization framework, we write the constraints of problem (50) as below.

$$\underbrace{\begin{bmatrix} G_U \\ G_{\Delta U} \\ G_Y \end{bmatrix}}_{G} \Delta U \leq \underbrace{\begin{bmatrix} 0_{N_c\times 3} & G_1 & 0_{N_c\times 3N_p} \\ 0_{N_c\times 3} & -G_1 & 0_{N_c\times 3N_p} \\ 0_{N_c\times 3} & 0_{N_c\times 1} & 0_{N_c\times 3N_p} \\ 0_{N_c\times 3} & 0_{N_c\times 1} & 0_{N_c\times 3N_p} \\ F & & 0_{3N_p\times 3N_p} \\ -F & & 0_{3N_p\times 3N_p} \end{bmatrix}}_{E} \underbrace{\begin{bmatrix} x(k) \\ u(k-1) \\ Ref \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} -U^{min} \\ U^{max} \\ -\Delta U^{min} \\ \Delta U^{max} \\ -Y^{min} \\ Y^{max} \end{bmatrix}}_{\mathcal{W}} \tag{50}$$

$$G\Delta U \leq Ex + \mathcal{W}$$

Finally, to solve the optimization problem by the quadratic multiparameter method, we map the constraints of the problem to the z variable using the change of variable $\Delta U = z - \hat{H}^{-1}\mathcal{F}^T x$.

$$G\left(z - \hat{H}^{-1}\mathcal{F}^T x\right) \leq Ex + \mathcal{W}$$

$$Gz \leq \underbrace{\left(E + G\hat{H}^{-1}\mathcal{F}^T\right)}_{S} x + \mathcal{W} \tag{51}$$

By defining $S = E + G\hat{H}^{-1}\mathcal{F}^T$, the constraints of the quadratic multiparameter problem can be rewritten in the following simple form.

$$Gz \leq Sx + \mathcal{W} \tag{52}$$

Therefore, the quadratic multi-parameter problem related to an explicit predictive model control scheme for missile guidance with the ability to predict or preview the reference signal is expressed as follows.

$$\bar{J}^*(x) = \min_z \frac{1}{2} z^T \hat{H} z$$
$$s.t. \quad Gz \leq Sx + \mathcal{W} \tag{53}$$

The optimization variable of this problem is the vector $z \in \mathbb{R}^{N_c}$, which is the result of the linear mapping of the vector of control input changes. Therefore, the number of design variables of the problem will be N_c. The parameters of the optimization problem also include the state vector at the current moment, the control input at the previous time step, and the reference signal during the forecast horizon. Therefore, the number of problem parameters is equal to $8 + N_p$. It should be noted that unlike online optimization, the solution to this problem is not an optimal vector, but rather it is the optimizer function $z^*\big(x(k), u(k-1), r(k+1|k), \dots, r(k+N_p|k)\big): \mathbb{R}^{8+N_p} \to \mathbb{R}^{N_c}$, and the optimal value of the objective function is not a scalar, but rather it is the function $\bar{J}^*\big(x(k), u(k-1), r(k+1|k), \dots, r(k+N_p|k)\big): \mathbb{R}^{8+N_p} \to \mathbb{R}$. Obviously, to calculate the vector of optimal changes of control commands during the control horizon, the variable change $\Delta U^* = z^* - \hat{H}^{-1}\mathcal{F}^T x$ can be used. Therefore, the optimal control command in this case will be equal to:

$$u^*(k) = \Delta u^*(k|k) + u(k-1) = [1 \quad 0 \quad \cdots \quad 0]\left[z^* - \hat{H}^{-1}\mathcal{F}^T x\right] + u(k-1) \tag{54}$$

Additionally, if we are interested in calculating the optimal cost function of the main problem, we can calculate it as follows.

$$\bar{J}^* = -\frac{1}{2} x^T \mathcal{F} \hat{H}^{-1} \mathcal{F}^T x + \hat{N} \tag{55}$$

3-3-4 Solution of the multi-parameter optimization problem

In the optimization problem (53) where $\hat{H} = \hat{H}^T \succ 0$, the goal is to find the optimal cost function $\bar{J}^*(x)$ and the optimizer $z^*(x)$ in terms of the parameters of the problem. In this section, we show that the optimizer is a piecewise affine vector function of the problem parameters in the form $z^* = F_i x + g_i$, and therefore having the state vector values at each moment along with the values of the control command of the previous time step and the reference vector in the horizon prediction, the optimal acceleration value of the host vehicle can be easily calculated with the help of equation (55).

Using the constraints of the boundaries of the problem space, we define the multifaceted closed bounded polynomial set K of parameters of the optimization problem as follows.

$$K = \{x \in \mathbb{R}^{8+N_p} | Tx \leq N\} \tag{56}$$

Matrix T and vector N in the above relationship are used to determine the upper and lower limits of state variables and control commands at the present time along with the bounds on the reference signal vector in the prediction horizon. It is clear that the polyhedron $K \subset \mathbb{R}^{8+N_p}$ is of full dimensions since equality constraints were not used in its definition. We define the set of feasible parameters $K^* \subseteq K$ as a region of parameters in which the optimization of the problem is possible.

$$K^* = \{x \in K | \exists z \in \mathbb{R}^{N_c}: Gz \leq Sx + \mathcal{W}\} \tag{57}$$

In the quadratic multiparameter problem stated in (61), let $I = \{1, \dots, 6N_p + 4N_c\}$ be the set of constraint indices. the optimal partition of the set of constraint indices I for a

hypothetical vector such as $x \in K^*$ is the following distribution.

$$\begin{aligned} &A(\bar{x}) \\ &= \{j \in I | \forall z^*(x) \in Z^*(x): G_j z^*(x) - S_j x = \mathcal{W}_j\} \\ &NA(\bar{x}) \\ &= \{j \in I | \exists z^*(x) \in Z^*(x): G_j z^*(x) - S_j x < \mathcal{W}_j\} \end{aligned} \quad (58)$$

In terminology, A(x) and NA(x) are called active and passive constraints. Obviously, $A(x) \cup NA(x) = I$ and $A(x) \cap NA(x) = \emptyset$. In addition, the critical region associated with the set of active constraints $A \subset I$ is defined as $CR_A = \{x \in |A(x) = A\}$.

