Real-time midcourse guidance with consideration of the impact condition

Eun-Jung Song* and Miok Joh*
Korea Aerospace Research Institute, Daejon, Korea 305-333

Abstract

The objective of this study is to enhance neural-network guidance to consider the impact condition. The optimal impact condition in this study is defined as an head-on attack. Missile impact-angle error, which is a measure of the degree to which the missile is not steering for a head-on attack, can also have an influence on the final miss distance. Therefore midcourse guidance is used to navigate the missile, reducing the deviation angle from head on, given some constraints on the missile g performance. A coordinate transformation is introduced to simplify the three-dimensional guidance law and, consequently, to reduce training data. Computer simulation results show that the neural-network guidance law with the coordinate transformation reduces impact-angle errors effectively.

Key Word : neural networks, optimal guidance law, impact-angle condition

Introduction

The feasibility of the neural-network approximation has been investigated for a midcourse guidance problem [1-4]. In this paper, the neural-network approach is extended to consider the impact-angle condition of intercept problems of non-maneuvering targets decelerated by atmospheric drag (Fig. 1). One of the important functions of midcourse guidance is to reduce the impact-angle error. For ballistic targets, the closer the intercept geometry is to head on, the shorter the miss distance is (the probability of collision increases). A coordinate transformation is employed to extend the previous approach to consider the impact condition. The three-dimensional (3D) guidance law is decomposed into two commands; one is to guide the vertical missile motion in the transformed coordinate (TC) and the other is to regulate the horizontal motion. The neural network for the vertical motion is trained by optimal trajectories generated for the same flight-path angle condition but other terminal position conditions. Then the network outputs for the not-trained terminal conditions are obtained by transforming the input of the network. Therefore optimal trajectories obtained for various terminal flight-path angles are not needed for training data. The

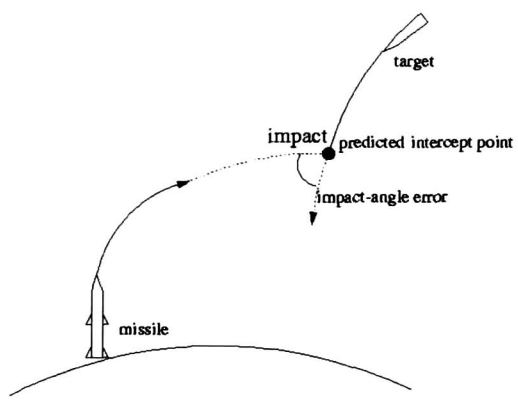


Fig. 1. Problem definition

* Senior Researcher
E-mail : ejsong@viva.kari.re.kr, TEL : 042-860-2994, FAX : 042-860-2233

estimation of the missile time to go, which is needed to predict the intercept point, is also simplified by considering the vertical motion in TC.

This paper is organized as follows: the mathematical model of typical medium-range missile is reviewed first. Then the midcourse guidance law using the coordinate transformation is explained. Third, the prediction algorithm is described. Finally, simulation results are presented to compare the performance of the proposed scheme with that of the optimal trajectory. The results are evaluated in terms of the impact-angle error as well as interception time and miss distance.

Mathematical model

The missile is modeled as a point mass and the state variables are the missile position in the ECEF (Earth Centered Earth Fixed) frame (r, τ, λ), the missile velocity relative to the NED (Navigation) frame v, and the flight-path angles γ and ψ. The control variables are the angle of attack α and the angle of total lift direction ϕ, called the bank angle. The equations of motion are given by

$$\dot{r} = v\sin\gamma \tag{1}$$

$$\dot{\tau} = \frac{v\cos\gamma\sin\psi}{r\cos\lambda} \tag{2}$$

$$\dot{\lambda} = \frac{v\cos\gamma\cos\psi}{r} \tag{3}$$

$$\dot{v} = \frac{(T\cos\alpha - D)}{m} - g\sin\gamma + r\Omega^2(\cos^2\lambda\sin\gamma - \cos\lambda\sin\lambda\cos\gamma\cos\psi) \tag{4}$$

$$\dot{\psi} = \frac{(T\sin\alpha + L)\sin\phi}{mv\cos\gamma} + \frac{v\sin\lambda\cos\gamma\sin\psi}{r\cos\lambda} + \frac{r\Omega^2\sin\lambda\cos\lambda\sin\psi}{v\cos\gamma} - \frac{2\Omega\cos\lambda\sin\gamma\cos\psi}{\cos\gamma} + 2\Omega\sin\lambda \tag{5}$$

$$\dot{\gamma} = \frac{(T\sin\alpha + L)\cos\phi}{mv} - \frac{g\cos\gamma}{v} + \frac{v\cos\gamma}{r} + \frac{r\Omega^2}{v}(\cos^2\lambda\cos\gamma + \sin\lambda\cos\lambda\sin\gamma\cos\psi) + 2\Omega\sin\psi\cos\lambda \tag{6}$$

