Evaluating Learned Pursuit Policies Against Classical Guidance Laws in a 1v1 Pursuit-Evasion Game

William Schafer
University of Illinois Urbana-Champaign, Champaign, IL 61820

This paper compares learning-based and analytical pursuit guidance in a 1v1 planar pursuit–evasion game. A Deep Q-Learning pursuer is evaluated against classical homing, deviated, and constant-bearing pursuit laws across deterministic and stochastic evasion strategies. Analytical methods dominate structured scenarios, while learning-based pursuit improves robustness under uncertainty at the cost of efficiency.

I. Introduction

Pursuit–evasion games describe a class of scenarios in which one or more agents actively pursue one or more evading agents with the objective of interception. Such problems arise across a wide range of engineering applications, including missile guidance and autonomous robotics, and have therefore been extensively studied. As a result, numerous classical guidance laws have been developed, many of which are computationally efficient, straightforward to implement, and highly effective. These methods remain in use today, even within modern engineering systems.

Despite their success, classical guidance laws are inherently model-driven and are typically derived under restrictive assumptions. Many assume known or simplified target motion models, are tailored to specific engagement geometries, and can struggle to maintain performance in the presence of highly unpredictable or adversarial behavior. Consequently, their effectiveness may degrade when these underlying assumptions are not held.

In recent years, learning-based approaches to guidance and control have emerged as a promising alternative within pursuit–evasion applications. Such methods have demonstrated the ability to implicitly adapt to adversarial behavior, optimize objectives that are difficult to encode analytically, and operate effectively in stochastic environments. However, learning-based policies are often difficult to interpret, and their inherent indeterminism can make it challenging to understand the decision-making processes that underlie their behavior [1]. While learning-based methods have shown promise in pursuit–evasion settings, it is often unclear under which conditions they offer meaningful advantages over well-established analytical approaches, and what tradeoffs they introduce in terms of efficiency and robustness.

This work seeks to address these questions by evaluating learned pursuit policies trained via deep reinforcement learning against classical pursuit guidance laws within a controlled two-dimensional pursuit–evasion environment. A single pursuer is tasked with intercepting an evader attempting to reach a fixed objective, and both learned and analytical pursuers are evaluated under identical initial conditions and evader behaviors. Multiple evasion strategies are considered, ranging from deterministic homing behavior to stochastic and adversarial motion, enabling a comprehensive assessment across a spectrum of engagement scenarios.

II. Problem Formulation

This section covers the pursuit–evasion game environment and evasion strategies used to train and test the learned pursuit policy against classical guidance laws.

A. Pursuit-Evasion Game Setup

The pursuit–evasion environment mimics a surface-to-air missile interception scenario consisting of three main components: a command center C, an evader E, and a pursuer P (Fig. 5). The environment is two-dimensional and evolves in discrete time with a fixed step size Δt. The command center is fixed at (x_C, y_C) with a goal radius r_{goal}, acting as the target for the evader and the launch point for the pursuer. The evader acts as the aggressor with the objective of reaching the command center before interception, while the pursuer acts as a defensive interceptor whose objective is to capture the evader before it reaches the command center.

Each agent has full-state observation of the other, including position (x_i, y_i), heading θ_i, and constant linear velocity v_i. Both agents are modeled as point-mass vehicles with fixed linear velocities v_P and v_E, and bounded turn rates $\omega_{P_{\max}}$

and $\omega_{E_{\max}}$, restricting their control authority to steering only. The pursuer is assigned a capture radius r_{capture}. The dynamics of each agent $i \in \{P, E\}$ follow simple planar kinematics:

$$\begin{cases} x_i(t+\Delta t) = x_i(t) + v_i \cos(\theta_i(t))\Delta t, \\ y_i(t+\Delta t) = y_i(t) + v_i \sin(\theta_i(t))\Delta t, \\ \theta_i(t+\Delta t) = \theta_i(t) + \omega_i(t)\Delta t, \quad |\omega_i(t)| \le \omega_{i_{\max}}. \end{cases} \tag{1}$$

Each episode begins with the pursuer located at the command center $(x_{P_0}, y_{P_0}) = (x_C, y_C)$ and initialized with a heading θ_{P_0} aligned with the line-of-sight (LOS) to the evader. The evader is initialized at a fixed distance d_{evader} from the command center in a uniformly random direction, with an initial heading θ_{E_0} aligned with the LOS to the pursuer.

An episode terminates if one of the following conditions is met: (i) the evader is captured by the pursuer, defined by $\|x_P - x_E\| \le r_{\text{capture}}$; (ii) the evader reaches the command center before interception, defined by $\|x_E - x_C\| \le r_{\text{goal}}$; or (iii) a maximum episode length is reached.

B. Markov Decision Process Formulation

The pursuit–evasion problem is formulated as a Markov Decision Process (MDP) defined by $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$. At each discrete time step t, the pursuer selects an action $a_t \in \mathcal{A}$ based on its observation of the environment state $s_t \in \mathcal{S}$. The state space $\mathcal{S}$ consists of the full state configuration of the pursuer and evader,

$$s_t = [x_P, y_P, \theta_P, v_P,\ x_E, y_E, \theta_E, v_E]_t\,. \tag{2}$$

The pursuer action space $\mathcal{A}$ is discrete and consists of three steering commands,

$$\mathcal{A} = \{\text{turn left},\ \text{go straight},\ \text{turn right}\} = \{-\omega_{\max}, 0, +\omega_{\max}\}, \tag{3}$$

corresponding to bounded angular velocity commands applied over a single time step. The evader follows a fixed analytical policy and is not a learning agent in this formulation.

State transitions are governed by the deterministic kinematic equations described in Section II.A, combined with the evader strategy. The transition function $\mathcal{P}(s_{t+1} \mid s_t, a_t)$ is therefore fully determined by the current state, the pursuer action, and the evader policy. The reward function $\mathcal{R}(s_t, a_t)$ is designed to encourage successful interception, with a positive terminal reward $r_T = +1.0$ granted upon evader capture. A discount factor $\gamma \in (0, 1)$ is used to prioritize rapid interception. Episodes terminate when the evader is captured, the command center is reached, or a maximum episode length is exceeded.

III. Guidance Strategies

A. Evader Guidance Strategies

To evaluate pursuer guidance strategies under a range of adversarial behaviors, several fixed analytical evader policies are considered. The evader is not a learning agent and follows a predefined control law throughout each episode. These strategies are designed to span deterministic, stochastic, and reactive behaviors of increasing complexity, thereby testing the robustness and adaptability of the pursuer.

1. Homing Evader

The homing evader steers directly toward the command center at all times. Its control input is determined by the LOS angle from the evader to the command center,

$$\theta_{\text{LOS}} = \arctan 2(y_C - y_E,\ x_C - x_E),$$

and the evader applies a proportional steering command to align its heading with this direction. This strategy represents a predictable, goal-directed threat with no pursuer-avoidant behavior and serves as a baseline evasion policy.

2. Random Evader

The random evader selects its steering commands stochastically. At each time step, the evader applies a uniformly random turn-rate command unless a clear path to the command center exists, at which point it switches to homing behavior following

$$\omega_E = \begin{cases} \text{Homing Evader}, & \dfrac{d_{EC}}{v_E} < \dfrac{d_{PC}}{v_P}, \\ \{-\omega_{\max},\, 0,\, \omega_{\max}\} \text{ with equal probability}, & \text{otherwise.} \end{cases}$$

This produces erratic motion patterns while preserving goal-seeking behavior by capitalizing on pursuer-overshoot.

3. Alpha-Blend Evader

The alpha-blend evader combines goal-seeking and pursuer-avoidant behaviors. Its commanded velocity direction is computed as a weighted combination of a vector pointing toward the command center and a vector perpendicular to the LOS from the evader to the pursuer (Fig. 1),

$$\mathbf{v}_{\text{target}} = (1-\alpha)\mathbf{v}_{\text{goal}} + \alpha\mathbf{v}_{\text{avoid}},$$

where $\alpha \in [0, 1]$ controls the tradeoff between aggressive goal pursuit and pursuer-avoidance. This strategy produces reactive behavior that adapts continuously to the pursuer's position and represents a more sophisticated adversary.

