



Semnan University



Generalized Incremental Predictive Guidance and Control: Design, Stability, Real-Time Validation, and Simulation

Husam Hasan¹, Nemat Allah Ghahremani² and Saeed Nasrollahi^{3*}

Abstract--This paper presents a new integrated guidance and control system for an air vehicle based on incremental predictive control. Integrating guidance and control loops improves reliability, enhances performance, and reduces costs. The integration is built using the linear system, with the command signal being the pursuer's deflection angle and the output being the miss distance. The generalized incremental model predictive control is used as the commanding block to control and guide the pursuer to its target. The goal is to minimize a quadratic cost function with a cost associated with the relative displacement between the target and the pursuer and the deflection angle. At first, the dynamical system model is derived, and the target acceleration is added to the system to provide additional information to reduce the control effort. Then, the guidance-control algorithm is designed and implemented, and the stability of the proposed algorithm is proven. After that, the influence of prediction and control horizons on the integrated system is analyzed. The results show the effectiveness of the predictive integrated system. Finally, to ensure the implementation capability of the proposed algorithm, a Processor-in-the-Loop experiment is conducted using Arduino Duo, and it yielded good results.

Index Terms- Autopilot, Generalized Incremental Predictive Control, Guidance, Processor in the Loop, Stability analyses.

NOMENCLATURE

V_P	pursuer velocity	q	the pitch angular rate of the pursuer
γ_P	pursuer flight-path angle	M_α	pitch moment generated from the angle of attack

a_P	pursuer acceleration	M_q	pitch moment generated from pitch rate
λ	Line-of-Sight (LOS) angle	M_δ	pitch moment generated from deflection angle
r	the range between pursuer and target	L_α	aerodynamic force generated by the angle of attack
f_1, f_2	present standard saturation functions	L_δ	aerodynamic force generated by deflection angle

I. INTRODUCTION

THE familiar way of designing guidance and autopilot systems for a pursuer is to separate the autopilot system from the guidance system and design each individually [1], for instance, in a recently published paper [2]. In this cascaded control structure, the autopilot is the inner loop and deals with high bandwidth processes, while the outer loop, which is the guidance loop, processes lower bandwidth processes. Because of the bandwidth separation, the full potential of weapons is not fully realized, which significantly reduces the pursuer's ability to strike precisely. To overcome this problem, Integrated Guidance and Control (IGC) provides an attractive concept by combining the guidance and autopilot subsystems into one system and using the relative state of the pursuer to generate fin deflection commands to drive the pursuer to intercept the target. In addition, in this approach, the output of the Inertial Navigation System (INS)

Received; 2024-08-15 Revised; 2025-09-23 Accepted; 2025-09-27

¹. Faculty of Electrical & Computer Engineering, Malek-Ashtar University of Technology, Tehran, Iran.

². Faculty of Electrical & Computer Engineering, Malek-Ashtar University of Technology, Tehran, Iran.

³. Faculty of Electrical & Computer Engineering, Malek-Ashtar University of Technology, Tehran, Iran.

* Corresponding author Email: nasrollahi.saeid@gmail.com

Cite this article as:

Hasan, H., Ghahremani, N, A. and Nasrollahi, S. (2025) Generalized Incremental Predictive Guidance and Control: Design, Stability, Real-Time Validation, and Simulation. *Journal of Modeling & Simulation in Electrical & Electronics Engineering (MSEEE)*, 5(2), 19-28.

DOI: <https://doi.org/10.22075/MSEEE.2025.38679.1224>

sensors is delivered to the autopilot, which reduces the cost, increases the reliability (by using fewer components), and increases the performance because it is an accurate sensor [3].

Since 1983, when IGC was introduced, numerous researchers have applied a wide range of control strategies.

Sliding Mode Control was considered in [12] because it can deal with the nonlinearity and modeling error, where the zero-effort miss distance was considered as a sliding surface instead of using the line-of-sight rate. For a certain impact angle, [17] used the linear quadratic regulator to demonstrate the IGC lead over the traditional guidance and control by using a parameterized linear system; in addition, controllability and equilibrium point analysis were conducted. In [18], a comparison between three guidance and control configurations (1-separated guidance and control, 2-an integrated guidance and control, 3-an integrated two-loop autopilot-guidance), where the thrust vector control missile was conducted. The optimal control theory was used to minimize a quadratic cost function with a terminal cost on the miss distance based on a first-order autopilot model; the simulation was based on the Pareto front.

The work in [19] took a six-degree-of-freedom model, using MPC with an IGC to produce the optimal closed-form control law. The extended-state observer was used for the target acceleration estimation to add the ability to hit maneuverable targets. The study [20] featured online trajectory generation for gliding at a specified dynamic pressure by modulating the bank angle. The closed-loop system demonstrated robust tracking and managed constraints. For the flying vehicle in [21], the MPC controller's objective was to simultaneously minimize both miss distance and time to collision with a target in a 3D engagement. Simulations demonstrated its performance over PID and LQR controllers. The authors in [22] used MPC to control the IGC system with noise in its measurement. To solve this problem, the Moving Horizon Estimation algorithm technique was implemented. They implemented a pseudo-spectral method to solve the two online optimizations, which improved the solution-finding accuracy. Also, a non-linear programming sensitivity-based optimization method was applied. The real-time computation problem of MPC for a linear IGC system was also handled in [23]. To minimize the computation time, the authors divided the primal-dual interior point method into four sections for solving convex optimization in MPC, which showed a reduced computation time for this condition.

This paper presents a 2-dimensional interception problem for the IGC system based on Generalized Incremental Predictive Control (GIPC). First, a linear engagement model is derived. Subsequently, the GIPC law for a linear system is implemented. The computed incremental control law is then utilized to find the IGC guidance command. The contribution of this work is to implement the GIPC with IGC, which gives the new GIP-GC method by considering the target acceleration. The proposed IGC algorithm, based on a generalized incremental predictive control technique, can complete the mission with little miss distance by adding the miss distance and deflection angle to the cost function so that it will be minimized through an optimization process. A simulation is conducted to analyze the performance considering target maneuvers (Step, Sinusoidal, and Pull-Up

For example: Model Predictive Control (MPC) [4], [5]; Feedback Linearization [1], [6]; Back-Stepping Control [7], [8], [9]; Game Theory [10], [11]; Sliding Mode Control [12], [13], [14]; Linear Optimal Control theory [15], [16]. These approaches were employed to design the IGC law.

maneuvering targets). The proposed method is compared with PN, Continuous Adaptive Sliding Mode Guidance (CASMG) [24], and Robust Optimal Guidance [25]. Finally, an analysis of the effectiveness of the control and predictive horizons on the system is performed with a laboratory experiment of the proposed algorithm on an Arduino Duo board to test real-time performance. MATLAB's Simulink is used to create the C code for the Arduino, with a serial link to establish a connection between the hardware device and the IGC model. The experiment indicates a successful algorithm implementation on the processor, where the deflection angle generated by Simulink matched that generated from the test board. Also, altitude effects are discussed. It's worth mentioning that although numerous researchers have applied to IGC and MPC, a specific algorithm with the estimation of target acceleration was not performed.

The remainder of this paper is organized as follows: Section 2 contains the issue statement from which the integrated system will be derived. Section 3 demonstrates the integrated control and guidance algorithm. The asymptotic stability is derived in Section 4. In Section 5, numerical simulation outcomes are presented; analyses of prediction and control horizons and a processor-in-the-loop experiment are conducted to check the algorithm's capability to work in real situations. Finally, the results are given in Section 6.

II. PROBLEM FORMULATION

The integrated system is based on [26], [27], [28]. Fig. 1 shows the planar homing engagement geometry, where the pursuer (P) and the Target (T) are presented in the polar coordinate system (r, λ) attached to the pursuer, where λ is the Line-Of-Sight (LOS) angle, and the range between the pursuer and the target is r .

