Defence Science Journal, Vol. 73, No. 5, September 2023, pp. 602-611, DOI : 10.14429/dsj.73.5.18481

Hybrid Deep Neural Network for Data-Driven Missile Guidance with Maneuvering Target

Junaid Farooq* and Mohammad Abid Bazaz

Electrical Engineering Department, National Institute of Technology Srinagar, Srinagar – 190 006, India

**E-mail: junaid_phd017@nitsri.ac.in*

ABSTRACT

Missile guidance, owing to highly complex and non-linear relative movement between the missile and its target, is a challenging problem. This is further aggravated in case of a maneuvering target which changes its own flight path while attempting to escape the incoming missile. In this study, to achieve computationally superior and accurate missile guidance, deep learning is employed to propose a self-tuning technique for a Fractional-Order Proportional Integral Derivative (FOPID) controller of a radar-guided missile chasing an intelligently maneuvering target. A multi-layer two-dimensional architecture is proposed for a deep neural network that combines the prediction feature of recurrent neural networks and estimation feature of feed-forward artificial neural networks. The proposed deep learning based missile guidance scheme is non-intrusive, data-based, and model-free wherein the parameters are optimized on-the-run while predicting the target's maneuvering tactics to correct for processing time and loop delays of the system. Using deep learning for online optimisation with minimal computational burden is the core feature of the proposed technique. Dual-core parallel simulations of missile-target dynamics and the control system were performed to demonstrate superiority of the proposed scheme in feasibility, adaptability, and the ability to effectively minimize the miss-distance in comparison with traditional and neural offline-tuned PID and FOPID based techniques. Compared to state-of-the-art offline-tuned neural control, the miss-distance was reduced by 68.42 % for randomly maneuvering targets. Furthermore, a minimum miss-distance of 0.97 m was achieved for intelligently maneuvering targets for which the state-of-the-art method failed to hit the target. Overall, the proposed technique offers a novel approach for addressing the challenges of missile guidance in a computationally efficient and effective manner.

Keywords: Deep learning; FOPID; Missile; Guidance; Neural network

NOMENCLATURE

V_M : Missile velocity
V_T : Target velocity
V_r : Relative target velocity w.r.t the missile along line-of-sight (LOS)
V_θ : Relative target velocity w.r.t the missile in the direction perpendicular to LOS
a_M : Latex

1. INTRODUCTION

The problem of guidance control is as old as the prehistoric concept of arrows and spears. In context of the modern developments, the control of guided missile is a complex problem as it involves an unmanned aerial vehicle with highly nonlinear dynamics, intelligently and accurately guiding itself towards a moving and possibly maneuvering target. The fast moving missile is fired in the approximate direction of a moving target while the goal of controller is to generate guidance system steering commands to provide lateral acceleration to the vehicle body to manipulate its flight path to hit the target with minimum miss-distance. The control of guided missile is a challenging problem owing to the highly complex and non-linear relative movement between the missile and its target which is further aggravated if the target employs maneuvering tactics to change its own flight path to escape an incoming missile. This paper proposes a deep-learning based flexible, adaptive, and predictive guidance architecture for radar-guided surface-to-air cruise missile chasing a maneuvering target.

The missile attempts to accurately hit the target. However, that is rarely possible as there is always a miss-distance between the target and the point where the missile bursts[1]. The goal of missile control is to minimize the miss-distance to avoid hitting the wrong target and ensure that maximum damage is inflicted on the intended target. Several techniques have been proposed to improve the miss-distance performance of guided missiles. Coupling between altitude control and coordinator stable tracking was proposed[2]. A predictive control strategy for linearised models of the missile was also proposed[3]. In addition, an observer-based method for adaptive nonlinear guidance while considering target uncertainties was proposed[4]. Nonlinear adaptive guidance control methods have also been proposed with compensation for both control loop dynamics and target acceleration[5-6]. An integrated control and guidance system using sliding mode algorithm was suggested using adaptive fuzzy-neural network[7]. An analytic method for solving the distribution of miss distance has been proposed,

Received : 28 September 2022, Revised : 14 March 2023
Accepted : 12 May 2023, Online published : 31 August 2023

in which the system is presumed to use a bang-bang control strategy[8]. However, the control of guided missile still remains challenging especially in case of an intelligently maneuvering target due to the highly nonlinear dynamics and stochastic nature of the problem.

Deep learning (DL) has emerged as a leading candidate for enabling scalable and data-driven neural network architectures to deal with highly nonlinear dynamics. Many DL based control techniques have been used to deal with the problem of missile guidance and control[9-11]. The work of Yaghi and Onder[12] is a significant contribution towards this area. The authors have shown with the help of simulations that fractional order proportional integral derivative (FOPID) based control scheme is superior to the traditional PID controller for radar-guided missile. They have used offline neural tuning for parameter optimization of the FOPID controller with further fine tuning by using H_2/H_∞ optimization method. However, as shown in this paper, the method proposed by Yaghi and Onder[12] has limited performance if the target moves in a direction that is significantly different from the one for which the system has been initially optimized or if the target employs maneuvering tactics to escape the incoming missile.

This study proposes a DL based adaptive predictive guidance (DLAPG) technique as a non-intrusive, online, on-the-run optimization algorithm for adaptive optimization of controller variables by using a multi-layered, two-dimensional deep neural network (DNN). DNNs have immense potential for auto-tuning controller variables.[13] Utilizing the DNN characteristics of latent inference, adaptive learning, and time-series forecasting, this study's algorithm design outperforms existing strategies. A FOPID controller is integrated into the proportional navigation (PN) system of the guided missile. Genetic algorithm (GA) is used to generate the training data. The suggested method is universal and may be utilized for online self-tuning of any controller variables. However, to compare with the work of Yaghi and Onder[12], it has been applied to FOPID. The main contributions of this work are:

- The first integration of deep learning architecture with genetic algorithm for self-tuning of an FOPID controller
- Implementation of an innovative deep learning architecture that integrates prediction feature of Recurrent Neural Network (RNN) and estimation property of feed-forward Artificial Neural Network (ANN)
- Model-free, optimized, data-based, adaptive and predictive control of radar-guided missile is achieved wherein the controller parameters are intelligently updated according to the maneuvering tactics of the intended target with minimization of miss-distance as the physically realizable objective function
- The algorithm is able to predict the target's maneuvering tactics in terms of change in its flight direction with prediction horizon equal to the algorithm run-time and other system delays of intrinsic nature to compensate for the target's maneuvering movements that take place during the time-lag in real-life conditions
- As an online optimization technique, although significant computational resources are required during training, it imposes minimal computational burden during deployment. Hence, it is suitable for time-sensitive applications like missile guidance.

Therefore, the current work integrates state-of-art artificial intelligence and automatic control algorithms to achieve an intelligent and adaptive self-tuning system for radar-guided missile.