where

$$L = \frac{1}{2}\rho v^2 S C_L, \quad C_L = C_{L_\alpha}(\alpha - \alpha_o)$$

$$D = \frac{1}{2}\rho v^2 S C_D, \quad C_D = C_{D_o} + k\,C_L^2$$

The aerodynamic derivatives C_{L_α}, C_{D_o}, and k are given as functions of Mach number M, which is a function of v and the altitude h:

$$C_{L_\alpha} = C_{L_\alpha}(M), \quad C_{D_o} = C_{D_o}(M), \quad k = k(M)$$

The missile mass and thrust are given as functions of time t:

$$m = m(t), \quad T = T(t)$$

Guidance law using coordinate transformation

The optimal flight trajectory formulated to reduce the impact-angle error is not always within a vertical plane, and the optimal missile motion cannot be always approximated by that on the vertical

plane. Therefore the guidance law designed in the two-dimensional (2D) space can not be easily extended to the 3D flight as the previous study [4]. In this study, a coordinate transformation is introduced to apply the previous extension method to this case. The transformation is determined by which the final missile velocity desired for the reduction of the error is within a vertical plane of TC. Then the neural network designed in the 2D space is employed to produce the reference vertical flight-path angle to satisfy the desired velocity direction in TC and the reference horizontal flight-path angle is calculated to regulate the missile motion within the plane. The reference angles are transformed inversely to the original coordinate (OC) and guidance commands to follow them are calculated by feedback linearization [5-7]. As long as the difference between OC and TC is small, the approximation by the coordinate transformation results in the optimal performance for the given impact condition.

Determination of coordinate transformation

Fig. 2 shows the relationship between TC and OC (the variable in TC is denoted as $(\cdot)'$. $\hat{i}'\,\hat{k}'$-plane is spanned by the final desired velocity and the desired position vector. The vertical flight-path angle of the velocity in TC is $\gamma_f{}^{\mathrm{ref}}$, those of optimal trajectories for training data. The vertical plane corresponds to the guidance plane in the previous study [4] and will be called guidance plane likewise.

The procedure of determining three vectors $\hat{i}'$, $\hat{j}'$, and $\hat{k}'$ is as follows.

1. First, $\hat{i}'$-axis is aligned with the final velocity required for a head-on attack :
   $\hat{i}'_o = \overrightarrow{v_f} \equiv (\cos\gamma_f \cos\psi_f,\ \cos\gamma_f \sin\psi_f,\ -\sin\gamma_f)$
2. $\hat{j}'$ is a normal vector to the guidance plane spanned by $\hat{i}'_o$ and
   $\overrightarrow{d} = (x_I^N, x_I^E, x_I^D) - (x_M^N, x_M^E, x_M^D)$ (7)
3. $\hat{k}'$ is obtained by rotating $\hat{k}_o'(=\hat{i}_o' \times \hat{j}')$ around $\hat{j}'$-axis by $-\gamma_f{}^{\mathrm{ref}}$. Then, $\hat{i}' = \hat{j}' \times \hat{k}'$

From $\hat{i}'$, $\hat{j}'$, and $\hat{k}'$, the coordinate transformation matrix is computed by

$$T = \begin{bmatrix} \hat{i}\cdot\hat{i}' & \hat{j}\cdot\hat{i}' & \hat{k}\cdot\hat{i}' \\ \hat{i}\cdot\hat{j}' & \hat{j}\cdot\hat{j}' & \hat{k}\cdot\hat{j}' \\ \hat{i}\cdot\hat{k}' & \hat{j}\cdot\hat{k}' & \hat{k}\cdot\hat{k}' \end{bmatrix} \quad (8)$$

and, $T^{-1} = T^T$ from the orthogonality condition of coordinate transformation matrix.

Fig. 3 shows the relationship between the guidance plane $\hat{i}'\,\hat{k}'$-plane and the vertical plane

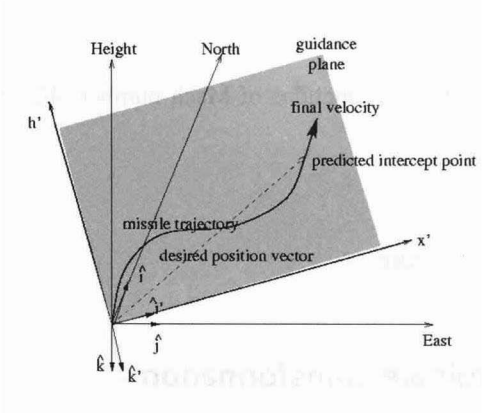


Fig. 2. Relationship between TC and OC

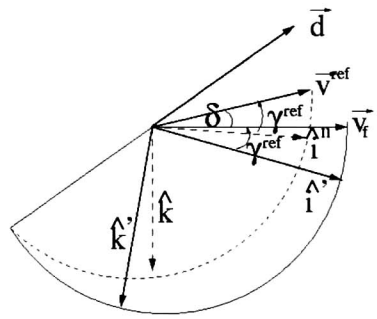


Fig. 3. Relationship between the guidance plane and the vertical planeof OC

of OC where neural-network guidance is obtained, $\hat{i}^n\hat{k}$-plane. Here, the direction of $\vec{v}^{\,ref}$ is defined by γ^{ref} and $\psi^{ref}=0$, $\vec{v}^{\,ref}=(\cos\gamma^{ref}, 0, -\sin\gamma^{ref})$. The difference δ between $\vec{v}_f$ and $\vec{v}^{\,ref}$, causes the difference between $\hat{i}^n\hat{k}$-plane and $\hat{i}'\hat{k}'$-plane. This induces the variation of flight environment in TC from that trained, such as the changes of launch conditions, gravity, and so on. Therefore to output the correct commands in TC neural networks must not be sensitive to the changes.