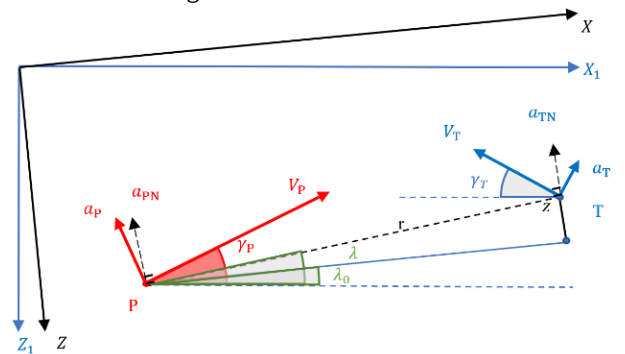


Fig. 1. Planar engagement geometry [26]

Assume that the pursuer's velocity is V_P and the target velocity is V_T . They are moving with fixed velocities; the gravitational force is neglected. The pursuer's flight-path angle is γ_P , and the target flight-path angle is γ_T . The equations of engagement kinematics are:

$$\dot{r} = -[V_P \cos(\gamma_P - \lambda) + V_T \cos(\gamma_T + \lambda)] \quad (1)$$

$$\dot{\lambda} = -V_P \sin(\gamma_P - \lambda) + V_T \sin(\gamma_T + \lambda) \quad (2)$$

$$\dot{\gamma}_P = \frac{a_P}{V_P} \quad (3)$$

$$\dot{\gamma}_T = \frac{a_T}{V_T} \quad (4)$$

The pursuer's acceleration is a_P , given by:

$$a_P = (L_\alpha - L_\delta)f_1(\alpha) + L_\delta f_2(\alpha + \delta) \quad (5)$$

where L_α, L_δ are the aerodynamic forces generated by the angle of attack and the deflection angle, respectively. Assuming U_m the maximum deflection angle, f_1 and f_2 present standard saturation functions as:

$$\text{sat}(u) = \begin{cases} U_m & U_m < u \\ u & -U_m \leq u \leq U_m \\ -U_m & u < -U_m \end{cases} \quad (6)$$

Consider the pursuer's longitudinal dynamics by neglecting gravity. It is assumed that the pursuer has no thrust during the last phase. The pitch plane dynamics can be shown as follows:

$$\dot{\alpha} = q - (L_\alpha - L_\delta)f_1(\alpha) + L_\delta f_2(\alpha + \delta)/V_P \quad (7)$$

$$\dot{q} = (M_\alpha - M_\delta)f_1(\alpha) + M_q q + M_\delta f_2(\alpha + \delta) \quad (8)$$

$$\dot{\delta} = (\delta^c - \delta)/\tau_s \quad (9)$$

Equation (9) presents the actuator that controls δ , the canard deflection angle, where τ_s is the actuator time constant; q is the pitch angular rate of the pursuer, and M_α, M_q, M_δ are the pitch moment acting on the pursuer generated from the angle of attack, pitch rate, and deflection angle, respectively. Superscript C represents the commanded signal. So, δ^c is the controlled deflection.

From Fig. 1, the initial LOS is aligned with the X-axis. Assume that z and $\dot{z}$ are the relative displacement between the target and the pursuer, vertical to the initial LOS direction, and its derivative, respectively. The target and pursuer accelerations vertical to LOS become a_{TN} and a_{PN} . Subscript N represents the projection vertical to the LOS.

$$a_{PN} \approx a_P \cos(\gamma_{P0} - \lambda_0) \quad (10)$$

$$a_{TN} \approx a_T \cos(\gamma_{T0} + \lambda_0) \quad (11)$$

$$\ddot{z} = a_{TN} - a_{PN} \quad (12)$$

The subscript 0 stands for the initial value around which linearization has been performed. The control signal is the deflection angle $U = \delta^c$. In this case, the integrated guidance with the control became:

$$\begin{bmatrix} \dot{z} \\ \ddot{z} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & L_\alpha \omega & 0 & L_\delta \omega \\ 0 & 0 & \frac{-L_\alpha}{V_P} & 1 & \frac{-L_\delta}{V_P} \\ 0 & 0 & M_\alpha & M_q & M_\delta \\ 0 & 0 & 0 & 0 & \frac{-1}{\tau_s} \end{bmatrix} \begin{bmatrix} z \\ \dot{z} \\ \alpha \\ q \\ \delta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau_s} \end{bmatrix} \delta^c + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} a_{TN} \quad (13)$$

Assuming $\omega = \cos(\gamma_{P0} - \lambda_0)$, from this state space model, the interception becomes a regulation problem.

Assuming that the change rate of the target acceleration is zero, the target acceleration (which is an unknown input) can be added as an extended state to the overall system.

$$\begin{bmatrix} \dot{z} \\ \ddot{z} \\ \dot{\alpha} \\ \dot{q} \\ \dot{\delta} \\ \dot{a}_{TN} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & L_\alpha \omega & 0 & L_\delta \omega & 1 \\ 0 & 0 & \frac{-L_\alpha}{V_P} & 1 & \frac{-L_\delta}{V_P} & 0 \\ 0 & 0 & M_\alpha & M_q & M_\delta & 0 \\ 0 & 0 & 0 & 0 & \frac{-1}{\tau_s} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \dot{z} \\ \alpha \\ q \\ \delta \\ a_{TN} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau_s} \\ 0 \end{bmatrix} \delta^c \quad (14)$$

III. GENERALIZED INCREMENTAL PREDICTIVE FOR INTEGRATED GUIDANCE AND CONTROL (GIP-GC)

In the GIPC method, both present and previous states are considered in the j -step ahead prediction of the outputs and states. To illustrate this method, let's take a linear state-space model representing one step ahead state/output predictions [29].

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A} \mathbf{x}_k + \mathbf{B} \mathbf{u}_k \\ \mathbf{Y}_k &= \mathbf{C} \mathbf{x}_k \end{aligned} \quad (15)$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{U} \in \mathbb{R}$ and $\mathbf{Y} \in \mathbb{R}$ represent the state vector, control input, and system output, respectively, and $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^n$ and $\mathbf{C} \in \mathbb{R}^n$ are system matrices (introduced in (14)). The incremental state prediction is:

$$\Delta \mathbf{x}_{k+1} = \mathbf{A} \Delta \mathbf{x}_k + \mathbf{B} \Delta \mathbf{u}_k \quad (16)$$

$$\Delta \mathbf{x}_k = \mathbf{x}_k - \mathbf{x}_{k-1}$$

$$\Delta \mathbf{x}_{k+1} = \mathbf{x}_{k+1} - \mathbf{x}_k \quad (17)$$

$$\Delta \mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1}$$

Combining (15) and (16) gives the state-space predictions in incremental form for one step ahead:

$$\mathbf{x}_{k+1} = (\mathbf{A} + \mathbf{I}) \mathbf{x}_k - \mathbf{A} \mathbf{x}_{k-1} + \mathbf{B} \Delta \mathbf{u}_k \quad (18)$$

The prediction of the output $\mathbf{Y}$ can be written as compact matrix/vector form, assuming that N_c and N_p are the control horizon and the prediction horizon, respectively:

$$\mathbf{Y} = \mathbf{F}_0 \mathbf{x}_k - \mathbf{F}_1 \mathbf{x}_{k-1} + \Phi \Delta \mathbf{u} \quad (19)$$

