Electronic Supplementary Materials

For https://doi.org/10.1631/jzus.A2500144

Time control entry guidance method for hypersonic glide vehicles based on deep reinforcement learning

Zhenyu LIU, Gang LEI, Yong XIAN, Leliang REN, Shaopeng LI, Daqiao ZHANG

Department of Missile Engineering, Rocket Force University of Engineering, Xi'an 710025, China

Section S1 Entry problem formulation

S1.1 Entry dynamics

The aerodynamic lift and drag forces L and D are expressed as follows:

$$\begin{cases} L = \dfrac{1}{2}\rho v^2 C_L S_{ref} \\ D = \dfrac{1}{2}\rho v^2 C_D S_{ref} \end{cases} \tag{S1}$$

where ρ is the atmospheric density; S_{ref} is the reference area of the vehicle; and C_L, C_D are the lift and drag coefficients, respectively. Additionally, the delay of actuator is also considered; that is,

$$\begin{cases} \dot{\Delta\alpha} = \dfrac{\Delta\alpha_c - \Delta\alpha}{\tau_a} \\ \dot{\Delta\sigma} = \dfrac{\Delta\sigma_c - \Delta\sigma}{\tau_b} \end{cases} \tag{S2}$$

where τ_a, τ_b are constants and the subscript c represents the command value.

Furthermore, in practice, the aerodynamic parameters of the vehicle and the atmospheric density are not precisely equal to the nominal values. Instead, deviations arise from the nominal values, so the precise values of the deviations cannot usually be determined in advance. Hence, we modeled the aerodynamic parameters and atmospheric density as follows:

$$\begin{cases} \rho = \rho^*(1+\Delta\rho) \\ C_i = C_i^*(1+\Delta C_i), i = L, D \end{cases} \tag{S3}$$

where superscript $*$ indicates the corresponding nominal value. $\Delta\rho$ and $\Delta C_i \left(i = L, D\right)$ are the percentage deviations of the atmospheric density and aerodynamic parameters from the nominal value, respectively. Here, we assume that the deviations of these parameters are uniformly distributed.

Section S2 Time control entry guidance algorithm

S2.1 Tracking the reference profile via DRL

S2.1.1 Observation space

The observation vector is defined as follows:

$$\boldsymbol{o}_t = \left[V_{err}, s, v, \alpha, \sigma, h_{err}, \Delta\psi_u, \Delta\psi_l, n_{ac} \right]^T \tag{S4}$$

where $V_{err} = V^N - V_{ref}^N$ is the reference-velocity: tracking error, which can be calculated as follows:

$$\begin{cases} \boldsymbol{V}^N = v \cdot \left[\cos\gamma\cos\psi, \sin\gamma, \cos\gamma\sin\psi \right]^T \\ \boldsymbol{V}_{ref}^N = v_{ref} \cdot \left[\cos\gamma_{ref}\cos\psi_{ref}, \sin\gamma_{ref}, \cos\gamma_{ref}\sin\psi_{ref} \right]^T \end{cases} \tag{S5}$$

And s represents the range-to-go, v denotes the velocity of the HGV, $h_{err} = h - h_f^*$ is the error relative to the desired terminal altitude. $\Delta\psi_u, \Delta\psi_l$ represent the differences between the heading angle ψ and the upper and lower boundary values of heading angle ψ_u, ψ_l, respectively (As shown in Fig. S1). Their expressions are as follows:

$$\begin{cases} \Delta\psi_u = \psi_u - \psi = \psi_{LOS} + \delta\psi_{thr}(v) - \psi \\ \Delta\psi_l = \psi - \psi_l = \psi - \psi_{LOS} + \delta\psi_{thr}(v) \end{cases} \tag{S6}$$

The primary purpose of introducing $\Delta\psi_u, \Delta\psi_l$ is to indicate to the agent when to perform a bank angle reversal. Specifically, these parameters provide a quantitative measure for how close the heading error is to the lower or upper bounds of the permissible error corridor. When $\Delta\psi_u$, $\Delta\psi_l$ approaches zero, it signals that the heading error has reached an edge of the corridor, thereby triggering the agent to consider reversing its bank angle to correct the heading error.

