Defence Science Journal, Vol. 71, No. 6, November 2021, pp. 826-835, DOI : 10.14429/dsj.71.16295

CMAC Trained Optimum Mid-course Guidance for Tactical Flight Vehicle

G.N. Kumar[#] and A.K. Sarkar[*]

[#]*DRDO-Defence Research and Development Laboratory, Hyderabad - 500 058, India*
[*]*Department of Aeronautical Engineering, ACS College of Engineering, Bengaluru - 560 060, India*
[*]*E-mail: achintya.drdl@gmail.com*

ABSTRACT

This paper discusses design and validation of neural network based mid-course guidance law of a surface to air flight vehicle. In present study, initially different optimal trajectories have been generated off-line of different pursuer-evader engagements by ensuring minimum flight time, maximum terminal velocity and favorable handing over conditions for seeker based terminal guidance. These optimal trajectories have been evolved by nonlinear programming based direct method of optimisation. The kinematic information of both pursuer and evader, generated based on these trajectories have been used to train cerebellar model articulate controller (CMAC) neural network. Later for a given engagement scenario an on-line near optimal mid-course guidance law has been evolved based on output of trained network. Training has been carried out by CMAC type supervisory neural network. The tested engagement condition is within input/output training space of neural network. Seeker based homing guidance has been used for terminal phase. Complete methodology has been validated along pitch plane of pursuer-evader engagement. During mid-course phase, the guidance demand has been tracked by attitude hold autopilot and during terminal phase, the guidance demanded lateral acceleration has been tracked by acceleration autopilot. System robustness has been studied in presence of plant parameter variations and sensor noise under Monte Carlo Platform

Keywords: Neural network; Optimal trajectory; Real time mid-course guidance; Suboptimal guidance; Seeker based terminal guidance

NOMENCLATURE

GCF	Gain crossover frequency
GM	Gain margin
NLP	Nonlinear programming
PM	Phase margin

1. INTRODUCTION

The surface to air flight vehicle (FV) trajectory consists of 1) launch 2) mid-course and 3) terminal homing phase. Present study is mainly to evolve an optimal mid-course guidance of a surface launched tactical flight vehicle (TFV) as FV seeking to intercept an airborne target during launch and mid-course phase. FV is guided by vehicle borne inertial navigation system (INS) along with ground radar tracked target kinematic information. Terminal phase homing is achieved through seeker based autonomous proportional navigation (PN) guidance (Fig. 1). For a target at greater distance, it is preferred to maximise terminal velocity for better endgame and also to minimise total flight time. However, direct formulation of mid-course guidance based on optimal control theory results in solution of two point boundary value problem (TPBVP) which can not be solved in real time on any present day on board computer (Song[1], *et al.*). Some of the authors have arrived at design of mid-course guidance law of an air-to-air engagement based on trajectory optimisation[2]. Later this algorithm has been successfully ported on FV on-board computer and successfully flight tested also[3]. This trajectory shaping during mid-course phase where the TPBVP is solved in real time is called singular perturbation (SP) guidance. In this paper they have reported that for a specific engagement mid-course SP guidance along with terminal PN guidance enhances launch range to 90 km whereas PN guidance throughout mid-course and terminal phase gives only 52 km.

In this paper the authors' aim is to replace SP based mid-course guidance by artificial neural network (ANN) technique which is mainly for range enhancement. In earlier days, this optimum mid-course trajectory used to be stored in an on-board computer and guidance law used to be implicit type. Present day this SP guidance as one popular mid-course guidance law

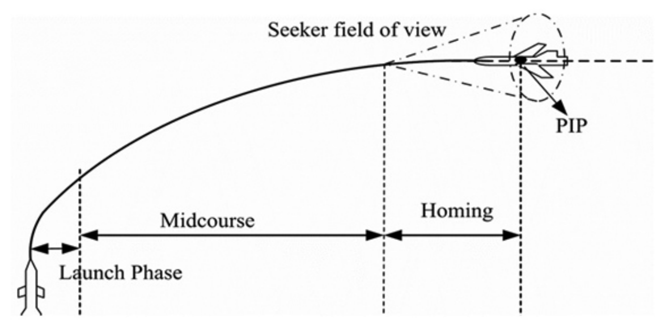

Figure 1. Schematic trajectory of flight vehicle target engagement (TFV framework).

Received : 11 September 2020, Revised : 08 March 2021
Accepted : 17 March 2021, Online published : 22 October 2021

is used in real time where the fast to slow variables are solved one by one along with boundary layer correction. This is a near optimal explicit guidance law in feedback form to cater for all uncertainties during flight . This guidance law is bench marked and fine tuned based on off-line evolved optimal trajectory as reference.

In nineties artificial neural network (ANN) has been introduced by Balakrishnan[4], *et al.* where it improves the optimum control problem solution. It enables learning, adaptation and faster throughput and equips the guidance system with capability akin to intelligent biological organisms. They have used dynamic programming technique off-line to evolve the optimal control histories for an entire bracket of initial conditions and entire range of interest. Later they trained ANN with the optimal control history and used trained network for a specified initial condition and range. Lee[5], *et al.* proposed ANN based guidance algorithm. They used ANN to train inverse dynamics of the vehicle and used PN guidance law directly on non-linear dynamic characteristics. Song[6], *et al.* proposed a guidance law based on ANN which was trained using the bunch of optimal trajectories obtained for various terminal points distributed over the region of interest. Song[1,7,8], *et al.* proposed a hybrid guidance scheme with two guidance laws. Here also they have generated optimal control in terms of initial state and terminal condition off-line, trained this set of data by ANN and evolved trained guidance law. In their proposed hybrid guidance scheme, γ-correction guidance law steers the vehicle to track the optimal trajectory generated from the nominal launch conditions and feedback guidance law reduces terminal miss distance caused by initial launch errors. Choi[9], *et al.* proposed a differential game guidance law using ANN for an air-to-air engagement. Here three dimensional time optimal pursuit-evasion game solutions are obtained off-line using a gradient based optimiser. Here they have used feed forward neural network (FFNN) with FV-target relative position components, their velocities and line of sight (LOS) rates along yaw and pitch as input and angle of attack along yaw and pitch plane as output respectively. Later they enhanced same algorithm by considering additional input variables such as range, range rate, LOS rate and heading error in FFNN[10].

Now let us discuss some current state of the art mid-course guidance laws based on deep learning (DL) cited in recent literature. Mahammad[11], *et al.* used off-line generated optimal trajectories of a surface to air missile for training of a regression model for estimation of the optimal guidance commands for mid-course application. In this paper they utilised the support vector machine (SVM) and the relevance vector machine (RVM) in their regression model to evaluate the ability and benefits of this approach. This approach kept them away from the over fitting problem while training. Through several numerical experiments with different terminal conditions they demonstrate the ability of developed models to guide the missile with satisfactory accuracy during the mid-course phase. Huang[12], *et al.* evolved a mid-course guidance method of interceptor missile based on long short-term memory DL networks. In their study they have compared present DL based guidance law with traditional ANN based approach and claimed significant reduction in miss dance using proposed approach. Very recently Shaumov[13,14] carried out simulation of Target-Missile Defender (TMD) engagement using Deep Reinforced Learning topology. In TMD a defending or intercepting missile is launched by target to neutralise the attacking missile which is similar to air defence problem, which is also a nonlinear optimal control problem. The optimal selection of the defender launch time on-line can be formulated as switched system optimisation problem. This is crucial for improving the performance of the target defender team. Using DL based neural network he has proposed estimation of optimal launch time even with small number of measurements. The DL method consists of training a deep neural network (DNN) (ANN with deep architecture such as several hidden layers) on a given data set[14] .