Where:

$$\mathbf{Y} = \begin{bmatrix} \mathbf{Y}_{k+1} \\ \mathbf{Y}_{k+2} \\ \mathbf{Y}_{k+3} \\ \vdots \\ \mathbf{Y}_{k+N_p} \end{bmatrix}, \Delta \mathbf{u} = \begin{bmatrix} \Delta \mathbf{u}_k \\ \Delta \mathbf{u}_{k+1} \\ \Delta \mathbf{u}_{k+2} \\ \vdots \\ \Delta \mathbf{u}_{k+N_c-1} \end{bmatrix}$$

$$\mathbf{F}_0 = \begin{bmatrix} \mathbf{C}(\mathbf{A} + \mathbf{I}) \\ \mathbf{C}(\mathbf{A}^2 + \mathbf{A} + \mathbf{I}) \\ \mathbf{C}(\mathbf{A}^3 + \mathbf{A}^2 + \mathbf{A} + \mathbf{I}) \\ \vdots \\ \mathbf{C}\left(\sum_{i=0}^{N_p} \mathbf{A}^i\right) \end{bmatrix} \quad (20)$$

$$\mathbf{F}_1 = \begin{bmatrix} \mathbf{C}\mathbf{A} \\ \mathbf{C}(\mathbf{A}^2 + \mathbf{A}) \\ \mathbf{C}(\mathbf{A}^3 + \mathbf{A}^2 + \mathbf{A}) \\ \vdots \\ \mathbf{C}\left(\sum_{i=1}^{N_p} \mathbf{A}^i\right) \end{bmatrix}$$

$$\Phi = \begin{bmatrix} \mathbf{C}\mathbf{B} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}\mathbf{B} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}(\mathbf{A}^2 + \mathbf{A} + \mathbf{I})\mathbf{B} & \mathbf{C}(\mathbf{A} + \mathbf{I})\mathbf{B} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}\left(\sum_{i=0}^{N_p-1} \mathbf{A}^i\right)\mathbf{B} & \mathbf{C}\left(\sum_{i=0}^{N_p-2} \mathbf{A}^i\right)\mathbf{B} & \cdots & \mathbf{C}\left(\sum_{i=0}^{N_p-N_c} \mathbf{A}^i\right)\mathbf{B} \end{bmatrix} \quad (21)$$

So, the predicted output using incremental prediction control is (19). Now, the incremental control is derived by defining the cost function of the receding-horizon control problem with the fault as:

$$J = (\mathbf{Y}_d - \mathbf{Y})^T \mathbf{Q} (\mathbf{Y}_d - \mathbf{Y}) + \Delta \mathbf{u}^T \mathbf{R} \Delta \mathbf{u} \quad (22)$$

where $\mathbf{Y}_d = [Y_{d,k+1}, Y_{d,k+2}, \dots, Y_{d,k+N_p}]^T$ is the desired output, $\mathbf{Q} = q * \mathbf{I}$ is an error weighting matrix, $\mathbf{R} = r * \mathbf{I}$ is a control weighting matrix, and $\mathbf{I}$ is the identity matrix. From (22), it is clear that the cost function consists of two penalty terms. The first one is the weighted error square between the output value and the desired value, and by minimizing the weighted square, one could force the output value to be close to the desired one. The second one reflects the control that must be applied for the correction process and should be as small as possible. The cost function (22) could be optimized by solving the quadratic programming problem for the GIP-GC algorithm.

By substituting the predicted output into the cost function and assuming that the desired output $\mathbf{Y}_d = \mathbf{0}$, the cost function becomes:

$$J = (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1} - \Phi \Delta \mathbf{u})^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1} - \Phi \Delta \mathbf{u}) + \Delta \mathbf{u}^T \mathbf{R} \Delta \mathbf{u} \quad (23)$$

By expanding the brackets:

$$J = \Delta \mathbf{u}^T \mathbf{R} \Delta \mathbf{u} - \mathbf{x}_k^T \mathbf{F}_0^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1} - \Phi \Delta \mathbf{u}) + \mathbf{x}_{k-1}^T \mathbf{F}_1^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1} - \Phi \Delta \mathbf{u}) - \Delta \mathbf{u}^T \Phi^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1} - \Phi \Delta \mathbf{u}) \quad (24)$$

To find the optimal $\Delta \mathbf{u}$ that minimizes J , the first derivative of the cost function must be taken, and the first derivative must be equal to zero; this gives $\Delta \mathbf{u}$, which minimizes J as:

$$\Delta \mathbf{u}_{k+1} = (\Phi^T \mathbf{Q} \Phi + \mathbf{R})^{-1} \Phi^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1}) \quad (25)$$

This control strategy is implemented in the integrated

guidance and control system, which will be called Generalized Incremental Predictive Guidance and Control (GIP-GC); this is done by considering the cost function, the output signal as $Y = z$ (the first state), and the control rate as $\Delta \mathbf{u} = \Delta \delta^c$ the canard deflection angle. By doing so, (25) represented as:

$$\Delta \delta_{k+1}^c = (\Phi^T \mathbf{Q} \Phi + \mathbf{R})^{-1} \Phi^T \mathbf{Q} (-\mathbf{F}_0 \mathbf{x}_k + \mathbf{F}_1 \mathbf{x}_{k-1}) \quad (26)$$

The most important characteristic of this method is that it uses more information to build the $\Delta \delta^c$ signal, where it uses the current and the previous state.

IV. GIP-GC STABILITY

The asymptotic stability is derived in this section based on [30]. In sample time k , the future guidance trajectory is optimized using the cost function (22), where $\mathbf{Y}_d$ is the desired output, and in our case, it is equal to zero, and $\mathbf{Y}_{k+i}$ is the output of the system and can be replaced with (14), so by assuming $Q_X = C^T Q C$, (22) can be written as:

$$J(x_{k+i}) = \sum_{i=1}^{N_p} x_{k+i}^T Q_X x_{k+i} + \sum_{i=0}^{N_p-1} \Delta u_{k+i}^T R \Delta u_{k+i} \quad (27)$$

To prove the asymptotic stability, two assumptions are considered:

Assumption 1: The terminal state is constrained as $x_{k+N_p} = 0$, resulting from the optimal solution in (25).

Assumption 2: for each sample time k , there is an optimal solution $\Delta \mathbf{u}_k^*$ which minimizes the cost function and obeys the terminal state condition $x_{k+N_p} = 0$.

Theorem 1. Given the cost function in (22) and the above assumptions, the GIPC is asymptotically stable.

Proof. By taking the cost function as the Lyapunov function, the stability happens when the change in the cost function is negative, meaning its value becomes smaller with time. Assuming that $V_{x(k)}$ Lyapunov function at the sample time k equal to the cost function $J_{x(k)}$:

$$V_{x(k)} = \sum_{i=1}^{N_p} x_{k+i}^T Q_X x_{k+i} + \sum_{j=0}^{N_p-1} \Delta u_{k+i}^T R \Delta u_{k+i} \quad (28)$$

One can realize that $V_{x(k)}$ is a positive definite function and can be a Lyapunov function. According to the second assumption, $\Delta \mathbf{u}_k^*$ guarantees the optimal solution. At the next sample time $k + 1$, the Lyapunov function $V(x_{k+1})$ can be written as:

$$V(x_{k+1}) = \sum_{i=1}^{N_p} x_{k+i+1}^T Q_X x_{k+i+1} + \sum_{j=0}^{N_p-1} \Delta u_{k+i+1}^T R \Delta u_{k+i+1} \quad (29)$$