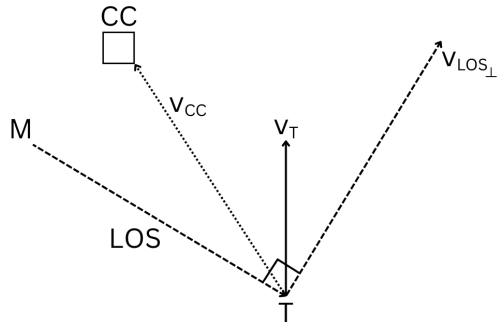


Fig. 1 Schematic of the alpha-blend evasion strategy where M is the pursuer, T is the evader, and CC is the command center.

B. Analytical Pursuit Guidance Laws

Several classical pursuit guidance laws described by Ghose [2] are implemented as analytical baselines for comparison against learned policies. These guidance laws are deterministic, geometry-based strategies that compute steering commands using instantaneous relative state information.

1. Homing Pursuit

Also called pure pursuit, homing pursuit is one of the simplest pursuit guidance laws. The pursuer continuously steers to align its velocity vector with the instantaneous LOS to the evader. The LOS angle from the pursuer to the evader is given by

$$\theta_{\text{LOS}} = \text{atan2}(y_E - y_P,\, x_E - x_P),$$

and the pursuer applies a proportional steering command to align its heading with this direction. Homing pursuit reacts solely to the current relative geometry and does not account for evader velocity or future motion, often resulting in lagged pursuit trajectories and prolonged interception times.

2. Deviated Pursuit

Also called lead pursuit, deviated pursuit extends pure pursuit by incorporating a simple prediction of the evader's future position. Assuming constant evader velocity over a finite prediction horizon $T_{\text{horizon}} = 0.75s$, the evader's

predicted position is approximated as

$$\hat{x}_E(t + T_h) = x_E(t) + \vec{v}_E(t)T_h.$$

The pursuer then computes the LOS angle to this predicted position and applies a proportional steering command analogous to homing pursuit. By leading the target, deviated pursuit can reduce interception time relative to pure pursuit. However, its performance is sensitive to the assumed prediction horizon and degrades when evader motion deviates from constant-velocity behavior.

3. Constant-Bearing Pursuit

Constant-bearing pursuit seeks to maintain a constant LOS angle between the pursuer and the evader. Under the assumption of constant pursuer and evader velocities, maintaining a constant bearing guarantees interception [3]. The constraints are defined by the LOS from the pursuer to the evader λ as

$$\lambda(t) = \lambda(0) \quad \text{and} \quad \frac{d\lambda(t)}{dt} = (\vec{v}_T - \vec{v}_M) \cdot \lambda_\perp = 0, \quad \forall t \geq 0$$

The evader's heading relative to the LOS is denoted by $\beta = \theta_E - \theta_{\text{LOS}}$. The pursuer computes a lead angle

$$\alpha = \arcsin\left(\frac{v_E}{v_P} \sin\beta\right)$$

and commands its heading toward $\theta_{\text{target}} = \theta_{\text{LOS}} + \alpha$. The pursuer then applies a proportional steering law to align its heading with this direction. When the assumptions of constant velocity and sufficient pursuer speed are satisfied, constant-bearing pursuit yields near-optimal interception trajectories. However, its effectiveness deteriorates in the presence of evasive or stochastic target maneuvers.

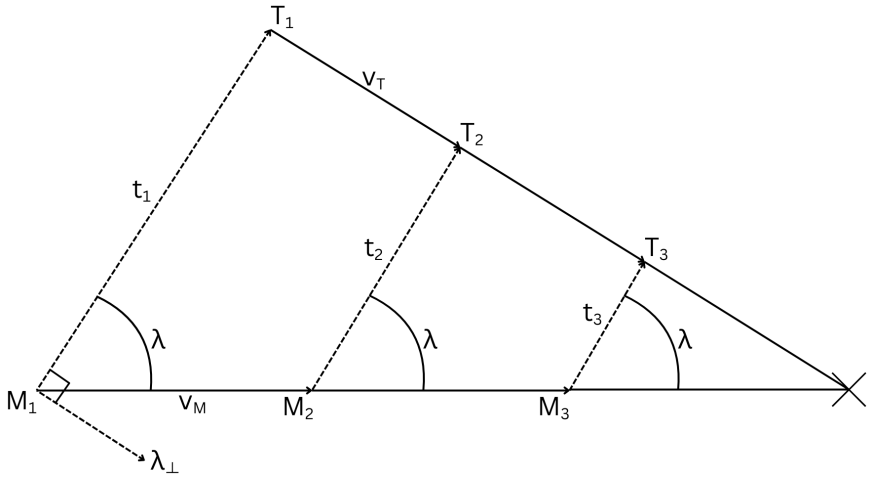


Fig. 2 Schematic of the constant-bearing pursuit strategy where M is the pursuer, T is the evader, and λ is the LOS from M to T over three time steps before collision [2], [3].

C. Learned Guidance via Deep Q-Learning

In addition to analytical pursuit laws, a learning-based guidance strategy is considered in which the pursuer policy is represented by a deep neural network trained using Deep Q-Learning. Rather than explicitly enforcing geometric constraints like the analytical laws, the learned policy maps the observed state directly to discrete steering commands. The policy outputs bounded turn-rate actions consistent with the pursuer kinematics in Eq. 1, allowing for direct comparison with analytical guidance laws under identical dynamics. The learning formulation, network architecture, and training procedure are described in detail in Section IV.

IV. Learning Framework

This section describes the reinforcement learning framework used to learn pursuit guidance policies. The Deep Q-Learning algorithm is first summarized, followed by details on the network architecture used to approximate the action-value function and the training procedure employed to ensure stable and reproducible learning.

A. Deep Q-Learning

Deep Q-Learning (DQN) extends the classical Q-learning algorithm, an off-policy temporal-difference method that iteratively updates an action-value function $Q(s, a)$ using observed transitions. While classical Q-learning is effective for small, discrete state spaces, tabular methods scale poorly and are infeasible for high-dimensional environments such as the pursuit–evasion game considered here. DQN uses a neural network parameterized by weights θ, which approximates the action-value function $Q(s, a; \theta)$. This enables learning in a continuous state space while retaining the core Bellman optimality principle of Q-learning.

To stabilize training, DQN introduces two key mechanisms: a target network and an experience replay buffer. The target network, parameterized by θ^-, is a periodically updated copy of the online network and is used exclusively to compute the temporal-difference target

$$y = r + \gamma \max_{a'} Q(s', a'; \theta^-),$$

which prevents the learning target from changing rapidly during optimization. The experience replay buffer $\mathcal{D}$ stores transitions of the form (s_t, a_t, r_t, s_{t+1}). Rather than learning from instantaneous transitions, small batches of historical transitions are drawn uniformly at random from $\mathcal{D}$, improving data efficiency and reducing variance. The network parameters are optimized by minimizing the mean-squared Bellman error

$$L(\theta) = \mathbb{E}_{(s,a,r,s')\sim U(\mathcal{D})}\left[\left(r + \gamma \max_{a'} Q(s', a'; \theta^-) - Q(s, a; \theta)\right)^2\right],$$

using stochastic gradient descent. For a short period following learning initialization, network updates are delayed to allow the replay buffer to accumulate a diverse set of experiences [4].

B. Network Architecture

The action-value function $Q(s, a; \theta)$ is approximated using a multilayer perceptron (MLP). The network takes as input the pursuer's observation of the environment state (Eq. 2) and outputs an action-value for each discrete steering action (Eq. 3). During training, actions are selected using an ϵ-greedy policy derived from the network outputs, while evaluation is performed greedily according to

$$a_t = \arg\max_{a} Q(s_t, a; \theta).$$

The network consists of two fully connected hidden layers with rectified linear unit (ReLU) activation functions, which provide sufficient representational capacity for the low-dimensional but nonlinear dynamics considered here. This architecture was selected to capture nonlinear pursuit behaviors while maintaining training stability and computational efficiency.