Reference flight-path angles

For the vertical missile motion in TC the γ-correction guidance law is used. As shown in the previous study, this guidance law is not sensitive to the changes of launch conditions, which can be induced by the coordinate transformation in this study. In this study, the previous γ-correction guidance is modified to consider the characteristic shape of optimal flight trajectories when final flight-path angles are constrained to γ_f^{ref} [8]; the optimal missile motion in the vicinity of the intercept point comes near to a straight line $h=h_f+(x-x_f)\tan\gamma_f$, as will be represented in the numerical results.

$$\gamma^* = \gamma_f + \{(h-h_f)-(x-x_f)\tan\gamma_f\}N(t, v, x-x_f, h-h_f) \tag{9}$$

where $N(\cdot)$ represents a neural network, which learns the optimal missile motion. Then the γ-correction command in TC is given by

$$\gamma^{*\prime} = \gamma_f^{ref} + (d_3' + d_1'\tan\gamma_f^{ref})N(t, v, -d_1', d_3') \tag{10}$$

where $\vec{d}' = T\vec{d}$. The horizontal flight-path angle to regulate the missile motion within the guidance plane in TC is given by

$$\psi^{*\prime} = \tan^{-1}\left(\frac{d_2'}{d_1'}\right) \tag{11}$$

From $\gamma^{*\prime}$ and $\psi^{*\prime}$, the reference flight-path angles in OC, γ^* and ψ^*, are computed as follows.

$$\vec{v}^{\,*\prime} \equiv (\cos\gamma^{*\prime}\cos\psi^{*\prime}, \cos\gamma^{*\prime}\sin\psi^{*\prime}, -\sin\gamma^{*\prime})$$

$$\vec{v}^{\,*} = T^T\vec{v}^{\,*\prime}$$

$$\gamma^* = \tan^{-1}\left(\frac{-v_3^*}{\sqrt{v_2^{*2}+v_1^{*2}}}\right)$$

$$\psi^* = \tan^{-1}(v_2^*, v_1^*)$$

Derivation of guidance commands

Two commands α and ϕ are designed to track γ^* and ψ^*. By assuming small α,

$$\dot{\gamma} = \frac{(T\sin\alpha+L)\cos\phi}{mv} + \Delta_\gamma \approx N_\alpha\alpha\cos\phi + \Delta_\gamma, \qquad N_\alpha = \left(\frac{1}{2}\rho v^2 S C_{L_\alpha} + T\right)/(mv) \tag{12}$$

$$\dot{\psi} = \frac{(T\sin\alpha+L)\sin\phi}{mv\cos\gamma} + \Delta_\psi \approx \frac{N_\alpha\alpha\sin\phi}{\cos\gamma} + \Delta_\psi \tag{13}$$

where Δ_γ and Δ_ψ represent the last terms of the RHS of Eqs. (6) and (5), respectively. The control inputs for γ and ψ correction are chosen as

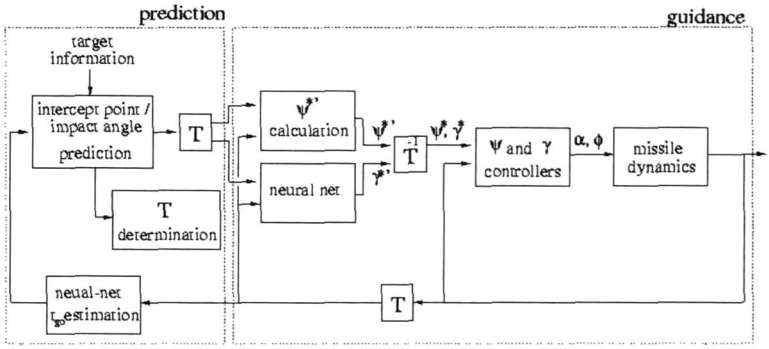


Fig. 4. Guidance loop using coordinate transformation

$$n_v \equiv a\cos\phi = \frac{k_{c_1}(\gamma^* - \gamma)}{N_a} \tag{14}$$

$$n_h \equiv a\sin\phi = \frac{k_{c_2}\cos\gamma(\psi^* - \psi)}{N_a} \tag{15}$$

where k_{c_1} and k_{c_2} are the inverse of the time constants of the closed-loop γ and ψ dynamics, respectively. Substitution of Eqs. (14) and (15) into Eqs. (12) and (13), respectively, gives the linearized γ and ψ dynamics