Based on above literature survey it is inferred that ANN has been applied successfully for both mid-course and terminal guidance of FVs. In all the research carried out so far, ANN based guidance law design consists of three major steps such as 1) Generation of optimal input in terms of FV initial conditions, path constraints and different terminal constraints which is obtained by solving TPBVP off-line as trajectory optimisation problem. 2) Selection of ANN topology and training methodology of huge input-output pair database. 3) Extraction of near optimal guidance command for a given engagement scenario using trained ANN. It is worth to mention at this juncture that based on above cited literature, generation of training patterns and selection of the input and output spaces of the neural network for obtaining ANN based guidance law is very crucial and is open till date for research. The authors have understood this lacuna and tried to bridge this gap on ANN training by following study.

In current research it is assumed that target predicted intercept point (PIP) is always available though ground based tracking radar. Once FV is launched, it need to be guided towards the PIP by itself using INS during the mid-course with desirable features with minimum time along with high terminal velocity to intercept the target for best no escape zone. The aim of this paper is to evolve ANN based mid-course guidance to generate optimum trajectory on-line for a given engagement. So based on the off-line optimised trajectories, an ANN has been trained to generate the on-line optimum trajectory suitable to given PIP and in-flight conditions as depicted in Fig. 5. This is a novel two stage cascaded network, the first network output feed to second along with its input to generate the optimum FV attitude angle (θ). Input to trained ANN needs only FV kinematic information available from INS and PIP initially obtained once through tracking radar. Present algorithm requires no need of datalink during mid-course guidance. It is worth to mention at this juncture that current state of the art is to guide the FV in mid-course based on the target information using datalink. Purpose of present research is to eliminate the need of datalink using ANN based mid-course guidance. At present this concept has been demonstrated for surface to air engagement. Later the study has to be extend for air-to-air engagement without datalink, eliminating the vulnerability of pilot and mother aircraft against enemy.

This paper is organised as follows:

(1) Off-line trajectory optimisation with all path and terminal constraints has been carried out by direct method using Nonlinear Programming (NLP)[15]
(2) Later present proposed novel cascaded ANN has been trained by cerebellar model articulation controller (CMAC) neural network (NN)
(3) The trained NN has been used to evolve mid-course guidance demanded θ for a given engagement condition.
(4) This guidance demand during mid-course is tracked by attitude tracking autopilot with PN guidance during terminal homing where the latax (lateral acceleration) autopilot tracks the guidance demanded latax.

During terminal guidance Extended Kalman Filter (EKF) based estimator is used to process the noisy seeker measurements. Complete methodology has been demonstrated by a FV target engagement along pitch plane. System robustness has been studied in presence of plant parameter variations and sensor noise under Monte Carlo (MC) Platform. The paper ends with immediate future activities to be addressed to mimic a realistic FV-target engagement.

2. OFF-LINE TRAJECTORY OPTIMISATION

Plant model considered here is described pictorially in Fig. 2. The attitude, θ is treated as a control variable. Acceleration along the forward and normal axis of the TFV excluding gravity component are:

$$a_F = \frac{T - QSC_D}{m}; \qquad a_N = \frac{QS\,C_{N\alpha}\alpha}{m}; \tag{1}$$

here, (T, Q, m) are (thrust, dynamic pressure, pursuer mass). S is reference area as body cross sectional area, α angle of attack, $(C_D, C_{N\alpha})$ are (drag coefficient, normal force curve slope) with respect to angle of attack.

2.1 State Equation

$$\begin{bmatrix} \dot{V}_h \\ \dot{V}_v \\ \dot{r} \\ \dot{h} \end{bmatrix} = \begin{bmatrix} a_F\cos(\theta) - a_N\sin(\theta) \\ a_F\sin(\theta) + a_N\cos(\theta) - g \\ V_h \\ V_v \end{bmatrix} \tag{2}$$

where $[V_h, V_v, r, h]$ are (horizontal and vertical velocity, range and altitude) along inertial frame (Fig. 2).

2.2 Decision Vector

$$[u] = [\theta] \tag{3}$$

Present trajectory optimisation has been carried out by NLP based direct method. The complete procedure is available in Betts [15] and not reported here for brevity. Recently Mukherjee[16], *et al.* carried out an air-to-air trajectory optimisation using NLP based direct transcription method. Present problem formulation is subset of that with fixed initial condition, terminal constraint of meeting specified evader (R_d, H_d) and in-flight path constraint of $|\alpha| \le \alpha_{max}$.

The cost function for present problem to be minimised is

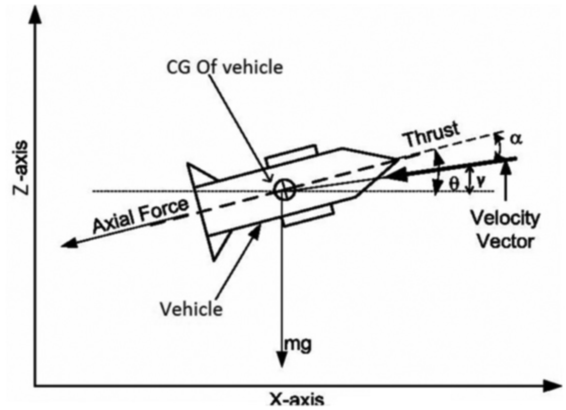

Figure 2. Free body diagram of present TFV.

$$J = \underbrace{K_t t_f^2 + K_h V_h^2(t_f) - K_v V_f^2(t_f)}_{\text{Performance parameter 1}} + \underbrace{\int_{t_0}^{t_f} K_\alpha Q\alpha^2 dt}_{\text{Performance parameter 2}} \tag{4}$$

where $(K_t, K_h, K_v, K_\alpha)$ are the weights of variables to be optimised. The performance parameters consist of minimisation of total flight time, maximisation of terminal horizontal and vertical velocity components for better end-game scenario. The performance parameter-2 consists of minimisation of total inflight path constraint which physically means reduction of aerodynamic load on the FV. So all $(K_t, K_h, K_v, K_\alpha)$ should be positive. In present problem these values are (1.0*E*05*,* 5.0*,* 2.0) and $(K_\alpha = 1)$ which have been evolved by trial and error.