C. Training Procedure

Each learned pursuer policy is trained using the Deep Q-Learning framework described in Section IV.A. All models are trained for 20,000 episodes, except for the pursuer trained against the alpha-blend evader with randomized velocities, which trained for 30,000 episodes. All models are trained with a discount factor of $\gamma = 0.9$ to encourage speedy interception. See Table 2 for all training parameters.

The pursuer action-selection mechanism is implemented as an ϵ-greedy policy; however, the exploration parameter is fixed to $\epsilon = 0$ for the entirety of training. This design choice is motivated by the specific structure of the pursuit–evasion environment and the sparse terminal reward. Because the evader consistently converges toward the command center, the pursuer is guaranteed to encounter interception opportunities even without stochastic exploration. Empirically, introducing random exploratory actions caused the pursuer to diverge from the evader, delaying interception and significantly degrading learning performance. Acting greedily from initialization ensured that the pursuer reliably experienced terminal rewards early in training, which was critical for bootstrapping learning under the sparse reward

structure. While an ϵ-decay schedule was retained in the implementation for completeness, it is redundant in this application due to ϵ being fixed at zero.

The Q-network is trained using mini-batch stochastic gradient descent with a learning rate of $\alpha = 10^{-4}$. Experience replay is employed with a buffer capacity of 50,000 transitions and a batch size of 64. To ensure sufficient experience is gathered prior to learning, gradient updates are delayed until the replay buffer contains at least 1,000 transitions. The target network parameters θ^- are synchronized with the online network every 5,000 environment steps.

V. Methodology

This section describes the experimental methodology used to train, evaluate, and compare learned pursuit policies against classical analytical guidance laws.

A. Experimental Design

All training and evaluation experiments were conducted within the pursuit–evasion environment described in Section II.A. The simulation time step was set to $\Delta t = 0.1$ s, and each episode was limited to a maximum of 500 steps. The command center was fixed at the origin with $r_{\text{goal}} = 3.0$ m, and the evader was initialized at $d_{\text{evader}} = 40.0$ m in a uniformly random direction about the command center with $r_{\text{capture}} = 1.0$ m. The maximum turn rates were set to $\omega_P = \omega_E = \pi/2$ rad/s, and the proportional steering gains for both agents were fixed at $k_P = k_E = 1.0$. Unless otherwise specified, both the pursuer and evader used fixed linear velocities of $v_P = v_E = 10.0$ m/s. See Table 3 for all environment parameters.

Three learned pursuer models were trained independently against fixed analytical evader strategies: a homing evader, a random evader, and an alpha-blend evader with $\alpha = 0.5$. Each learned model was trained exclusively against a single evader policy, allowing the pursuer to specialize its behavior to that adversarial strategy. In addition, a fourth learned pursuer was trained against the alpha-blend evader with randomized linear velocities for both agents. In this configuration, v_P and v_E were independently sampled at the start of each episode from the interval $[5.0, 20.0]$ m/s, introducing kinematic uncertainty and testing robustness to varying engagement speeds.

Following training, each learned pursuer was evaluated against three analytical pursuit guidance laws: constant-bearing pursuit, deviated pursuit, and homing pursuit. For each comparison, both pursuers were evaluated against the same evader strategy using identical random seeds, ensuring that each pursuer experienced identical initial conditions and evader behavior. Each learned–analytical pairing was evaluated over 1000 independent episodes. In total, this resulted in 12 direct comparison cases, corresponding to three analytical pursuit strategies evaluated against learned pursuers trained on homing, random, and alpha-blend evasion strategies, with the randomized-velocity alpha-blend case evaluated separately.

B. Evaluation Metrics

Learned and analytical pursuit guidance strategies were evaluated using multiple complementary metrics designed to quantify interception effectiveness, guidance efficiency, and control effort effort.

The primary performance metric is the success rate, defined as the fraction of episodes in which the pursuer successfully intercepts the evader before the evader reaches the command center. This metric directly captures the fundamental objective of the pursuit task. To quantify interception efficiency, the steps per episode metric is recorded, representing the number of simulation time steps required for an episode to terminate. This metric is reported in two forms: the average number of steps over all episodes and the average number of steps over successful episodes only. The latter isolates interception efficiency from failed engagements or episodes that terminate at the maximum episode length. Control effort is measured using an energy usage proxy, defined as the time-integrated absolute angular displacement of the pursuer,

$$E = \sum_{t=0}^{T} |\Delta\theta_P(t)|,$$

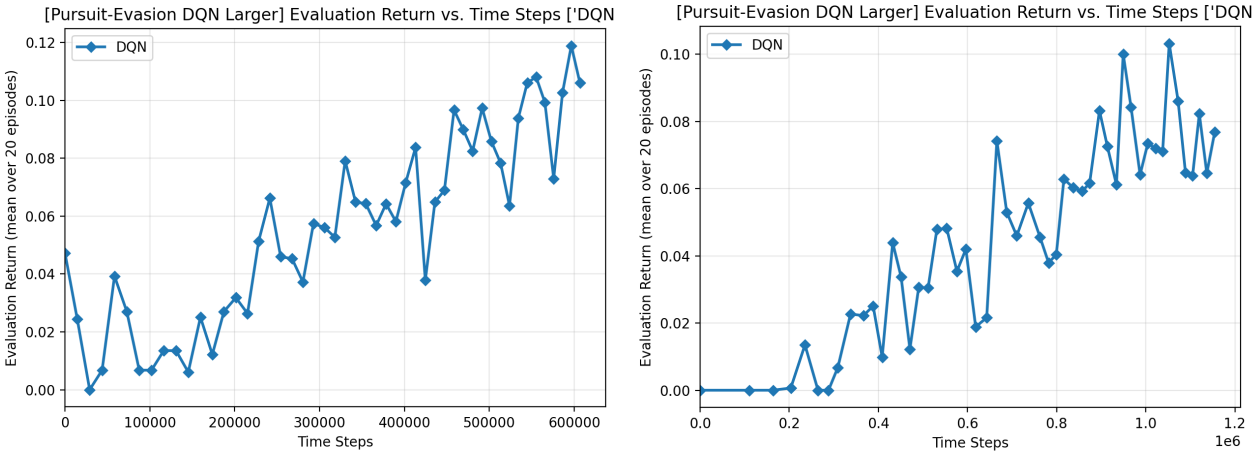
where $\Delta\theta_P(t)$ is the pursuer's applied heading change at time step t, and T denotes the episode termination time. Similar to the step metric, energy usage is reported both over all episodes and over successful episodes only, with the successful-only metric providing a fair comparison of guidance efficiency during effective interceptions. Together, these metrics enable a comprehensive comparison between learned and analytical pursuit strategies in terms of reliability, speed, and control efficiency.

VI. Results

This section presents the empirical results of training the learned pursuit policies and their comparison against classical analytical guidance laws.

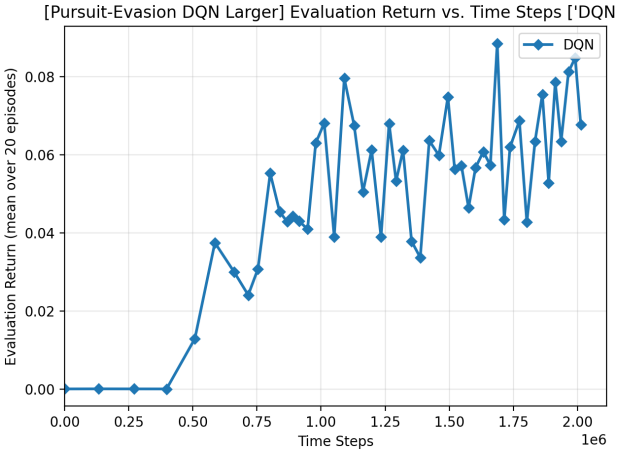
A. Training Performance

Training performance is evaluated using 50 policies sampled uniformly throughout training, with each policy assessed by its average evaluation return computed over 20 independent episodes.