2. MISSILE DYNAMICS

2.1 Proportional Navigation System

As shown in Fig. 1, the missile dynamics are governed by the following Eqns:

$$V_r = \dot{R} = V_T\cos(\alpha_T - \theta) - V_M\cos(\alpha_M - \theta) \tag{1}$$

$$V_\theta = R\dot{\theta} = V_T\sin(\alpha_T - \theta) - V_M\sin(\alpha_M - \theta) \tag{2}$$

where, V_r is the target velocity with respect to the missile along line-of-sight (LOS), V_θ is the target velocity w.r.t the missile in the direction perpendicular to LOS, and R is the LOS vector. θ is the angle between R and horizontal reference. V_M and V_T are missile and target velocities, respectively. PN guidance law in its simple form defines a lateral acceleration guidance command (latex) such that the rate of rotation of missile velocity vector is proportional to the rate of rotation of the LOS:

$$\dot{\alpha}_M = N\dot{\theta} \tag{3}$$

where, N is the navigation constant. In its more advanced form, PN law is called as Pure Proportional Navigation *(PNN)* given by:

$$\dot{\alpha}_M = \frac{a_M}{V_M} \tag{4}$$

where, a_M is the latex. From (3) and (4) we get,

$$a_M = NV_M\dot{\theta} \tag{5}$$

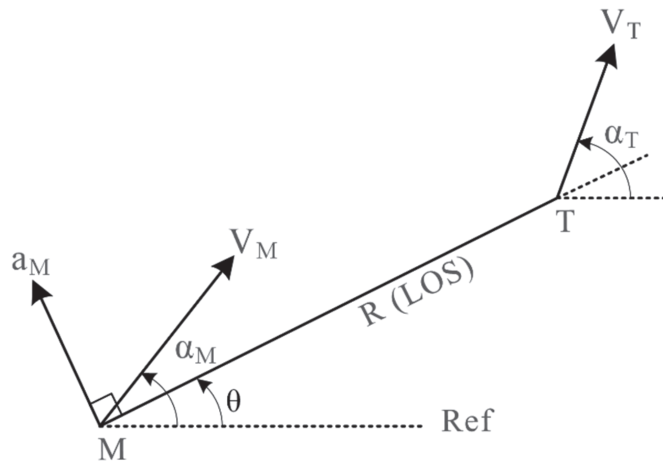

Figure 1. Geometry of missile-target engagement

Yaghi & Onder[12] used the same PNN guidance law. However, Eqn. (4) is valid only under the assumption that the latex is applied perpendicular to the missile velocity. Although this direction is the most natural direction of the lift force generated by airframe and lifting surfaces of the vehicle to manipulate its flight path; however, it is not realistic as it ignores the limitations posed by the angle-of-attack

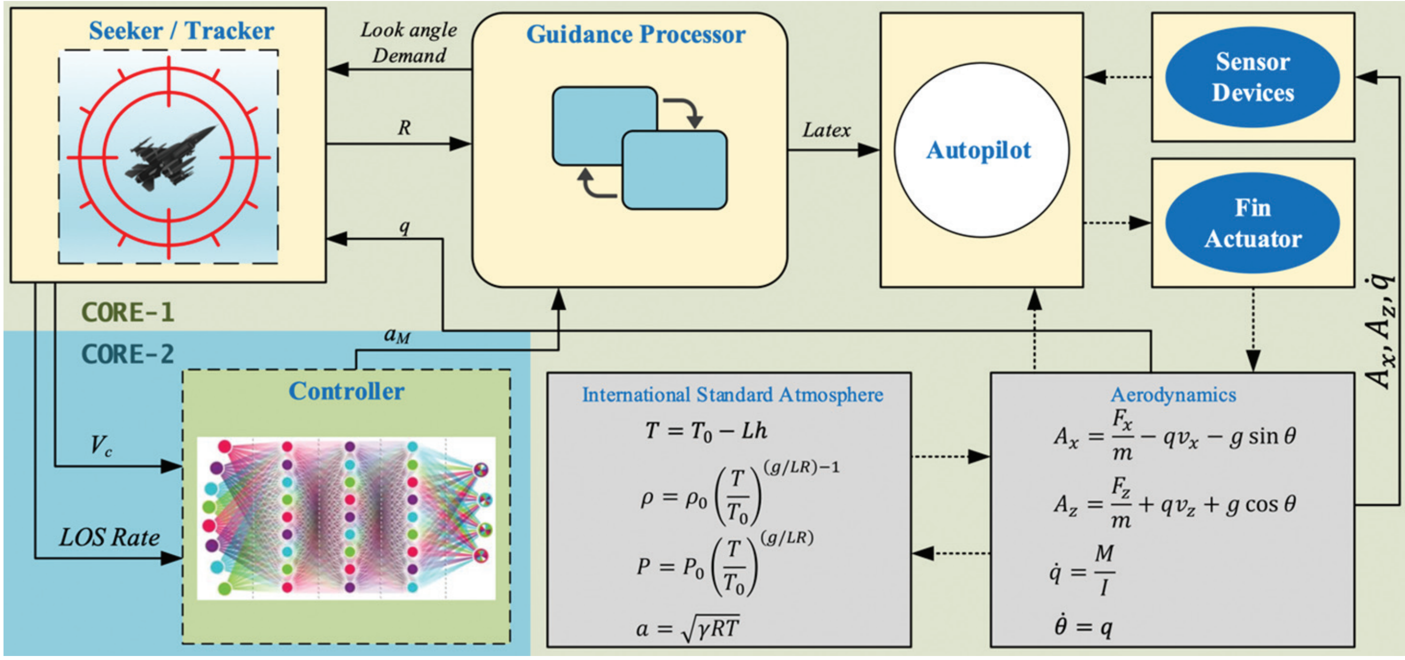

Figure 2. System model.

(AOA) which is never zero. AOA is quite high, especially for maneuverable missile-target systems. Therefore, this work uses True Proportional Navigation (TPN) in which latex is applied proportional and perpendicular to the closing velocity (V_c) between the missile and the target (which is the direction of LOS). Thus,

$$a_M = N' V_C \dot{\theta} \tag{6}$$

where, $N' = NV_M / V_C$ is the effective navigation ratio. Another advantage of TPN over PNN is that V_M is not readily available unless an inertial navigation unit is embedded in the missile system, but V_c can be found directly from the seeker's doppler data.

Eqn. (6) can be considered as a simple proportional controller. The current study and the work of Yaghi and Onder[12] implement a FOPID controller in place of the proportional controller due to its various advantages as discussed in Section 3.

2.2 System Modeling

The simulation model used in this work represents a tail-controlled radar-guided cruise missile with Mach 2-4 speed, 3,050 m (10,000 ft) to 18,290 m (60,000 ft) altitude and -20° to $+20^\circ$ AOA.