First, we show that the critical regions of quadratic multiparameter optimization are multimodal. We use the KKT conditions to obtain the H representation of the polyhedra of the critical regions, and calculate the optimizer function $z^*(x)$and the value of the optimal function $\bar{J}^*(x)$ within each critical region. For an $\bar{x} \in K^*$whose active and passive constraints are related to it in the form $(A, NA) = (A(\bar{x}), NA(\bar{x}))$it can be shown that:

1. CR_A is a polyhedron.
2. $z^*(\bar{x})$ is an affine function of the state variables in the critical region of CR_A. In other words, for all $\bar{x} \in CR_A$we can write $z^*(\bar{x}) = F_i \bar{x} + g_i$.
3. $\bar{J}^*$ is a quadratic function of the state variables inside the region CR_A. In other words, for each $\bar{x} \in CR_A$, we have $\bar{J}^*(\bar{x}) = x^T M_i \bar{x} + c_i^T \bar{x} + d_i$.

For this purpose, we use the first-order KKT conditions for problem (53).

$$\begin{aligned} &\nabla_z \square = \hat{\mathrm{H}} z^* + G^T \lambda^* = 0\,, \qquad \lambda \in \mathbb{R}^{(4N_c+6N_p)} \\ &\lambda_i^*(G_i z^* - \mathcal{W}_i - S_i x) = 0\,, \qquad i = 1, \dots, 4N_c + 6N_p \\ &\lambda_i^* \geq 0 \\ &Gz^* - Sx - \mathcal{W} \leq 0 \end{aligned} \quad (59)$$

We know that $\hat{H} \succ 0$ and therefore it is invertible. Therefore, we obtain $z^*$from the necessary condition for optimality.

$$z^* = -\hat{H}^{-1} G^T \lambda^* \quad (60)$$

And we put the result in the redundant condition of the complement.

$$\lambda_i^*\left(-G_i \hat{H}^{-1} G^T \lambda^* - S_i x - \mathcal{W}_i\right) = 0 \quad (61)$$

We introduce λ_{NA}^* and λ_A^* as Lagrange coefficients associated with the passive and active constraints, respectively. Since $\lambda_{NA}^* = 0$ for passive constraints, the following relationship holds for active constraints.

$$-G_A \hat{H}^{-1} G_A^T \lambda_A^* - S_A\, x - \mathcal{W}_A = 0 \quad (62)$$

Assume that the rows of $G_A$are linearly independent. In this case, λ_A^*can be obtained from the above relationship and then λ^*can be calculated.

$$\lambda_A^* = -\left(G_A \hat{H}^{-1} G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) \quad (63)$$

It is clear that λ_A^* is an affine function of the problem parameters. On the other hand, we know that $\lambda_{NA}^* = 0$, and therefore λ^* is an affine function of the vector of parameters. Now, by placing the value of λ^*in equation (60), z^*can be calculated according to the parameters of the problem.

$$z^* = \hat{H}^{-1} G_A^T \left(G_A \hat{H}^{-1} G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) \quad (64)$$

This relationship shows that the optimizer is also an affine function of the parameters, and as a result $J^*(x) = \frac{1}{2} z^{*T}(x) H z^*(x)$ is a quadratic function of the vector of parameters. The value of z^*obtained from equation (64) must satisfy the initial feasibility condition. Therefore, we can write:

$$G\hat{H}^{-1} G_A^T \left(G_A \hat{H}^{-1} G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) < Sx + \mathcal{W} \quad (65)$$

In this way, a subspace of the parameters space is determined.

$$\begin{aligned} &P_{primal} \\ &= \left\{x \in K^* \middle| G\hat{H}^{-1}\, G_A^T \left(G_A \hat{H}^{-1} G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) < Sx\right\} \end{aligned} \quad (66)$$

On the other hand, the vector of Lagrange coefficients calculated in (63) should satisfy the dual feasibility condition. Hence we have:

$$-\left(G_A \hat{H}^{-1} G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) \geq 0 \quad (67)$$

This relation specifies the subspace in which the dual feasibility condition holds.

$$P_{dual} = \left\{x \in K^* \middle| -\left(G_A \hat{H}^{-1}\, G_A^T\right)^{-1}(S_A\, x + \mathcal{W}_A) \geq 0\right\} \quad (68)$$

By removing the redundant constraints from relations (65) and (67), one can express the compact form of representation H related to CR_A.

$$CR_A = P_{primal} \cap P_{dual} \quad (69)$$

Therefore, CR_A is a polyhedron in x-space. Since P_{primal} is an open and non-zero set (it contains at least the point $\bar{x}$), this set is full-dimensional in x space and can be written as $dim(CR_A) = di\,m(P_{primal})$. In addition, in the case that the index set of active constraints, i.e. set A, is null, $\lambda^* = \lambda_{NA}^* = 0$ and $z^* = 0$, which means that the corresponding critical area is $CR_A = \{x | 0 < Sx + \mathcal{W}\}$.

To this point, it has been assumed that the rows of the

matrix G_A are linearly independent and the criterion of linearly independent constraints has been established. Therefore, the matrix $G_A\widehat{H}^{-1}G_A$ is invertible. If the rows of the matrix G_A do not have linear independence, for example, when more than N_c constraints are active in the minimizer $z^* \in \mathbb{R}^{N_c}$ which is called the primal degeneracy state, in this case it is possible the vector of Lagrange coefficients λ^*may not be uniquely defined because the dual problem in this case is not strictly convex. Note that the double degeneracy (less than N_c of the constraint to be active in the minimizer $z^* \in \mathbb{R}^{N_c}$) and the non-uniqueness of the minimizer $z^*(x)$ are not possible in quadratic multi-parameter problems because the matrix $\widehat{H}$ is positive definite.

In the case of primal degeneracy, suppose that relation (69) holds for two arbitrary vectors $\bar{x} \in CR_A$ and $\bar{\bar{x}} \in CR_A$, and that it can be written $-G_A\widehat{H}^{-1}\,G_A^T\bar{\lambda}_A^* - S_A\bar{x} - \mathcal{W}_A = 0$ and $-G_A\widehat{H}^{-1}G_A^T\bar{\bar{\lambda}}_A^* - S_A\bar{\bar{x}} - \mathcal{W}_A = 0$. Therefore, for the disturbance $\tilde{x} = \bar{\bar{x}} - \bar{x}$, it can be stated that $G_A\widehat{H}^{-1}\,G_A^T\tilde{\lambda}_A^* = -S_A\tilde{x}$. Obviously, this relation can be solved when S_A is in the column space of the matrix $G_A\widehat{H}^{-1}G_A^T$. Therefore, as long as the relation $rank\left[\left(G_A\widehat{H}^{-1}G_A^T\right) : S_A\right] = rank[G_A\widehat{H}^{-1}G_A^T]$ holds, the critical region will be of full dimensions. In fact, the dimensions of the critical area can be obtained from the following relationship.