The function *fmincon* of MATLAB optimisation toolbox, a general purpose performance index minimisation routine under linear and nonlinear equality and inequality constraints has been used to solve present problem. The optimisation was started initially in the first pass with only five grid points N_b during boost phase and twelve grid points $N_c = 3\times(N_b - 1)$ during coast phase. So in the first pass of trajectory optimisation, there is total number of 17 grid points. In the second pass, $N_b = 2 \times 5 - 1 = 9$ and $N_c = 3\times(N_b - 1)$ has been chosen. The algorithm for choice of (N_b, N_c) for different iterations is given below. In the first pass, for 17 grid points $(N_b, 1 = 5)$, the NLP constraints are for during coast phase. So in the first pass of trajectory optimisation, there is total number of 17 grid points. In the second pass,

$N_b = 2 \times 5 - 1 = 9$ and $N_c = 3\times(N_b - 1)$ has been chosen. The algorithm for choice of (N_b, N_c) for different iterations is given in Mukherjee[16], *et al.* In the first pass, for 17 grid points $(N_b, 1 = 5)$, the NLP constraints are formed using trapezoidal discretisation. Total five passes are required for achieving convergence in present problem. In fifth and last pass the grid points are 65 and Hermite-Simpson discretisation is used. One typical trajectory optimisation of present TFV is discussed.

Present TFV is launched vertically from ground. In launcher it is for 0.5 s. At launch exit initial states for trajectory optimisation are (V_h, V_v, r, h) as (0, 25 m/s, 0 m, 300 m). Pursuer has to reach at mid-course end (r_f, h_f) as (15, 5) km. The path constraints are α < 15 deg and Qα < 15000 N rad/m[2] from structural design consideration. For the present case study normalised optimal trajectory profile along with optimal -profile are shown in Figs. (3 and 4), respectively. Coincidence

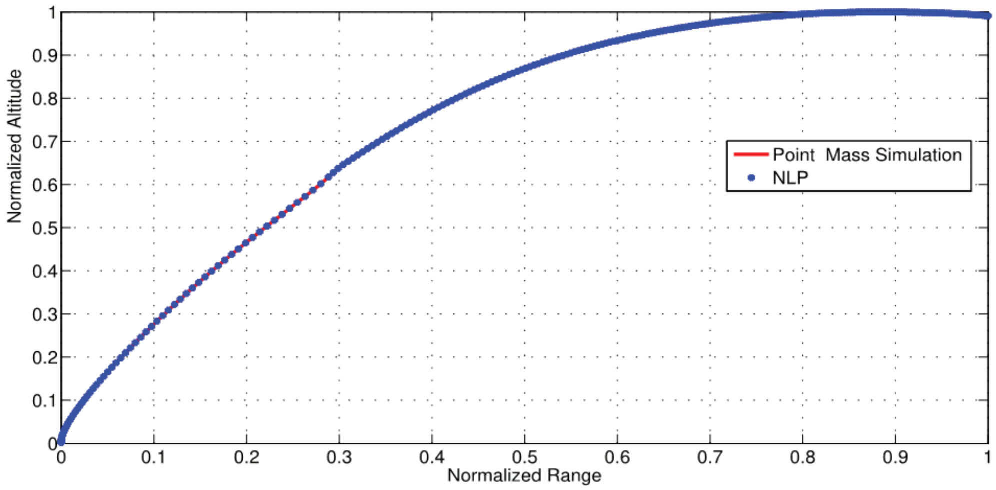

Figure 3. Normalised FV optimal trajectory.

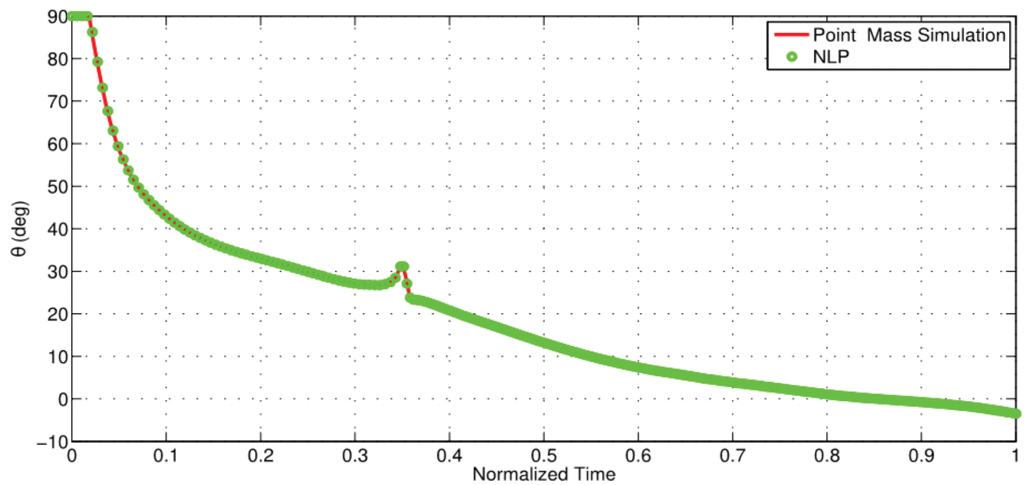

Figure 4. Corresponding optimal θ-profile time history.

of point mass continuous simulation with NLP discrete points indicate convergence of optimisation algorithm.

3. ANN BASED ON-LINE OPTIMUM TRAJECTORY GENERATION

In the previous section, generation of optimal trajectory has been discussed. Now ANN will be discussed to train a bunch of optimal trajectories and to obtain the control input _ from trained ANN for on-line application. The ANN training topology for on-line trajectory generation is shown in Fig. 5. Based on literature survey it is imperative to state that the researchers have used position and velocity information of both pursuer and evader for generating training data set. The aim of the present study is to take the Pursuer only up to PIP. Based on evader kinematic information and knowing time to go (t_{go}) approximately, PIP can be calculated apriori. The aim of the mid-course guidance is to guide the pursuer only up to PIP which can be approximate. The mid-course errors can be corrected in terminal seeker based guidance. For guiding the pursuer up to PIP no evader kinematic information is required. Generally evader is tracked by ground radar and data link is used to transmit the evader kinematic information to pursuer. Aim of present investigation is to guide the pursuer up to PIP

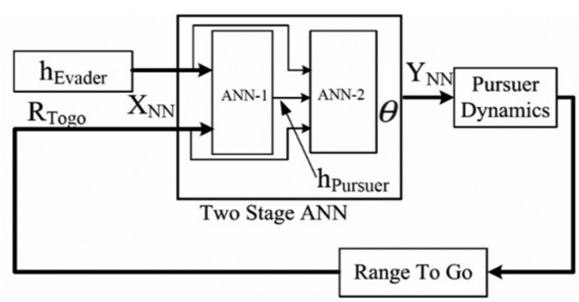

Figure 5. On-line trajectory generation scheme.

without use of tracking radar and data link during mid-course to make the weapon system cost effective and simple. So present ANN topology has been evolved with minimum number of inputs (X_{nn}). It consists of two stages ANN1 and ANN2. Inputs to ANN1 is (h_{PIP} , r_{togo}) where $r_{togo} = x_{PIP} - x_p(t)$. The output of ANN1 is estimated instantaneous pursuer height $\hat{h}_p(t)$. Input to ANN2 is (h_{PIP} , $\hat{h}_p$, r_{togo}) and output of ANN2 is estimated $\hat{\theta}(t)$. The present topology has been evolved keeping in mind that ANN should function satisfactory with no evader and minimum pursuer kinematic information. An in house developed CMAC developed for training bunch of optimal trajectories and brief description of CMAC algorithm is given below.