A nonlinear model of airframe rigid body dynamics are considered as shown in Fig. 2. The change in atmospheric environment with changing altitude is modeled by atmosphere subsystem according to International Standard Atmosphere (ISA)[14] described by the following equations:

$$T = T_0 - Lh \tag{7}$$

$$\rho = \rho_0 \left(\frac{T}{T_0}\right)^{(g/LR)-1} \tag{8}$$

$$P = P_0 \left(\frac{T}{T_0}\right)^{(g/LR)} \tag{9}$$

$$b = \sqrt{\gamma RT} \tag{10}$$

where h and L are missile altitude (m) and lapse rate (K/m) respectively. T and T_0 represent the absolute temperature at altitude h and at mean sea level, ρ and ρ_0 represent air density (Kg/m^3) at h and at mean sea level, P and P_0 represent the air pressure at h and at mean sea level respectively. b, R, and g are speed of sound at h, characteristic gas constant (J/Kg/K), and acceleration due to gravity (m/s^2) respectively. Aerodynamics subsystem models the equations defining missile trajectory based on calculation of all forces and moments acting on the vehicle body [12] as follows:

$$A_x = \frac{F_x}{m} - qv_x - g\sin\theta \tag{11}$$

$$A_z = \frac{F_z}{m} + qv_z + g\cos\theta \tag{12}$$

$$\dot{q} = \frac{M}{I} \tag{13}$$

$$\dot{\theta} = q \tag{14}$$

where, A_x and A_z are the horizontal and vertical components of missile acceleration, respectively. q and m represent the rate of rotation of missile body and total missile mass, respectively. I represents the missile inertia, v_x and v_z are the horizontal and vertical components of missile velocity, respectively. M represents the pitch moment of the missile. The forces acting on the two axes are given by the following Eqns:

$$F_x = C_x\left(0.5\rho V_M^2 S_{ref}\right) \tag{15}$$

$$F_z = C_z\left(0.5\rho V_M^2 S_{ref}\right) \tag{16}$$

$$M = C_m\left(0.5\rho V_M^2 S_{ref} D_{ref}\right) \tag{17}$$

where, S_{ref} is the reference cross-sectional area of the missile, D_{ref} is the diameter of the missile's reference circular body, V_M is the missile speed, C_x and C_z are constants that depend on speed and AOA.

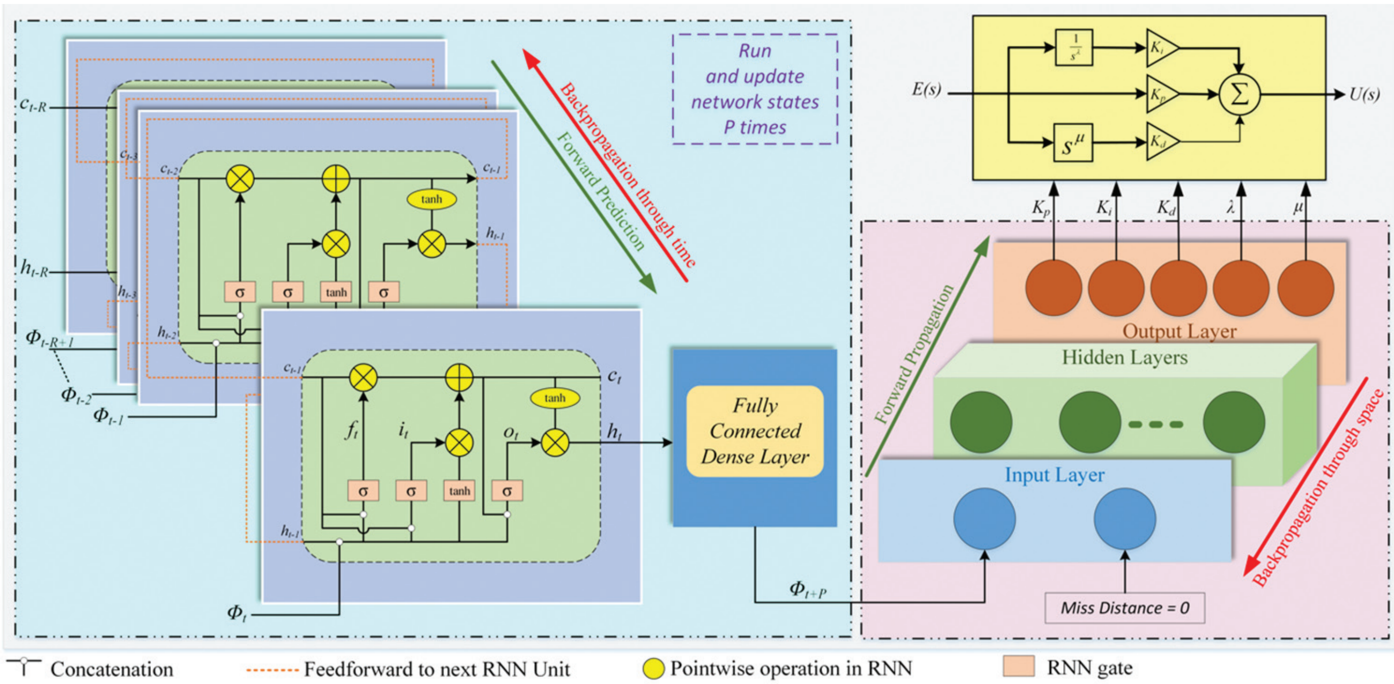

Figure 3. DNN architecture and FOPID control scheme.

Autopilot subsystem models a three-loop controller[15] which governs normal acceleration. Sensor subsystem containing rate gyro and accelerometer along with the fin actuator subsystem provide the coupling between the autopilot system and airframe. A homing guidance system[16] is modeled consisting of seeker/tracker subsystem which drives the gimbals and estimates the sightline rate; and guidance subsystem which implements the TPN guidance. Compensation for radome aberrations is considered in seeker/tracker modeling. To ensure comparison with the work of Yaghi and Onder[12], we reproduced their test system without modifying any missile specifications, which were taken from previous papers[16-18] as shown in Fig. 2.

3. PROPOSED SCHEME

3.1 FOPID

FOPID controllers are described by fractional-order integro-differential equations[20] as follows:

$$u(t) = K_p e(t) + K_i I^{-\lambda} e(t) + K_d I^{\mu} e(t) \tag{18}$$

where, λ and μ are positive real. Applying Laplace transform on (18) gives:

$$C(s) = K_p + K_i s^{-\lambda} + K_d s^{\mu} \tag{19}$$

Thus, the FOPID controller is a generalized integer-order PID controller with added flexibility, robustness and better adjustment capability[21-22]. A FOPID controller has two extra parameters compared to a PID controller which provide extra degrees of freedom to the system dynamics and makes it less sensitive[23]. In general, FOPID controller is known to outperform PID controller[24-25]. The superiority of FOPID-based guidance law over the proportional navigation and proportional–integral–differential navigation guidance laws has also been established[26].

Various approaches have been offered in the literature to simplify the realization of real order fractal elements that include the approximations given by Oustaloup[27], Khoichi[28], AbdelAty[29], and El-Khazali[30]. Yaghi and Onder[12] used Oustaloup's approximation which is popular for customizable bandwidth. However, the current study implements El-Khazali's integro-differential approximation using a biquadratic algorithm owing to the latter's straightforward realization of fractal elements which depends only on the order of differentiation or integration. Further, El-Khazali's approximation gives a better frequency response with center frequency as flat phase response under narrower bandwidth compared to Oustaloup's approximation, with improved steady-state response under higher orders of approximation.[31] El-Khazali's approximation also has smaller parameter values requiring less expensive circuit design in hardware implementation.