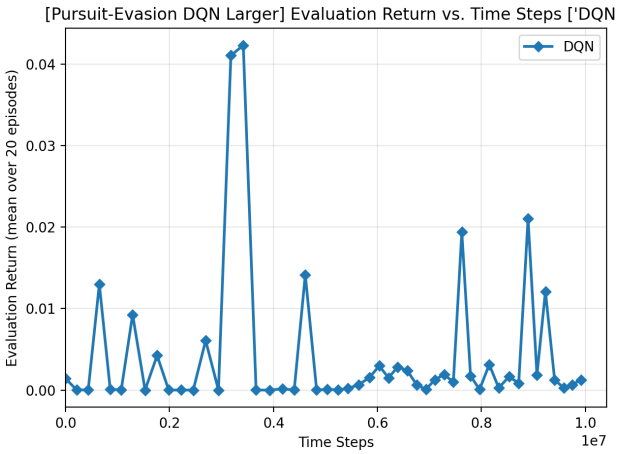


(a) Pursuer vs. Homing Evader

(b) Pursuer vs. Random Evader



(c) Pursuer vs. Alpha-Blend Evader



(d) Pursuer vs. Alpha-Blend Evader (Random Velocities)

Fig. 3 Evaluation return over training for DQN pursuers trained against different evader strategies.

The pursuer trained against the homing evader (Fig. 3a) exhibited the most stable learning behavior. Evaluation return increased steadily and approximately linearly from the start of training to approximately 6×10^5 time steps, with only minor transient dips. This model achieved the highest final evaluation return of 0.1059, indicating consistent interception performance.

The pursuer trained against the random evader (Fig. 3b) demonstrated slower initial learning, with no measurable improvement in evaluation return during the first 2×10^5 time steps. Beyond this point, learning progressed in a generally linear manner but with more frequent and pronounced performance fluctuations. Despite this variability, the model converged to a final evaluation return of 0.0768.

Training against the alpha-blend evader without velocity randomness (Fig. 3c) resulted in delayed but structured learning behavior. Evaluation return remained near zero until approximately 3×10^5 time steps, after which performance increased rapidly in a near-exponential manner for roughly 10^6 time steps before gradually leveling off. Although

frequent dips were observed, performance gains were consistent over time, yielding a final evaluation return of 0.0677.

In contrast, the pursuer trained against the alpha-blend evader with randomized velocities (Fig. 3d) failed to learn a stable interception policy. Evaluation return remained near zero for the duration of training, with only rare and transient spikes. The highest observed evaluation return occurred near 3.5×10^6 time steps and remained below 0.05. This behavior suggests that velocity randomization significantly hindered the pursuer's ability to receive a consistent learning signal under the given reward structure.

B. Comparative Performance Evaluation of Pursuit Guidance Strategies

1. Quantitative Analysis

Quantitative performance is evaluated using metrics across all pursuit–evasion pairings. For each trained pursuer, performance is compared against three analytical pursuit laws under four evader strategies using $1,000$ evaluation episodes with identical random seeds. Metrics reported include interception success count, mean steps to capture, and a mean energy proxy, all computed over successful episodes only. Metrics computed over all episodes are provided in Table 4 for completeness.

Table 1 Success-only performance comparison between learned and analytical pursuit strategies out of $1,000$ episodes. Lower step counts and lower energy usage indicate more efficient interception.

Analytical Pursuit	Evader	Success		Steps		Energy	
		Learned	Analytic	Learned	Analytic	Learned	Analytic
Homing	Homing	783	1000	20.2	20.0	1.37	0.00
Homing	Random	626	215	20.0	21.6	2.41	0.07
Homing	Alpha-Blend	905	0	30.3	Fail	2.80	Fail
Homing	Alpha-Blend (Rand. Vel.)	349	111	122.3	68.8	14.87	5.68
Deviated	Homing	765	1000	20.2	20.0	0.47	1.33
Deviated	Random	636	595	22.4	20.6	2.53	0.84
Deviated	Alpha-Blend	911	1000	31.3	59.0	2.89	4.75
Deviated	Alpha-Blend (Rand. Vel.)	373	280	127.8	44.3	14.95	3.43
Constant-Bearing	Homing	761	1000	20.2	20.0	1.34	0.00
Constant-Bearing	Random	594	511	21.9	21.2	2.41	0.38
Constant-Bearing	Alpha-Blend	898	1000	30.5	275.0	2.77	12.71
Constant-Bearing	Alpha-Blend (Rand. Vel.)	388	187	124.0	66.2	14.86	2.94

From Table 1, several consistent trends emerge. Against the homing and alpha-blend evaders, analytical pursuit laws achieve perfect interception performance (with the exception of the homing pursuer vs. the alpha-blend evader, a scenario in which interception is geometrically impossible unless $v_P > v_E$), reflecting the favorable, non-stochastic assumptions under which they are derived. However, under stochastic evasion (random and alpha-blend w/ rand. vel.), the learned pursuer exceeds analytical success rates in all scenarios. In cases where analytical pursuit laws exhibit superior success rates, the learned pursuer maintains a comparable level of interception performance.

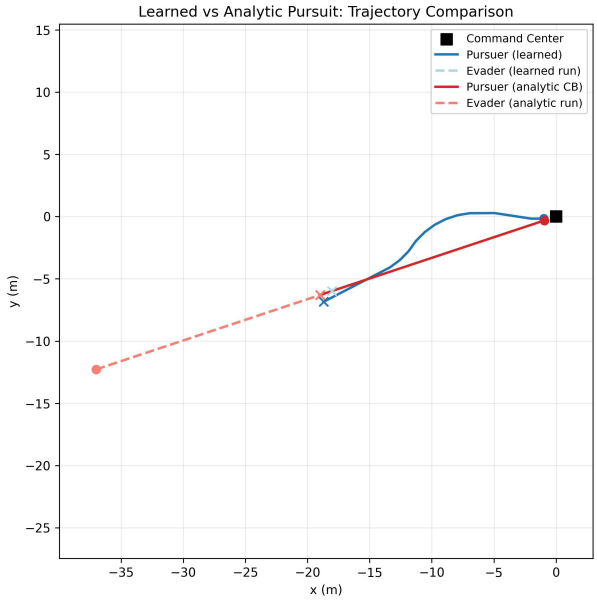
The learned pursuer generally requires a greater number of steps to capture the evader than the analytical pursuit laws. Against homing and random evaders, however, this difference is minimal, with analytical strategies achieving only marginally faster interceptions. The most pronounced timing disparity occurs under alpha-blend evasion, where the learned pursuer intercepts the evader substantially faster than the analytical baselines, indicating improved adaptation to reactive avoidance behavior. In contrast, under alpha-blend evasion with randomized velocities, the learned pursuer requires significantly more steps to capture than analytical pursuit, highlighting a sensitivity to dynamic uncertainty despite maintaining higher overall success rates.