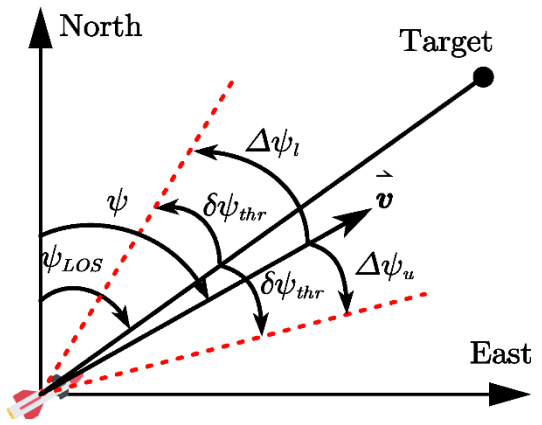

Fig. S1: The schematic diagram of $\Delta\psi_u$ and $\Delta\psi_l$

Furthermore, n_{ac} represents the actual aerodynamic overload, which can be directly obtained from the apparent acceleration output of the inertial measurement unit (IMU). That is

$$n_{ac} = \frac{\left\| \boldsymbol{a}_{appr} \right\|}{g_0} \tag{S7}$$

where $\boldsymbol{a}_{appr}$ is the apparent acceleration output from the IMU.

S2.1.2 Reward function design

To ensure that HGVs accurately tracks the reference flight profile and consistently meets path constraints, the reward function is formulated as follows:

$$r_t = r_{shaping} + r_{alt} + r_{ctrl} + r_{proc} + r_{rev} + r_{bonus} \tag{S8}$$

Here, $r_{shaping}$ represents the shaping reward, which primarily serves to incentivize the agent to track a designed reference flight profile. It can be calculated as follows:

$$r_{shaping} = \beta_{shaping} \exp\left(-\frac{\left\| \boldsymbol{V}_{err} \right\|^2}{\kappa_v^2} \right) \tag{S9}$$

where $\beta_{shaping}$ is the shaping reward weights, and $\kappa_v = 0.1v$ is the shaping-reward scaling factor.

r_{alt} represents the altitude error reward signal, which is designed to incentivize the agent to control the HGV to meet the terminal altitude constraints. Its expression is given as follows:

$$r_{alt} = -\beta_{alt} \left| \frac{\sin\gamma - \sin\gamma_{ref}}{\kappa_{alt}} \right|^2 \tag{S10}$$

where β_{alt} is the altitude reward weights, and $\kappa_{alt} = 0.1\sin\gamma$ is the altitude reward scaling factor.

r_{ctrl} denotes the penalty for the control effort, designed to limit the magnitude of actions to avoid violations of the restrictions. It can be calculated as follows:

$$r_{ctrl} = -\beta_{ctrl} \left\| \boldsymbol{a}_t \right\|^2 \tag{S11}$$

where β_{ctrl} is a positive constant.

r_{proc} represents the path constraint penalty signal, aimed at penalizing states that might violate constraints to incentivize the agent to avoid such violations as much as possible. Its expression is as follows:

$$r_{proc} = -\beta_{proc} \left[\exp\left(\frac{\dot{Q} - \dot{Q}_{max}}{\dot{Q}_{max}} \right) + \exp\left(\frac{q - q_{max}}{q_{max}} \right) + \exp\left(\frac{n - n_{max}}{n_{max}} \right) \right] \tag{S12}$$

where β_{proc} is a positive constant.

r_{rev} denotes the penalty for bank angle reversal, applying a negative reward signal for each instance of bank angle reversal to discourage the agent from frequently reversing the bank angle of the HGV. It can be calculated as follows:

$$r_{rev} = -\beta_{rev} \left(B_t - B_{t-1} \right) \tag{S13}$$

where β_{rev} is a positive constant. and B_t is the number of the reverses of the bank angle up to the *t*-th time step.

r_{bonus} represents the terminal bonus reward, which incentivizes the agent to control the HGV to reach the terminal state with a specified precision. Its expression is as follows:

$$r_{bonus} = \begin{cases} \xi & \text{if Eq. (8) met and done.} \\ 0 & \text{others} \end{cases} \tag{S14}$$

where ξ is a positive constant.