$$\begin{aligned}\dim(CR_A) = 4 + 2\,N_p \\ - \left(rank\left[\left(G_A\widehat{H}^{-1}G_A^T\right) : S_A\right]\right. \\ \left. - rank[G_A\widehat{H}^{-1}G_A^T]\right)\end{aligned} \quad (70)$$

To obtain the critical region in the case that the rows of the matrix G_A are not linearly independent, suppose we have $l = rank(G_A)$. Then we calculate the QR decomposition of the matrix $-G_A\widehat{H}^{-1}G_A^T$ as follows.

$$-G_A\widehat{H}^{-1}G_A^T = QR = Q\begin{bmatrix}u_1 & u_2\\ 0 & 0\end{bmatrix} \quad (71)$$

In the above relationship, $Q \in \mathbb{R}^{|A|\times|A|}$ is the orthogonal matrix, $R \in \mathbb{R}^{|A|\times|A|}$ is the upper triangular matrix, $u_1 \in \mathbb{R}^{l\times l}$ is the full rank square matrix, and $u_2 \in \mathbb{R}^{l\times(|A|-l)}$. By replacing the QR decomposition of the matrix $-G_A\widehat{H}^{-1}G_A^T$ in equation (62), the following equation is obtained.

$$[R \quad -Q^{-1}S_A]\begin{bmatrix}\lambda^*\\ x\end{bmatrix} = Q^{-1}\,\mathcal{W}_A \quad (72)$$

By defining the new matrices $[P^T \quad D^T]^T = -Q^{-1}S_A$ and $[q^T \quad r^T]^T = -Q^{-1}\mathcal{W}_A$, the following relationship results.

$$\begin{bmatrix}u_1 & u_2 & P\\ 0 & 0 & D\end{bmatrix}\begin{bmatrix}\lambda_{A,1}^*\\ \lambda_{A,2}^*\\ x\end{bmatrix} = \begin{bmatrix}q\\ r\end{bmatrix} \quad (73)$$

By block multiplication of matrices, we relation results.

$$u_1\lambda_{A,1}^* + u_2\lambda_{A,2}^* + Px = q$$

Hence,

$$\lambda_{A,1}^* = u_1^{-1}\left(-u_2\lambda_{A,2}^* - Px + q\right) \quad (74)$$

Now, using the value obtained for ${\lambda_{A,1}}^*$ and consequently λ_A^*, followed by placing λ_A^* in relation (70), we can calculate the z^*function as,

$$z^* = -\widehat{H}^{-1}G_A^T\lambda^* = \widehat{H}^{-1}G_A^T\begin{bmatrix}u_1^{-1}\left(u_2\lambda_{A,2}^* + Px - q\right)\\ -\lambda_{A,2}^*\end{bmatrix} \quad (75)$$

Since the minimizer is unique in this case, its value should be independent of the choice of $\lambda_{A,2}^*$. Therefore, $\lambda_{A,2}^* = 0$ is considered. By dividing the matrix G_A in the form $[G_{A,1}^T \quad G_{A,2}^T]^T$, where $G_{A,1} \in \mathbb{R}^{l\times N_c}$, we reach at the following equation.

$$z^* = \widehat{H}^{-1}G_{A,1}^T u_1^{-1}(Px - q) \quad (76)$$

The last relation shows that, in the case where the rows of the G_A matrix are not linearly independent, the optimizer is an affine function of the optimization parameters. As such, one can conclude that in this case as well the optimal cost function $\bar{J}^*(x) = \frac{1}{2}z^{*T}(x)\widehat{H}z^*(x)$ is a quadratic function of the parameters.

To calculate the critical region, similar to what was described in the case of linear independence of the rows of the matrix G_A, we use the replacement of z^* in the initial feasibility condition.

$$\begin{aligned}&P_{primal}\\ &= \left\{x \in K^* \middle| G\widehat{H}^{-1}G_{A,1}^T u_1^{-1}(Px - q) < Sx + \mathcal{W}\right\}\end{aligned} \quad (77)$$

On the other hand, the value of $\lambda_{A,1}^*$ obtained from equation (73) must satisfy the dual feasibility condition.

$$\begin{aligned}&P_{\lambda_{A,2}^*}\\ &= \left\{x \in K^*\ , \lambda_{A,2}^* \in \mathbb{R}^{|A|-l} \middle| u_1^{-1}\left(-u_2\lambda_{A,2}^* - Px + q\right) \geq 0\right\}\end{aligned} \quad (78)$$

And finally, the critical area is obtained by sharing the image $P_{\lambda_{A,2}^*}$ in the vector space x with P_{primal} and the subspace determined through the device of equations $Dx = r$.

$$\begin{aligned}&CR_A\\ &= \left\{x \in K^* \middle| Dx = r, x \in P_{primal}, x \in proj_x(P_{\lambda_{A,2}^*})\right\}\end{aligned} \quad (79)$$

It is obvious that the subspace obtained from this sharing is a polyhedron.

Now we have to calculate the optimizer function and the optimal cost function for the rest of the problem parameter space. For this purpose, the subspace $K^{rest} = K\backslash CR_A$ should be searched to find new critical regions.

if $X \subseteq \mathbb{R}^n$ is a polyhedron and $Cr_0 = \{x \in X | Ax \leq b\}$ is a non-empty subset of X and if we also have $R_i =$

$\{x \in X | \forall j < i, i = 1, \dots, m : A_i x > b_i, A_j x \leq b_j\}$ where $m = \dim(b)$and $R^{rest} = \cup_{i=1}^{m} R_i$ holds, then in this case, first $R^{rest} \cup CR_0 = X$, secondly, $CR_0 \cap R_i = \Phi$, and for each i≠j we have $CR_0 \cap R_i = \Phi$. In other words$\{CR_0, R_1, \dots, R_m\}$ is a subset of the set X. In this analysis, only one of the constraints of the problem is evaluated during each step.

The above theorem also provides a solution for dividing non-convex sets $X \backslash CR_0$ into polyhedral subsets R_i. For each R_i a new vector x_i is determined by calculating the center of the Chebyshev sphere.

$$\begin{aligned} &\max_{x,z,\varepsilon} \varepsilon \\ &s.t. \quad T_i x + \varepsilon \|T_i\| \leq N_i \\ &Gz \leq Sx + \mathcal{W} \end{aligned} \tag{80}$$

The value of the optimizer z_i^* and those of the active constraints $G_{A(x_i)}$, $S_{A(x_i)}$, $W_{A(x_i)}$, and finally CR_i are then determined. This theorem is used to divide $R_i \backslash CR_i$ into multifaceted subsets by iterative algorithm method.