$$\dot{\gamma} \approx k_{c_1}(\gamma^* - \gamma) + \Delta_\gamma \tag{16}$$

$$\dot{\psi} \approx k_{c_2}(\psi^* - \psi) + \Delta_\psi \tag{17}$$

If $|\Delta_\gamma|$ and $|\Delta_\psi|$ are small, the proper choice of parameter k_{c_1} and k_{c_2} enables the missile to track γ^* and ψ^*. From Eqs. (14) and (15),

$$a = \text{sign}(n_v)\sqrt{n_v^2 + n_h^2} \tag{18}$$

$$\phi = \tan^{-1}\left(\frac{n_h}{n_v}\right) \tag{19}$$

Fig. 4 represents the guidance loop using the coordinate transformation. It shows that reduction of impact-angle errors requires the determination of the coordinate transformation and three times coordinate transformations.

Prediction of the intercept point using coordinate transformation

The target is a freely falling body in the central gravitational field. The intercept geometry is illustrated in Fig. 5. The method in Ref. [4] is used to predict the intercept point. The procedure required to compute the solution of the following equation is represented in Fig. 6.

$$t^m_{go}(\theta) - t^t_{go}(\theta) = 0 \tag{20}$$

The computation procedure in each block is summarized as follows.

- Blocks 1 through 3 : Given the target current states and the central angle θ,
  - ▸ predicted target position $I(\theta)$ [9] :

$$\frac{r_T}{r^{I(\theta)}} = \frac{1-\cos\theta}{\Lambda\cos^2\gamma_T} + \cos(\gamma_T + \theta)\cos\gamma_T, \qquad \Lambda = \frac{v_T^2}{\mu_\oplus / r_T}$$

  - ▸ predicted target flight-path angles at $I(\theta)$:

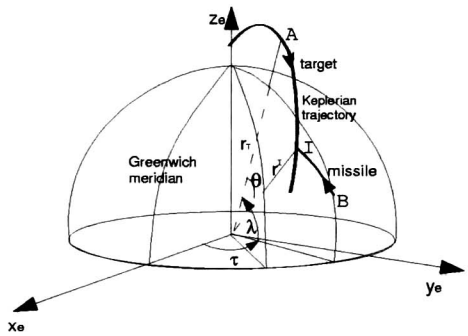


Fig. 5. Intercept geometry

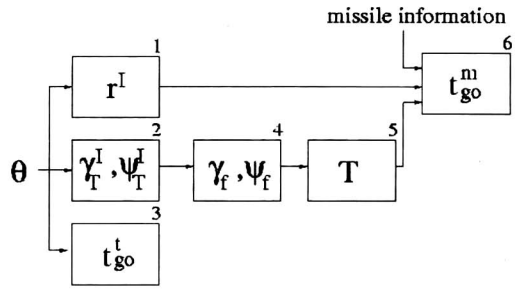


Fig. 6. Procedure required to compute Eq. (20)

$$\gamma_T^I(\theta) = \tan^{-1}\left[\frac{1}{\sin\theta}\left\{\frac{r^I}{r_T} - \frac{\mu_\oplus r^I(1-\cos\theta)}{(r_T v_T \cos\gamma_T)^2} - \cos\theta\right\}\right]$$

$$\psi_T^I(\theta) = \psi_T$$

▸ t_{go}^t :

$$t_{go}^t = \frac{r_T\{\tan\gamma_T(1-\cos\theta)+(1-\Lambda)\sin\theta\}}{v_T\cos\gamma_T\{(2-\Lambda)(1-\cos\theta)/(\Lambda\cos^2\gamma_T)+\cos(\gamma_T+\theta)/\cos\gamma_T\}} + \frac{2r_T}{v_T\Lambda(2/\Lambda-1)^{3/2}}\tan^{-1}\left[\frac{(2/\Lambda-1)^{1/2}}{\cos\gamma_T\cot(\theta/2)-\sin\gamma_T}\right]$$

- Block 4 : For a head-on collision,

$$\gamma_f(\theta) = -\gamma_T^I(\theta)$$

$$\psi_f(\theta) = \psi_T^I(\theta) - \mathrm{sign}(\psi_T^I(\theta))\cdot\pi$$

- Block 5 : Given $I(\theta)$, $\overrightarrow{v_f}$, and current missile position, the coordinate transformation matrix T is computed.
- Block 6 : A neural network is employed for estimating t_{go}^m. It is approximately estimated by considering only the vertical motion in TC. However, it has disadvantage that the procedure of determining T is required whenever θ is updated.

$$t_{go}^m = t_{go}^m(v, \gamma', -d_1', d_3')$$

Numerical Results

Neural-network training

Using the missile data in Ref. [4], the neural-network guidance law and t_{go}-estimator are designed. Initially, optimal trajectories computed under the condition in Table 1 are used for training data.