CMAC in Algorithmic form

CMAC is a single layer and single neuron associative memory networks. First introduced by Albus[17], they are inspired by the models for the functions of human cerebellum have grown lot of interest since nineties among researchers in the field of learning control and robotics due to their attractive features such as high training speed, fast convergence to global optima, excellent functional representation etc. It can learn nonlinear relationship for broad category of functions $y = f(x_1, \cdots, x_M)$. To compute the output y, it carries out quantisation of corresponding inputs, gets a list of active memory address and sums the weights located at these addresses. The active addresses are determined by a three stage mapping of the corresponding input points, such as Quantisation, Segment Address and Associated Address mapping which is shown schematically in Fig. 6. Govindarao has developed a novel algorithmic generalisation of method of Albus for any number of inputs and applied successfully for approximation of many complex functions related to Chemical Process Engineering. In present problem also the training of kinematic data has been carried out by CMAC of Govindarao[18]. The authors wish to state that present generalised algorithmic form of CMAC is not available in any published literature on aerospace application till date.

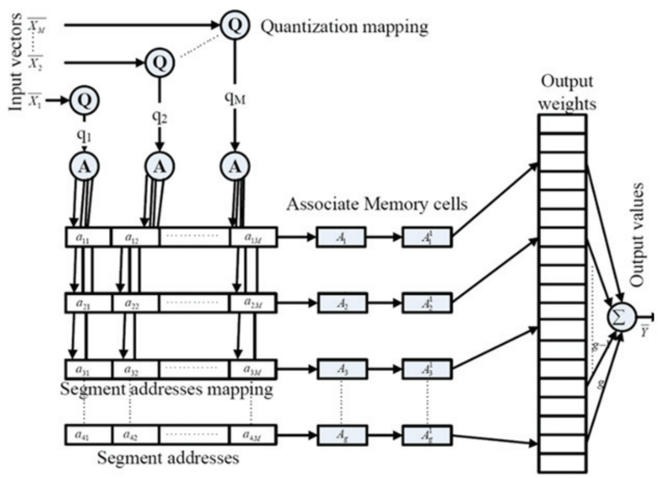

Figure 6. Schematic diagram of CMNN.

3.1 Quantisation Mapping

Let the n[th] input-output pair to be trained be ($x_1^{(n)}, \cdots, x_m^{(n)}; t^{(n)}$). The mapping can be defined by

$$I_m^{(n)} = INT\left(\frac{x_m^{(n)} - x_{m\min}}{r_m}\right) + 1$$

for ($m = 1, \cdots, M$; and $n = 1, \cdots, N$) where

(M,N) = Total inputs variables and points

$x_m^{(n)}$ = m[th] input variable of nth input

$x_{m,\min}$ = Lower limit of m[th] input variable

$x_{m,\max}$ = Upper limit of m[th] input variable

r_m = Resolution of input variable

$I_m^{(n)}$ = Quantisation interval (m[th] variable of n[th] input)

3.2 Segment Address Mapping

The numbers $I_m^{(n)}$ to $(I_m^{(n)} + (g\text{-}1))$ constitute g segment addresses of m[th] input variable for n[th] input point, where g is generalisation parameter, which constitutes the number of weights whose sum is considered to represent the corresponding target value.

- Arrange these g numbers in the increasing order of the residue with respect to g as ($a_{1m}^{(n)}, \cdots, a_{gm}^{(n)}$)
- Same way get M such sets,($a_{1m}^{(n)}$ for $i = 1, \cdots, g$) For all variables ($m = 1, \cdots, M$).
- Arrange the M sets into g sets with each element of a set having same residue with respect to g.
- Repeat the above steps for each of the inputs ($\forall\ n = 1, \cdots, N$).

3.3 Associated Address Mapping

Involves concatenation of the above segmentation addresses sets to obtain g associative cell address, $A_i(n)$, for ($i = 1, \cdots$, g) for each input/output pair. The algorithm for n[th] input/ output pair (to be repeated $\forall\ n = 1, \cdots, N$) in brief is as follows:

begin

for $m = 1, \ldots, M$

$$q_m = INT\left(\frac{x_{m\max} - x_{m\min}}{r_m}\right) + 1$$

for $l = 1, \ldots, g$ & *for* $m = 1, \ldots, M$

$$C_{lj} = INT\left(\frac{q_j - l - 1 + g}{g} + 1\right)$$

for $i = 1, \ldots, g$ & *for* $m = 1, \ldots, M$ & *for* $l = 1, \ldots, g$

$$\alpha_{iml}^{(n)} = INT\left(\frac{a_{im}^{(n)} - l - 1 + g}{g}\right)$$

for $i = 1, \ldots, g$

begin

$$k = MOD\left(\alpha_{i1}^{(n)}, g\right);$$

$$if\ MOD\left(\alpha_{i1}^{(n)}, g\right) = 0, k = g;$$

$$For\ M = 1;\ A_i^{(n)} = \alpha_{i1}^{(n)}$$

$$For\ M = 2;\ A_i^{(n)} = \left(\sum_{l=1}^{g}\left(\alpha_{i1}^{(n)} C_{12} + \alpha_{i2k}^{(n)} + 1\right)\right)$$

$$For\ M > 2;\ A_i^{(n)} = \sum_{l=1}^{g}\left(\alpha_{i1}^{(n)} \prod_{j=2}^{M} C_{lj}\right) + \sum_{m=2}^{M-1}\left(\alpha_{imk}^{(n)} \prod_{j=m+2}^{M} C_{kj}\right) + \alpha_{imk}^{(n)} + 1$$

END

END

3.4 CMAC Training Algorithm

1. Specify g.
2. Construct the input space.
3. Initialise the weights $w_i = 0$ for ($i = 1, \cdots, P$) where P is the total number of associated cells.
4. Get cell addresses $A_i(n)$ for each input / output pair using the algorithm described above.
5. Compute outputs

$$y^{(n)} = \Sigma_{i=A_1^{(n)}}^{A_g^{(n)}} \omega_i \tag{5}$$

and calculate $e^{(n)} = t^{(n)} - y^{(n)}$.

6. Learning: Update weights by using

$$\omega_i^{(new)} = \omega_i^{(old)} + \eta \frac{e^{(n)}}{g} \tag{6}$$

Based on least mean square (LMS) learning using steepest descent update rule where (η) is learning coefficient of the order of 0.1 to 0.5.

7. Calculate RMS error

$$\Phi=\sqrt{\frac{\sum_{n=1}^{N}\left(e^{(n)}\right)^{2}}{N}} \qquad \text{for each epoch} \tag{7}$$

8. Checking: If Φ is less than ε or if the number of epochs is greater than pre-specified limit, completion of training occurs. Otherwise repeat from step 5 till convergence. Here the associated addresses $A_i^{(n)}$ have to be generated and the weights have to be evaluated $\forall\ i = 1, \cdots, g$ & $n = 1, \cdots, N$.