With respect to control effort, the learned pursuer consistently expends more energy than analytical pursuit laws, even in scenarios where the analytically optimal strategy involves no turning, such as homing evasion. Reduced energy consumption by the learned pursuer is observed only in cases where it achieves markedly faster interceptions or where the analytical guidance law suffers from a geometric disadvantage, such as homing pursuit against alpha-blend evasion

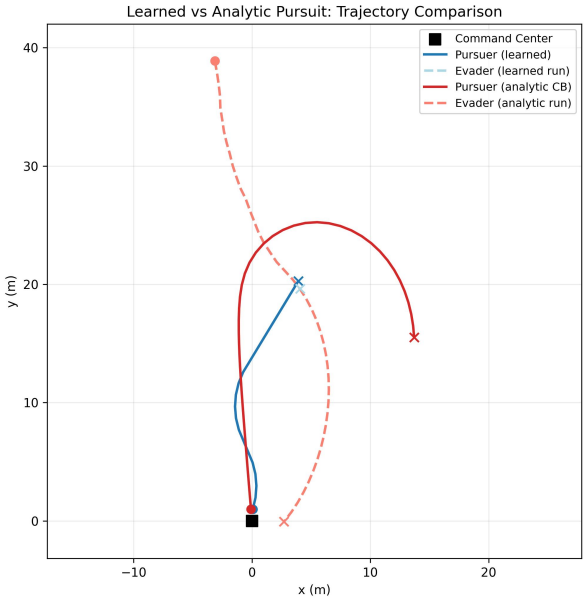
or deviated pursuit against homing evasion.

2. Qualitative Analysis

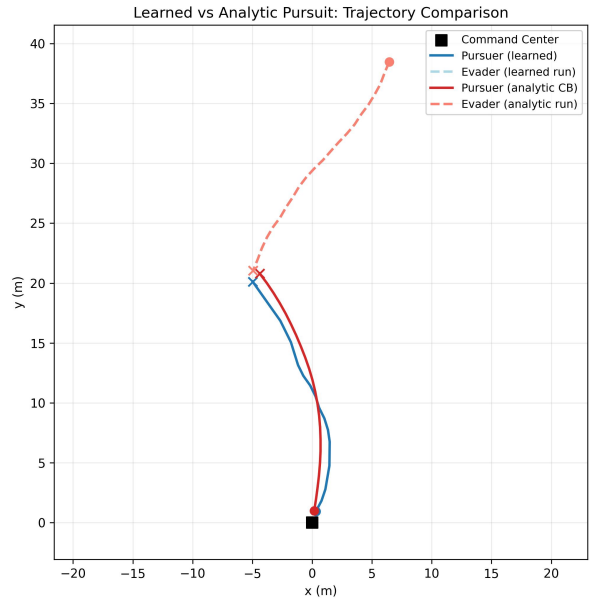
This subsection qualitatively examines pursuit–evasion trajectories to provide insight into the behavioral differences between learned and analytical pursuit strategies that underlie the quantitative performance trends observed in Section VI.B.1. Additional trajectories are provided in Section VIII.E.



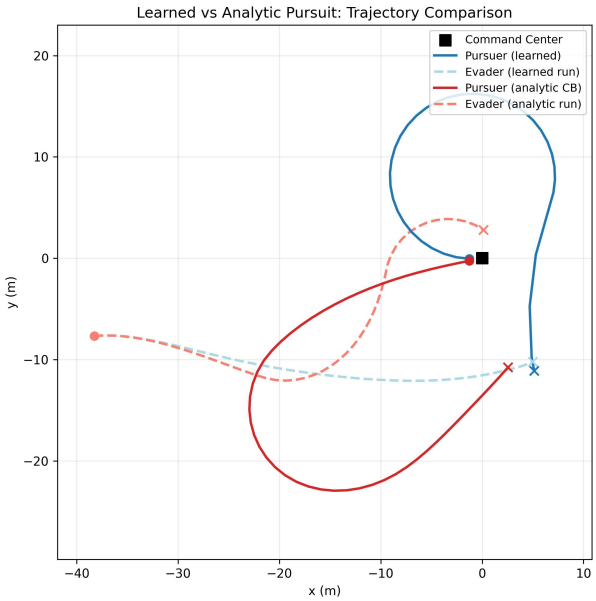
(a) Homing Evader: Learned vs. Constant-Bearing Pursuit



(b) Random Evader: Learned vs. Homing Pursuit



(c) Random Evader: Learned vs. Constant-Bearing Pursuit



(d) Alpha-Blend Evader (Random Velocities): Learned vs. Constant-Bearing Pursuit

Fig. 4 Representative pursuit–evasion trajectories comparing learned and analytical guidance laws.

In Fig. 4a, a homing evader scenario is shown in which the analytical constant-bearing pursuer outperforms the

learned pursuer. Because the evader travels directly toward the command center without attempting to evade, the optimal interception strategy is a straight-line pursuit. The analytical pursuer follows this geometry exactly, while the learned pursuer executes an unnecessary lateral deviation prior to interception. Although capture is still achieved, this deviation results in increased energy expenditure, consistent with the higher control effort observed quantitatively for the learned policy in this scenario.

Figure 4b illustrates a random evader scenario in which the learned pursuer clearly outperforms the analytical homing pursuer. The evader follows a mildly curved trajectory, causing the homing pursuer to overshoot and fail to intercept. In contrast, the learned pursuer exhibits a rapid oscillatory steering behavior, suggestive of frequent action switching, which allows it to remain responsive to evader motion uncertainty and successfully intercept. This behavior reflects the learned policy's ability to adapt online to stochastic evasion patterns rather than committing to a fixed geometric rule.

In Fig. 4c, both the learned pursuer and the analytical constant-bearing pursuer perform similarly against a random evader approaching at an oblique angle. In this case, the geometry of the engagement is favorable to both strategies, and the resulting trajectories nearly overlap. This qualitative similarity aligns with the comparable success rates and interception times observed in the quantitative analysis for this scenario.

Figure 4d highlights a nontrivial behavior in the most challenging setting: alpha-blend evasion with randomized linear velocities. Here, the analytical constant-bearing pursuer follows the expected direct interception path but fails due to the evader's alpha-blended guidance and velocity variation. The learned pursuer, however, adopts a non-intuitive strategy by initially maneuvering away from the evader and rotating around the command center. This maneuver appears to disrupt the evader's alpha-blended objective, ultimately enabling a successful interception. Although unconventional, this behavior demonstrates the learned policy's capacity to exploit weaknesses in the evader's decision structure that are not captured by fixed analytical pursuit laws.

Overall, these trajectory comparisons illustrate that while analytical strategies remain optimal in simple, well-structured engagements, learned pursuit policies can discover adaptive and occasionally non-intuitive behaviors that improve robustness against stochastic and adversarial evasion strategies.

VII. Discussion

This work set out to examine when learning-based pursuit strategies offer meaningful advantages over classical analytical guidance laws, and what tradeoffs arise in terms of efficiency, robustness, and interpretability. The results demonstrate that neither approach universally dominates. Instead, performance depends strongly on the structure and predictability of the evasion strategy, highlighting complementary strengths between learned and analytical pursuit.

When Learned Pursuit Excels

The learned pursuer consistently outperforms analytical guidance laws in scenarios characterized by stochasticity, adversarial behavior, or incorrect modeling assumptions. Against random and alpha-blend evasion with randomized velocities, the learned pursuer achieves substantially higher success rates than all analytical strategies considered. These scenarios violate the core assumptions underlying classical pursuit laws, such as smooth target motion, known engagement geometry, and predictable evader intent.

In such scenarios, the learned policy benefits from its experiential, adaptive nature. Rather than committing to a fixed geometric strategy, the neural policy adapts continuously to observed evader behavior, enabling it to exploit transient vulnerabilities and non-obvious engagement geometries. This is most evident in alpha-blend evasion with randomized velocities, where the learned pursuer discovers non-standard interception behaviors, including indirect approaches that manipulate the alpha-blended evader's inherent goal–avoidance tradeoff. These behaviors are not encoded explicitly and would be difficult to derive analytically, yet they emerge naturally through reinforcement learning.

When Analytical Pursuit Dominates

Conversely, analytical pursuit laws consistently outperform the learned pursuer in simple, deterministic scenarios, particularly against homing and alpha-blend evasion. In these cases, the optimal interception strategy is geometrically deterministic. Classical guidance laws exploit this structure inherently and efficiently, achieving rapid interception with minimal control effort.

