aerospace

MDPI

Article

Multi-Missile Curvature Cooperative Guidance Method Based on LSTM-IMM Target Tracking

Hong Zhao and Wenjun Yi *

National Key Lab of Transient Physics, Nanjing University of Science and Technology, Nanjing 210094, China; zhaohong@njust.edu.cn

* Correspondence: yi_wj@njust.edu.cn

Abstract

Multi-missile interception of maneuvering targets requires accurate prediction and impact-time coordination under fixed missile speed. This study couples LSTM-IMM tracking with active curvature planning. LSTM generates a time-varying IMM transition matrix from measurement features, and the guidance law converts fixed-speed coordination into curvature and normal-acceleration allocation. Comparisons, sensitivity tests, Monte Carlo simulations, and five engagement scenarios verify tracking, mode recognition, real-time feasibility, and interception without speed regulation.

Keywords: multi-missile cooperative guidance; maneuvering target tracking; LSTM-IMM; time coordination; active curvature planning

1. Introduction

Multi-missile cooperative interception improves engagement probability against maneuvering targets [1]. Simultaneous arrival further increases penetration and damage effects [2,3], but impact-time coordination remains difficult when missile speed is fixed and target motion is uncertain [4].

The problem involves three coupled difficulties: estimating target states under unknown maneuvers [5,6], predicting evasive trajectories [7], and coordinating impact time under planning and guidance constraints [8,9].

Existing studies provide useful but partial solutions. ESO and FTDO observers estimate disturbances but face convergence or noise-sensitivity limits [10–12]. LSTM and IMM methods support temporal prediction and multi-model tracking [13,14], while transition-matrix selection remains important [15–17]. Bayesian, strong-tracking, and adaptive IMM variants improve switching behavior but can be sensitive to residual noise [18,19]. Cooperative guidance methods based on virtual targets, trajectory shaping, and time-to-go estimation address arrival coordination but often assume adjustable speed or weak coupling with target prediction [20–22].

These studies do not fully close the prediction-guidance loop. Observer methods estimate disturbances without providing mode-transition information; residual-adaptive IMM methods react locally; standalone deep predictors do not enforce IMM probability structure; and many cooperative guidance laws do not use prediction confidence in fixed-speed coordination.

The remaining gaps can be summarized as follows. Frequent model switching can destabilize estimation when only instantaneous observations are used [23,24]. A single predictor has limited adaptability to diverse maneuvers [25]. Fixed-speed cooperative

Check for updates

Academic Editor: Mingzhe Hou

Received: 25 June 2026
Revised: 23 July 2026
Accepted: 27 July 2026
Published: 28 July 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

guidance still lacks an integrated mechanism that links tracking confidence, path-length regulation, and terminal-hit priority.

This paper addresses the lack of an integrated method that learns maneuver-mode transitions from history and uses that information for fixed-speed cooperative guidance.

The framework uses IMM for multi-model robustness [26,27], LSTM for temporal transition learning [28,29], and curvature-based path-length regulation for fixed-speed time coordination [30,31].

The contribution is the closed-loop use of an LSTM-generated, row-normalized IMM transition matrix for both prediction and cooperative guidance. The guidance module then adjusts arrival time through curvature-induced path-length regulation rather than missile-speed variation.

To make the distinction from representative state-of-the-art studies explicit, Table 1 summarizes the technical focus, remaining limitations, and corresponding advantages of the proposed method.

Table 1. Comparative analysis of representative literature and the proposed method.