We implemented cascaded multiple 2nd-order biquadratic transfer functions as follows:

$$\left(\frac{s}{\omega_g}\right) = \prod_{i=1}^{n} H_i\left(\frac{s}{\omega_i}\right) = \prod_{i=1}^{n} \frac{N_i\left(\frac{s}{\omega_i/\omega_g}\right)}{D_i\left(\frac{s}{\omega_i/\omega_g}\right)} \tag{20}$$

where, ω_i is the center frequency of i^{th} biquadratic module. $\omega_g = \sqrt[n]{\prod_{i=1}^{n}\omega_i}$ is the corresponding geometric mean. Supposing ω_1 to be the first section's first center frequency, a constant-phase element is obtained by calculating the subsequent frequencies using the recursive formulation as follows:

$$\omega_i = \omega_x^{2(i-1)}\omega_1; \quad i = 2,3,\ldots,n \tag{21}$$

where, ω_x is the maximum real solution for the polynomial shown as follows:

$$a_0 a_2 \eta\gamma^4 + a_1\left(a_2 - a_0\right)\gamma^3 + \left(a_1^2 - a_2^2 - a_0^2\right)\eta\gamma^2 + a_1\left(a_2 - a_0\right)\gamma + a_0 a_2 \eta = 0 \tag{22}$$

```
1. Initialize the values of Kp,Ki,λ,Kd,μ as 0.9434, 0.0133, 0.0421,
   0.1122, and 0.278, respectively.
2. Define the lower bound (LB) and upper bound (UB) of [Kp,Ki,λ,Kd,μ] as
   [0.5, 0.005, 0.05, 0.01, 0.1] and [5, 0.5, 0.2, 0.1, 0.5],
   respectively.
3. Define the fitness function (FF) of GA as miss-distance output of
   the Simulink model (SM).
4. for φ = 1 to 360 with an increment of 1, do:
       [Kp,Ki,Kd,λ,μ] = GA(SM, FF, LB, UB)
       % Save all the values of the iteration, not only the final
       optimal values.
   end;
```

Pseudo-code.

where, $\eta = \tan(a\pi/4)$. Thus, a biquadratic module is represented by:

$$H_i\left(\frac{s}{\omega_i}\right) = \frac{N_i\left(\frac{s}{\omega_i}\right)}{D_i\left(\frac{s}{\omega_i}\right)} \cong \frac{a_0\left(\frac{s}{\omega_i}\right)^2 + a_1\left(\frac{s}{\omega_i}\right) + a_2}{a_2\left(\frac{s}{\omega_i}\right)^2 + a_1\left(\frac{s}{\omega_i}\right) + a_0} \tag{23}$$

where,

$a_0 = \alpha^\alpha + 2\alpha + 1$,

$a_2 = \alpha^\alpha - 2\alpha + 1$, and

$$a_1 = \left(a_2 - a_0\right) tan\left(\frac{(2+\alpha)\pi}{4}\right) = -4\alpha\tan\left(\frac{(2+\alpha)\pi}{4}\right) \tag{24}$$

3.2 DNN for Self-Tuning of Controller Parameters

This section deals with the problem of online self-tuning of the five parameters (K_p, K_i, K_d, λ, μ) of the FOPID controller using a deep learning approach. The missile-target system under consideration is highly non-linear and stochastic. Therefore, fixed offline tuning of controller parameters cannot be expected to perform well as shown in section 4. Furthermore, a maneuvering target can manipulate its flight path by changing the direction of motion to escape the incoming missile. Therefore, the missile controller parameters must be tuned using an intelligent and adaptive approach in accordance with the target's current direction of motion and anticipated future direction as well.

3.2.1 Generation of Training Data

The simplest method for generating training data is to vary the five controller parameters and compute miss-distance. The training data thus obtained reflects miss-distance for various combinations of controller parameters, enabling a neural network to estimate miss-distance for different parameter values. Augmenting the training data can enhance network accuracy. However, the data generated via this method would be dispersed across a range of miss-distance values, with only few data points representing optimal values. To concentrate training data around optimal values where miss-distance approaches zero, we propose using genetic algorithm (GA). As an evolutionary optimization technique, GA generates input-output pairs where most correspond to near-zero miss-distance for different controller parameter values.

Thus, GA is used to generate the training data. Owing to the offline nature of the data generation process, it is not subjected to any time constraints or computational limitations. All possible target flight directions are considered with an increment of 1° with minimization of the miss-distance as the physically realizable objective function. Multiple numerical experiments were performed for different values of the increment. The value of 1° was selected based on the observation that further reduction in the interval value beyond 1° produced only marginal improvements in the training performance at the cost of training time. Hence, GA is executed 360 times by executing the Simulink model of the system while saving the miss-distance, relative target flight direction, and FOPID controller variables produced during each iteration to generate training data. In each iteration of GA, a large number of input-output pairs are generated and all of them, not just the final optimal values, are stored. Owing to the evolutionary nature of GA, the number of data points generated in each run varies according to multiple factors including the miss-distance tolerance. In this study, we set a threshold of tolerance of 0.1 m.

The Matlab code for the GA executes the Simulink model and outputs the miss-distance corresponding to different combinations of the controller parameters and target flight directions, which evolve in the direction of optimal values. The pseudo-code for this process is mentioned.

3.2.2 Neural Network Architecture

To reduce the miss-distance, the DNN uses the training data to estimate the missile-target system dynamics and adaptively changes the controller settings according to the target's maneuvering tactics. However, owing to the time gap between the initial sensing of variables and the compilation of controller settings, the parameter tuning procedure is susceptible to errors. The delay is caused by the processing times of *(a)* the neural network itself, *(b)* control loop dynamics, and *(c)* sensors and other electronic components. The controller parameters determined at time *t* are actually tuned for the target flight direction and other system variables at a previous time $t-T_d$, where T_d represents the time lag. The current direction of motion of the target may vary from what it was at time $t - T_d$.

To address this issue, the proposed architecture of DNN, as shown in Fig. 3, has two dimensions: space (estimation), and time (prediction). The *space* dimension models the nonlinear relationship of the target direction, controller variables, and miss-distance. It is built on a deep feedforward neural network with three hidden layers of 50, 30, and 20 neurons, and supposes the following dynamics:

$$D_m = f_{NL}\ (\phi,\ K_p,\ K_i,\ K_d,\ \lambda,\ \mu) \tag{25}$$

where, D_m is the final miss-distance when the missile closes upon the target and f_{NL} is estimated by the space dimension as a highly nonlinear function. ϕ is the target direction.

$$\begin{bmatrix} K_p \\ K_i \\ K_d \\ \lambda \\ \mu \end{bmatrix} = \begin{bmatrix} g_{1,NL}(\phi, D_m) \\ g_{2,NL}(\phi, D_m) \\ g_{3,NL}(\phi, D_m) \\ g_{4,NL}(\phi, D_m) \\ g_{5,NL}(\phi, D_m) \end{bmatrix} = g_{NL}(\phi, D_m) \tag{26}$$

From Eqn. 26, where, g_{NL} is vector function of highly nonlinear nature. As shown in Fig. 3, $D_m \approx 0$ and the real-time forecasted ϕ are fed to the space dimension with FOPID controller parameters as outputs for corresponding target direction and zero miss-distance.