The learned pursuer, while still successful in many such cases, often exhibits unnecessary steering and longer interception times. This inefficiency stems from the lack of explicit optimization for energy or time in the reward function,

as well as from the inherent approximation error of the learned policy. These results underscore that learning-based methods do not inherently guarantee efficiency in scenarios where the optimal solution is simple and well understood.

Tradeoffs of Learning-Based Guidance

Across all scenarios, learning-based pursuit showcases clear tradeoffs. Most notably, the learned pursuer consistently consumes more control energy than analytical methods, even in cases where zero steering is optimal. This behavior reflects both the absence of energy penalties in the reward function and the tendency of learned policies to rely on frequent corrective actions rather than smooth motion.

Additionally, learned policies generally require slightly longer times to intercept compared to analytical laws in nominal settings. While these differences are small for homing and random evasion, they highlight that improved robustness often comes at the cost of efficiency. Finally, learned policies exhibit reduced interpretability. Successful behaviors may lack clear intuition, and failure modes are more difficult to diagnose, complicating validation and assurance of performance requirements.

Implications for Guidance System Design

Taken together, these results suggest that learning-based pursuit should not be viewed as a complete replacement for classical guidance laws. Instead, learning offers the greatest benefit in regimes where analytical assumptions break down, such as under adversarial, stochastic, or highly uncertain evasion. In contrast, analytical pursuit remains superior in structured, predictable environments where efficiency and interpretability are critical.

This naturally motivates hybrid guidance architectures in which analytical laws serve as a baseline controller, while learned policies are invoked selectively when uncertainty or adversarial behavior is present. Such an approach could combine the efficiency and reliability of classical guidance with the adaptability and robustness of learning-based methods.

Limitations and Future Work

Several limitations of the present study stem from deliberate modeling and implementation choices made to enable hardware-efficient experimentation. The pursuit–evasion environment is restricted to two-dimensional, planar kinematics with three degrees of freedom per agent and point-mass dynamics. While this approach captures essential geometric and decision-making aspects of interception, it omits important nonlinearities associated with real aerial or ground vehicular applications, including aerodynamic effects, nonholonomic constraints, and dynamic-coupling. Consequently, the results should be interpreted as representative of abstract guidance scenarios, rather than as direct proxies for full-scale missile, aircraft, or robotic platforms. This simplification was intentionally motivated by computational and hardware constraints, enabling extensive training and evaluation with explainable behavioral effects.

The learning function itself introduces additional limitations. The reward structure exclusively incentivizes successful interception and does not explicitly penalize energy expenditure, path length, or excessive maneuvering. While this design highlights differences in strategic effectiveness, it also explains the consistently higher control effort observed for the learned pursuer. Moreover, training was conducted without exploration noise ϵ, with the action-selection policy operating greedily throughout learning. Although this choice empirically improved convergence in the present setting, it would likely bias the learned policy toward aggressive exploitation and reduce robustness to unmodeled dynamics or unseen operating conditions in other settings.

From a comparative standpoint, the analytical baselines considered in this work are classical pursuit laws with well-understood geometric properties. While these methods remain highly relevant due to their interpretability and low computational cost, they do not represent the full spectrum of modern optimal control or game-theoretic approaches. Advanced techniques such as Hamilton-Jacobi-Isaacs (HJB) control, receding-horizon optimal control, or differential game formulations may close some of the observed performance gaps under certain conditions. However, such methods typically require substantially greater modeling fidelity and offline computation, and this work hypothesizes that the qualitative trends observed here would persist under fair computational scaling.

Despite the limitations, several promising directions emerge from this study. First, extending the environment to higher-dimensional dynamics with realistic vehicle models would enable evaluation of learned pursuit strategies under more physically representative constraints. Incorporating control limits, stochastic sensor systems, and realistic actuator dynamics would further test policy robustness and generalization. Second, denser reward formulations that balance interception success with energy efficiency and trajectory optimality may yield more practical learned behaviors while

preserving the observed adaptability.

From an algorithmic perspective, future work should prioritize improving the practical feasibility of reinforcement learning for guidance applications, particularly by reducing training cost and narrowing the sim-to-real gap that currently limits scalability to high-dimensional, high-fidelity systems. Recent surveys of reinforcement learning for pursuit–evasion games emphasize that addressing these challenges is critical for handling strategic uncertainty, adversarial behavior, and complex engagement geometries at scale [1]. Hybrid frameworks that embed analytical guidance laws as priors, constraints, or safety layers within learning-based controllers represent a promising direction, as they may significantly reduce training burden while preserving performance guarantees and interpretability. Finally, systematic environment randomization and sim-to-real transfer techniques will be essential for translating learning-based pursuit strategies from abstract simulation environments to practical, deployable guidance systems.

VIII. Conclusion

This paper evaluated when learning-based pursuit policies provide meaningful advantages over classical analytical guidance laws, and what tradeoffs they introduce in efficiency and robustness, using a controlled 1v1 two-dimensional pursuit–evasion game. Deep Q-Learning was used to train pursuer policies against multiple evader behaviors spanning homing evasion, random motion evasion, and reactive alpha-blend evasion, and the resulting learned policies were compared against homing, deviated, and constant-bearing analytical pursuit laws under identical initial conditions.

The results indicate that analytical guidance laws remain highly effective in structured engagements that align with their underlying assumptions, achieving near-perfect success and low control effort in deterministic scenarios. In contrast, learned pursuit policies demonstrate clear advantages in adversarial and stochastic settings, where analytical methods degrade or fail due to inaccurate modeling assumptions. In particular, against random and velocity-randomized alpha-blend evasion, the learned pursuer consistently achieved higher interception success rates and exhibited adaptive behaviors that exploited weaknesses in the evader's guidance algorithm.

These robustness gains came with identifiable costs. The learned pursuer generally required comparable or slightly longer capture times than analytical baselines and expended consistently higher control effort, reflecting both the sparse terminal reward structure and the absence of explicit penalties on maneuvering. Moreover, while learned policies can discover effective non-intuitive strategies, their reduced interpretability complicates diagnosis and validation relative to deterministic analytical laws.

Overall, the findings suggest that learning-based pursuit is most valuable as a robustness-enhancing component in environments characterized by uncertainty, stochasticity, or adversarial evasion, rather than as a universal replacement for classical guidance. A practical implication is that hybrid guidance architectures may offer the most compelling path forward for deployable learning-enhanced guidance system design.

References

[1] Yang, K., Shen, A., Xu, N., Deng, F., Lu, M., and Chen, C., "A review of reinforcement learning approaches for pursuit-evasion games," *Chinese Journal of Aeronautics*, 2025, p. 103940. https://doi.org/https://doi.org/10.1016/j.cja.2025.103940, URL https://www.sciencedirect.com/science/article/pii/S1000936125005461.

[2] Ghose, D., "Guidance of missiles," 2012.

[3] Cross, M., and Shtessel, Y., "A Single-Loop High-Order Sliding Mode Controller for a Missile Interceptor," 2018, pp. 331–336. https://doi.org/10.1109/VSS.2018.8460335.

[4] Mnih, V., Kavukcuoglu, K., Silver, D., et al., "Human-level Control Through Deep Reinforcement Learning," *Nature*, Vol. 518, No. 7540, 2015, pp. 529–533. https://doi.org/10.1038/nature14236, URL https://doi.org/10.1038/nature14236.

Appendix

A. Simulation Environment Snapshot

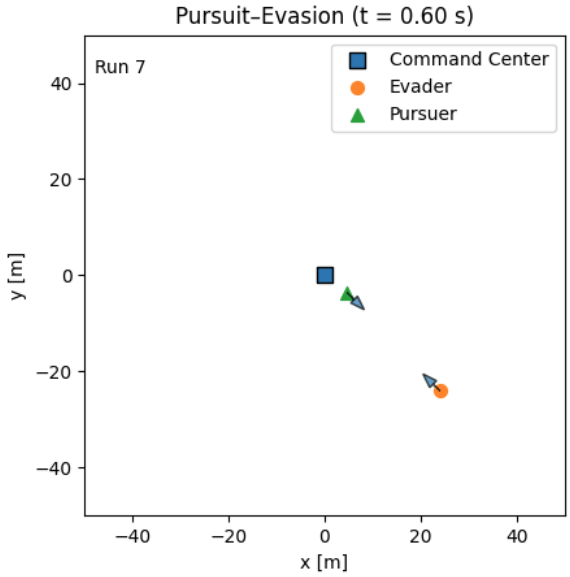


Fig. 5 Pursuit–evasion environment overview.