Research Category	Representative Technical Focus	Remaining Limitation for Fixed-Speed Cooperative Interception	Relation to the Proposed Method
ESO/FTDO observer-based tracking and guidance [10–12]	Estimate lumped disturbance or maneuver acceleration and compensate for model mismatch.	They do not provide maneuver-mode transition probabilities, and convergence or noise sensitivity can affect prediction during abrupt maneuvers.	The proposed LSTM-IMM uses an IMM model set and a history-dependent transition matrix to support both filtering and prediction confidence.
Conventional IMM and LSTM-aided target prediction [13–17,26–29]	Combine multiple motion models or learn temporal trajectory features for maneuvering target estimation.	Fixed or manually tuned transition matrices may lag during mode changes, while standalone deep predictors do not preserve the IMM probability structure.	The LSTM outputs a row-normalized time-varying transition matrix that is reused in IMM interaction and short-term prediction.
Bayesian, strong-tracking, and residual-adaptive IMM methods [18,19]	Adjust model probabilities using innovation, residual, or Bayesian consistency information.	Local residual fluctuations can cause abrupt switching under measurement noise, and the tracking result is often weakly coupled with guidance.	The historical feature window reduces sensitivity to isolated innovations and provides predicted intercept-point and confidence information for guidance.
Cooperative impact-time guidance with virtual targets, trajectory shaping, or time-to-go coordination [20–22,30,31]	Coordinate arrival time by shaping paths, assigning virtual targets, or regulating time-to-go.	Many methods rely on speed adjustment or do not explicitly link uncertain target prediction with overload-feasible time planning.	The proposed guidance law keeps missile speed fixed and converts time coordination into curvature-path-length and normal-acceleration allocation.
Modern sequence predictors such as GRU-IMM and Transformer-style predictors [25,28,29]	Use recurrent or attention-based sequence models to learn nonlinear maneuver history.	They can increase implementation cost and may require additional constraints to produce interpretable model-transition probabilities.	The LSTM module is constrained by row-wise Softmax before IMM interaction, and the simulation section reports runtime and comparison results.

Table 1. *Cont.*

Research Category	Representative Technical Focus	Remaining Limitation for Fixed-Speed Cooperative Interception	Relation to the Proposed Method
Communication-aware cooperative guidance studies [1–4]	Use multi-agent coordination to improve simultaneous interception or engagement probability.	Communication delay, loss, and synchronization are often treated separately from learned target tracking and prediction.	This paper establishes the fixed-speed prediction-guidance baseline and discusses communication impairments as practical engineering limitations.
Proposed integrated framework	LSTM-IMM transition learning, prediction confidence, feasible common-arrival-time planning, and curvature cooperative guidance.	The present study assumes ideal communication and point-mass missile dynamics.	The main contribution is the closed-loop use of learned mode-transition information in fixed-speed time planning, hit-priority overload allocation, and terminal reference fusion.

The main contributions of this study are summarized as follows:

1. First, an LSTM-driven transition-probability generator is used in the IMM tracking loop. In contrast to fixed-matrix IMM or residual/threshold-based adaptive IMM, the LSTM maps historical measurement features to a constrained time-varying transition matrix for the next IMM interaction and prediction cycle. This formulation preserves the multi-model filtering structure and supplies history-dependent maneuver-mode probabilities.
2. Second, fixed-speed time coordination is formulated through active trajectory curvature. Relative to velocity-regulation, virtual-target, or purely consensus-based cooperative guidance approaches, this formulation derives the curvature-path length-remaining time mapping and converts simultaneous arrival into normal-acceleration allocation under overload limits.
3. Third, an integrated prediction-guidance decision framework is constructed. LSTM-IMM prediction generates predicted intercept points and confidence information, while the guidance law performs feasible common-arrival-time assessment, minimum-variance allocation for infeasible cases, hit-priority overload allocation, and terminal reference-state fusion. This structure differs from loosely coupled tracking-plus-guidance schemes because prediction confidence directly affects time feasibility, overload allocation, and terminal reference-state selection.

The remainder of the paper presents the model, LSTM-IMM prediction method, curvature-based cooperative guidance law, simulations, and conclusions.

2. Modeling the Problem of Multi-Missile Cooperative Interception

This section defines the interception geometry, missile and target kinematics, relative motion, and fixed-speed cooperative interception objective.

2.1. Interception Scenarios and Overall System Framework

Consider N missiles intercepting one maneuvering target in a three-dimensional inertial frame. The state variables are defined in meters and seconds.

The *i-th* missile is denoted as M_i, where $i \in \{1, 2, \cdots, N\}$. The missile position vector and velocity vector are defined as follows:

$$r_i(t) = [x_i(t) \;\; y_i(t) \;\; z_i(t)]^T \in \mathbb{R}^3, v_i(t) = \left[v_{xi}(t) \;\; v_{yi}(t) \;\; v_{zi}(t)\right]^T \in \mathbb{R}^3, \quad (1)$$

where $r_i(t)$ represents the position of the *i-th* missile in the inertial coordinate system, $x_i(t), y_i(t), z_i(t) \in \mathbb{R}$, and usually $z_i(t) \geq 0$; $v_i(t)$ represents the missile's velocity vector.