Table 1. Optimal trajectories for initial training data

cost : $\min J = t_f$
initial conditions : v_o=27m/s, γ_o= 90 °, $(x_o, h_o)=(0,0)$km
terminal conditions: γ_f= 40 °,
(x_f, h_f)={(40, 40), (40, 60), (40, 80),
(60, 40), (60, 60), (60, 80),
(80, 40), (80, 60), (80, 80)}km
inequality constraint : $\lvert\alpha\rvert \le 10$ °

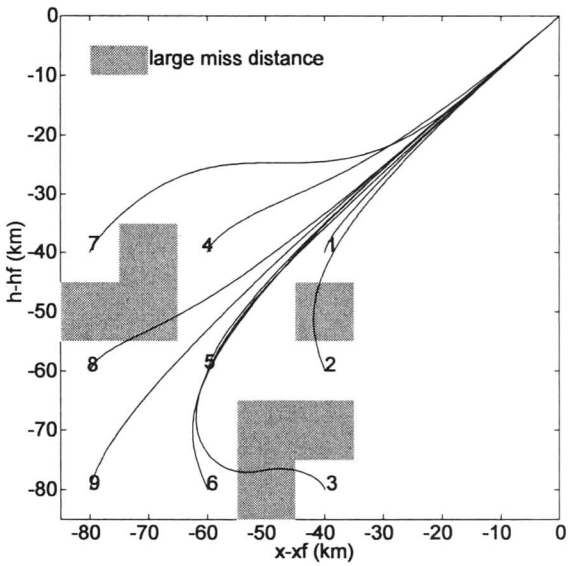


Fig. 7. Optimal flight trajectories for initial training data

Fig. 7 shows the optimal flight trajectory for each terminal condition. As the missile approaches the final point, the flight trajectories approximate to the straight line $h = h_f + (x - x_f)\tan\ \gamma_f$. Using this data, a neural network for guidance is trained. The neural network accepts t, v, $x - x_f$, $h - h_f$ as input variables, and outputs the value of $(\gamma - \gamma_f^{ref})/\{(h - h_f) - (x - x_f)\tan\ \gamma_f^{ref}\}$ (see Fig. 8). The pattern vectors which satisfy $(h - h_f) = (x - x_f)\tan\ \gamma_f^{ref}$ are excluded from the training set. The procedure of neural-network training is same as to the method in Ref. [1]. However, the obtained neural-network guidance law shows the bad generalization capability: large miss distances (> 2 km) as shown in Fig. 7. The different results from the previous study [4] are caused by the final constraint $\gamma_f = \gamma_f^{ref}$, which diminishes the similarity between optimal trajectories. The optimal γ trajectories in Fig. 9 also show this tendency. The trajectories 3 and 7 are more remarkable than others. To solve this problem, more optimal trajectories are added to the training data, and the trajectory 3 is excluded because many optimal trajectories are required to obtain good performance in the region near the point. Fig. 10 shows new optimal trajectories to improve training data, where the trajectories a1 through a6 represent the additional trajectories The more dense trajectories will improve intercept performance. A new neural network is trained again, and that for t_{go}-estimator is also trained by the same training data. The structure of the neural network for guidance is chosen as 2 hidden layers, where 12 and 10 units in each layer, respectively, and that of t_{go}-estimator consists of the same number of hidden layers with 8 and 7 units in each layer.