Thus, g_{NL} on-the-run updates the controller variables by implementing a simple feedforward neural network with minimal computational burden. The weights and biases of this dimension are tuned offline by applying backpropagation learning based on Levenberg-Marquardt algorithm on the training data as follows:

$$\text{Input} = \begin{bmatrix} D_m & \phi \end{bmatrix} \tag{27}$$

$$\text{Output} = \begin{bmatrix} K_p & K_i & K_d & \lambda & \mu \end{bmatrix} \tag{28}$$

$$\omega_{ij}^{[l]}(k+1) = \omega_{ij}^{[l]}(k) - \eta \frac{\partial J}{\partial \omega_{ij}^{[l]}} \tag{29}$$

$$v_i^{[l]}(k+1) = v_i^{[l]}(k) - \eta \frac{\partial J}{\partial v_i^{[l]}} \tag{30}$$

where, $\omega_{ij}^{[l]}$ represents the weight connecting layer l's *ith* neuron with *jth* neuron of layer $l-1$ and v represents *lth* layer's *ith* neuron's bias. η and k represent the learning rate and the iteration instant, respectively. The objective function J is optimised as follows:

$$\min \quad J = \frac{1}{2}[D_m(k) - \widehat{D}_m(k)]^2 \tag{31}$$

To make up for the overall delay caused by system processing, the time or prediction dimension forecasts the target's flight path relative to the missile. It accomplishes this by extracting patterns from the target's flight trajectory and using those patterns to anticipate future directions with the total system delay equal to the prediction horizon. This dimension has multiple RNN layers based on the LSTM network[32] featuring online learning capability using accurate time-series forecasting. We updated the LSTM network by including peephole connections that produce nonlinear and precisely timed spikes without sacrificing performance and while maintaining stability[33]. There are three sublayers for each network, which are:

$$f_t = \sigma(W_f \cdot [c_{t-1}, h_{t-1}, \phi] + b_f) \quad : \quad \text{forget gate} \tag{32}$$

$$i_t = \sigma(W_i \cdot [c_{t-1}, h_{t-1}, \phi] + b_i) \quad : \quad \text{input gate} \tag{33}$$

$$o_t = \sigma(W_o \cdot [c_{t-1}, h_{t-1}, \phi] + b_o) \quad : \quad \text{output gate} \tag{34}$$

where, c_t and h_t are the cell state and the prediction at time t, respectively. The network output is as follows:

$$c_t = f_t * c_{t-1} + i_t * (\tanh(W_c \cdot [h_{t-1}, x_t] + b_c)) \tag{35}$$

$$h_t = o_t * \tanh(c_t) \tag{36}$$

A sigmoid function is used by the input gate in (33) to choose fresh values to be saved in the cell state. In (35), the cell state undergoes change by multiplication of the previous state by a forget gate that only allows the necessary data to be sent forward. The information supplied by the input gate is then processed using a *tanh* layer. The data supplied by the input gate is then processed using another *tanh* layer. The updated cell state comes across another *tanh* layer in equation (36) to guarantee that the values fall in [-1,1] range. Finally, multiplication by the output gate provides the final forecasts.

A 200 layer network was used that outputs the forecasted values of ϕ_{t+P} based on the sequence of past 200 values of ϕ as inputs to correct for P instants by running it P times while updating the network states at each iteration. P is the prediction horizon of the network, which is set approximately equal to the run-time of the space dimension.

The online backpropagation through time (BPTT) procedure was applied for updating the network states to prevent vanishing gradients. The gradient is calculated after the RNN outputs the prediction $h(k)$ as follows:

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W} \rightarrow 0 \tag{37}$$

where, E represents the prediction error and W is the parameter to be optimized. If (37) is considered as a series of functions, then it converges to zero if the series of its partial sums tends to zero:

$$S_n = \sum_{t=1}^{n} \frac{\partial E_t}{\partial W} \rightarrow 0 \tag{38}$$

The error gradient for time step t is defined as:

$$\frac{\partial E_k}{\partial W} = \frac{\partial E_k}{\partial h_k} \frac{\partial h_k}{\partial c_k} \cdots \frac{\partial c_2}{\partial c_1} \frac{\partial c_1}{\partial W} = \frac{\partial E_k}{\partial h_k} \frac{\partial h_k}{\partial c_k} \left(\prod_{t=2}^{k} \frac{\partial c_t}{\partial c_{t-1}} \right) \frac{\partial c_1}{\partial W} \tag{39}$$

For prediction, the RNN output is supplied to a fully connected dense regression layer as:

$$\phi_{t+1} = \omega_{dense} h_t + b_{dense} \tag{40}$$

where, ω_{dense} and b_{dense} represent the respective weights and bias terms. Algorithm 1 shows the step-wise process of data generation and neural network training in the first dimension. Algorithm 2 explains time-series prediction and estimation of optimised FOPID variables in the second dimension.

4. PERFORMANCE ANALYSIS

Owing to the involvement of missile systems, hardware implementation of the entire system couldn't be realized. However, the proposed algorithm was implemented in real-time on a suitable hardware (Intel Xeon Processor 3.70 GHz, 3696 Mhz, 8 Cores, 16 Logical Processors, 64 GB RAM) while considering simulated missile-target dynamics.

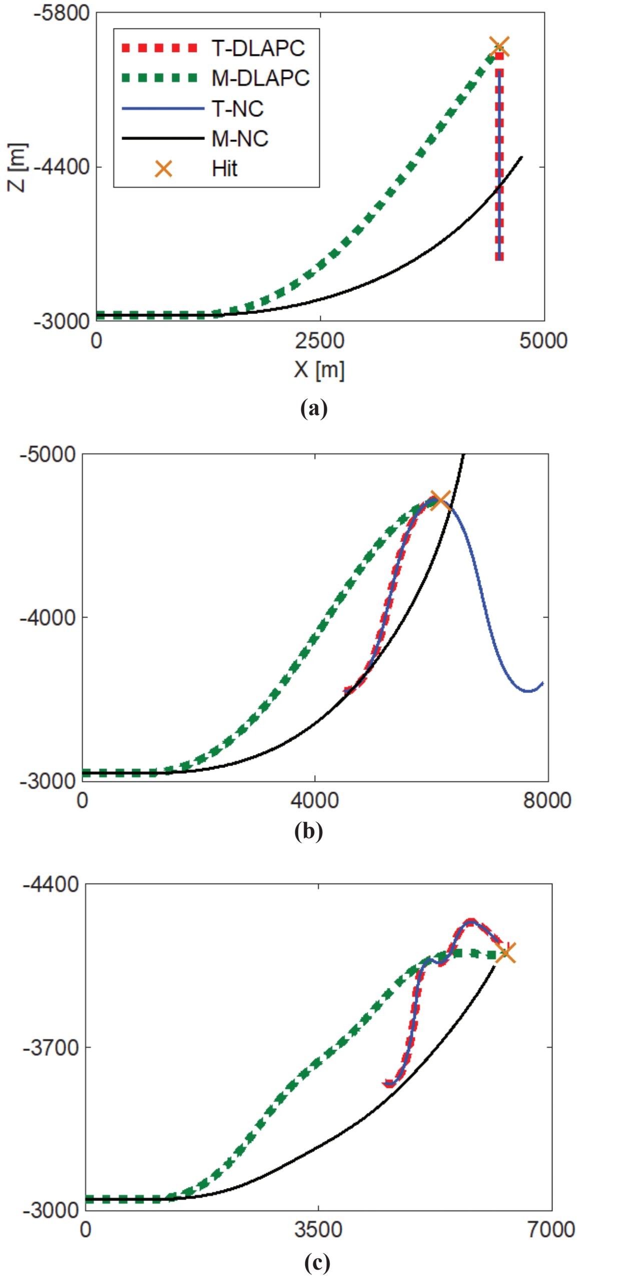

Figure 4. Target and missile trajectories: (a) Non-maneuvering target, (b) Randomly maneuvering target, and (c) Intelligently maneuvering target.