The multi-missile cooperative interception system consists of a target state acquisition module, a target state estimation and prediction module, a multi-missile cooperative guidance module, a terminal guidance module, and a missile motion update module. The overall system flow is shown in Figure 1.

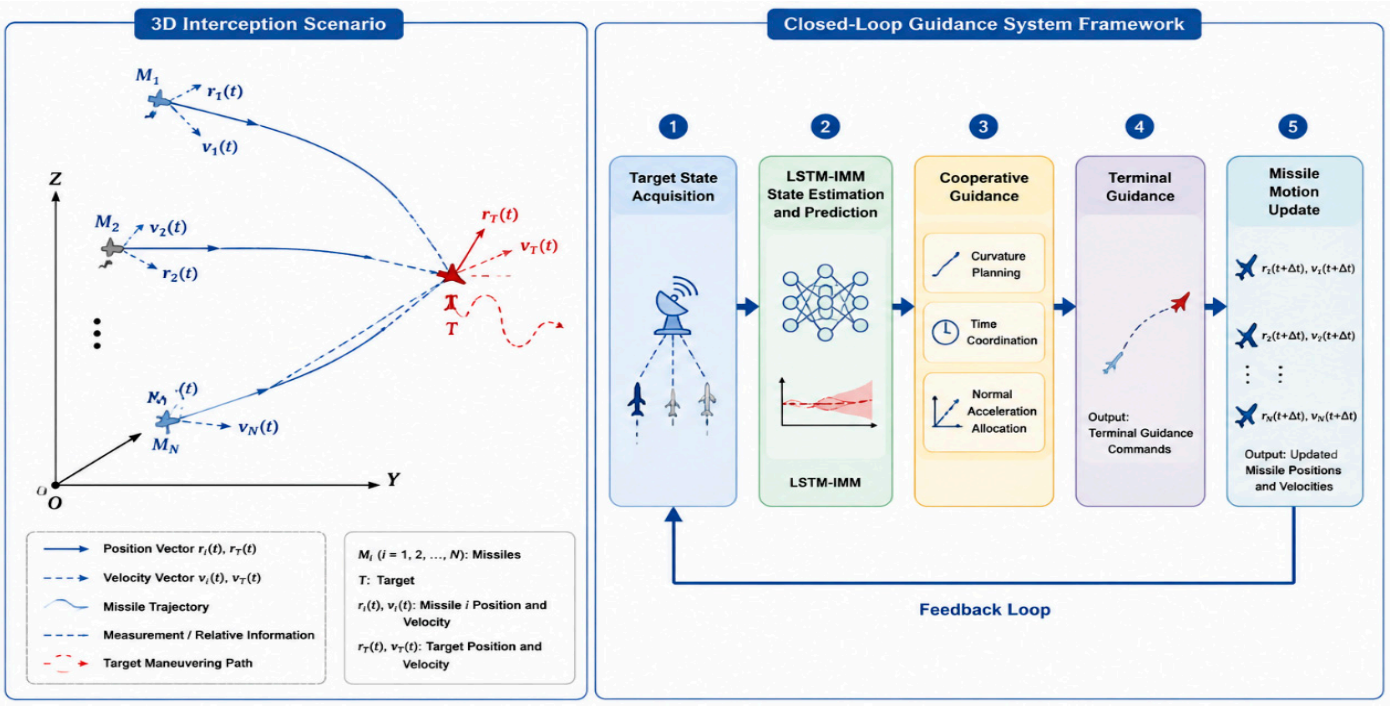


Figure 1. Overall framework of the three-dimensional multi-missile cooperative interception system for a maneuvering target.

2.2. Missile Kinematic Model

The missile is modeled as a point mass, and guidance commands are represented by equivalent normal acceleration.

For missile i, the speed, trajectory inclination, and heading angle are denoted by the corresponding velocity magnitude and angular variables.