Note that the last theorem may cause critical regions to be divided into several subsets. Thus, after the entire parameter space is covered, these polymorphic regions are detected. Since these areas have the same optimizer function, if the union of these critical areas form a convex set, the more compact form of the partition resulting from the combination of these two areas is presented [16]. The recursive algorithm of problem space extraction (53) is summarized as follows [16].

Algorithm 1- Algorithm of allocation of space to critical areas

1. Selection of the initial vector x_0 within the multifaceted set K by calculating the center of the Chebyshev sphere using equation (80)
2. Checking the value of ε
   2-1. If $\varepsilon \leq 0$, then solution of problem (53) is impossible for all x in K and this loop iteration stops.
   2-2. Otherwise, register the vector $x = x_0$
3. Determining active constraints for $z^*_{(x_0)}$through equation (58)
4. Using relations (60) to (63) and determining $G_{A(x_0)}$, $S_{A(x_0)}$, and $W_{A(x_0)}$
5. Checking the linear independence of $G_{A(x_0)}$
   5-1. Calculation of $\lambda^*_{(x_0)}$from equation (64) and $z^*_{(x_0)}$ from (65) in the absence of initial degeneracy
   5-2. Applying relations (66) to (67) to calculate the critical region $CR_{A(x_0)}$in the absence of degeneracy
   5-3. Obtaining $\lambda^*_{A(x_0),1}$ from the equation (75) and $z^*_{(x_0)}$from (77) in the case of initial degeneracy
   5-4. Using relations (78) to (80) to calculate the critical region $CR_{A(x_0)}$ in case of degeneracy
6. The partition of the space $K^{rest} = K \backslash CR_A$ through the one-to-one direction change of the constraints that played a role in the definition of $CR_{A(x_0)}$
7. Repeat the loop for all newly created areas

It was stated before that the main drawback of this algorithm is that sometimes a critical region with the same solution is divided into more than one polygon by the algorithm. Figure 2 shows the process of generating the space of a symbolic two-parameter problem by the mentioned algorithm.

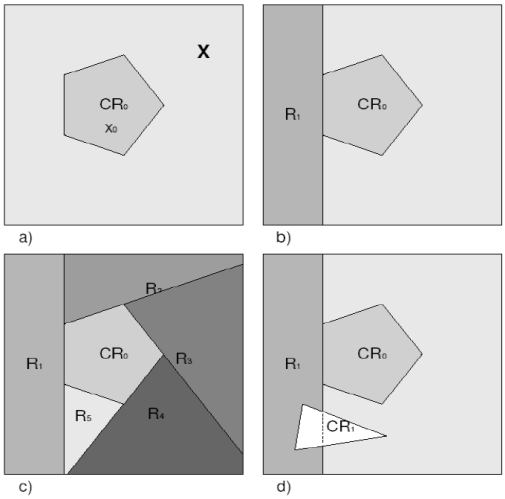

Fig.2 - The steps of partitioning the parameter space (two-dimensional) using the algorithm [15]: a) forming CR_0 by calculating the center of the Chebyshev sphere, b) obtaining the subspace R_1 by inverting the constraint number 1, c) extracting the entire parameter space, d) The major problems of the algorithm in dividing the critical region CR_1 into two different multipliers and the need to combine these two regions in the next step.

As it is well shown in this figure, after the completion of the algorithm implementation of the parameter space to the critical areas, it is necessary to examine the created polygons one by one, so that if the necessary conditions are met, these polygons one can be merged together. Usually, this process allocates the most time to solve quadratic multi-parameter optimization.

3-3-5 Controller stability analysis and solution feasibility condition

In general, the stability of the predictive model controller is

not guaranteed in advance. Furthermore, the controller may direct the state variables to a part of the state space where no optimal response satisfying the constraints of the problem can be computed in finite time. Therefore, the controller of the predictive model can be implemented when the stability and the condition of the existence of the solution in the entire problem space are examined.

The stability of the predictive model controller feedback loop has been investigated by several researchers. Most of the approaches to prove the stability of the predictive model control are essentially dependent on the arguments of Kirtshi and Gilbert, which show that under some conditions, the optimal objective function is actually a Lyapunov function. In these types of controllers, stability is generally a complex function of various adjustable parameters such as N_p ،N_c ،P ، , and R. If a short control horizon is selected, the controller can easily become unstable. To avoid this situation, the prediction horizon can be considered very large (and ideally infinite). It is clear that such a choice will lead to growing increase in the processing volume of the controller.

Another way to ensure the stability of the controller for an arbitrary prediction horizon is to apply the Terminal constraint on the last state vector in the prediction horizon . In this way, we will ensure that the state vector will converge to a certain vector at the end of the prediction horizon. The drawback of the mentioned method is that the equality condition of the Terminal constraint may cause inefficiency of the controller operation [17]. In addition, in order for such an approach to be possible, the open loop system must be achievable in addition to sustainability [18]. As it was shown earlier, the system studied in this research is not fully controllable and therefore all the state space vectors are not accessible in this problem and this method cannot be used to ensure the stability of the controller.

However, it has been shown that the stability of the controller with a limited prediction horizon is also possible in the absence of the. Specifically, it is proved in that a closed-loop control system with predictive model controller is asymptotically globally stable if and only if the associated optimization problem is feasible. Therefore, by showing that the constrained optimization problem is possible in any situation, we can implicitly prove the stability of the controller.

4- Simulations and results

After completing the design of the controllers used, the performance of the designed controllers is investigated in this section. The parameters of the missile and the values of the initial parameters of the missile and target engagement, and the initial conditions in all these simulations are the same and according to Table 1.