S2.2 Terminal time control guidance method

S2.2.1 Feasible terminal range

It is evident that a wider heading error corridor leads to a larger heading error. Then, according to Eq.(10), a larger $\delta\psi_0$ corresponds to a wider heading error corridor. Therefore, for a given initial state $\tilde{\boldsymbol{z}}_0$, the minimum and maximum flight times can be estimated by setting $\delta\psi_0$ to its minimum value $\delta\psi_0^{\min}$ and maximum value $\delta\psi_0^{\max}$, respectively. That is,

$$\begin{cases} t_f^{min} = \Phi\left(\tilde{\boldsymbol{z}}_0, \delta\psi_0^{min}\right) \\ t_f^{max} = \Phi\left(\tilde{\boldsymbol{z}}_0, \delta\psi_0^{max}\right) \end{cases} \tag{S15}$$

Moreover, considering environmental uncertainties such as aerodynamic parameter errors, it is a challenge to ensure the terminal accuracy when set- ting the terminal time t_f^* close to t_f^{min} or $t_f^{\max}$. Therefore, a safety factor k_t is introduced to narrow the time range, that is,

$$\begin{cases} \tilde{t}_f^{min} = \left(1-k_t\right)t_f^{min} + k_t t_f^{max} \\ \tilde{t}_f^{max} = k_t t_f^{min} + \left(1-k_t\right)t_f^{max} \end{cases} \tag{S16}$$

where $k_t > 0$ is a positive constant.

Section S3 Experiments and results

S3.1 DRL training results and test results

To ensure the guidance policy exhibits good generalization ability, we trained it on a range of initial conditions, which were randomly generated at the start of each episode. The full set of initial conditions and terminal states are given in Table S1.

Notably, the target longitude λ_f^* was automatically generated as the follows:

$$\lambda_f^* = \arccos\left(\frac{\cos s_0 - \sin\phi_0 \sin\phi_f^*}{\cos\phi_0 \cos\phi_f^*}\right) + \lambda_0 \tag{S17}$$

Table S1: Initial and terminal state dispersion in the offline training

Parameter	Value	Parameter	Value
Init range s_0 (km)	[9500, 10500]	Init altitude h_0 (km)	[79.8, 80.2]
Init longitude λ_0 (deg)	[0, 20]	Init latitude ϕ_0 (deg)	[10, 20]
Init velocity v_0 (m/s)	[6750, 6850]	Init flight path angle γ_0 (deg)	[−0.5, 0]
Init heading error ψ_{err0} (deg)	[−0.1,0.1]	Heading offset parameter $\delta\psi_0$ (deg)	[5.0, 45.0]
Terminal latitude ϕ_f^*(deg)	[20, 30]	Terminal range-to-go s_f^*(km)	95
Terminal altitude h_f^*(km)	28	Terminal velocity v_f^*(m/s)	2000
Range error tolerance δs_f(km)	0.5	Altitude error tolerance δh_f(km)	0.3
Velocity error tolerance δv_f(m/s)	50	Heading error tolerance $\delta\psi_f$(deg)	5

Moreover, a four-layer neural network structure was used to implement the policy and value network (Henderson et al., 2018), wherein the first hid- den layer was a recurrent layer implemented using GRUs. The network architectures is shown in Table S2, and the specific parameters of PPO algorithm are listed in Table 4.The coefficients for these terms, listed in Table S3, were determined through an iterative and empirical tuning process. Our approach involved first establishing a baseline tracking policy using only the shaping and bonus rewards. Subsequently, the penalty weights were introduced and fine-tuned to improve the smoothness and efficiency of the flight trajectory without compromising the primary objectives. This systematic approach ensures a balanced reward signal that leads to a robust and effective guidance policy.

Table S2: Policy and value network structures

	Policy network		Value network	
Layer	Size	Activation	Size	Activation
Input	(11,110)	tanh	(11,110)	tanh
GRU	(110,46)	tanh	(110,46)	tanh
Hidden	(46,20)	tanh	(46,5)	tanh
Output	(20,2)	linear	(5,1)	linear

Table S3: Details parameters of the PPO algorithm

Parameter	Value	Parameter	Value
Maximum epochs	3000	Collected episodes per epoch	60
Discounting factor	0.99	Value network learning rate α_ω	2.0×10^{-3}