The missile velocity vector can be expressed as:

$$v_i(t) = V_i(t) \begin{bmatrix} \cos\gamma_i(t)\cos\psi_i(t) \\ \cos\gamma_i(t)\sin\psi_i(t) \\ \sin\gamma_i(t) \end{bmatrix}, \tag{2}$$

Therefore, the kinematic equation of position for the *i-th* missile is:

$$\dot{r}_i(t) = v_i(t), \tag{3}$$

This paper focuses on the cooperative guidance problem of multiple missiles under a fixed velocity constraint. Therefore, it is assumed that the magnitude of the missile's flight velocity remains constant.

$$V_i(t) = V_{i0}, 0 < V_{\min} < V_{i0} < V_{\max}, i = 1, 2, \cdots, N, \tag{4}$$

where V_{i0} is the constant flight velocity of the *i-th* missile, and $V_{\min}$ and $V_{\max}$ are the lower and upper bounds of the missile's allowed velocity, respectively, both positive real numbers.

Let the guidance acceleration of the *i-th* missile be $a_i(t) \in \mathbb{R}^3$, its direction being perpendicular to the current velocity vector. Since the velocity magnitude is fixed, the missile acceleration satisfies:

$$\dot{v}_i(t) = a_i(t), v_i^{\mathrm{T}}(t)a_i(t) = 0. \tag{5}$$

where $a_i(t)$ is the normal acceleration vector, in $\frac{m}{s^2}$. Considering the limited maneuverability of the missile, let its maximum usable normal acceleration be $a_{i,\max} > 0$, then we have

$$\|a_i(t)\| \leq a_{i,\max}, i = 1, 2, \cdots, N. \tag{6}$$

In angular form, let $a_{\gamma i}(t) \in \mathbb{R}$ be the normal acceleration component changing the trajectory inclination angle and $a_{\psi i}(t) \in \mathbb{R}$ be the normal acceleration component changing the heading angle. Both satisfy the following relationship with the rate of change of the trajectory angle:

$$\dot{\gamma}_i(t) = \frac{a_{\gamma i}(t)}{V_{i0}}, \ \dot{\psi}_i(t) = \frac{a_{\psi \mathrm{i}}(t)}{V_{i0}\cos\gamma_i(t)}, \tag{7}$$

To avoid heading angle singularity, $\cos\gamma_i(t) \neq 0$, i.e., $\gamma_i(t) \in \left(-\frac{\Pi}{2}, \frac{\Pi}{2}\right)$.

The missile acceleration amplitude constraint can be further written as:

$$a_{\gamma i}^2(t) + a_{\psi i}^2(t) \leq a_{i,\max}^2, \tag{8}$$

Since the missile's velocity is constant, the curvature of its trajectory is determined by its normal acceleration. Let $\kappa_i(t) \geq 0$ be the trajectory curvature of the *i-th* missile, then we have:

$$\kappa_i(t) = \frac{\|a_i(t)\|}{V_{i0}^2}, \tag{9}$$

Equation (9) shows that fixed-speed missiles adjust path length and arrival time through normal-acceleration-induced curvature.

2.3. Motion Model of Maneuvering Target

Targets typically possess uncertain maneuverability. Let the target acceleration vector be $a_T(t) \in \mathbb{R}^3$, in $\frac{m}{s^2}$, and its maximum maneuvering acceleration be $a_{T,\max} > 0$. The continuous motion model of the target can be written as:

$$\dot{r}_T(t) = v_T(t), \dot{v}_T(t) = a_T(t), \|a_T(t)\| \leq a_{T,\max}, \tag{10}$$

Here, $a_T(t)$ describes the target's maneuverability. The target's velocity magnitude $V_T(t) = \| v_T(t) \|$ usually satisfies

$$0 \leq V_{T,\min} \leq V_T(t) \leq V_{T,\max}, \tag{11}$$

where $V_{T,\min}$ and $V_{T,\max}$ are the lower and upper bounds of the target velocity, respectively.

To describe different types of target maneuvers, this paper adopts a multi-model description approach. The target motion mode is denoted as $m(t)$, and its value set is as follows:

$$m(t) \in \mathcal{M} = \{\mathrm{CV}, \mathrm{CA}, \mathrm{CT}\}, \tag{12}$$

where CV represents the uniform linear motion model, CA represents the uniform acceleration motion model, and CT represents the coordinated turning motion model.

2.4. Model of Missile-Target Relative Motion

To describe the relative motion between the *i-th* missile and the target, the missile-target relative position vector is defined as follows:

$$\rho_i(t) = r_T(t) - r_i(t), \rho_i(t) \in \mathbb{R}^3, \tag{13}$$

where $\rho_i(t)$ represents the relative position vector from the missile M_i to target T.