Table 1 - Parameters of the homing missile [13]

variable	value	Variable	value
S	$0.42\ m^2$	J_x	100 kg. m^2
L	0.68 m	J_y	5700 kg. m^2
m	1200 kg	J_z	5600 kg. m^2
ρ	1.1558 kg/m^3	m_y^{β}	-27.31
m_z^{α}	-28.16	$m_y^{\delta_y}$	-26.57
$m_z^{\delta_z}$	-27.92	m_x^{α}	0.46
c_y^{α}	57.16	m_x^{β}	-0.37
c_y^{β}	0.08	$m_x^{\delta_x}$	2.12
$c_y^{\delta_z}$	5.74	c_z^{α}	-56.31
c_z^{β}	-5.62	$c_z^{\delta_y}$	0.09

Table 2 - Initial parameters of missile and target engagement [13]

variable	value	Variable	value
$\theta(0)$	45 (deg)	V_m	600 m/s
$\Phi_c(0)$	0 rad	V_t	600 m/s
ω_x	0.1 rad/s	$x_t(0)$	1136 m
ω_y	0.1 rad/s	$y_t(0)$	8603 m
ω_z	0.2 rad/s	$z_t(0)$	5192.8 m
$x_m(0)$	0 m	θ_{Lf}	30 (deg)
$y_m(0)$	0 m	ϕ_{Lf}	-30 (deg)
$z_m(0)$	0 m	a_T	19.6*cos(t)

4-1 Linear equations of missile movement and interception in three-dimensional space

For a nonlinear system, consider the general equation (81):

$$\dot{x} = f(x,u) \tag{81}$$

To obtain the linear system $\dot{x} = Ax + Bu$ corresponding to the nonlinear system (81), we use relations (82):

$$A = \frac{\partial f(x,u)}{\partial x}|x = x_0,\ u = u_0$$
$$B = \frac{\partial f(x,u)}{\partial u}|x = x_0,\ u = u_0 \tag{82}$$

In relation (76), x_0 and u_0 are the operating points of linearization. According to this relationship, to find the two matrices $A$and$B$, we must find the partial derivatives of f with respect to $x$and$u$ at the operating point (i.e., the Jacobian matrix), respectively.

We now define a new linear system as Eq. (83):

$$\dot{x} = Ax + Bu \tag{83}$$

In Eq. (83), the state and control vectors are defined as $x = \left[x_0^T, x_1^T, x_2^T, x_3^T, \theta, \phi_v, R, \dot{R}\right]^T$ and $u = \left[\delta_x, \delta_y, \delta_z\right]^T$, respectively. In fact, we define a linear system in such a way that it includes all the state variables defined in the previous sections. According to the above equations, we will have the numerical values given in Table 2, and finally using the equation (82), we arrive at the following matrices for A and B.

$$A = \begin{bmatrix}
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-6.67 & 7.23 & 0 & 0 & -6.70 & -204.6 & 0 & 0 & 0 & 0 & -5.22 & -1.04 & -8.09 & 0 \\
2.41 & 2.41 & 0 & 0 & -3.46 & 0.682 & 0 & 0 & 0 & 0 & 4.8 & -2.41 & -4.17 & 0 \\
0 & 0 & 0 & 0 & -0.0116 & 0.0003 & -0.0003 & -0.0175 & 0.0003 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0114 & 0.0011 & 0.0142 & 0.0175 & 1 & 0 & -0.0001 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0063 & -0.0137 & 0 & 1 & -0.0175 & 0 & 0.0004 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.45 & -0.36 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -0.47 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -0.49 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0118 & 0 & 0.0002 & 0 & 0 & 0 & 0.0082 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.0112 & 0.0013 & -0.0002 & 0 & 0 & 0 & 0.0001 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & -0.1672 & -0.0609 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$$B = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
2.0994 & 0 & 0 \\
0 & -0.4616 & 0 \\
0 & 0 & -0.4937 \\
0 & -3.17e^{-7} & 0.0012 \\
0 & -2.10e^{-5} & -2.33e^{-5} \\
0 & 0 & 0 \\
0 & 9.6252 & -567.3836
\end{bmatrix}$$

In obtaining the numerical values of these matrices, the operating points are considered as follows (actually, all operating points should have been considered zero, but since the denominator of some ratios became zero and undefined, small near values are considered for some operating points instead).

$x_0^0 = [0,0]^T$, $x_1^0 = [0,0]^T$, $x_2^0 = \left[\frac{\pi}{180}, \frac{\pi}{180}, \frac{\pi}{180}\right]^T$, $x_3^0 = [0,0,0]^T$, $\theta^0 = \frac{\pi}{6}$, $\phi_v^0 = 0$, $R^0 = 1$, $dR^0 = 0$

4-2 PID controller simulation performance

In this section, the performance of the PID controller, whose parameters are adjusted by the Ziegler-Nichols method according to Table 3, is examined.

Table 3 – Parameters of the PID controller

gain type	gain amount
	0. 732
	0. 417
	1. 669

The simulation results for PID controller performance are shown in Fig. 3 to 6.