B. Training Parameters Used in Experiments

Table 2 Deep Q-Learning training parameters used across all learned pursuer experiments.

Parameter	Value
Number of training episodes	20,000
Discount factor	$\gamma = 0.9$
Exploration rate	$\epsilon = 0.0$
Final exploration rate	$\epsilon_{\text{final}} = 0.0$
Exploration decay steps	10,000
Learning rate	$\alpha = 1 \times 10^{-4}$
Hidden layer sizes	(64, 64)
Replay buffer capacity	50,000 transitions
Mini-batch size	64
Target network update frequency	5,000 steps
Replay start size	1,000 transitions

C. Full Environment Parameters Used in Experiments

Table 3 Summary of pursuit–evasion environment parameters used across all training and evaluation.

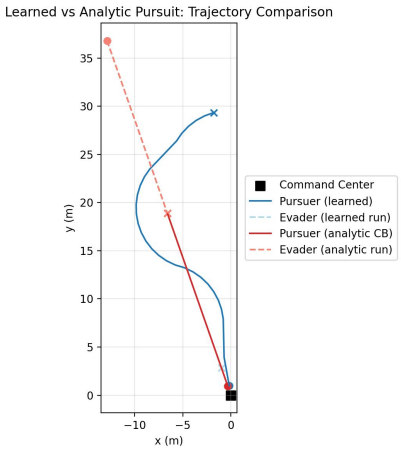
Parameter	Symbol	Value
Simulation time step	Δt	0.1 s
Maximum episode length	$T_{\max}$	500 steps
Command center position	(x_C, y_C)	$(0, 0)$
Command center radius	r_{goal}	3.0 m
Evader capture radius	r_{capture}	1.0 m
Evader initial distance	d_{evader}	40.0 m
Pursuer linear velocity	v_P	10.0 m/s
Evader linear velocity	v_E	10.0 m/s
Pursuer velocity (randomized case)	v_P	[5.0, 20.0] m/s
Evader velocity (randomized case)	v_E	[5.0, 20.0] m/s
Maximum pursuer turn rate	ω_P	$\pi/2$ rad/s
Maximum evader turn rate	ω_E	$\pi/2$ rad/s
Pursuer steering gain	k_P	1.0
Evader steering gain	k_E	1.0

D. Complete Quantitative Results (All Episodes)

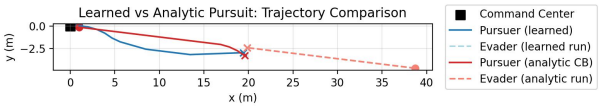
Table 4 All-episode performance comparison between learned and analytical pursuit strategies out of $1,000$ episodes. Lower step counts and lower energy usage indicate more efficient interception.

Analytical Pursuit	Evader	Steps (All Episodes)		Energy (All Episodes)	
		Learned	Analytic	Learned	Analytic
Homing	Homing	23.9	20.0	2.02	0.00
Homing	Random	35.3	37.0	4.59	2.39
Homing	Alpha-Blend	40.9	500.0	4.32	53.60
Homing	Alpha-Blend (Rand. Vel.)	251.4	180.8	31.59	18.58
Deviated	Homing	24.3	20.0	2.02	0.47
Deviated	Random	37.3	33.6	4.87	1.94
Deviated	Alpha-Blend	40.4	59.0	4.20	4.75
Deviated	Alpha-Blend (Rand. Vel.)	250.4	156.9	31.44	15.36
Constant-Bearing	Homing	24.3	20.0	2.05	0.00
Constant-Bearing	Random	37.5	33.0	4.91	1.77
Constant-Bearing	Alpha-Blend	38.9	275.0	4.00	12.71
Constant-Bearing	Alpha-Blend (Rand. Vel.)	248.5	143.8	31.12	14.26

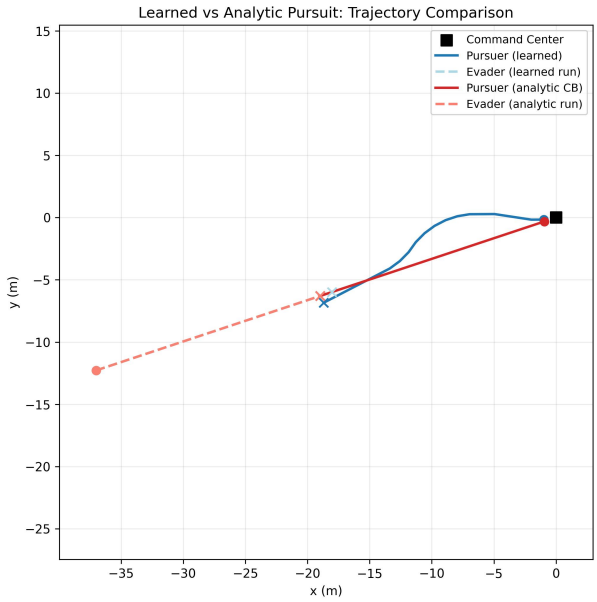
E. Additional Trajectory Plots



(a) Learned vs. Homing Pursuit

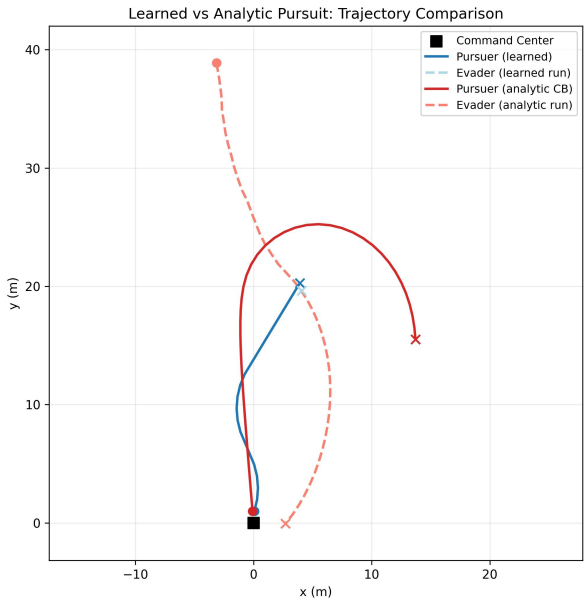


(b) Learned vs. Deviated Pursuit

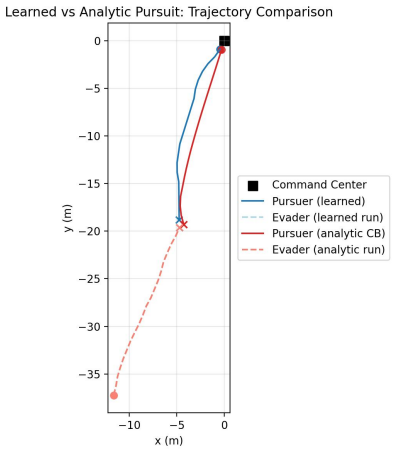


(c) Learned vs. Constant-Bearing Pursuit

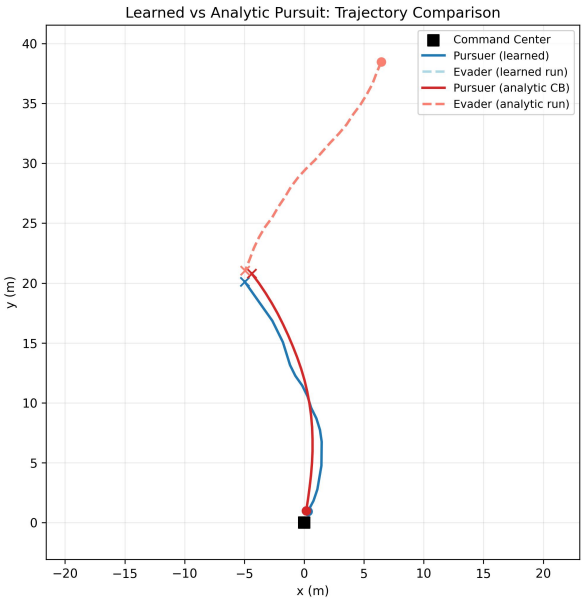
Fig. 6 Representative pursuit–evasion trajectories comparing learned and analytical guidance laws against homing evasion.