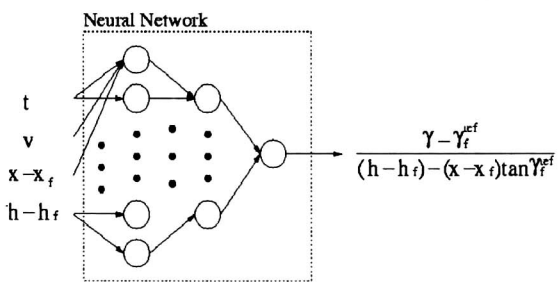


Fig. 8. Input elements and output variable of th eneural-net guidance law

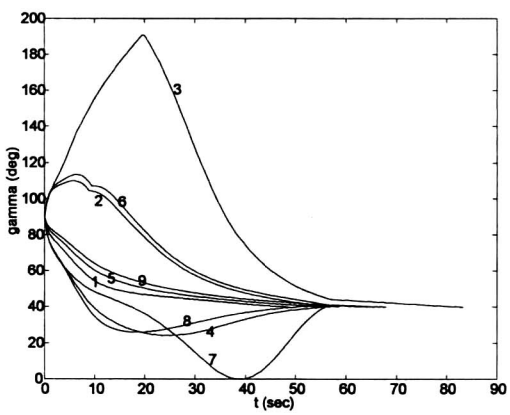
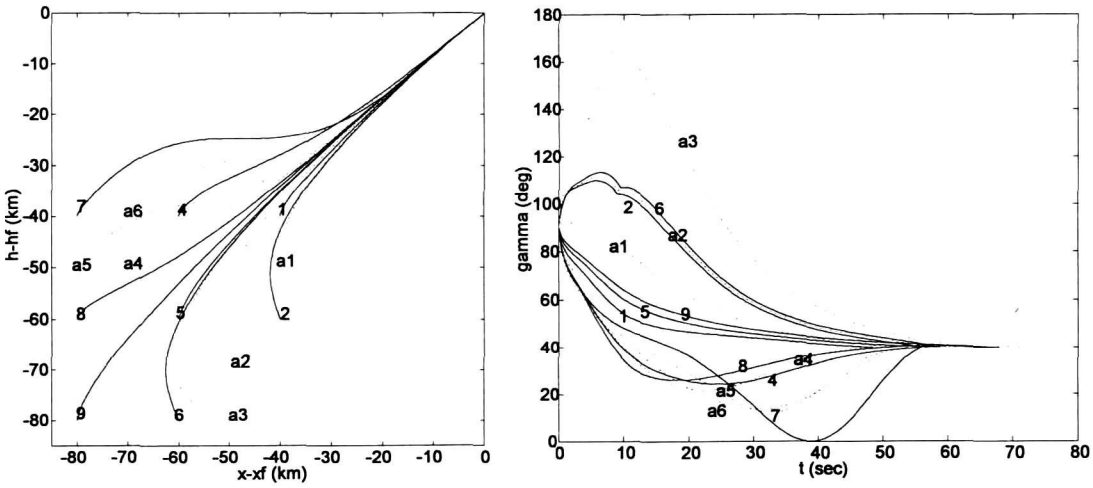


Fig. 9. Optimal vertical flight-path angles for initial training data



(a) flight trajectory (b) vertical flight-path angle

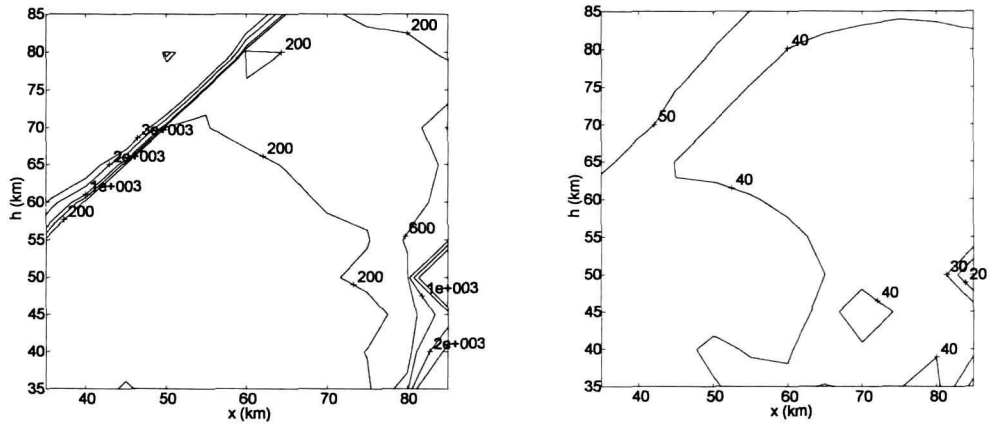
Fig. 10. Additional optimal trajectories to improve training data

Guidance law evaluation

The performance of the neural-network guidance law is investigated by computer simulation. The first simulation is used to check the degree of training and the generalization capability of neural networks in the 2D space. Fig. 11 shows the contour map of the miss distances and the terminal flight-path angles. These results are obtained by simulating the guidance loop for many points with 5km × 5km grid size when $\gamma_f = \gamma_f^{ref} (= 40^\circ)$. From this figures, it is observed the performance of the neural-network guidance is guaranteed in the region 40-80km × 40-80km except the surroundings of the point 3. In the region, the miss distances are less than 600m and the terminal flight-path angles are about 40°. Fig. 12 represents the time-to-go error distribution, which is defined by $\varepsilon(t_{go}) \equiv \frac{1}{t_f}\int_o^{t_f} |t_{go}^{true} - t_{go}^{estimated}| dt$. The terminal flight times obtained from the guidance-loop simulation are used to compute the true time to go t_{go}^{true}. The errors are less than 2 sec. From the test results in the 2D space, good performance is expected in the 3D space. It depends on the sensitives of the neural networks to the coordinate transformation.

Then, the guidance loop in Fig. 4 is evaluated; the simulation against the moving targets in the 3D space is performed. The feedback gains for γ and ψ correction are chosen as $k_{c_1} = 2$ and $k_{c_2} = 0.5$, respectively, and the predicted intercept point is updated at every 5 sec. If neural networks are not sensitive to the coordinate transformation, the region which shows the good performance in OC is preserved by the coordinate transformation. Three different scenarios are considered according to the initial target location and flight-path angles as shown in Fig. 13.