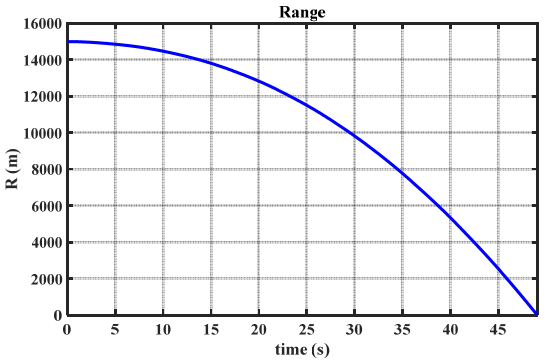

Fig. 3-The missile-target relative distance using the PID controller

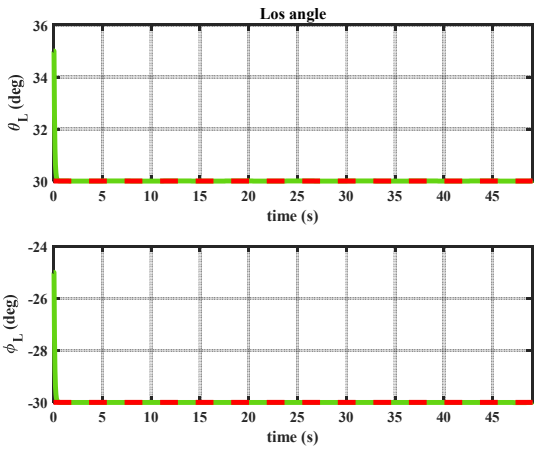

Fig. 4 -Changes of elevation and side angles with time using the PID controller

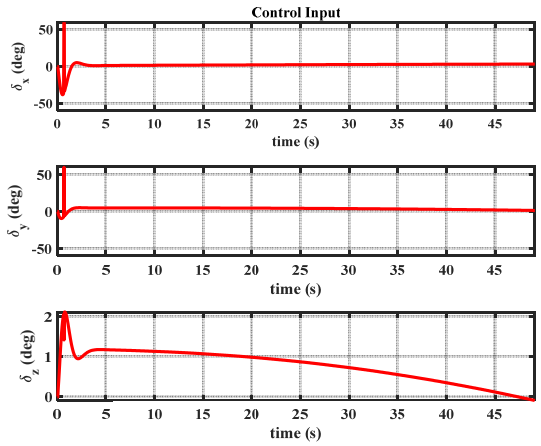

Fig. 5 -Missile control input with the PID controller

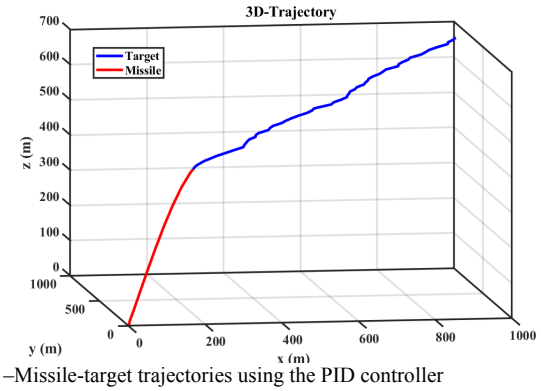

Fig.6 –Missile-target trajectories using the PID controller

The graphs in Fig. 3 and 4 show that in about 50 seconds, the relative distance between the missile and the target reaches zero. Also, the altitude and side angles reach their target values at the beginning of the flight. The control input in the diagram of Fig. 5 forced the missile to move towards the target according to the trajectories shown in Fig. 6 and hit the target at a height of 310 meters. In general, it can be said that the performance of the PID controller is poorly evaluated, since the flight time in this controller is rather too long for a short-range scenario, and the collision between the missile and the target occurred at a too low altitude, which is not considered an appropriate height for collision in air defenses. In addition, in the PID controller, the angles reach their reference values at the beginning of the missile's flight and do not converge during the flight, and this makes the target recognize the missile's path, hence lures the target to try to evade. The PID controller coefficients are obtained using Ziegler-Nicols method. The output of this method is normally fine and usually works better than that from optimization methods. Use of optimization tools in adjusting the PID controller gains are time-consuming and the desired result may not be quite reasonable due to spending too long a time. Moreover, the linearization required for the PID controller will make PID responsive only in limited operating points. According to the dynamics of the missile, linearization is not guaranteed for all modes, and even if the PID controller gains were designed using the neural and fuzzy network method, because the linearization was done, it is not valid in all domains. In addition, by changing the coefficients of the PID controller, its results are not improved in comparison with those from other controllers. Even if it is possible to achieve such a state, its control effort will be greatly increased and the system will be in a saturated state. As a result, PID is not considered as an appropriate controller for this system. In addition, the missile-target collision time with the PID controller is so high that PID is not a proper controller for the missile to hit the target in a reasonable time span. In air defense systems, particularly in short range scenarios, time plays a very key role, even if some control effort is increased. In the end, it can be said that although the PID controller is simpler from an implementation perspective, it does not respond properly to complex systems. As a result, a controller that is suitable for complex nonlinear systems will be more reliable.

4-3 Simulation of LQR controller performance

In order to simulate the LQR controller, first, the appropriate values for the Q and R matrices for target-missile engagement in this research were extracted and determined using the particle swarm optimization algorithm. The results of this optimization process for these two diagonal matrices are obtained as follows.

$Q = diag[0.0988,5.8532,0.0893,4.8708,9.3566,$
$0.8765,1.2646,3.1068,9.8594,4.4510,9.8985$
$,3.0687,0.1200,0.0380]$

$R = diag[9.1104,1.7283,1.3463]$

Using these two matrices in the cost function of relation (32), the changes in the values of the cost function were obtained and depicted in Fig 7. It is observed that the time to reach the end of the simulation is 1154 seconds for the PSO algorithm.

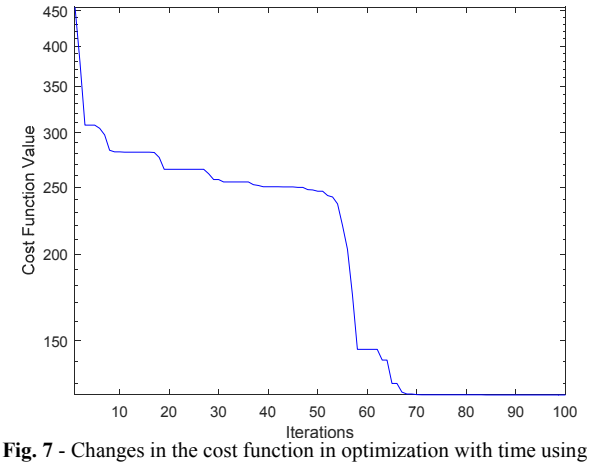

Fig. 7 - Changes in the cost function in optimization with time using the LQR controller and genetic algorithm

In the following, using the linearized state matrices and Riccati equation, the control input and gain matrix are calculated and the results from the simulation can be seen in Fig. 8 to 11.