Moreover, by replacing the optimized $\Delta \mathbf{u}_{k+1}^* = [\Delta u_{k+1}, \Delta u_{k+2}, \dots, \Delta u_{k+N_p-1}, 0]^T$ in the last equation. The previous equation can be written as:

$$V(x_{k+1}) \leq V'(x_{k+1}) \quad (30)$$

The function $V'(x_{k+1})$ similar to (28) except that the $\Delta \mathbf{u}_{k+1}$ sequence is replaced by the feasible sequence

$$\Delta u_{k+1}^* V(x_{k+1}) - V(x_k) \leq V'(x_{k+1}) - V(x_k) \quad (31)$$

The two functions on the right-hand side have the same state and control sequence from the sample $k+1, k+2, \dots, k+n-1$, so the right-hand side can be rewritten as:

$$V'(x_{k+1}) - V(x_k) = x_{k+Np}^T Q_X x_{k+Np} - x_{k+1}^T Q_X x_{k+1} - \Delta u_k^T R \Delta u_k \quad (32)$$

From the first assumption, $x_{k+Np} = 0$, the last equation can be written as:

$$V'(x_{k+1}) - V(x_k) = -x_{k+1}^T Q_X x_{k+1} - \Delta u_k^T R \Delta u_k \quad (33)$$

Then:

$$V(x_{k+1}) - V(x_k) \leq -x_{k+1}^T Q_X x_{k+1} - \Delta u_k^T R \Delta u_k < 0 \quad (34)$$

This is proof that the change in the cost function is negative, then $V(x_k)$ has asymptotic stability.

V. SIMULATION

Because of the benefits of the IGC system, such as reduced cost, increased reliability, and improved performance, it has been considered a suitable solution for tactical pursuers.

The effectiveness of the proposed GIP-GC strategy is evaluated through simulation scenarios using the system model presented in (15) and the canard deflection angle described by (25). A performance comparison is made against the Proportional Navigation (PN) law, assuming an autopilot time constant of 0.3 seconds. Further comparison is made with the Robust Optimal Guidance (GESO) method presented in [25] and with continuous adaptive sliding mode guidance (CASMG) [24]. Sensitivity analysis of the prediction and control horizons is carried out. Finally, a PIL experiment was performed. Simulations are performed using MATLAB R2020b.

A. Scenarios

The motion model of the target in the inertial coordinate system is based on the simulation introduced in [31] and can be described as $\dot{\gamma}_T = \frac{a_T}{V_T}$, $\dot{X}_T = -V_T \cos(\gamma_T)$, $\dot{Z}_T = V_T \sin(\gamma_T)$. For pursuer $\dot{\gamma}_P = \frac{a_P}{V_P}$, $\dot{X}_P = V_P \cos(\gamma_P)$, $\dot{Z}_P = V_P \sin(\gamma_P)$. The target is located at (2500,0), and the pursuer is at (0,0). To intercept the target, the desired output $\mathbf{Y}_d = \mathbf{0}$, the initial pursuer acceleration was 0[g], and the autopilot time constant was $\tau_s = 0.02[s]$. TABLE I shows the initial conditions for the pursuer and target.

TABLE I
Pursuer and Target Initial Conditions

	Pursuer	Target	unit
Velocity	400	300	[m/s]
flight-path angle	20	25	[deg]

The time constant is $T_s = 0.01s$. Pursuer aerodynamic parameters are $L_\alpha = 1270 [m/s^2]$, $L_\delta = 80 [m/s^2]$, $M_\alpha =$

$-74 [1/s^2]$, $M_\delta = -5 [1/s]$, $M_q = 160 [1/s^2]$ and the GIP-GC parameters are $N_p = 80$, $N_c = 2$, $r = 1000$, $q = 0.02$. The saturation function in (6) is set to be $U_m = 30 [deg]$.

1) Step-maneuvering Target.

In this scenario, the target is assumed to be performing a step maneuver ($a_T = 10 m/s^2$). The engagement trajectories for the pursuer and target under PN, GESO, and GIP-GC are illustrated in Fig. 2. The miss distance under GIP-GC is about 1 meter, whereas the PN method results in a miss distance exceeding 9m, GESO is 1.8m, and CASMG is 1.6m.

The pursuer accelerations under the algorithms are plotted in Fig. 3. Under PN, the acceleration increases gradually and may not be sufficient during the final engagement phase. GESO-generated acceleration first oscillates with a large amplitude and then converges around the target acceleration. While CASMG increases steadily, it then takes a constant value. In contrast, GIP-GC leads to a rapid increase in acceleration, followed by a decrease towards the near target acceleration. Fig. 4 shows the relative distance between the pursuer and target.

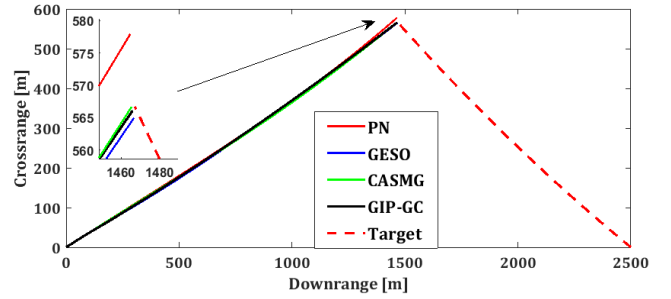


Fig. 2 The engagement trajectory for the step target maneuver

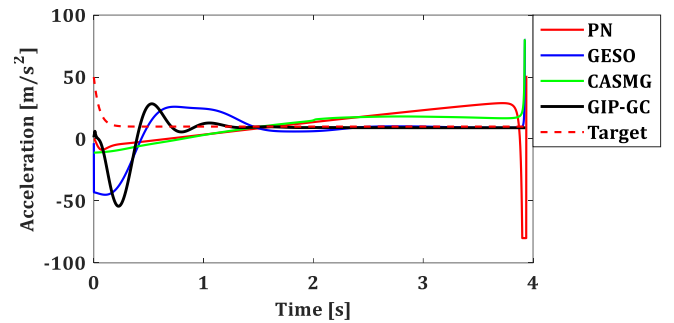


Fig. 3 Pursuer acceleration for step target maneuver

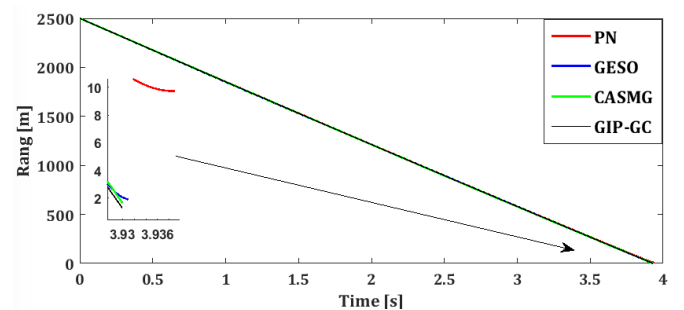


Fig. 4 Pursuer-Target relative distance for step target maneuver

2) Maneuvering Target – Sinusoidal Profile

In this scenario, the target executes a "sinusoidal" maneuver $a_T = 50 \sin(3 \cdot t) m/s^2$. The performance of GIP-GC is compared to that of PN, GESO, and CASMG. Fig. 5 shows the pursuer-target engagement trajectories. All approaches result in the successful interception of the target.

Fig. 6 compares the accelerations generated by PN, GESO, CASMG, and GIP-GC. The GIP-GC method produces accelerations, starting from zero and increasing before decaying and flipping direction with sinusoidal waving. GIP-GC exhibits a lower final acceleration, which is a favorable characteristic, and all the other algorithms saturated at the end of the simulation. Fig. 7 illustrates the relative distance for the methods. The GIP-GC approach is significantly shorter and closer to the target compared with the other approaches, demonstrating more efficient behavior. Table II shows that GIP-GC has the lowest control effort, miss distance, and final time.