Table 2 shows the simulation results, where Optimal (free) represents the optimization results without the constraints of flight-path angles and Optimal (IA) represents the optimization results constrained by the terminal impact angles of NN. Reductions in the impact-angle error $\varepsilon(\delta)$ of NN is observed in all cases of NN. Since Case 3 has very small impact-angle error in Optimal (free), the differences between Optimal (free) and NN are small. The impact-angle errors of Case 1 and Case 2 decrease by 11° approximately, and this induces longer flight times than those of Optimal (free). When compared with Optimal (IA), the flight times of NN increase slightly, with the error smaller than 3.4%. This is because the 3D motion is simplified by the 2D in NN. From this results, it can be concluded that the 2D guidance law is extended to the case of the 3D flight using the coordinate transformation.



(a) miss distance (m) (b) final flight-path angle (deg)

Fig. 11. Contour map

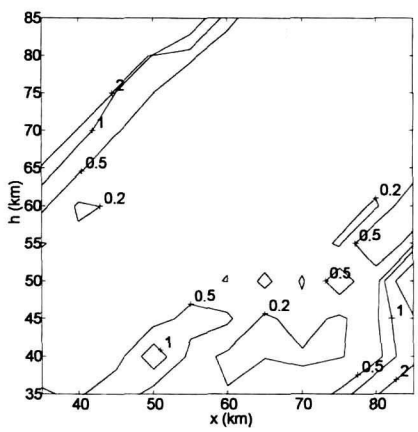


Fig. 12. time-to-go error (sec)

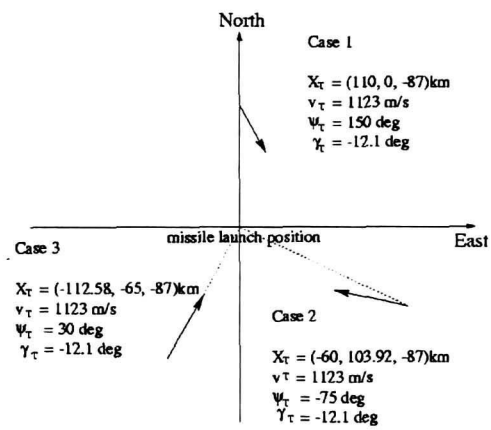
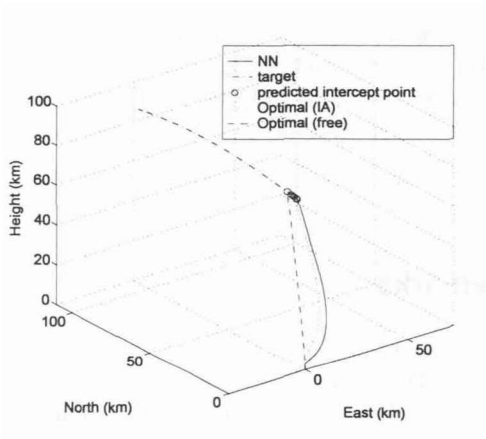


Fig. 13. Target initial conditions

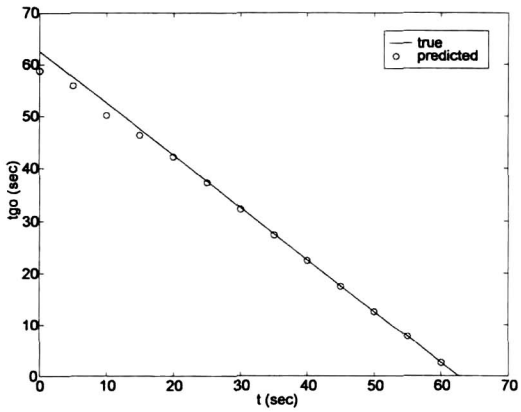
Table 2. Simulation results

target	criterion	Optimal (free)	Optimal (IA)	NN
Case 1	t_f (sec)	58.70	60.59	62.62 (3.35)
	MD (m)	-	-	101.72
	$\varepsilon(\delta)$ (deg)	48.13	37.02	37.01
Case 2	t_f (sec)	59.24	59.81	61.18 (2.23)
	MD (m)	-	-	226.11
	$\varepsilon(\delta)$ (deg)	25.52	14.02	14.02
Case 3	t_f (sec)	59.17	59.21	59.25 (0.07)
	MD (m)	-	-	121.60
	$\varepsilon(\delta)$ (deg)	1.63	-	0.16

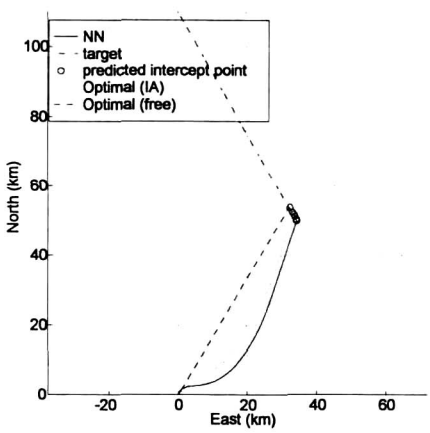
(·) : error(%) in t_f from that of Optimal (IA)