Policy network learning rate α_θ	1.5×10−4	Clipped factor ϵ	0.1
Shaping reward coefficient $\beta_{shaping}$	0.25	Altitude reward weight β_{alt}	0.05
Control reward weights β_{ctrl}	0.02	Path constraints reward weights β_{path}	0.05
Bank angle reversing reward weights β_{rev}	0.1	Bonus reward ξ	10.0

The training results are shown in Fig. S2-Fig. S4. In Fig. S2, the reward curve is plotted on the left y-axis, while the terminal velocity error curve is depicted on the right y-axis. Fig. S3 illustrates the terminal range and altitude error curves on the left and right y-axes, respectively. As illustrated in Fig. S3 ,and Fig. S4, after collecting 60,000 episodes of data, the terminal velocity, altitude, and range errors have converged, and the reward curve has stabilized. This confirms the effectiveness of the method proposed in Sect. 3.2.

Moreover, Fig. S4 illustrates the variation curve of episode length during the training process. It is evident that the episode length rises quickly in the initial training stage and subsequently stabilizes with convergence. This shows that the agent initially acquires the ability to meet various path constraints to prevent episode termination in the early training period, and then gradually learns to minimize the terminal error in the later training period. This confirms the effectiveness of the episode early termination strategy employed in the training phase of this study.

It is worth noting that during the training process, once the path constraints are violated, the current episode is immediately terminated, which also leads to a significant increase in terminal error. However, as can be observed from Fig. S2 and Fig. S3, with the continuous progress of training, the terminal error gradually converges to near zero. This further demonstrates that the agent has not only learned how to track the designed reference flight profile, but also acquired the ability to satisfy the path constraints.