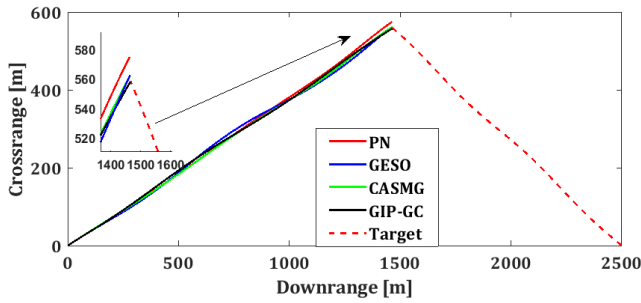


Fig. 5 The engagement trajectory for sinusoidal target maneuver

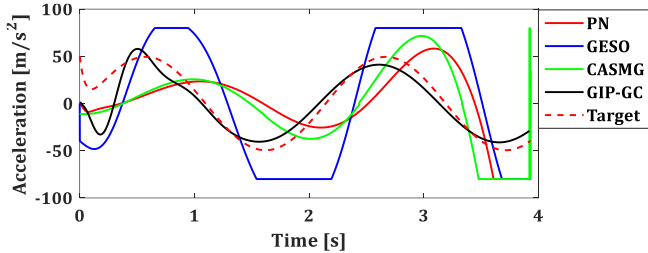


Fig. 6 Pursuer acceleration for sinusoidal target maneuver

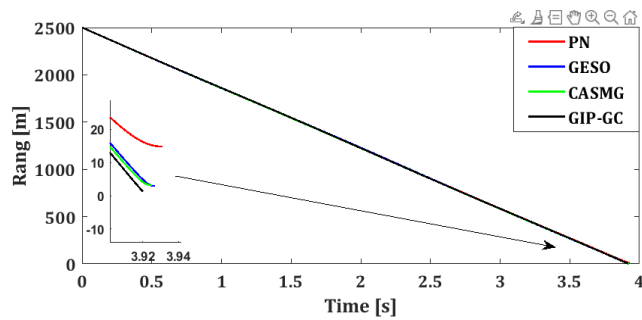


Fig. 7 Pursuer-Target canard deflection for sinusoidal target maneuver

3) Pull-Up-maneuvering Target.

The pull-up target acceleration mathematical model is used as [32]. The target acceleration is considered as $a_T = 5t + 20 \text{ m/s}^2$. All the algorithms guide the pursuer to intercept the target Fig. 8. Their differences in the acceleration profile, miss distance, and interception time. Fig. 9 presents the lateral acceleration profiles over time. The PN method generates the highest acceleration and reaches the saturation, which may be impractical, especially in the last phase. GESO oscillates at the beginning of the engagement, reflecting sensitivity to the target's maneuver. CASMG reduces this oscillation but still requires relatively high acceleration. In contrast, the GIP-GC method provides the most stable acceleration, closely matching the target's acceleration profile. This highlights the ability of GIP-GC to achieve interception with lower control effort. The relative

range between pursuer and target during the engagement is shown in Fig. 10. PN leaves a residual miss distance, while GESO and CASMG reduce it further but still fall short of perfect convergence. GIP-GC demonstrates the best performance. A numerical comparison is shown in Table II to demonstrate the algorithms' performances.

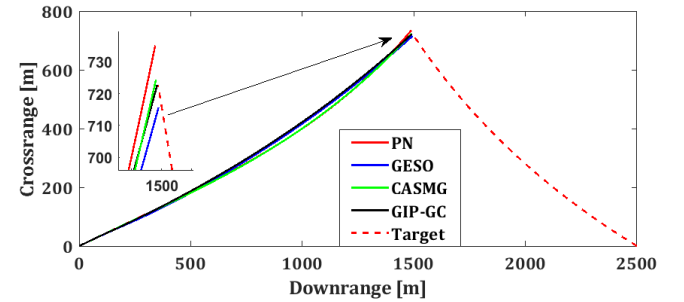


Fig. 8 The engagement trajectory for the pull-up target maneuver

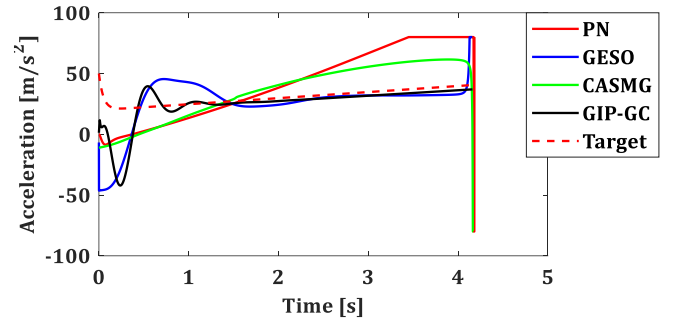


Fig. 9 Pursuer acceleration for pull-up target maneuver

TABLE II
Comparing the Guidance Methods

maneuver		CE [m^2/s^4]	MD [m]	Time [s]
Step	PN	1316	9.7	3.939
	GESO	1225	1.897	3.931
	CASMG	988	1.64	3.93
	GIP-GC	871	1.27	3.93
Sinusoidal	PN	4547	14.83	3.931
	GESO	16989	2.99	3.927
	CASMG	6514	3.27	3.925
	GIP-GC	3651	1.27	3.92
Pull-Up	PN	10662	7.2	4.182
	GESO	4591	6.682	4.158
	CASMG	7461	1.823	4.168
	GIP-GC	3612	1.46	4.16

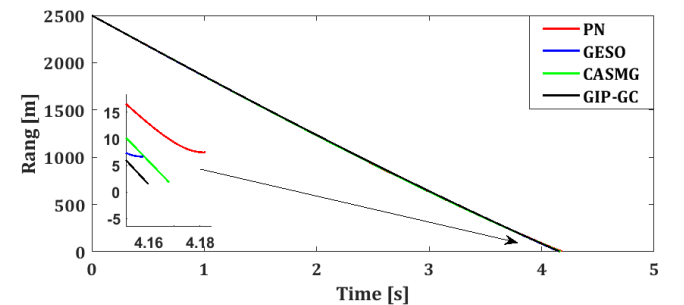


Fig. 10 Pursuer-Target relative distance for pull-up target maneuver

B. The parameter sensitivity

All system parameters are constant except for the parameter under examination to analyze the sensitivity of the

prediction and control horizons.

1) Prediction horizon.

Here is the result of changing the prediction horizon $N_p = [40, 60, 80]$. From Fig. 11, a relatively small prediction horizon of 40 results in a significantly larger canard deflection compared to other values. The canard deflection oscillates significantly until it reaches the final value. As N_p increases, the maximum deflection decreases, indicating smoother control behavior. A value of $N_p = 80$ offers an optimal trade-off, resulting in both reduced deflection and improved smoothness. The system output corresponding to different prediction horizons is shown in Fig. 12. For all values of N_p , the output signal starts at zero, increases to a peak, and then settles into a steady state. Notably, $N_p = 40$ achieves its peak faster than the other configurations. Pursuer acceleration, shown in Fig. 13, reveals further insights.