9. Schematic block diagram for CMNN for multiple input and single output is shown in Fig. 6. Present CMAC has two stages as ANN1 and ANN2 (Fig. 5). Inputs to ANN1 for training are (h_{pip}, R_{to-go}). Range to go is $R_{to-go} = x_{pip} - x_p$. Output is estimated pursuer altitude $\left(\hat{h}_p\right)$. ANN2 is trained with three inputs $\left(h_{pip}, \hat{h}_p, R_{to\text{-}go}\right)$ and the output is network estimated pursuer attitude $\hat{\theta}$. Various optimal trajectories with given initial and terminal conditions have been generated as described before (Section 2). PIP is the points for pursuer to reach optimally and from there it should lock on to evader for intercepting with terminal homing guidance. Spread of the PIP is mainly dictated by seeker field of view.

The optimal trajectory data used for training both ANN1 and ANN2 corresponds to the four trajectories of evader PIP pertaining to h_{pip} = (3.5, 4.5, 5.5, 6.5) km and x_{pip} = 15 km have generated initially. This data set has been used to train CMAC as discussed above. Then for PIP of (x_{pip}, h_{pip}) = (15, 5) km altitude $\hat{\theta}$ has been evaluated from trained network. This particular trajectory was not in the bunch used for ANN training. Estimated $\left(\hat{h}_p\right)$ from trained ANN1 along with estimation error is shown in Fig. 7. Then inputs to ANN2 are $\left(h_{pip}, \hat{h}_p, R_{to\text{-}go}\right)$ and the corresponding output is $\hat{\theta}$. Estimated $\hat{\theta}$ along with estimation error is shown in Fig. 8. In present study g = 300, maximum RMS error for convergence = 1.0E − 05 and total epochs for training convergence is 2000. Here g is associated memory address and generalisation parameter. The value of g has been arrived as tuning parameter and for this value the performance of trained network is best in terms of RMS error of estimated output with respect to true value.

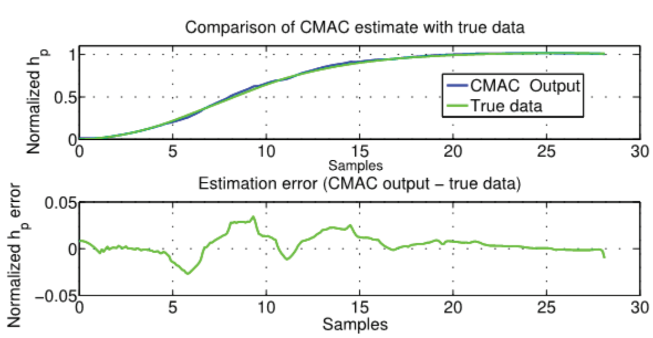

Figure 7. Normalised FV estimated altitude from trained CMAC.

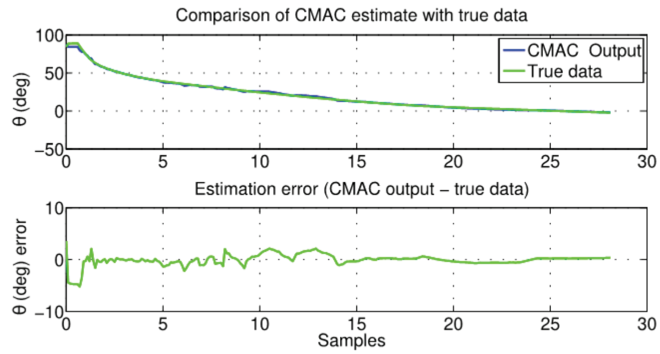

Figure 8. FV estimated attitude from trained CMAC.

4. GUIDANCE DEMAND TRACKING AUTOPILOT

During mid-course guidance $\hat{\theta}$ from ANN output is tracked by θ-tracking (Fig. 9). The attitude tracking autopilot is done by thrust vectoring control (TVC) and Aerodynamic Control (ADC) both, starting from launch till dynamic pressure builds up ADC can not be used. During initial phase of flight both TVC and ADC are used till booster end. Later with dynamic pressure build up ADC becomes effective. This section explains the modelling of the jet vane phase and the design of autopilot. The TFV is vertically launched and its attitude has to be controlled from the time it comes out of canister. The propulsion consists of both booster and sustainer till 8 seconds and design has been carried out to use TVC till propulsion end and then the TVC mechanism is ejected. According to the design control deflection for TVC as well as ADC are same due to same actuator. The state and output equations for controller design along pitch plane are [19]

$$\begin{bmatrix}\dot{\alpha}\\ \dot{q}\end{bmatrix}=\begin{pmatrix}-\dfrac{QSC_{n\alpha}}{mV_m} & 1\\ -\dfrac{QSC_{n\alpha h}}{I_{yy}} & 0\end{pmatrix}\begin{bmatrix}\alpha\\ q\end{bmatrix} + \begin{bmatrix}-\left(\dfrac{QSC_{n\delta a}}{mV_m}+\dfrac{C_{n\delta jv}}{mV_m}\right)\\ -\left(\dfrac{QSC_{n\delta a}l_c}{I_{yy}}+\dfrac{C_{n\delta jv}l_c}{I_{yy}}\right)\end{bmatrix}\delta_p \tag{8}$$

$$q_s=\begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}\alpha\\ q\end{bmatrix}+\begin{bmatrix}0\end{bmatrix}\delta_p \tag{9}$$

here, the transfer function (TF) of body pitch rate to pitch deflection is

$$\frac{q_s}{\delta_p}=\frac{K_b\left(1+\tau_\alpha s\right)}{\left(\dfrac{s^2}{\omega_{af}^2}+\dfrac{2\varsigma_{af}s}{\omega_{af}}+1\right)} \tag{10}$$

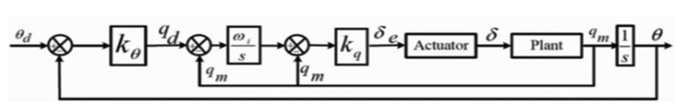

Figure 9. Schematic diagram of Θ-tracking autopilot (mid-course guidance).

where $\left(C_{n\delta a}, C_{n\delta jv}\right)$ are (ADC, TVC) control characteristics, $\left(\alpha, q_s, \delta_P\right)$ are (angle of attack, sensed pitch rate, fin deflection),

$$\left(V_m, I_{yy}\right) are\,(Velocity, moment\ of\ inertia)$$
$$along\ pitch\ plane$$

$$K_b = -\frac{1}{mV_m}\left(QSC_{n\delta a} + C_{n\delta jv}\right)\left(\frac{l_c}{h} - 1\right)$$
$$T_\alpha = \frac{mV_m}{QSC_{n\delta a} + \left(1 - \frac{h}{l_c}\right)}$$
$$\omega^2_{af} = \frac{QSC_{n\alpha}}{I_{yy}} \tag{11}$$
$$m_\eta = -\frac{QSC_{n\delta eqv}\left(x_{cp\delta} - x_{cg}\right)}{I_{yy}}$$
$$C_{n\delta eqv} = C_{n\delta a} + \frac{C_{n\delta jv}}{QS}$$
$$K_q = \frac{\omega_c}{m_\eta}$$