The missile-target relative velocity vector is defined as:

$$\dot{\rho}_i(t) = v_T(t) - v_i(t), \tag{14}$$

The relative distance between the missile and the target is defined as follows:

$$R_i(t) = \|\rho_i(t)\|, R_i(t) \geq 0, \tag{15}$$

When $R_i(t) > 0$, the unit vector of the missile-target line of sight is defined as

$$\lambda_i(t) = \frac{\rho_i(t)}{R_i(t)}, \|\lambda_i(t)\| = 1 \tag{16}$$

The rate of change of the distance between the missile and the target is

$$\dot{R}_i(t) = \frac{\rho_i^T(t)\dot{\rho}_i(t)}{R_i(t)} \tag{17}$$

Closing velocity is defined as the velocity at which the missile-target distance decreases, denoted as $V_{ci}(t)$, and has:

$$V_{ci}(t) = -\dot{R}_i(t) = -\frac{\rho_i^T(t)\dot{\rho}_i(t)}{R_i(t)} \tag{18}$$

where $V_{ci}(t) > 0$ indicates that the distance between the missile and the target is decreasing, and $V_{ci}(t) < 0$ indicates that the distance is increasing.

The missile-target line-of-sight angular velocity vector is defined as:

$$\omega_{Li}(t) = \frac{\rho_i(t) \times \dot{\rho}_i(t)}{R_i^2(t)} \tag{19}$$

where $\omega_{Li}(t) \in \mathbb{R}^3$, in units of rad/s, and $\times$ denotes the vector cross product. This quantity reflects the rate of change of the line-of-sight direction and is an important input quantity for terminal proportional guidance laws.

Let the hit criterion distance be $R_{hit} > 0$. The hit time of the *i-th* missile is defined as

$$t_{hi} = \inf\left\{ t \in \left[0 \middle| t_f\right] \middle| R_i(t) \leq R_{hit} \right\} \tag{20}$$

where t_{hi} represents the moment when the *i-th* missile first satisfies the hit criterion. If there is no moment satisfying Equation (20) within $\left[0, t_f\right]$, then the missile is considered to have missed the target.

The miss distance of the *i-th* missile is defined as

$$d_i = \min_{t \in [0, t_f]} R_i(t), d_i > 0 \tag{21}$$

When $d_i \leq R_{hit}$, the *i-th* missile can be considered to have achieved a valid hit.

2.5. Description of the Problem of Multi-Missile Cooperative Guidance

The goal of multi-missile coordinated interception is not only to ensure that each missile hits the target but also to require that multiple missiles arrive near the target as simultaneously as possible, thereby improving the reliability of intercepting maneuvering targets.

Let the set of all missile impact times be $\{t_{h1}, t_{h2}, \cdots, t_{hN}\}$. The constraint of simultaneous arrival of multiple missiles can be expressed as:

$$\max_{1\le i\le N} t_{hi} - \min_{1\le i\le N} t_{hi} \le \varepsilon_t \tag{22}$$

where $\varepsilon_t \ge 0$ is the allowable hit-time error threshold, in seconds. If $\varepsilon_t = 0$, it indicates a strictly simultaneous hit; in practical engineering, $\varepsilon_t > 0$ is usually chosen.

The average hit time is defined as:

$$\bar{t}_h = \frac{1}{N}\sum_{i=1}^{N} t_{hi} \tag{23}$$

The consistency of hit time can be described by the standard deviation of hit time:

$$\sigma_t = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(t_{hi} - \bar{t}_h\right)^2} \tag{24}$$

where $\sigma_t \ge 0$, the smaller the value, the better the time coordination of multiple missiles.

Under fixed speed, the remaining flight time is changed by path length rather than velocity magnitude:

$$\tau_i(t) = \frac{L_i(t)}{V_{i0}}, L_i(t) \ge \|p_i - r_i(t)\| \tag{25}$$

where $\tau_i(t) \ge 0$ represents the remaining flight time of the *i-th* missile, and p_i represents the desired interception point or capture point corresponding to that missile. Equation (25) shows that when the velocity V_{i0} is fixed, the missile can only adjust its remaining flight time by changing the path length $L_i(t)$.

As shown in Equation (9), the adjustment of the path length essentially depends on the planning of the trajectory curvature $\kappa_i(t)$. Therefore, the time coordination problem under a fixed velocity constraint can be transformed into an active curvature planning problem.