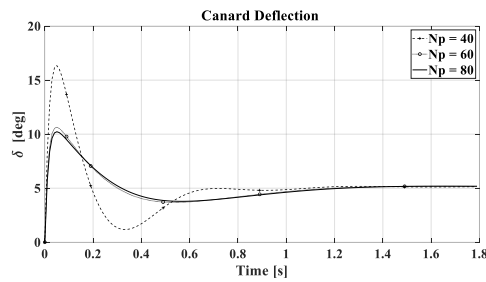


Fig. 11 Canard deflection N_p variation

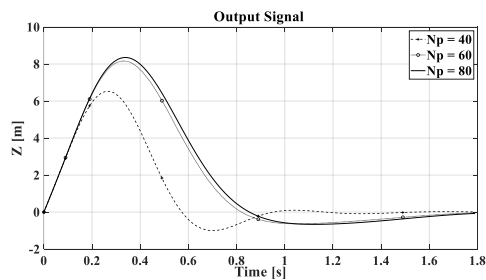


Fig. 12 Output signal N_p variation

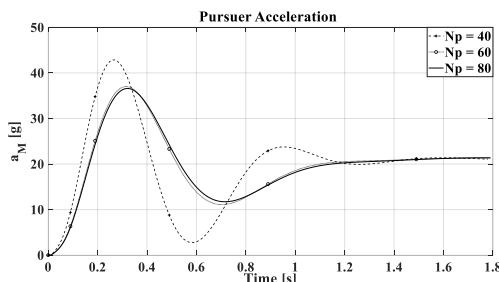


Fig. 13 Pursuer acceleration N_p variation

At $N_p = 40$, acceleration rises rapidly to a peak of approximately 43g, followed by oscillations before reaching steady-state. As N_p increases, the acceleration becomes more progressive and controlled. All tested horizons result in a miss distance of less than 1.5 meters. These observations suggest that $N_p = 80$ represents a suitable compromise between predictive capability and computational load, enabling effective control with manageable complexity.

2) Control horizon.

The effect of varying the control horizon $N_c = [2, 15, 30]$ on system output, canard deflection, and acceleration is also investigated. As shown in Fig. 14, for $N_c = 2$, the system output reaches a distinct peak before decaying to its final

value. For larger horizons $N_c = [15, 30]$, the output curves are nearly identical, with only minor differences in oscillatory behavior. $N_c = 30$ may show slightly more oscillations before settling. Fig. 15 illustrates the canard deflection profiles. For $N_c = 2$, the deflection is notably smoother and has a lower magnitude, indicating a more conservative control strategy. As the control horizon increases, deflection tends to fluctuate more, with $N_c = 30$ displaying the most pronounced oscillations. The pursuer's acceleration is presented in Fig. 16. A shorter control horizon, $N_c = 2$, leads to significantly lower acceleration and a more gradual deflection response, suggesting a smoother system behavior. In contrast, longer control horizons produce higher accelerations, implying more aggressive control actions.

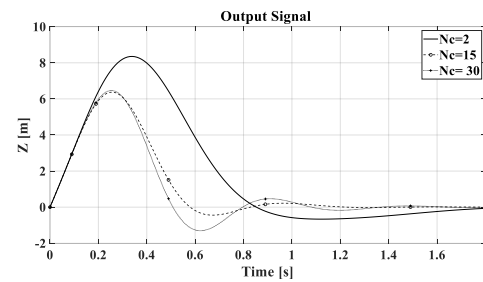


Fig. 14 Output signal N_c variation

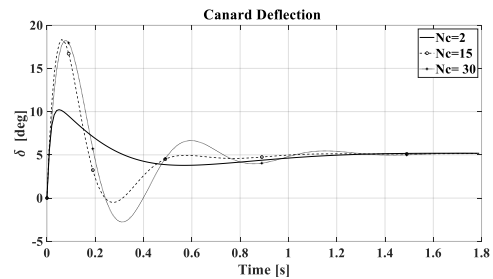


Fig. 15 Canard deflection N_c variation

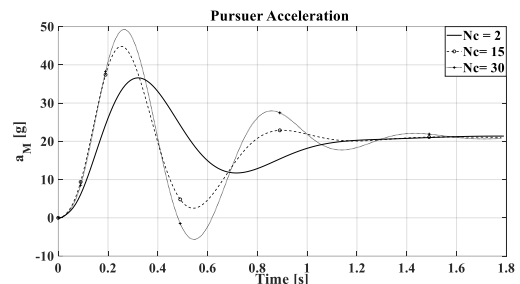


Fig. 16 Pursuer acceleration N_c variation

In summary, small prediction horizons might lead to more reactive but less stable control, while excessively large horizons might introduce computational burden without significantly improving control performance. A smaller control horizon results in smoother control signals, lower acceleration, and a distinctive system response with a peak followed by decay. Larger control horizons might lead to more oscillatory control signals, higher acceleration, and a more consistent or stable system response without distinct peaks. The choice of control horizon affects the aggressiveness and stability of control actions applied to the system. Stability is guaranteed for a range of $N_p = 80$. The role of $N_c = 2$ is a trade-off between computational effort and performance.

C. Processor in the loop experiment

Applying the same initial conditions and the parameters as presented in the first case, where the target has no acceleration, the "Processor-in-the-loop" (PIL) method is performed by deploying the algorithm onto a physical processor - specifically, the Arduino Due - to evaluate its real-time performance.

MATLAB Simulink is used to design the control model, and Simulink Coder is employed to automatically generate C code from the Simulink model (Fig. 17). This code is deployed directly to the Arduino Due using hardware support packages, allowing real-time execution of the proposed IGC algorithm.

The Arduino executes the algorithm in real time, with Fig. 18 illustrating the physical connection between Simulink and the Arduino board in the laboratory setup. A serial communication link is established between the hardware and the Simulink-based IGC model. In this implementation, the algorithm generates the deflection angle command necessary to guide the pursuer toward the target.

Fig. 19 compares the deflection angle generated in Simulink and that produced by the Arduino. The close match between the two curves confirms the successful implementation of the algorithm on the hardware.

The pursuer acceleration can be seen in Fig. 20, while the relative distance between pursuer and target can be seen in Fig. 21. This experiment demonstrates the feasibility and effectiveness of using Simulink in conjunction with Arduino hardware to implement and validate control algorithms in real time, offering a fast and reliable path toward practical deployment. The accuracy of the PIL implementation is checked by comparing the PIL acceleration output with Simulink. The RMSE is 0.18321, which is very small compared to the acceleration range $\min = -6.35$, $\max = 19.01$. This confirms that the PIL results closely match the simulation.