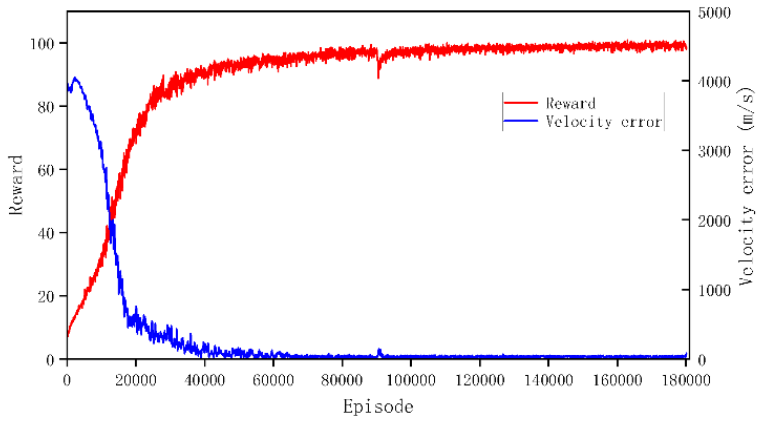

Fig. S2: Reward and velocity error curves

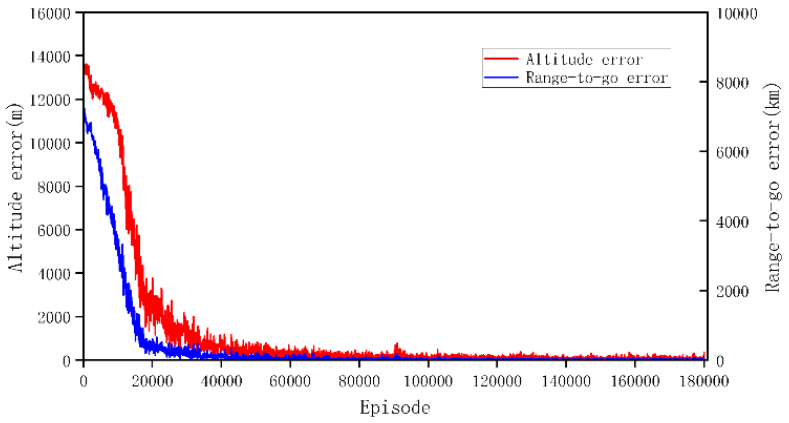

Fig. S3: Altitude and range-to-go error curves

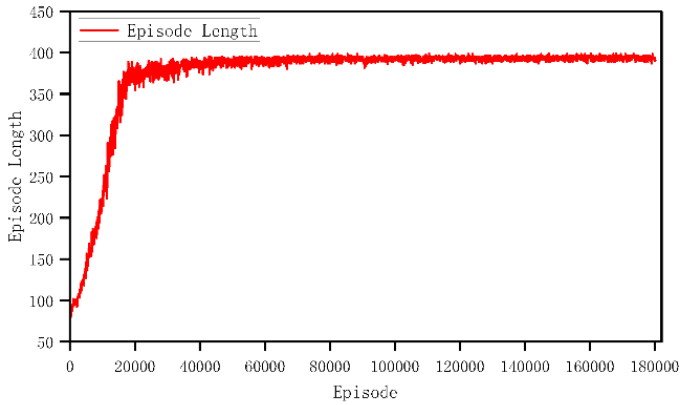

Fig. S4: Episode length curves

S3.2 Results for remaining flight time prediction

To generate the training data for the residual prediction network proposed in Sect. 3.3.1, we utilized the trained agent based on the results from Sect. 4.1 and collected 4000 flight trajectories under the initial conditions shown in Table S1, resulting in a total of 1688461 data samples. Additionally, Fig. S5. illustrates the statistical distribution histogram of the terminal times for the 4000 collected trajectories. The entire data set was divided into a training set and a validation set in a 9:1 ratio. The reason for not partitioning a separate test set is that new trajectories can be generated as test data by resetting the initial conditions and utilizing the well-trained agent from Sect 4.1.

The residual prediction network adopts a fourlayer network structure with the ReLU activation function, where the second layer is an LSTM layer. Among them, the input layer contains 128 neurons, the LSTM layer contains 64 neurons, the hidden layer contains 32 neurons, and the output layer directly outputs the estimated residual of time-to-go. The loss function used is the Mean Squared Error (MSE) loss. The parameters of the training process are as follows: learning rate lr = 0.001, batch size B = 4096, and the network was trained for 100 epochs. The curves of the loss function during the training process is shown in Fig. S6. As can be seen from Fig. S6, the loss functions for both the training set and the validation set converged rapidly during the training process, achieving the desired performance.

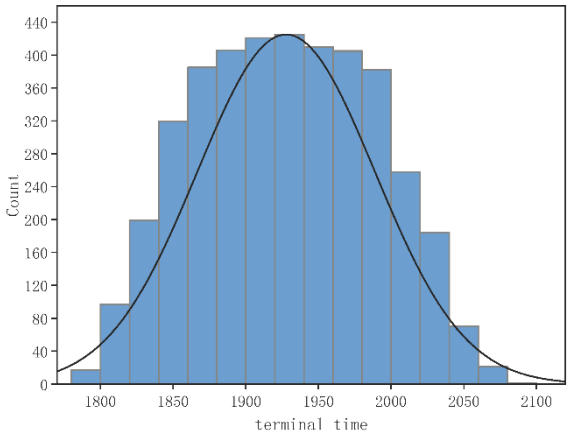

Fig. S5: The histogram of the terminal times for the collected 4000 trajectories.

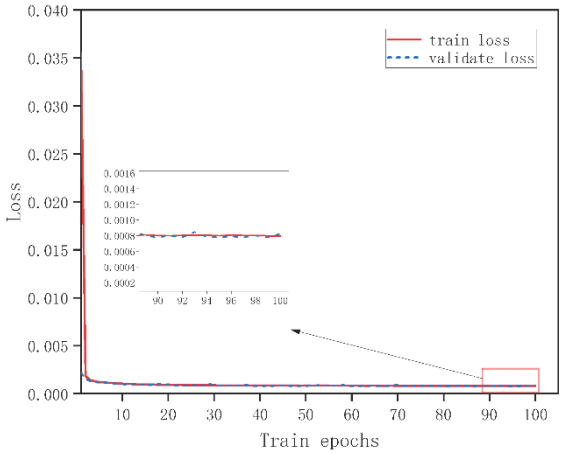

Fig. S6: The loss curves during the training process.