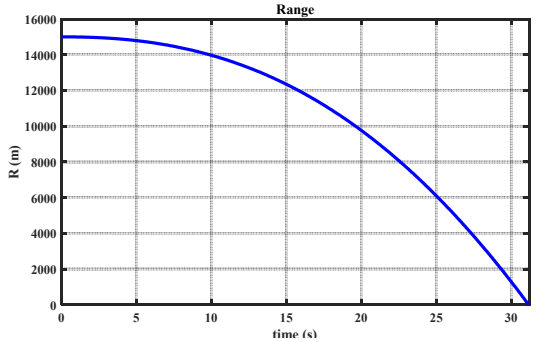

Fig. 8 - The missile-target relative distance using the LQR controller

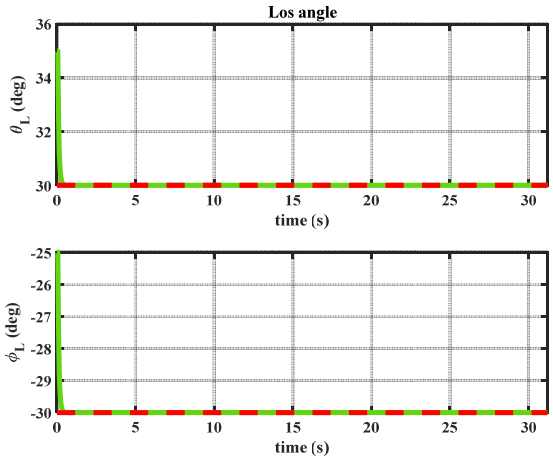

Fig. 9 - Changes of elevation and side angles with time using the LQR controller

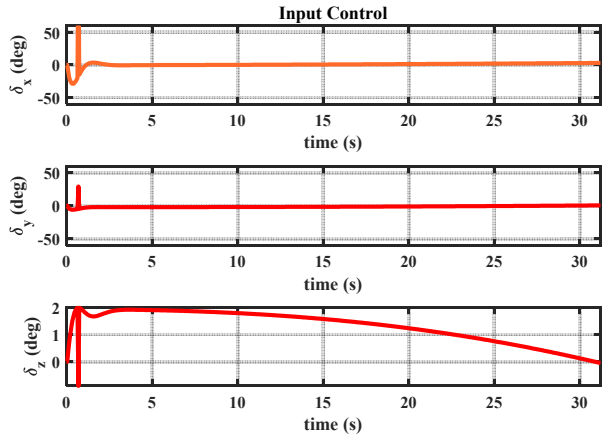

Fig. 10 Missile control input with the LQR controller

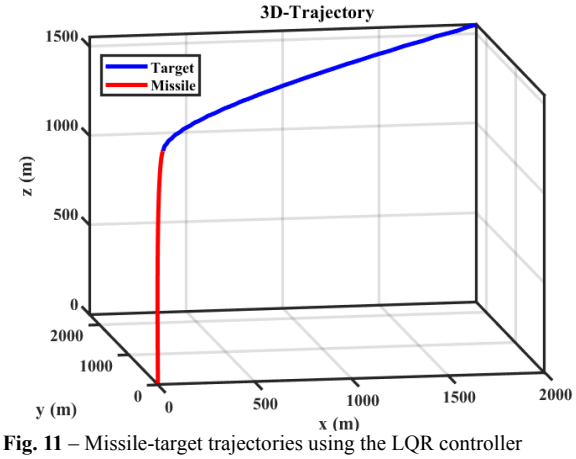

Fig. 11 – Missile-target trajectories using the LQR controller

The graphs in Fig. 8 shows that in about 32.3 seconds, the relative distance between the missile and the target reaches zero, resulting in collision. Also, the altitude and side angles reach their target values at the beginning of the flight, as shown in Fig. 9. The control input in the diagram of Figure 10 forced the missile to move towards the target according to the trajectory drawn in Figure 11, and hit the target at a height of 823 meters. As mentioned earlier, in the design of this LQR controller, the particle swarm optimization method was employed to calculate the weight matrices, rendering this controller outperform the PID controller. Although a better result was obtained, it can be stated that the performance of the LQR controller is yet evaluated as fair at best, because the flight time in this controller is still rather large for a short-range surface-to-air engagement, plus the fact that the height at which missile-target collision occurred is still rather low, making this controller still inappropriate for air defense. As stated earlier, in cases of short-range air defense, it is essential that the missile and the target collide at a higher altitude, and the target should not come close to the defense site positions. Also, in the LQR controller, the angles reach their reference

values at the beginning of the missile's flight rather than converge throughout the flight, which offers the target chances to recognize or foresee the missile's path and try to evade the missile and avoid collision. The weight matrices of this LQR controller are obtained using the particle swarm optimization method. Also, for the LQR controller, linearization must be done, with the linearized model effective only in limited operating points. Due to missile dynamics, linearization is not guaranteed for all modes. The time to collision using this LQR controller is rather high, which is not desirable, since time plays a key and decisive role in a missile-target conflict.

4-4. Simulation of the explicit linear predictive model controller

In this part, the performance of the predictive controller of the model is examined.

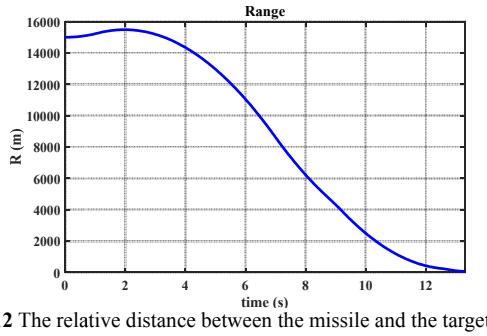

Fig. 12 The relative distance between the missile and the target-explicit linear predictive model controller

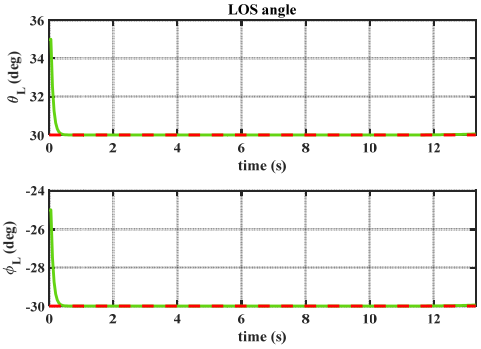

Fig. 13 Changes of elevation and side angles - explicit linear predictive model controller

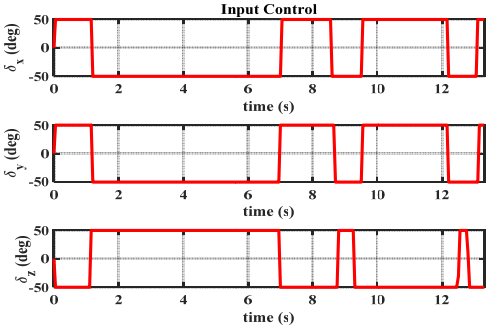

Fig. 14 missile control input - explicit linear predictive model controller

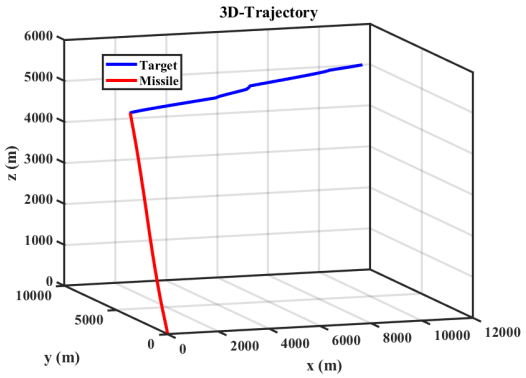

Fig. 15 Missile trajectory and target - explicit linear predictive model controller

The diagram in Figure 12 shows the relative distance between the missile and the target, and this relative distance gets close to zero in about 12.75 seconds and the collision takes place. This time is an appropriate time for air defenses in short ranges, because after firing, the missile quickly moves towards the target in this rather short time, and the missile does not leave a chance for the target to maneuver and avoid collision with the missile. Figure 13 shows that the flight angles converge at a proper time. According to Figure 14, the control input has also been applied well. This control input caused the missile to move towards the target according to the trajectory shown in Figure 15, intercepting the target at a height of 4160 meters. Online linearization method was used in MPC controller design and according to the simulations, this controller has performed much better than PID and LQR controllers. It can be said that the performance of the MPC controller is evaluated as superior, because the flight time with this controller is quite suitable for a short-range surface-to-air engagement. In addition, the height at witch collision took place is also quite proper for short-range defenses. As stated earlier, in short-range air defenses, it is better for the missile and the target to collide at a higher altitude so that the target should not come close to the defense site.