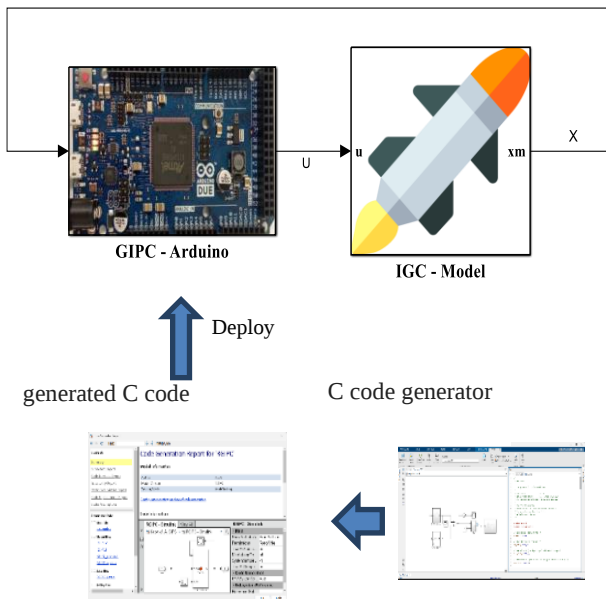


Fig. 17 The connections between the Simulink and Arduino board in the lab

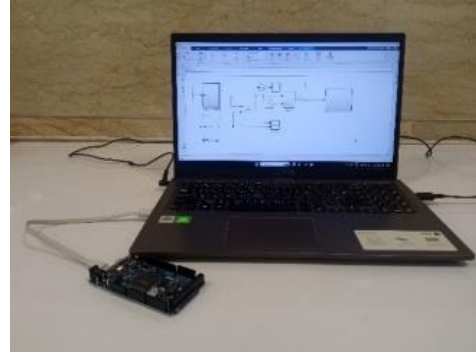


Fig. 18 The physical connection between Simulink and the board

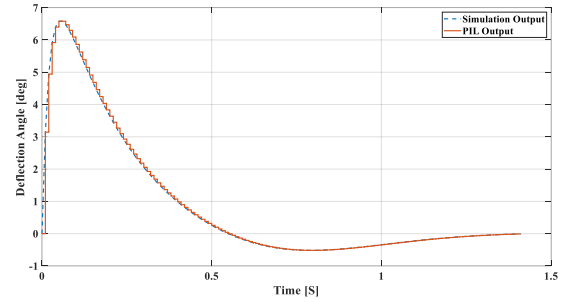


Fig. 19 Deflection angle

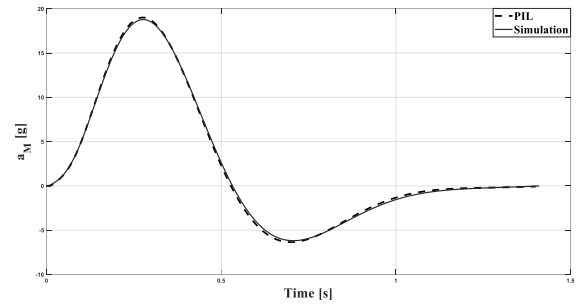


Fig. 20 Pursuer acceleration

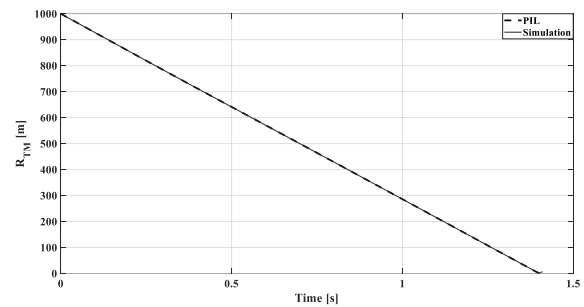


Fig. 21 Pursuer-Target relative distance

D. Altitude effects

Constant target acceleration is considered for better visualization to check the effects of altitude on the pursuit. It is known that at higher altitudes, air density decreases, which reduces aerodynamic forces. This phenomenon causes the pursuer to require larger deflection angles to generate the same maneuvering force. The control effort at sea level is $1083 \text{ m}^2/\text{s}^4$, while it becomes $1647 \text{ m}^2/\text{s}^4$ at 3000m above sea level. It is worth noticing that, for 3000m, the aerodynamic force and the moment are decreased by about 75% due to a change in air density.

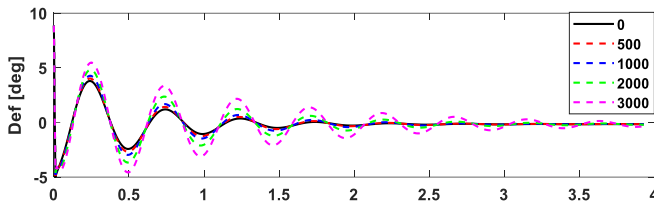


Fig. 22 Deflection angle with altitude variations

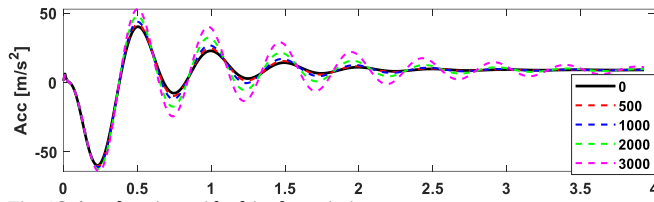


Fig. 23 Acceleration with altitude variations

VI. CONCLUSION

This study proposes GIPC-based IGC for linear systems using the deflection angle of the pursuer as a control action and the miss distance as an output. A quadratic optimization method was employed to solve the optimization problem. The first control input (deflection angle) was applied to the integrated system as a guidance command, and this process was repeated until interception with the target was achieved. It was shown that the GIP-GC had compensated for the target acceleration, which could be viewed as a disturbance to the system. The GIP-GC method was compared with the well-known Proportional Navigation and Robust Optimal Guidance approaches. Numerical simulations indicated that the GIP-GC method provides improved accuracy and acceleration performance compared to PN. Compared to GESO, the GIP-GC showed comparable tracking performance but required a smaller deflection angle to generate similar acceleration, indicating more efficient control behavior. A sensitivity analysis was conducted, leading to several insights. Notably, a trade-off was observed when selecting the prediction horizon. Shorter horizons resulted in more reactive but less stable control behavior. An optimal balance between predictive power, stability, and computational efficiency seems to be around $N_p = 80$. Similarly, a shorter control horizon leads to a smoother control signal, lower magnitude, and reduced acceleration, possibly indicating a more conservative or less aggressive control approach. So, N_c was chosen to be equal to 2. Furthermore, the asymptotic stability was proven. The PIL experiment showed that the proposed algorithm could work well in real-time conditions. It is important to acknowledge the limitations of the proposed method. The accuracy of the linearized model decreases in scenarios involving large line-of-sight rates, strong target maneuvers, or long engagement durations. The model must be extended for engagements dominated by such significant nonlinear effects to incorporate the full nonlinear kinematics. The IGC algorithm proposed in this work will be extended to deal with constraints in future research.

ACKNOWLEDGMENT

No specific funding has been provided for the research.