In frequency domain design of autopilot second order actuator of (natural frequency, damping ratio) as $\left(\omega_a, \varsigma_a\right) = $ (30 *Hz,* 0.6) and loop delay of 5 mili seconds has been considered. These numbers are based on hardware characteristics. Generally inner loop gain crossover frequency $\omega_C = \frac{\omega_a}{5}$, $\omega_i = \frac{\omega_c}{3}$ and $K_\theta = \frac{\omega_i}{3}$ for maximum separation among different loops[19]. In present case $\left(\omega_c, \omega_i, K_\theta\right)$ is (6, 2, 1) *Hz*. Design ensures (*GM, PM*) to be more than (6 dB, 30 deg). The phase cross over frequency ω_P should be $2\omega_C$. We wish to state that present design is preliminary in nature and ω_P has not been fine tuned. Subsequently in detailed design all stringent specifications will be met through proper compensator design. The TFV pitch plane flight dynamical equations (Eqn 12) are based on Fig. 10. The designed autopilot as above has been integrated with pitch plane equation. During mid-course guidance the autopilot tracking of ANN estimated $\hat{\theta}$ is shown in Fig. 11. At present study realistic actuator model has not been used. So tracking

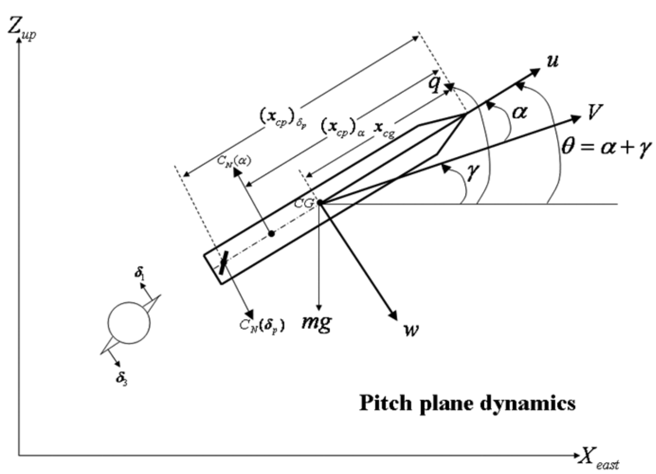

Figure 10. TFV equations of motion along pitch plane.

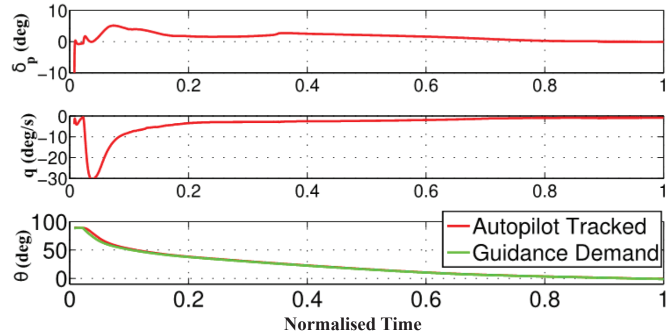

Figure 11. θ_d - tracking autopilot along with (δ_p , q).

performance along with $\left(\delta_p, q\right)$ time history is very smooth. The latax autopilot is used to track guidance demanded pitch latax[19].

$$m\left(\dot{u} + qw\right) = T_x - \frac{1}{2}\rho V^2 SC_X - mg\sin\theta$$
$$m\left(\dot{w} + qu\right) = T_z - \frac{1}{2}\rho V^2 SC_N\left(\alpha\right)$$
$$+\frac{1}{2}\rho V^2 SC_N\left(\delta_p\right) + mg\cos\theta \tag{12}$$
$$I_{yy}\dot{q} = M_y^p - \frac{1}{2}\rho V^2 SC_N\left(\alpha\right)\left(\left(x_{cp}\right)_\alpha - x_{cg}\right)$$
$$+\frac{1}{2}\rho V^2 SC_N\left(\delta_p\right)\left((x_{cp})_\delta - x_{cg}\right)$$
$$\dot{\theta} = q$$
$$\dot{x}_e = u\cos\theta + w\sin\theta$$
$$\dot{z}_u = u\sin\theta - w\cos\theta$$
$$\dot{\alpha} = q - \dot{\gamma}$$

5. SIMPLIFIED SEEKER MODEL AND PN GUIDANCE DURING HOMING

TFV is equipped with RF seeker and the available measurements are relative range, relative range rate, LOS angle and LOS rates $\left(r_l, \dot{r}_l, \gamma_l, \dot{\gamma}_l\right)$ between pursuer and evader. Present formulation being along pitch plane, corresponding state and measurement equations are given below. EKF has been used to solve present problem and are available in Sarkar[20], *et al.* The state equations consisting of (Δx, ΔV_x, a_{tx}, Δz, ΔV_z, a_{tz}) are

$$\Delta\dot{x} = \Delta V_x$$
$$\Delta\dot{z} = \Delta V_z$$
$$\Delta\dot{V}_x = a_{tx} - a_{mx} + \omega_{a_{mx}}$$
$$\Delta\dot{V}_z = a_{tz} - a_{mz} + \omega_{a_{mz}} \tag{13}$$
$$\dot{a}_{tx} = -\frac{1}{\tau_x} + \omega_{a_{tx}}$$
$$\dot{a}_{tz} = -\frac{1}{\tau_z} + \omega_{a_{tz}}$$

where state variables (Δx, ΔV_x, a_{tx}, Δz, ΔV_z, a_{tz}) are (relative position, velocity, target acceleration) along (*x, z*) axes (Fig. 10).

The target acceleration components are first order Gauss Markov with corelation time constant (τ_x, τ_z).

The random noise component is zero mean Gaussian noise $(\omega_{amx}, \omega_{amz})$. The corresponding process noise covariance's are $Q_{atx} = 2\frac{\sigma^2_{atx}}{\tau_x}$ and $Q_{atz} = 2\frac{\sigma^2_{atz}}{\tau_z}$. The measurements along the LOS

Frames are

$$\begin{aligned} r_{lm} &= \sqrt{\Delta x^2 + \Delta z^2} + \eta_1 \\ \dot{r}_{lm} &= \frac{\Delta x \Delta \dot{x} + \Delta z \Delta \dot{z}}{\sqrt{\Delta x^2 + \Delta z^2}} + \eta_2 \\ \gamma_{lm} &= \tan^{-1}\left(\frac{\Delta z}{\Delta x}\right) + \eta_3 \\ \dot{\gamma}_{lm} &= \frac{\Delta \dot{z} \Delta x - \Delta z \Delta \dot{x}}{\Delta x^2 + \Delta z^2} + \eta_4 \end{aligned} \tag{14}$$

here, $(\eta_1, \eta_2, \eta_3, \eta_4)$ are zero mean Gaussian measurement noise. So present estimation consists of Eqn (13) with six state variables $(\Delta x, \Delta z, \Delta V_x, \Delta V_z, a_{tx}, a_{tz})^T$ and measurement equation Eqn. (14) with four measurements $(r_{lm}, \dot{r}_{lm}, l_m, \gamma_m)^T$. Our aim is to obtain $V_c = \hat{\dot{r}}_1$ and LOS rate $\dot{\lambda} = \hat{\dot{\gamma}}_1$ and latax is generated along pitch plane as $\eta_z = N' V_c \dot{\lambda}$ For filter tuning $P_0 = diag$ (1.0E02, 1.0E02, 1.0E02, 1.0E02, 1.0E01, 1.0E01), $Q = diag$ (1$E-$05, 1E $-$Vc 05, 1$E-$05, 1$E-$05, 1$E-$01, 1$E-$01) have been used. The true measurement data has been contaminated by white noise with (*one σ*) measurement error as (10 m, 5 m/s, 0.1 deg, 0.5 deg/s). τ for process noise has been considered as 10 second. The numbers have been arrived at based on experience and RF seeker hardware manual. Time history of LOS rate estimation error corresponding to one MC sample of present engagement is shown in Fig. 12. This figure clearly indicates that noise attenuation is very good.