Combining hit accuracy, time coordination, and overload constraints, the multi-missile cooperative interception problem can be described as follows: Under the condition that the target motion satisfies Equation (10) and the missile motion satisfies Equations (3)–(9), design the guidance acceleration $a_i(t)$ of each missile such that

$$J = \sum_{i=1}^{N} d_i^2 + \eta\sum_{i=1}^{N}\left(t_{hi} - \bar{t}_h\right)^2 + \beta\sum_{i=1}^{N}\int_0^{t_{hi}} \|a_i(t)\|^2\, dt \tag{26}$$

where $J \ge 0$ is the cooperative interception performance index and $\eta \ge 0$ is the time-coordination weighting coefficient. The first term measures the miss rate and the second term measures the consistency of hit time.

The optimization problem must satisfy the following constraints:

$$\begin{cases} \dot{r}_i(t) = v_i(t), \\ \dot{v}_i(t) = a_i(t), \\ \|v_i(t)\| = V_{i0}, \\ v_i^T(t)a_i(t) = 0, \\ \|a_i(t)\| \le a_{i,\max}, \\ R_i(t_{hi}) \le R_{hit}, \end{cases} i = 1, 2, \cdots, N. \tag{27}$$

In Equation (27), the first four constraints describe the missile's kinematic characteristics at a fixed velocity, the fifth constraint describes the missile's overload capacity limitation, and the last constraint describes the hit condition.

Therefore, the core issues for subsequent research in this paper can be summarized as follows:

1. How to accurately estimate and predict the target state when the target exhibits maneuvering and measurement noise;
2. How to achieve simultaneous arrival of multiple missiles through trajectory curvature planning under the conditions of fixed missile velocity and limited overload;
3. How to ensure that the missile has a sufficiently strong convergence capability towards maneuvering targets during the terminal rendezvous process.

3. An LSTM-IMM-Based Method for Maneuvering Target Tracking and Prediction

Accurate target estimation and prediction are required before cooperative guidance can be planned. Because the target may switch among CV, CA, and CT modes, this section combines IMM filtering with an LSTM-generated transition matrix.

This section defines the discrete state and measurement models, introduces the CV, CA, and CT motion models, reviews IMM estimation, and then replaces the fixed transition matrix with an LSTM-generated time-varying matrix for prediction and guidance.

3.1. Target Discrete State and Measurement Model

Let the system sampling period be $T_s > 0$, and the discrete time be denoted as $t_k = kT_s,\ k = 0, 1, 2, \cdots$, where k is the discrete time index. To simultaneously describe the target's position, velocity, and acceleration information, the target discrete state vector is defined as:

$$x_{T,k} = \begin{bmatrix} x_{T,k} & y_{T,k} & z_{T,k} & v_{xT,k} & v_{yT,k} & v_{zT,k} & a_{xT,k} & a_{yT,k} & a_{zT,k} \end{bmatrix}^T \in \mathbb{R}^9 \tag{28}$$

where $[x_{T,k}, y_{T,k}, z_{T,k}]^T$ is the target's position vector in the inertial coordinate system, $[v_{xT,k}, v_{yT,k}, v_{zT,k}]^T$ is the target's velocity vector, and $[a_{xT,k}, a_{yT,k}, a_{zT,k}]^T$ is the target's acceleration vector.

The discrete motion model of the target can be uniformly represented as:

$$x_{T,k} = f_{m_k}(x_{T,k-1}) + w_{m_k,k-1} \tag{29}$$

Using Equation (12), $f_{m_k}(\cdot)$ is the state transition function for the corresponding motion mode; $w_{m_k,k-1}$ is the process noise, which is usually assumed to satisfy:

$$w_{m_k,k-1} \sim \mathcal{N}(0, Q_{m_k,k-1}) \tag{30}$$

where $Q_{m_k,k-1}$ is the process noise covariance matrix corresponding to the motion model m_k.

During target tracking, the sensor can obtain information about the target's position, velocity magnitude, and velocity direction angle. The target measurement vector is defined as:

$$z_{T,k} = \begin{bmatrix} x_{T,k}^m & y_{T,k}^m & z_{T,k}^m & V_{T,k}^m & \theta_{T,k}^m & \psi_{T,k}^m \end{bmatrix}^T \in \mathbb{R}^6 \tag{31}$$

where $x_{T,k}^m, y_{T,k}^m, z_{T,k}^m$ are the measured values of the target position, $V_{T,k}^m$ is the measured value of the target velocity magnitude, and $\theta_{T,k}^m$ and $\psi_{T,k}^m$ are the measured values of the target velocity tilt angle and heading angle, respectively.