Parallel dual-core simulations were performed to evaluate the impact of system processing latency and its correction via prediction. Core-1 of the processor mimics the dynamics of the missile-target environment, while Core-2 executes the DLAPG algorithm and shares the input (V_c,θ,ϕ) and output (a_m command) data with Core-1. Core-2's prediction horizon is equivalent to Core-1's simulation processing duration.

In this study, the horizontal and vertical components of the initial range between the target and the missile are 4500 m and 535 m, respectively. Yaghi and Onder[12] used the H_2/H_∞ neural control (NC) technique where the controller parameters are tuned offline and achieved a minimum miss-distance of 0.262 m for ϕ = 180°. The same technique is reproduced in this paper with miss-distance value of 0.269 m. The slight

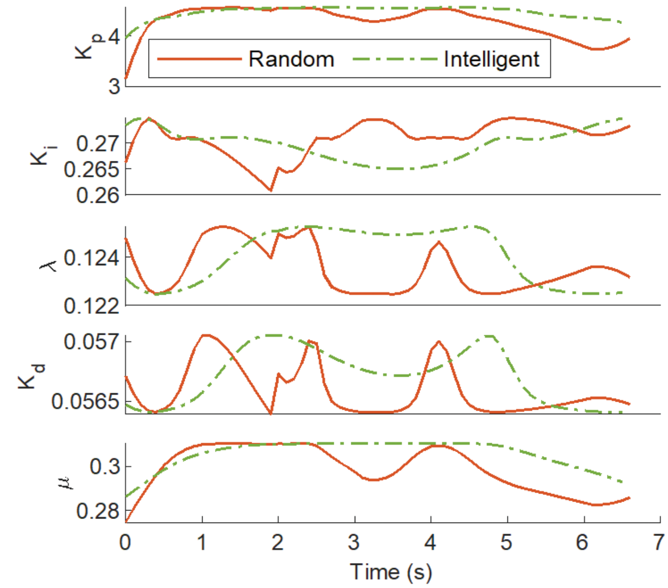

Figure 5. Adaptively changing FOPID parameters for the cases of randomly and intelligently maneuvering target.

difference in the miss-distance values can be attributed to the differences in FOPID approximation techniques, PN guidance law modifications, and system modeling approaches used in the two papers. DLAPG is applied for the same ϕ = 180° and a comparable miss-distance value of D_m = 0.47 m is achieved which is slightly more than what is achieved by NC algorithm. This can be attributed to the fact that the NC algorithm is exclusively tuned for ϕ = 180° prior to implementation whereas the DLAPG algorithm is an online-self tuning technique which is not pre-tuned for any particular value of ϕ. However, when the two techniques are compared for ϕ values drastically different from 180° or for the case of maneuvering target, the DLAPG technique shows higher performance and robustness as discussed below.

In Fig. 4(a), T and M stand for target and missile, respectively, shows the comparison for ϕ = 90° which is drastically different from ϕ = 180°. With DLAPG, the missile is able to hit the target with a miss-distance value of 0.86 m while as with NC, the missile misses the target.

The performance of the proposed technique is further evaluated for randomly and intelligently maneuvering targets. A randomly maneuvering target changes its course or speed in a random or unpredictable manner to avoid being hit by the missile. This type of target is typically used in training exercises or simulations, as it is easy to program and does not require advanced decision-making capabilities. However, an intelligently maneuvering target employs more sophisticated evasion tactics and decision-making algorithms to avoid being hit by the missile. Thus, an intelligently maneuvering target is likely to be more difficult to hit than a randomly maneuvering target, as it is better able to anticipate and respond to the missile's movements; therefore, it is often used in more advanced missile defense training scenarios.

Figure 4(b) shows a randomly maneuvering target which is hit by the missile with miss-distance equal to 1.2 m for DLAPG and 3.8 m for NC. Figure 4(c) shows an intelligently

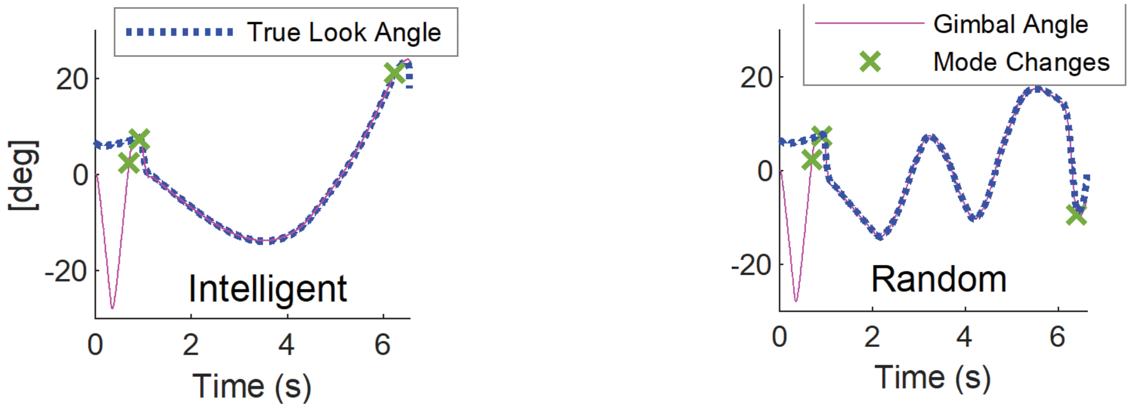

Figure 6. Gimbal angle vs look angle stability for the cases of intelligently and randomly maneuvering targets.