6. VALIDATION OF PROPOSED GUIDANCE THROUGH SIMULATION

Now let us validate above concepts through pitch plane simulation. First all different optimal trajectories upto PIP are generated (Sections (2-3)). Based on training of different optimal trajectories as ANN output $\hat{\theta}_d$ profile for (x_{pip}, h_{pip}) = (15, 5) km has been evolved. Initially based on evader velocity and altitude information from ground radar data PIP is known. Generally during mid-course phase evader does not maneuver. It starts maneuvering after seeker locks on during terminal guidance. So without loss of generality pursuer can reach PIP based on $\hat{\theta}_d$ tracking during ANN based mid-course guidance. This ensures no need of data link during mid-course. So initially evader is at (21, 5) km away from pursuer and it is approaching towards PIP at speed of 200 m/s. Once pursuer reaches PIP, seeker will lock on within range of 6 km range to go and system will switch over to terminal guidance. PN guidance is used from estimated seeker outputs (Section 5). Normalised Pursuer-evader trajectory is shown in Fig. 13.

Total 300 MC runs have been taken for ensuring system robustness. In present study seeker noise model and aero data uncertainty within bound of ±10 % has been considered. Histogram of handing over error before seeker lock on is depicted in Fig. 14. This error (157 m) is due to aerodynamic uncertainties. For a seeker lock on range of 6 km it leads to 1.5° which is well within seeker beam width ± 4° to acquire the target for switching over to terminal guidance. Cumulative Distribution Function (CDF) plot of miss distance is shown in Fig. 15. The miss distance as $(\bar{x}, \sigma)$ is (1.0, 0.6) m. This is due to seeker noise and aerodynamic uncertainty.

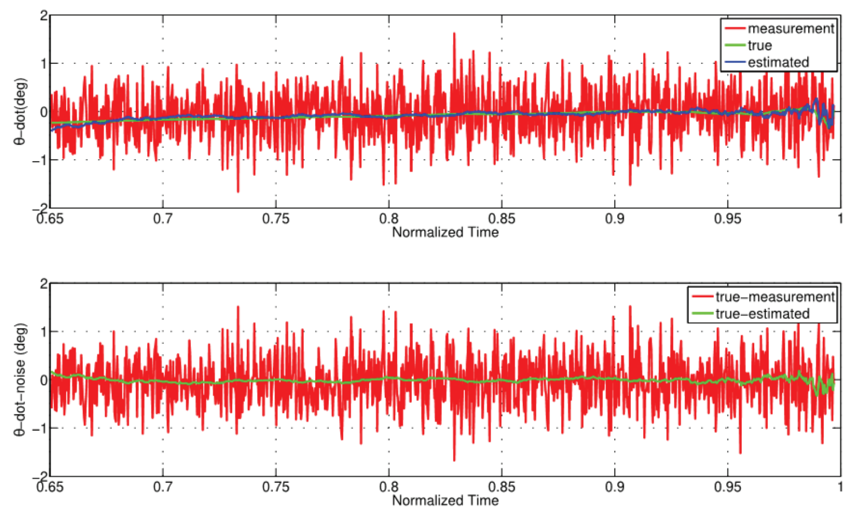

Figure 12. Time history of LOS rate estimation error (one MC sample of present engagement).

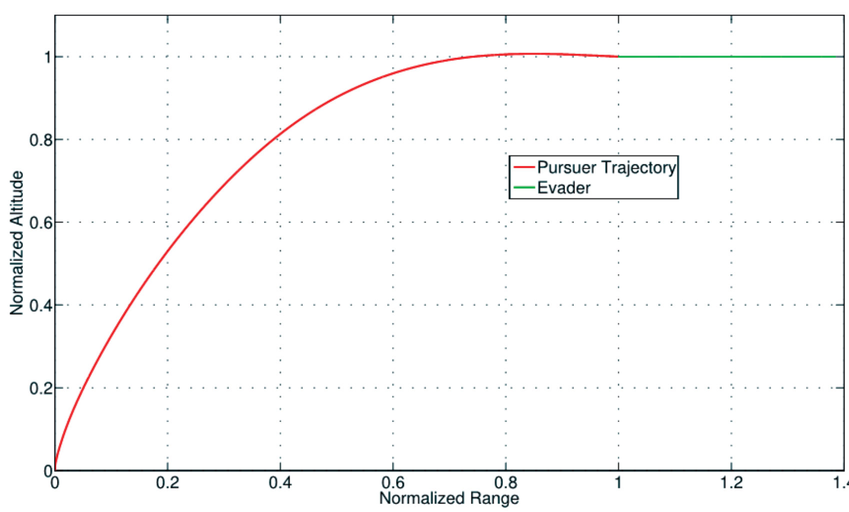

Figure 13. Normalised FV-target trajectory.

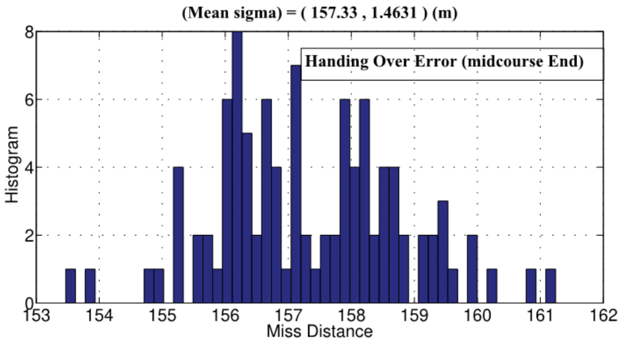

Figure 14. Histogram of handing over error (mid-course end).

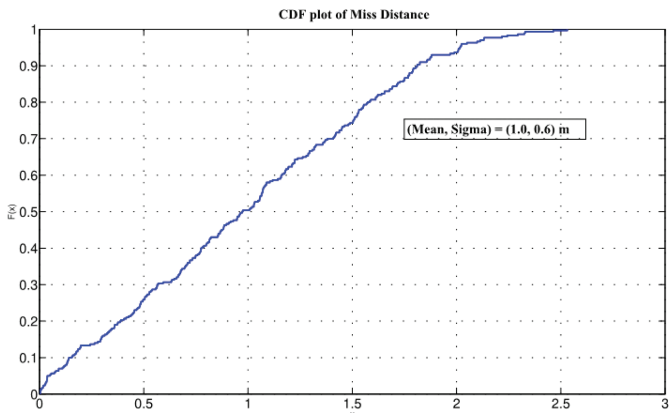

Figure 15. CDF plot of miss distance (m) (present engagement).