References

Bao C, Li X, Xu W, et al., 2025. Coordinated reentry guid- ance with a* and deep reinforcement learning for hy- personic morphing vehicles under multiple no-fly zones. Aerospace, 12(7).
https:// doi.org/10.3390/aerospace12070591

Chai R, Tsourdos A, Savvaris A, et al., 2021. Review of advanced guidance and control algorithms for space/aerospace vehicles. Progress in Aerospace Sciences, 122:100696.
https://doi.org/10.1016/j.paerosci.2021.100696

Cheng L, Jiang F, Wang Z, et al., 2021. Multiconstrained real-time entry guidance using deep neural networks. IEEE Transactions on Aerospace and Electronic Systems, 57(1):325-340.
https://doi.org/10.1109/TAES.2020.3015321

Christopher WB, Lu P, 2012. Comparison of fully numerical predictor-corrector and apollo skip entry guidance algorithms. The Journal of the Astronautical Sciences, 59:517–540.
https://doi.org/10.1007/s40295-014-0005-1

Chung JY, Gulcehre C, Cho K, et al., 2015. Gated feedback recurrent neural networks. ICML'15: Proceedings of the 32nd International Conference on International Conference on Machine Learning, 37:2067-2075.
https://doi.org/10.5555/3045118.3045338

Gao Y, Zhou R, Chen J, 2024. Integrated entry guidance with no-fly zone constraint using reinforcement learning and predictor-corrector technique. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 238(5):728 - 741. https://doi.org/10.1177/09544100241236995

Gaudet B, Drozd K, Furfaro R, 2022. Adaptive approach phase guidance for a hypersonic glider via reinforcement meta learning. AIAA SCITECH 2022 Forum, p.1-19. https://doi.org/10.2514/6.2022-2214

Guo Y, Li X, Zhang H, et al., 2020. Entry guidance with terminal time control based on quasi-equilibrium glide condition. IEEE Transactions on Aerospace and Elec- tronic Systems, 56(2):887-896.
https://doi.org/10.1109/TAES.2019.2921213

Harpold JC, Gavert DE, 1983. Space shuttle entry guidance performance results. Journal of Guidance, Control, and Dynamics, 6(6):442-447.
https://doi.org/10.2514/3.8523

Henderson P, Islam R, Bachman P, et al., 2018. Deep re- inforcement learning that matters. Proceedings of the Thirty-Second AAAI Conference on Artificial Intelli- gence, New Orleans, Louisiana, USA, p.1-8.
https://doi.org/10.5555/3504035.3504427

Hu Q, Cao R, Han T, et al., 2021. Field-of-view limited guidance with impact angle constraint and feasibility analysis. Aerospace Science and Technology, 114:106753.
https://doi.org/10.1016/j.ast.2021.106753

HU Y, GAO C, LI J, et al., 2022. A novel adaptive lateral reentry guidance algorithm with complex distributed no-fly zones constraints. Chinese Journal of Aeronautics, 35(7):128–143.
https://doi.org/10.1016/j.cja.2021.06.016

Kim HG, Lee JY, Kim HJ, et al., 2020.Look-angle-shaping guidance law for impact angle and time control with field-of-view constraint. IEEE Transactions on Aerospace and Electronic Systems, 56(2):1602–1612.
https://doi.org/10.1109/taes.2019.2924175

Lee S, Lee Y, Kim Y, et al., 2023. Impact angle control guidance considering seeker's field-of-view limit based on reinforcement learning. Journal of Guidance, Control, and Dynamics, 46(11):2168-2182.
https://doi.org/10.2514/1.G007715

Li J, Zhang G, Shan Q, et al., 2022. A novel cooperative design for usv–uav systems: 3-d mapping guidance and adaptive fuzzy control. IEEE Transactions on Control of Network Systems, 10(2):564-574.
https://doi.org/10.1109/TCNS.2022.3220705

Li Z, Hu C, Ding C, et al., 2018. Stochastic gradient particle swarm optimization based entry trajectory rapid planing for hypersonic glide vehicles. Aerospace Science and Technology, 76:176–186.
https://doi.org/10.1016/j.ast.2018.01.033

Li Z, He B, Wang M, et al., 2019. Time-coordination entry guidance for multi-hypersonic vehicles. Aerospace Science and Technology, 89:123-135.
https://doi.org/10.1016/j.ast.2019.03.056

Liang Z, Li Q, Ren Z, 2017. Virtual terminal–based adaptive predictor–corrector entry guidance. Journal of Aerospace Engineering, 30(4):04017013.
https://doi.org/10.1061/(asce)as.1943-5525.0000716

Liang Z, Lv C, Zhu S, 2023. Lateral entry guidance with terminal time constraint. IEEE Transactions on Aerospace and Electronic Systems, 59(3):2544-2553.
https://doi.org/10.1109/TAES.2022.3215554