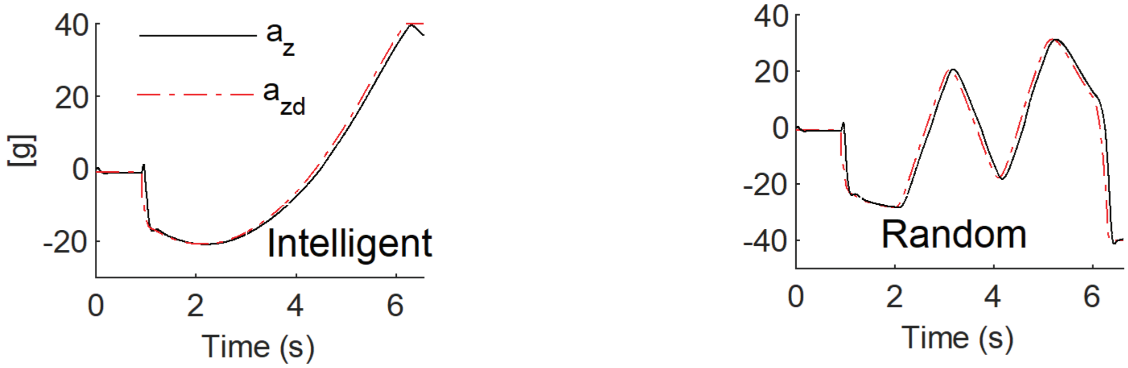

Figure 7. Normal acceleration command, a_z, vs acceleration demand, z_d, for maneuvering targets.

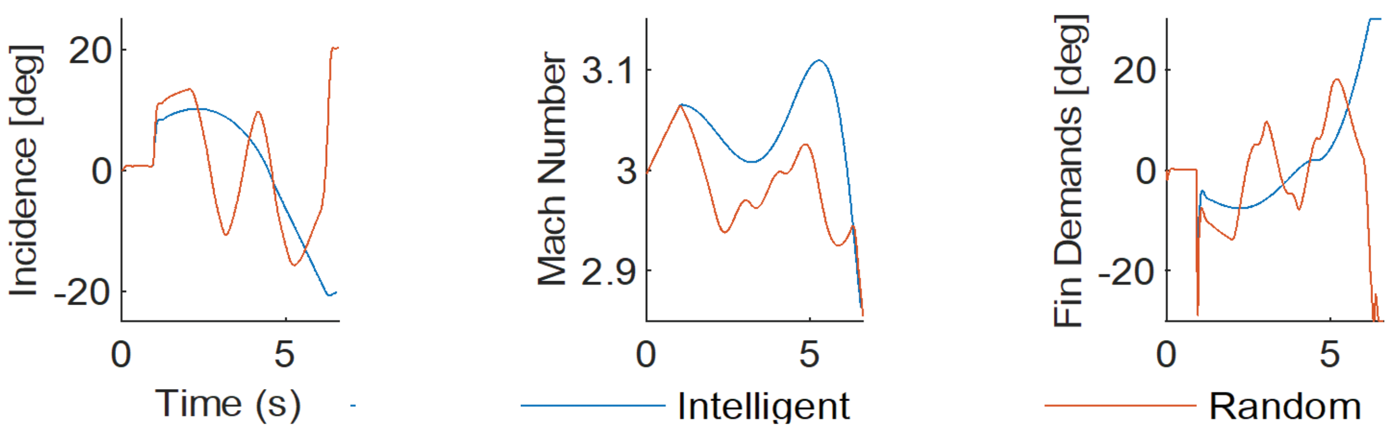

Figure 8. Incidence angle, mach number, and fin demand for different types of targets.

maneuvering target which turns upwards initially so that missile too orients towards upward direction, and then the target moves downwards to deceive the incoming missile and escape it. This upward-downward movement is continued until the missile escapes the target. The missile with DLAPG is able to hit this target with miss-distance equal to 0.97 m while as the missile with NC misses the target. Many other random experiments were performed and it was observed that the DLAPG algorithm ensures that the missile always hits the target with $D_m \leq 1.5$ m even if it is maneuvering to escape the missile. The offline-tuned NC was found to show high performance for non-maneuvering targets with $\phi = 180^{\circ} \pm 30^{\circ}$ and few other random target directions. However, for maneuvering targets and targets moving in directions drastically different from $\phi = 180$, the NC technique either misses the target or achieves a higher miss-distance value compared to DLAPG. These results clearly demonstrate the superiority of DLAPG over NC and by extension other techniques like PID controller optimized using particle swarm optimization (PSO) and Ziegler-Nichols (ZN) techniques that are compared with NC by Murad and Yaghi[12]. Figure 5 shows the adaptively changing FOPID controller parameters in accordance with the maneuvering tactics of randomly maneuvering and intelligently maneuvering targets respectively. In case of random maneuver, the target erratically changes the direction in a zig-zag pattern, leading to significant fluctuations in the parameter values. Figure 6 shows the stability of gimbal angle and its conformity with true look angle for both the cases. It is shown in Fig. 7 that the normal acceleration (a_z) produced is in accordance with the demanded acceleration (a_{zd}). Figure 8 shows that the DLAPG equipped missile does not lose performance or violate bounds in terms of critical system parameters like incidence angle, mach number, and fin actuator demand.

5. CONCLUSION

The main outcome of this study is the development of a deep learning based, partially physics-informed, model-free, and adaptive predictive guidance for radar-guided missile. The study also establishes the use of deep learning to predict the maneuvering tactics of the target to compensate for the system processing delays of intrinsic nature during which the target could potentially modify its flight path direction. Adaptability of controller variables is an additional advantageous aspect of the research, since the parameters are continually adjusted on-the-fly in response to the changing flight trajectory of the desired target. The superior performance of the proposed technique in terms of minimization of miss-distance and adaptability over other methods is demonstrated.

In future, additional parameters such as target speed, multiple targets, etc., could be included. In addition to miss-distance, other metrices such as impact angle constraints, which play a significant role in the mission success, must be evaluated in future to ensure onboard feasibility of the proposed technique. Furthermore, multiple missiles for multiple targets can be used intelligently and coherently if the control algorithm is implemented in the launching station with a communication link with the missiles. Moreover, supercomputers can be used to generate more training data by considering multiple parameters or data from actual missiles in action can be used to further enhance this technique. This technique can also be potentially considered for different time-sensitive applications requiring online optimization, such as robotic path planning, autonomous driving, drone control, and gamming.