7. CONCLUSION AND FUTURE WORK

In the present investigation ANN based mid-course guidance law has been designed for the pursuer to reach PIP optimally with minimum flight time and maximum velocity at mid-course end. A cascaded CMAC based ANN has been developed for the pursuer to reach predesignated PIP with only pursuer kinematic information of range and altitude alone. Present algorithm needs no evader information eliminating the need of tracking radar and data link for tracking during mid-course guidance. The training set of optimal trajectories have been evolved by trajectory optimisation using direct method. ANN estimated $\hat{\theta}_d$ has been tracked by attitude hold autopilot during mid-course guidance and during seeker based terminal homing guidance demand has been tracked by latax autopilot. Complete concept has been validated along pitch plane 3 degrees of freedom model.

In present simulation actuator and time delay has not been included. The complete engagement study has to be carried out using high fidelity six degrees of freedom FV dynamical model in presence of seeker noise, actuator non-linearity, autopilot lag and aerodynamic as well as thrust uncertainty. Also for practical implementation it is required to train lot of input/output data set corresponding to different engagement conditions at different altitudes and engagement orientations, using DL technique[11–14]. At present this study is under progress.

REFERENCES

1. Song, E. J. & Tahk, M. J. Real-time neural network midcourse guidance. *Control Engineering Practice*, 2001, **9**(10), 1145-1154.
doi: 10.1016/s0967-0661(01)00058-2
2. Singh, Anup; Ghose, D. & Sarkar, A. K. Launch envelope optimization of virtual sliding target guidance scheme. *IEEE Trans. Aerospace and Electron. Syst.*, 2009, **45**(1), 899-918.
doi: 10.1109/taes.2009.5259173
3. Manickavasagam, M.; Sarkar, A.K. & Vaidyanathan, V. A singular perturbation based midcourse guidance law for realistic air -to-air engagement. *Def. Sci. J.*, 2017, **67**(1), 108-118.
doi: 10.14429/dsj.1.9236
4. Balakrishnan, S. & Biega, V. Near optimal midcourse guidance law for flight vehicle. *In* Proceedings of The American Control Conference, Washington, 1995, 2148-2152.
5. Lee, H.; Lee, Y. I.; Song, E. J.; Sun, B. C. & Tahk, M. J. Missile Guidance Using Neural Networks, *Control Engineering Practice,*1997, **5**, 753-762.
doi: 10.1016/s0967-0661(97)00059-2
6. Song, E.J. & Tahk, M.J, Real-time midcourse guidance with intercept point prediction. *Control Engineering Practice,* 1998, **6**(8), 957-967.
doi: 10.1016/s0967-0661(98)00041-0
7. Song, E.J. & Tahk, M.J. Real-time midcourse missile guidance robust against launch conditions. *Control Engineering Practice,* 1999, **7**(4), 507-515.
doi: 10.1016/s0967-0661(99)00010-6
8. Song, E.J. & Tahk, M.J. Three-dimensional midcourse guidance using neural networks for interception of ballistic targets. *IEEE Transaction of Aerospace and Electronics Systems,* 2002, **38**(1), 404-414.
doi: 10.1109/ TAES.2002.100897
9. Choi, M.L.; Tahk, M.J. & Bang, H.C. Neural network guidance based on pursuit-evasion games with enhanced performance. *Control Engineering Practice* 2006, **14**(7), 735-742.
doi: 0.1016/j.conengprac.2005.03.001.
10. Choi, H.L.; Parkt, Y.; Lee, H.G. & Tahk, M.J. A Three-dimensional diffrential game missile guidance law using neural networks. *AIAA Guidance, Navigation and Control Conference and Exhibit, Canada*, AIAA-2001-4343-CP, 2001.
doi: 10.2514/6.2001-4343
11. Mohammad, Reza Mortazavi & Farshad, Almasgan: Optimal midcourse guidance of an air-to-air missile via SVM and RVM. *Soft Computing, Springer*, 2018.
doi: 10.1007/s00500-018-3308-5.
12. Huang, L. & Chen, W. Deep learning midcourse guidance for interceptor missile. *In* IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC 2019), 2019.
doi: 10.1109/ITNEC.2019.8729311
13. Shalumov, V. Online Launch-Time Selection Using Deep Learning in a Target Missile Defender Engagement. *J. Aerospace Information Syst.*, 2019, **16**(6), 224-236.
doi: 0.2514/1.I010668
14. Shalumov, V. Cooperative online guide-launch-guide policy in a target-missile-defender engagement using

deep reinforcement learning. *Aerospace Sci. Technol.*, 2020, **104**, 1-12.
doi: 10.1016/j.ast.2020.105996

15. Betts, J.T. Practical Methods for Optimal Control Using Nonlinear Programming. *Advances in Design and Control, SIAM Publisher,* second edition, 2010.
16. Mukherjee, A.; Kar, P.K.; Sarkar, A.K. & Padhi, R. Range Extension Of An Air-to-air Engagement By Offline Trajectory Optimization. *AIAA Atmospheric Flight Mechanics Conference, Portland, Oregon,*Paper No: AIAA-CP-2011- 6339, August 8-11, 2011.
doi: 10.2514/6.2011-6339
17. Albus, J.S. A new approach to manipulator control: The cerebellar model articulation controller (CMAC). *J. Dynamic Systems, Measurements Control,* 1975, **97**(3), 220-227.
doi: 10.1115/1.3426922
18. Govindarao, V. Short term course in neural network in chemical process engineering. Technical Report, Indian Institute of Science, Bangalore, India, 1996, 29-49.
19. Zarchan, P. Tactical and strategic missile guidance. *AIAA Inc., Sixth edition,* 2002, 529-566.
20. Sarkar, A. K.; Ananthasayanam, M.R.; Srinivasan, T. & Kar, P.K. Comparison of the radar and seeker modes of pursuer guidance. *J. Guidance, Control Dynamics,* 2009, **32**(6), 1912-1920.
doi: 10.2514/1.40404

CONTRIBUTORS

Dr G. Naresh Kumar had graduate in Electrical and Electronics Engg. from JNTUC, Hyderabad in 1998 and post graduate in Electrical Engg. (Specialised in High Voltage Engg.) from Indian Institute of Sciences (IISc), Bangalore in 2000. He obtained PhD from DIAT, Pune in Guidance of Hypersonic Flight Vehicles. He joined DRDL in year 2000 as Scientist 'B' and currently is the Scientist-F and Head of "Special Weapon System Division" in Directorate of Systems of DRDL. His area of interest in Guidance, Navigation, Image Processing and System Modelling and Simulation. Artificial Intelligence (AI) techniques are applied for Aerospace applications.

His contribution in present study is on neural network training, complete problem formulation and generation of all results for paper writing.

Mr A.K. Sarkar had the graduate and post graduate education at IIT Kharagpur in Aeronautical Engineering (1975-80) and IIT Madras in Computer Science (1982-84). During 1982-84 he worked in design bureau of HAL Bangalore. He has worked as Scientist in DRDL, Hyderabad since (1984-2018). Being in DRDL he pursued Doctoral study in Engineering from IISc Bangalore (1996-2004). His research interests are simulation and modelling, estimation, guidance, control and optimisation techniques for flight vehicle system design. He superannuated as Technology Director (Systems) from DRDL in 2018. At present he is faculty in Aeronautical engineering department of ACS College of Engineering Bangalore.

His contribution in current study is towards optimisation formulation and overall writing of the paper.