REFERENCES

- [1] P. K. Menon and E. J. Ohlmeyer, "Integrated design of agile missile guidance and autopilot systems," *Control Eng. Pract.*, vol. 9, no. 10, pp. 1095–1106, 2001, doi: [https://doi.org/10.1016/S0967-0661\(01\)00082-X](https://doi.org/10.1016/S0967-0661(01)00082-X).
- [2] M. Ebrahimi and S. Nasrollahi, "Fractional Guidance Law with Impact Angle Constraint and Seeker's Look Angle Limits," *Unmanned Syst.*, vol. 13, no. 01, pp. 261–277, Jan. 2025, doi: [10.1142/S2301385025500189](https://doi.org/10.1142/S2301385025500189).
- [3] D. WILLIAMS, J. RICHMAN, and B. FRIEDLAND, "Design of an integrated strapdown guidance and control system for a tactical missile," in *Guidance and Control Conference*, Reston, Virginia: American Institute of Aeronautics and Astronautics, Aug. 1983, p. 2169. doi: [10.2514/6.1983-2169](https://doi.org/10.2514/6.1983-2169).
- [4] S. Shamaghdari, S. K. Y. Nikraves, and M. Haeri, "Integrated guidance and control of elastic flight vehicle based on robust MPC," *Int. J. Robust Nonlinear Control*, vol. 25, no. 15, pp. 2608–2630, Oct. 2015, doi: [10.1002/rnc.3215](https://doi.org/10.1002/rnc.3215).
- [5] X. Hu, H. R. Karimi, L. Wu, and Y. Guo, "Model predictive control-based non-linear fault tolerant control for air-breathing hypersonic vehicles," *IET Control Theory Appl.*, vol. 8, no. 13, pp. 1147–1153, Sep. 2014, doi: [10.1049/iet-cta.2013.0986](https://doi.org/10.1049/iet-cta.2013.0986).
- [6] P. K. Menon, G. D. Sweriduk, E. J. Ohlmeyer, and D. S. Malyevac, "Integrated Guidance and Control of Moving-Mass Actuated Kinetic Warheads," *J. Guid. Control. Dyn.*, vol. 27, no. 1, pp. 118–126, Jan. 2004, doi: [10.2514/1.9336](https://doi.org/10.2514/1.9336).
- [7] F. Liao, K. Yang, and H. Ji, "Adaptive integrated guidance and control with actuator failures based on backstepping and input-to-state stability," in *2014 IEEE Chinese Guidance, Navigation and Control Conference, CGNCC 2014*, IEEE, Aug. 2015, pp. 49–54. doi: [10.1109/CGNCC.2014.7007218](https://doi.org/10.1109/CGNCC.2014.7007218).
- [8] A. Ashrafifar and M. F. Jegarkandi, "Adaptive fin failures tolerant integrated guidance and control based on backstepping sliding mode," *Trans. Inst. Meas. Control*, vol. 42, no. 10, pp. 1823–1833, Jun. 2020, doi: [10.1177/0142331219897430](https://doi.org/10.1177/0142331219897430).
- [9] S. Xingling and W. Honglun, "Back-stepping active disturbance rejection control design for integrated missile guidance and control system via reduced-order ESO," *ISA Trans.*, vol. 57, pp. 10–22, Jul. 2015, doi: [10.1016/j.isatra.2015.02.013](https://doi.org/10.1016/j.isatra.2015.02.013).
- [10] C. F. Lin, Q. Wang, J. L. Spayer, J. H. Evers, and J. R. Cloutier, "Integrated Estimation, Guidance, and Control System Design Using Game Theoretic Approach," in *1992 American Control Conference*, IEEE, Jun. 1992, pp. 3220–3224. doi: [10.23919/ACC.1992.4792744](https://doi.org/10.23919/ACC.1992.4792744).
- [11] M. Levy, T. Shima, and S. Gutman, "Full-state autopilot-guidance design under a linear quadratic differential game formulation," *Control Eng. Pract.*, vol. 75, no. 3, pp. 98–107, Jun. 2018, doi: [10.1016/j.conengprac.2018.03.009](https://doi.org/10.1016/j.conengprac.2018.03.009).
- [12] T. Shima, M. Idan, and O. M. Golan, "Sliding-mode control for integrated missile autopilot guidance," *J. Guid. Control. Dyn.*, vol. 29, no. 2, pp. 250–260, Mar. 2006, doi: [10.2514/1.14951](https://doi.org/10.2514/1.14951).
- [13] Z. Guo, J. Guo, X. Wang, J. Chang, and H. Huang, "Sliding mode control for systems subjected to unmatched disturbances/unknown control direction and its application," *Int. J. Robust Nonlinear Control*, vol. 31, no. 4, pp. 1303–1323, Mar. 2021, doi: [10.1002/rnc.5336](https://doi.org/10.1002/rnc.5336).
- [14] J. Guo, N. Lu, R. Jiang, and Z. Guo, "Novel Explicit Reference Governor-Based Adaptive Terminal Sliding Mode Control Design for Reentry Vehicles Equipped with Strapdown Seeker," *Int. J. Aeronaut. Sp. Sci.*, vol. 25, no. 1, pp. 200–212, Jan. 2024, doi: [10.1007/s42405-023-00647-8](https://doi.org/10.1007/s42405-023-00647-8).
- [15] T. Hughes and M. McFarland, "Integrated missile guidance law and autopilot design using linear optimal control," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Reston, Virginia: American Institute of Aeronautics and Astronautics, Aug. 2000, p. 4163. doi: [10.2514/6.2000-4163](https://doi.org/10.2514/6.2000-4163).
- [16] C.-F. Lin, J. Bibel, E. Ohlmeyer, and S. Malyevac, "Optimal design of integrated missile guidance and control," in *AIAA and SAE, 1998 World Aviation Conference*, Reston, Virginia: American Institute of Aeronautics and Astronautics, Sep. 1998, p. 5519. doi: [10.2514/6.1998-5519](https://doi.org/10.2514/6.1998-5519).
- [17] J. Yun and C.-K. Ryoo, "Integrated guidance and control law with impact angle constraint," in *2011 11th International Conference on Control, Automation and Systems*, IEEE, 2011, pp. 1239–1243.
- [18] M. Levy, T. Shima, and S. Gutman, "Linear Quadratic Integrated Versus Separated Autopilot-Guidance Design," *J. Guid. Control. Dyn.*, vol. 36, no. 6, pp. 1722–1730, Nov. 2013, doi: [10.2514/1.61363](https://doi.org/10.2514/1.61363).

- [19] R. Sheikhabaei and S. Khankalantary, "Three-dimensional continuous-time integrated guidance and control design using model predictive control," *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.*, vol. 237, no. 3, pp. 503–515, Mar. 2023, doi: 10.1177/09544100221103320.
- [20] D. Bhattacharjee, S. Jaeger, and M. Hemati, "Integrated Guidance and Control of Quasi-Equilibrium Hypersonic Gliding Using Model Predictive Control," in *AIAA AVIATION FORUM AND ASCEND 2025*, Reston, Virginia: American Institute of Aeronautics and Astronautics, Jul. 2025, p. 3550. doi: 10.2514/6.2025-3550.
- [21] M. M. Soori and S. H. Sadati, "Online model predictive integrated control and guidance to intercept maneuvering targets," *AUT J. Mech. Eng.*, 2025, doi: 10.22060/ajme.2025.23808.6156.
- [22] R. Chai, A. Savvaris, and S. Chai, "Integrated missile guidance and control using optimization-based predictive control," *Nonlinear Dyn.*, vol. 96, no. 2, pp. 997–1015, Apr. 2019, doi: 10.1007/s11071-019-04835-8.
- [23] S. Lee, H. Lee, Y. Kim, J. Kim, and W. Choi, "GPU-Accelerated PD-IPM for Real-Time Model Predictive Control in Integrated Missile Guidance and Control Systems," *Sensors*, vol. 22, no. 12, p. 4512, Jun. 2022, doi: 10.3390/s22124512.
- [24] J. Guo, Y. Li, and J. Zhou, "An observer-based continuous adaptive sliding mode guidance against chattering for homing missiles," *Trans. Inst. Meas. Control*, vol. 41, no. 12, pp. 3309–3320, 2019.
- [25] N. Mate, B. Panchal, and S. E. Talole, "GESO based robust optimal guidance," in *2015 International Conference on Industrial Instrumentation and Control (ICIC)*, IEEE, May 2015, pp. 187–192. doi: 10.1109/IIC.2015.7150735.
- [26] B. Panchal, N. Mate, and S. E. Talole, "Continuous-time predictive control-based integrated guidance and control," *J. Guid. Control. Dyn.*, vol. 40, no. 7, pp. 1579–1595, 2017, doi: 10.2514/1.G002661.
- [27] A. Zhurbal and M. Idan, "Effect of Estimation on the Performance of an Integrated Missile Guidance and Control System," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 47, no. 4, pp. 2690–2708, 2011, doi: 10.1109/TAES.2011.6034659.
- [28] A. Sinha, S. R. Kumar, and D. Mukherjee, "Impact time constrained integrated guidance and control design," *Aerosp. Sci. Technol.*, vol. 115, p. 106824, Aug. 2021, doi: 10.1016/j.ast.2021.106824.
- [29] N. O. Ghahramani and F. Towhidkhah, "Constrained incremental predictive controller design for a flexible joint robot," *ISA Trans.*, vol. 48, no. 3, pp. 321–326, 2009, doi: 10.1016/j.isatra.2009.01.010.
- [30] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB®*. London: Springer London, 2009. doi: 10.1007/978-1-84882-331-0.
- [31] A. Koren, M. Idan, and O. M. Golan, "Integrated Sliding Mode Guidance and Control for Missile with On-Off Actuators," *J. Guid. Control. Dyn.*, vol. 31, no. 1, pp. 204–214, Jan. 2008, doi: 10.2514/1.31328.
- [32] J. Park, Y. Kim, and J.-H. Kim, "Integrated Guidance and Control Using Model Predictive Control with Flight Path Angle Prediction against Pull-Up Maneuvering Target," *Sensors*, vol. 20, no. 11, p. 3143, Jun. 2020, doi: 10.3390/s20113143.