(a) Learned vs. Homing Pursuit

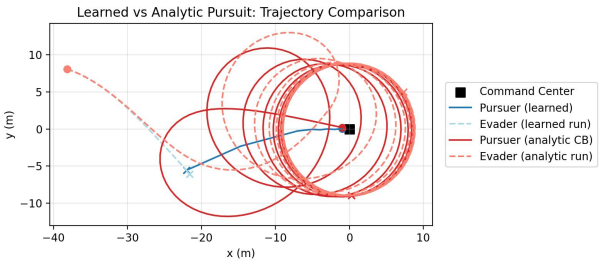


(b) Learned vs. Deviated Pursuit

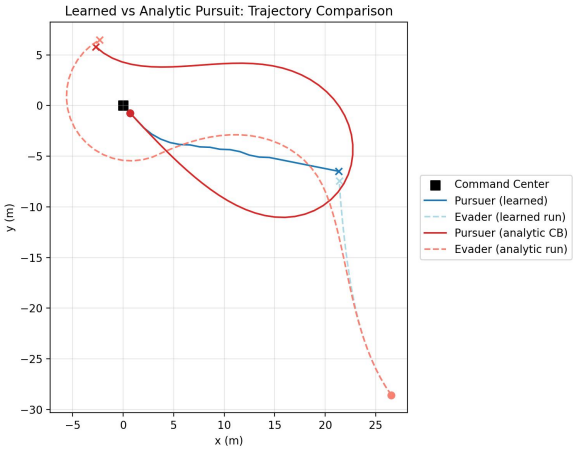


(c) Learned vs. Constant-Bearing Pursuit

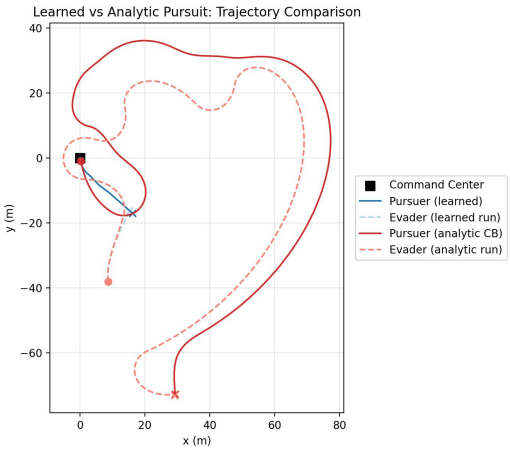
Fig. 7 Representative pursuit–evasion trajectories comparing learned and analytical guidance laws against random evasion.



(a) Learned vs. Homing Pursuit

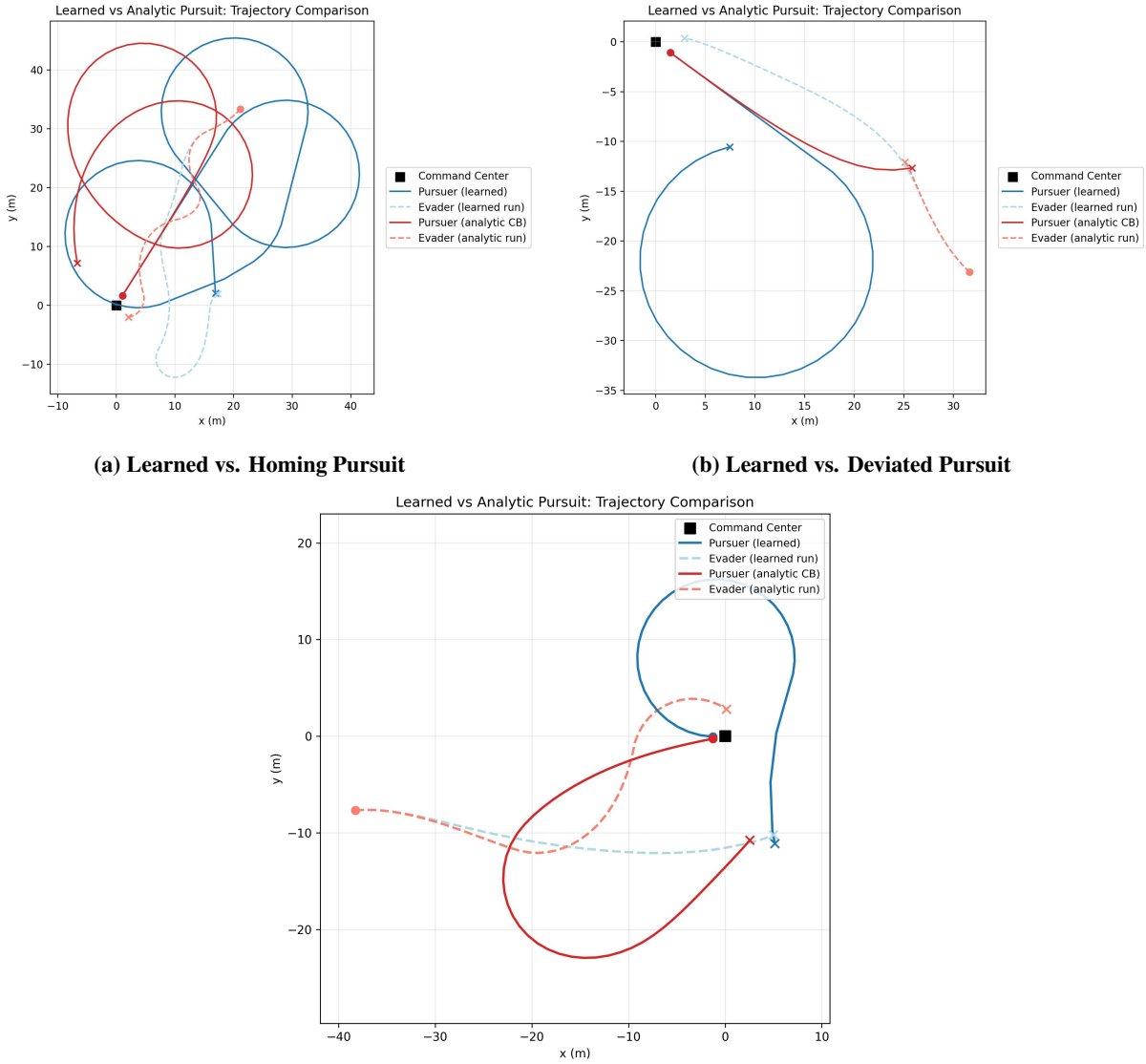


(b) Learned vs. Deviated Pursuit



(c) Learned vs. Constant-Bearing Pursuit

Fig. 8 Representative pursuit–evasion trajectories comparing learned and analytical guidance laws against alpha-blend evasion.



(a) Learned vs. Homing Pursuit

(b) Learned vs. Deviated Pursuit

(c) Learned vs. Constant-Bearing Pursuit

Fig. 9 Representative pursuit–evasion trajectories comparing learned and analytical guidance laws against alpha-blend with randomized velocities evasion.

F. Code Availability

The code used to generate all results is available at:

- Repository: Deep-Q-Learning-for-Control-in-a-Simple-Pursuit-Evasion-Game
- Key scripts: `TRAINING.py`, `TESTING.py`, `EVALUATION.py`, `pursuit_evasion_env.py`