A quantitative comparison between the results of the controllers studied in this article is presented in Table 4.

Table 4 – Quantitative performance comparison of the three controllers examined

	PID	LQR	MPC
missile flight time	49.66 (sec)	32.30 (sec)	12.75 (sec)
missile flight height	310 (m)	823 (m)	4160 (m)

5- Conclusion

In this paper, an innovative approach for the guidance and control of air defense missiles was proposed, leveraging an online linear model predictive control (MPC) framework for an integrated three-dimensional (3D) missile-target model. The study began with the complete derivation and development of the integrated guidance and control (IGC) equations for both the missile and the target, forming the foundation for designing advanced controllers. To assess the effectiveness of the proposed controller, benchmark comparisons were performed with traditional PID and LQR controllers. The performance of the PID controller, as demonstrated by the simulations, was found to be suboptimal. This was attributed to its limitations, such as longer flight times and an inadequately defined control law, which led to excessive flight altitudes, thereby reducing effectiveness in air defense scenarios. Subsequently, an LQR controller was designed and evaluated, showing improved performance compared to the PID controller, but still falling short of achieving optimal results. The primary contribution of this work lies in the design and implementation of the proposed online linear MPC. This controller was designed to optimize the missile's trajectory in real-time by minimizing the time to collision and ensuring precise control within a feasible range. According to the simulation results, the proposed controller achieved a time to collision of approximately 12.75 seconds, a value that meets the operational requirements for air defense systems. Furthermore, the control law generated by the online MPC enabled the missile to maneuver effectively, resulting in a successful interception at an altitude of 4160 meters. This altitude, being both operationally advantageous and tactically significant, ensures that the missile engages the hostile target within a range and height that minimizes the chances of evasion through unpredictable maneuvers by the target. The findings of this research highlight the superiority of the proposed online linear MPC over conventional PID and LQR controllers. The ability of the predictive controller to deliver precise, efficient, and adaptive control in real-time makes it highly suitable for air defense applications, where rapid decision-making and high accuracy are critical. This novel integration of online MPC with an IGC framework not only improves missile-target engagement performance but also paves the way for more advanced implementations in modern air defense systems.

6-References

[1] P. Zarchan, Tactical and strategic missile guidance. American Institute of Aeronautics and Astronautics, Inc. , 2012.

[2] Neil F. Palumbo, Ross A. Blauwkamp, and Justin M. Lloyd. Basic principlesof homing guidance. Johns Hopkins APL Technical Digest (Applied PhysicsLaboratory), 2010.

[3] Nathan Harl and S. N. Balakrishnan. Reentry terminal guidance through slidingmodecontrol. Journal of guidance, control, and dynamics, 33(1): 186—199, 2010. doi. org/10. 2514/1. 42654.

[4] Wang, X. H. , Tan, C. P. , & Cheng, L. P. (2020). Impact time and angle constrained integrated guidance and control with application to salvo attack. Asian Journal of Control, 22(3), 1211-1220.

[5] He, S. , Song, T. , & Lin, D. (2017). Impact angle constrained integrated guidance and controlfor maneuvering targetinterception. Journal of Guidance, Control, and Dynamics, 40(10), 2653-2661.

[6] Ma, J. , Guo, H. , Li, P. , & Geng, L. (2013). Adaptive integrated guidance and control design for a missile with input constraints. IFAC Proceedings Volumes, 46(20), 206-211.

[7] Michael A. Cross. Missile interceptor integrated guidance andcontrol: single loop higher order sliding mode approach, PhD thesis , The University of Alabama in Huntsville, 2020.

[8] Jiayi Tian, Neng Xiong, Shifeng Zhang, Huabo Yang, Zhenyu Jiang,Integrated guidance and control for missile with narrow field-of-view strapdown seeker, ISA Transactions, vol. 106,2020, Pages 124-137. https: //doi. org/10. 1016/j. isatra. 2020. 06. 012.

[9] Sinha, Abhinav & Kumar, Shashi & Mukherjee, Dwaipayan, Integrated Guidance and Control For Dual Control Interceptors Under Impact Time Constraint, AIAA scitech Forum Conference, January 2021. doi. org/10. 2514/6. 2021-1463.

[10] Li Z. , Dong Q. , Zhang X. , Zhang H. , Zhang F. Field-to-View Constrained Integrated Guidance and Control for Hypersonic Homing Missiles Intercepting Supersonic Maneuvering Targets, Aerospace. 2022; 9(11): 640. https: //doi. org/10. 3390/aerospace9110640.

[11] Xiaohui Liang, Bin Xu, Kunhao Xinglan LiuAdaptive NN control of integrated guidance and control systems based on disturbance observer, Journal of the Franklin Institute, Volume 360, Issue 1, 2023.

[12] Xiangyu Tang, Jianglong Yu, Xiwang Dong, Zhang Ren,Integrated guidance and control with impact angle and general field-of-view constraints, Aerospace Science and Technology, Volume 144,2024. DOI: 1 10. 1016/j. ast. 2023. 108809.

[13] S. Ma, A. Li و Z. Wang, “Integrated Guidance and Control for Homing Missiles with Terminal Angular Constraint in Three Dimension Space", IEEE International Conference on Artificial Intelligence and Information Systems, Dalian, China, 2020. DOI: 10.1109/ICAIIS49377.2020.9194808.

[14] Modern control Engineering 5th Edition, by Katsuhiko Ogata,2009.

[15] A. Bemporad, "Recent advances in embedded model predictive control," Michigan Engineering, 14 July 2016. [Online]. Available: https://www.youtube.com/watch?v=ugeCx1sytNU. [Accessed 1 June 2019].

[16] A. Bemporad, M. Morari, V. Dua and E. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," Automatica, vol. 38, no. 1, p. 3–20, 2002.

[17] R. Rajamani, "Vehicle dynamics and control," in Mechanical Engineering Series, Springer, 2012, pp. 87-109 & 241-162.

[18] J. H. Lee, "Model predictive control: review of the three decades of development," International Journal of Control, Automation and Systems, vol. 9, p. 415–424, 2011.