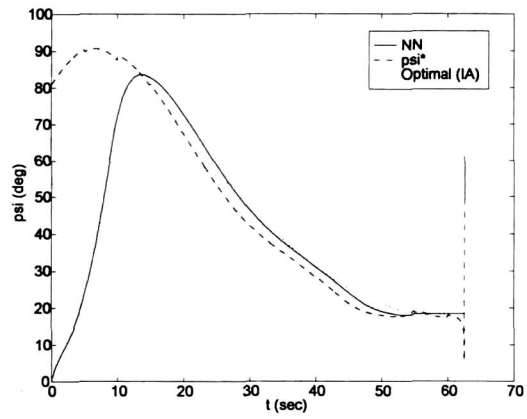
(a) 3D flight trajectory



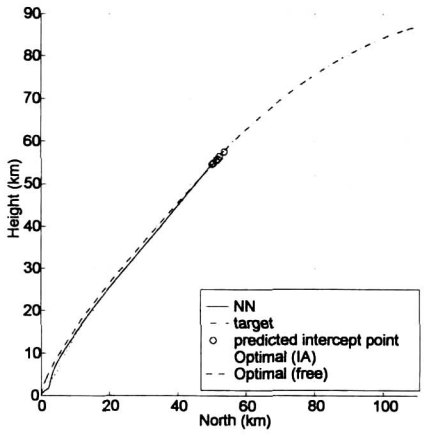
(d) time to go



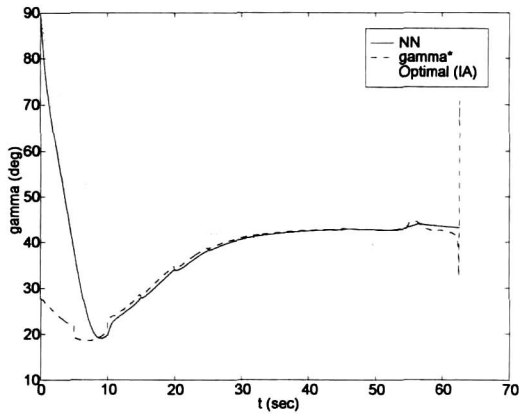
(b) flight trajectory in the N-E plane



(e) horizontal flight-path angle



(c) flight trajectory in the N-H plane



(f) vertical flight-path angle

Fig. 14. Simulation results of Case 1

Fig. 14 represents the simulation results of Case 1. The discrepancy from the flight trajectory of Optimal (IA) decreases as the missile approaches the intercept point in Figs. 14(a) through (c). To reduce the impact-angle error, the trajectory is more curved than that of Optimal (free). Although the initial errors of the flight-path angles induce the prediction error of t_{go}, it decreases as the flight time increases in Fig. 14(d). From Figs. 14(e) and (f), γ and ψ follow γ^* and ψ^*, respectively, and it enables the missile to reduce the impact-angle error.

Concluding Remarks

The neural-network guidance law is enhanced to reduce the impact-angle error. The coordinate transformation is introduced to simplify the design of the 3D guidance law. The missile motion in TC is decomposed into vertical and horizontal motion, respectively. The neural-network guidance law and t_{go}-estimator designed in the vertical plane are used. The neural networks are trained by optimal trajectories subject to the same terminal condition of the flight-path angle and other terminal position conditions. The terminal flight-path angle conditions which are not used in training are produced by the coordinate transformation. Computer simulation investigates the performance of the guidance law. The results show that the neural-network guidance law with the coordinate transformation reduces impact-angle errors and flight times are slightly larger than the optimal solution.

References

1. Song, E. J. and Tahk, M. J., "Real-time midcourse guidance with intercept point prediction," *Control Engineering Practice*, Vol. 6, No. 8, pp. 957-967, 1998.

2. Song, E. J. and Tahk, M. J., "Real-time midcourse guidance robust against launch conditions," *Control Engineering Practice*, Vol. 7, No. 4, pp. 507-515, 1999.

3. Song, E. J. and Tahk, M. J., "Real-time neural-network midcourse guidance," *Control Engineering Practice*, Vol.9, No.10, pp. 1145-1154, 2001.

4. Song, E. J. and Tahk, M. J., "Three-dimensional midcourse guidance using neural networks for interception of ballistic targets," *IEEE Trans. Aerosp. Electr. Syst.*, Vol.38, No.2, pp.404-414, 2002.

5. Slotine, J. J., "*Applied Nonlinear Control*," Prentice Hall, Englewood Cliffs, New Jersey, pp. 207-275, 1991.

6. Vidyasagar, M., "*Nonlinear Systems Analysis (2nd Ed.)*," Prentice-Hall, New Jersey, pp. 427-463, 1993.

7. Khalil, H. K., "*Nonlinear Systems*," Prentice-Hall, New Jersey, pp. 81-85, 1996.

8. Lagaris, I. E. and Likas, A. L., "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Trans. on Neural Networks*, Vol. 9, No. 5, pp. 987-1000, 1998.

9. Regan, F. J. and Anandarkrishnan, S. M., "Dynamics of Atmospheric Re-Entry," AIAA, Inc., Washington, DC, 1993.