REFERENCES

1. Rusnak, I. Bounds on the miss of radar-guided missiles with discrete guidance. *J. Guid. Control Dyn.*, 2020, **43**(4), 825–830. doi: 10.2514/1.g004811.
2. Yi, K.; Tang, K.; She, S.; Li, M. & Tan, Z. Integrated guidance and control for semi-strapdown missiles. *In* 2018 37th Chinese Control Conference (CCC); IEEE, 2018. doi: 10.23919/ChiCC.2018.8482808.
3. Kocpinar, T. & Kerestecioglu, F. Predictive control of aerodynamically driven missiles. *In* 1999 European Control Conference (ECC); IEEE, 1999. doi: 10.23919/ECC.1999.7099661.
4. Chwa, D.; Choi, J.Y. & Anavatti, S.G. Observer-based adaptive guidance law considering target uncertainties and control loop dynamics. *IEEE Trans. Control Syst. Technol.* 2006, **14**(1), 112–123 doi: 10.1109/tcst.2005.860529.
5. Choi, J.Y.; Chwa, D. & Cho, H.-P. Nonlinear adaptive guidance considering target uncertainties and control loop dynamics. *In* Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148); IEEE, 2001. doi: 10.1109/ACC.2001.945596.
6. Chwa, D. & Choi, J.Y. Adaptive nonlinear guidance law considering control loop dynamics. *IEEE Trans. Aerosp. Electron. Syst.* 2003, **39**(4), 1134–1143 doi: 10.1109/taes.2003.1261117.
7. Wenjin, G.; Ruchuan, Z. & Hongchao, Z. On sliding mode guidance algorithm for based on adaptive fuzzy-neural network approximation. *In* 2008 27th Chinese Control Conference; IEEE, 2008. doi: 10.1109/CHICC.2008.4605008.
8. Xiang, S.; Fan, H. & Fu, Q. Distribution of miss distance for pursuit-evasion problem. *IEEE/CAA J. Autom. Sin.* 2020, **7**(4), 1161–1168 doi: 10.1109/jas.2019.1911552.
9. Huang, L. & Chen, W. Deep learning midcourse guidance for interceptor missile. *In* 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC); IEEE, 2019. doi: 10.1109/ITNEC.2019.8729311.
10. Li, B.; Ma, F. & Wu, Y. Missile attitude control based on deep reinforcement learning. *In* 2020 IEEE 16th International Conference on Control & Automation (ICCA); IEEE, 2020. doi: 10.1109/ICCA51439.2020.9264391.
11. Liang, C.; Wang, W.; Liu, Z.; Lai, C. & Zhou, B. Learning to guide: Guidance law based on deep meta-learning and model predictive path integral control. *IEEE Access* 2019, **7**, 47353–47365 doi: 10.1109/access.2019.2909579.
12. Yaghi, M. & Onder Efe, M. H2/H∞-neural-based fopid controller applied for radar-guided missile. *IEEE Transact. Industrial Electron.* 2020, **67**(6), 4806–4814. doi: 10.1109/TIE.2019.2927196
13. Farooq, J. & Bazaz, M.A. Deep Learning for self-tuning of control systems. *In* 2021 International Conference on Emerging Techniques in Computational Intelligence (ICETCI); IEEE, 2021. doi: 10.1109/ICETCI51973.2021.9574048.
14. Standard atmosphere. ISO 2533: 1975. International Organization for Standardization, pages 11–12, 1975.
15. Mracek, C.; Cloutier, J.; Cloutier, J. & Mracek, C. Full envelope missile longitudinal autopilot design using the state-dependent Riccati equation method. *In* Guidance, Navigation, and Control Conference; American Institute of Aeronautics and Astronautics: Reston, Virigina, 1997. doi: 10.2514/6.1997-3767.
16. Bennani, S.; Willemsen, D. & Scheror, C. Robust LPV control with bounded parameter rates. *In* Guidance, Navigation, and Control Conference. Reston, Virigina: American Institute of Aeronautics and Astronautics., August 1997. doi: 10.2514/6.1997-3641
17. Mracek, C. & Cloutier, J. Full envelope missile longitudinal autopilot design using the state-dependent riccati equation method. *In* Guidance, Navigation, and Control Conference. Reston, American Institute of Aeronautics and Astronautics, Virigina, August 1997. doi: 10.2514/6.1997-3767
18. Shamma, J. & Cloutier, J. Gain-Scheduled Missile autopilot

design using linear parameter varying transformations. *J. Guid Control Dyn.*, 1993, **16**(2), 256-263. doi: 10.2514/3.20997

19. Lin, C.-F. Modern navigation, guidance, and control processing; Prentice Hall: Philadelphia, PA, 1991.
20. Podlubny, I. Fractional differential equations: An introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications; Academic Press, 2014.
21. Čech, M. & Schlegel, M. The fractional-order PID controller outperforms the classical one. *In* Process Control 2006; 2006.
22. Luo, Y. & Chen, Y. Fractional-order [proportional derivative] controller for robust motion control: Tuning procedure and validation. *In* 2009 American Control Conference; IEEE, 2009. doi: 10.1109/ACC.2009.5160284.
23. Monje, C.; Chen, Y.; Vinagre, B.M.; Xue, D. & Feliu, V. Fractional-order systems and controls: Fundamentals and applications; Springer: London, England, 2010.
24. Saleh, K.M. Fractional order PID controller tuning by frequency loop-shaping: Analysis and applications. 2017.
25. Gu, G. Linear feedback control - Analysis and design with MATLAB (by Dingyu Xue et al; 2007) [Bookshelf]. *IEEE Control Syst.* 2009, **29**(1), 128–129 doi: 10.1109/mcs.2008.930839.
26. Pan, B.; Fareed, U.; Qing, W. & Tian, S. A novel fractional order PID navigation guidance law by finite time stability approach. *ISA Trans.*, 2019, **94**, 80-92 doi: 10.1016/j.isatra.2019.04.019
27. Oustaloup, A.; Levron, F.; Mathieu, B. & Nanot, F.M. Frequency-band complex noninteger differentiator: Characterization and synthesis. *IEEE Trans. Circuits Syst. I. Fundam. Theory Appl.* 2000, **47**(1), 25–39 doi: 10.1109/81.817385.
28. Matsuda, K. & Fujii, H. H(infinity) optimized wave-absorbing control - Analytical and experimental results. *J. Guid. Control Dyn.* 1993, **16**(6), 1146–1153 doi: 10.2514/3.21139.
29. AbdelAty, A.M.; Elwakil, A.S.; Radwan, A.G.; Psychalinos, C. & Maundy, B.J. Approximation of the Fractional-Order Laplacian s^{α} as a weighted sum of first-order high-pass filters. *IEEE Trans. Circuits Syst. II Express Briefs,* 2018, **65**(8), 1114–1118 doi: 10.1109/tcsii.2018.2808949.
30. El-Khazali, R. On the biquadratic approximation of fractional-order Laplacian operators. *Analog Integr. Circuits Signal Process.* 2015, **82**(3), 503–517 doi: 10.1007/s10470-014-0432-8.
31. El-Khazali, R.; Batiha, I.M. & Momani, S. Approximation of fractional-order operators. *In* Springer Proceedings in Mathematics & Statistics; Springer Singapore: Singapore, 2019;121–151.
32. Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* 1997, **9**(8), 1735–1780 doi: 10.1162/neco.1997.9.8.1735.
33. Gers, F.A. & Schmidhuber, J. Recurrent nets that time and count. *In* Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium; IEEE, 2000. doi: 10.1109/IJCNN.2000.861302.

CONTRIBUTORS

Mr Junaid Farooq obtained his Masters in Electrical Power and Energy Systems from National Institute of Technology (NIT) Srinagar, India. His research interests include automation, machine learning, artificial intelligence, dynamical modeling, adaptive and intelligent control.
His contributions to this paper include: Conception of ideas, simulation experiments, data analysis, model building, and paper drafting.

Dr Mohammad Abid Bazaz obtained his PhD in Control and Automation from Indian Institute of Technology (IIT), Delhi, India. He is working as Professor with the Department of Electrical Engineering, NIT Srinagar. His research interests include: Adaptive control and automation, model order reduction, deep learning, artificial neural networks, and epidemiological modeling.
His contributions to this paper include: Conception of ideas, data validation, research supervision, and paper editing.