The measurement model can be written as follows:

$$z_{T,k} = h(x_{T,k}) + v_k \tag{32}$$

where $h(\cdot)$ is a nonlinear measurement function, and v_k is the measurement noise, satisfying:

$$v_k \sim \mathcal{N}(0, R_k) \tag{33}$$

where R_k is the measurement noise covariance matrix.

According to the definition of the state vector, the nonlinear measurement function can be expressed as:

The measurement function is obtained by converting the Cartesian velocity into speed, elevation, and azimuth. Equation (36) is the Euclidean norm of the velocity vector, and Equation (36) follows from the elevation angle between the velocity vector and its horizontal projection.

Equation (37) gives the azimuth of the horizontal velocity projection. The two-argument arctangent uses the horizontal velocity components simultaneously, so it preserves the correct quadrant and avoids the sign ambiguity of a single arctangent.

$$h(x_{T,k}) = \begin{bmatrix} x_{T,k} \\ y_{T,k} \\ z_{T,k} \\ V_{T,k} \\ \theta_{T,k} \\ \psi_{T,k} \end{bmatrix} \tag{34}$$

where

$$V_{T,k} = \sqrt{v_{xT,k}^2 + v_{yT,k}^2 + v_{zT,k}^2} \tag{35}$$

$$\theta_{T,k} = \arctan \frac{v_{zT,k}}{\sqrt{v_{xT,k}^2 + v_{yT,k}^2}} \tag{36}$$

$$\psi_{T,k} = \operatorname{atan} 2\left(v_{yT,k}, v_{xT,k}\right) \tag{37}$$

where $\operatorname{atan} 2(\cdot)$ denotes the two-argument arctangent function. Since the measurement equation includes velocity magnitude and angle information, the target tracking filtering process is nonlinear.

3.2. Development of Multi-Model Target Motion Models

To describe the different maneuvering characteristics of targets, this paper uses three types of models—CV, CA, and CT—to form a set of maneuvering target models. Each model uses the 9-dimensional state vector shown in Equation (28), but the state transition relationships are different.

3.2.1. Uniform Linear Motion Model

The CV model describes weak-maneuver or nearly uniform straight-line motion, with acceleration represented mainly by process noise.

$$x_{T,k} = F_{CV} x_{T,k-1} + w_{CV,k-1} \tag{38}$$

where $F_{CV} \in \mathbb{R}^{9\times 9}$ is the state transition matrix of the CV model, which can be expressed as:

$$F_{CV} = \begin{bmatrix} I_3 & T_s I_3 & 0_3 \\ 0_3 & I_3 & 0_3 \\ 0_3 & 0_3 & 0_3 \end{bmatrix} \tag{39}$$

Here, the deterministic acceleration term is zero, and acceleration uncertainty is absorbed into the process noise.

3.2.2. Uniformly Accelerated Motion Model

When a target exhibits continuous acceleration maneuvers, the CA model can be used to describe it. This model assumes that the target acceleration remains approximately constant within one sampling period, and its discrete state equation is:

$$x_{T,k} = F_{CA} x_{T,k-1} + w_{CA,k-1} \tag{40}$$

where

$$F_{CA} = \begin{bmatrix} I_3 & T_s I_3 & \frac{1}{2} T_s^2 I_3 \\ 0_3 & I_3 & T_s I_3 \\ 0_3 & 0_3 & I_3 \end{bmatrix} \tag{41}$$

The CA model can describe the acceleration or deceleration of a target in a short period of time. However, when the target makes a significant turning maneuver, the CA model alone may not be able to accurately characterize the change in velocity direction.

3.2.3. Coordinated Turning Motion Model

The CT model describes horizontal turning motion through the turning angular velocity, which couples horizontal position and velocity.