Liu X, Li X, Zhang H, et al., 2024. Entry guidance with terminal time constraint based on reduced-order dynam- ics. IEEE Transactions on Aerospace and Electronic Systems, 1(1):1-14.
https://doi.org/10.1109/TAES.2024.3524200

Lu P, 1997.Entry guidance and trajectory control for reusable launch vehicle. Journal of Guidance, Control, and Dynamics, 20(1):143-149.
https://doi.org/10.2514/2.4008

Lu P, 2014. Entry guidance: A unified method. Journal of Guidance, Control, and Dynamics, 37(3):713–728. https://doi.org/10.2514/1.62605

Qiu X, Lai P, Gao C, et al., 2024. Recorded recurrent deep reinforcement learning guidance laws for intercepting endoatmospheric maneuvering missiles. Defence Tech- nology, 31:457-470.
https://doi.org/10.1016/j.dt.2023.02.016

Ren L, Xian Y, Li S, et al., 2023. Robust depletion shutdown guidance algorithm for long-range vehicles with a solid divert control system in large deviation conditions. Ad- vances in Space Research, 72(9):3818-3841.
https://doi.org/10.1016/j.asr.2023.07.049

REN L, GUO W, XIAN Y, et al., 2024. Deep reinforcement learning based integrated evasion and impact hierarchi- cal intelligent policy of exo-atmospheric vehicles. Chi- nese Journal of Aeronautics, 38(1):103193.
https://doi.org/10.1016/j.cja.2024.08.024

Schulman J, Wolski F, Dhariwal P, et al., 2017. Proximal policy optimization algorithms. arXiv: Learning.

https://doi.org/arxiv.org/abs/1707.06347
Shen Z, Lu P, 2003. Onboard generation of three-dimensional constrained entry trajectories. Journal of Guidance, Control, and Dynamics, 26(1):111-121.
https://doi.org/10.2514/2.5021
Suresh M, Swar SC, Shyam S, 2023. Autonomous cooperative guidance strategies for unmanned aerial vehicles during on-board emergency. Journal of Aerospace Information Systems, 2(2):102-113.
https://doi.org/10.2514/1.I011095
T.H. P, 2003. A acommon aero vehicle (cav) model, descrip- tion, and employment guide. Schafer Corporation for AFRL and AFSPC, 27:1-9.
WANG C, WANG W, DONG W, et al., 2024. Multiplestage spatial–temporal cooperative guidance without time-to-go estimation. Chinese Journal of Aeronautics, 37(9):399-416.
https://doi.org/10.1016/j.cja.2024.05.026
Wang H, Guo J, Wang X, et al., 2022a. Time-coordination entry guidance using a range-determined strategy. Aerospace Science and Technology, 129:107842.
https://doi.org/10.1016/j.ast.2022.107842
Wang N, Wang X, Cui N, et al., 2022b. Deep reinforcement learning-based impact time control guidance law with constraints on the field-of-view. Aerospace Science and Technology, 128:107765.
https://doi.org/10.1016/j.ast.2022.107765
Xue S, Lu P, 2010. Constrained predictor-corrector entry guidance. Journal of Guidance, Control, and Dynamics, 33(4):1273-1281.
https://doi.org/10.2514/1.49557
Yang H, Liang H, Liu J, et al., 2024a.Analytical time- coordinated entry guidance for multi-hypersonic vehi- cles within three-dimensional corridor. Aerospace Sci- ence and Technology, 155:109639.
https://doi.org/10.1016/j.ast.2024.109639
Yang H, Hu J, Li S, et al., 2024b. Reinforcement-learning- based robust guidance for asteroid approaching. Jour- nal of Guidance, Control, and Dynamics, 47(10):2058- 2072.
https://doi.org/10.2514/1.G008085
Yu W, Chen W, Jiang Z, et al., 2019. Analytical entry guidance for coordinated flight with multiple no-fly- zone constraints. Aerospace Science and Technology, 84:273–290.
https://doi.org/10.1016/j.ast.2018.10.013
Zeng L, Zhang H, Zheng W, 2018. A three-dimensional predictor–corrector entry guidance based on reduced-order motion equations. Aerospace Science and Technology, 73:223–231.
https://doi.org/10.1016/j.ast.2017.12.009