Let the target's horizontal state be:

$$x_{h,k} = \begin{bmatrix} x_{T,k} & y_{T,k} & v_{xT,k} & v_{yT,k} \end{bmatrix}^T \tag{42}$$

The horizontal state transition of the CT model can then be written as:

Equation (44) follows from the constant-turn-rate assumption over one sampling interval. With a fixed turning rate, the horizontal velocity vector rotates during the interval; integrating the rotated velocity gives the sine-cosine position terms in the transition matrix. When the turning rate approaches zero, the limiting form reduces to the constant-velocity horizontal model, so the CT model remains consistent with the CV case.

$$x_{h,k} = F_{CT,h}(\omega_k) x_{h,k-1} + w_{CT,h,k-1} \tag{43}$$

where

$$F_{CT,h}(\omega_k) = \begin{bmatrix} 1 & 0 & \frac{\sin\omega_k T_s}{\omega_k} & -\frac{1-\cos\omega_k T_s}{\omega_k} \\ 0 & 1 & \frac{1-\cos\omega_k T_s}{\omega_k} & \frac{\sin\omega_k T_s}{\omega_k} \\ 0 & 0 & \cos\omega_k T_s & -\sin\omega_k T_s \\ 0 & 0 & \sin\omega_k T_s & \cos\omega_k T_s \end{bmatrix} \tag{44}$$

When $\omega_k \to 0$, the CT model reduces to an approximately uniform linear model. For vertical motion, a one-dimensional uniform acceleration approximation can be used, i.e.,

$$\begin{bmatrix} z_{T,k} \\ v_{zT,k} \\ a_{zT,k} \end{bmatrix} = \begin{bmatrix} 1 & T_s & \frac{1}{2}T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_{T,k-1} \\ v_{zT,k-1} \\ a_{zT,k-1} \end{bmatrix} + w_{CT,z,k-1} \tag{45}$$

For coordinated turning motion on a horizontal plane, the target's horizontal velocity changes continuously with respect to its angular velocity ω_k. According to the kinematics of coordinated turning, the horizontal acceleration is determined by the horizontal velocity and the turning angular velocity, i.e.,

$$a_{xT,k} = -\omega_k v_{yT,k} \tag{46}$$

Therefore, in the CT model, the target's horizontal acceleration is not an independently propagating state but is determined by the predicted horizontal velocity component and the turning angular velocity. Let $\Omega_k = \omega_k T_s$, then the one-step prediction of the horizontal position and velocity is:

$$x_{T,k} = x_{T,k-1} + \frac{\sin\Omega_k}{\omega_k} v_{xT,k-1} - \frac{1-\cos\Omega_k}{\omega_k} v_{yT,k-1} \tag{47}$$

$$y_{T,k} = y_{T,k-1} + \frac{1-\cos\Omega_k}{\omega_k} v_{xT,k-1} + \frac{\sin\Omega_k}{\omega_k} v_{yT,k-1} \tag{48}$$

$$v_{xT,k} = \cos\Omega_k v_{xT,k-1} - \sin\Omega_k v_{yT,k-1} \tag{49}$$

$$v_{yT,k} = \sin\Omega_k v_{xT,k-1} + \cos\Omega_k v_{yT,k-1} \tag{50}$$

After predicting the horizontal velocity, the horizontal acceleration is updated to:

$$a_{xT,k} = -\omega_k v_{yT,k} \tag{51}$$

$$a_{yT,k} = \omega_k v_{xT,k} \tag{52}$$

A one-dimensional uniform acceleration model is used in the vertical direction:

$$z_{T,k} = z_{T,k-1} + T_s v_{zT,k-1} + \frac{1}{2}T_s^2 a_{zT,k-1} \tag{53}$$

$$v_{zT,k} = v_{zT,k-1} + T_s a_{zT,k-1} \tag{54}$$

$$a_{zT,k} = a_{zT,k-1} \tag{55}$$

From Equations (47)–(55), the complete 9-dimensional CT state transition function is obtained:

$$x_{T,k} = f_{CT}(x_{T,k-1}, \omega_k) + w_{CT,k-1} \tag{56}$$

CT models can reflect continuous changes in the target's velocity direction and are suitable for local turning phases during target maneuvers such as hovering, evasion, or serpentine maneuvers.

3.3. Interactive Multi-Model State Estimation Method

IMM estimation runs multiple model filters in parallel, updates their probabilities from measurement consistency, and fuses the model-conditioned estimates.

Suppose there are M models in the model set, and in this paper, $M = 3$. The probability of the *j-th* model at time t_k is denoted as:

$$\mu_{j,k} = P(m_k = j \mid Z_k) \tag{57}$$

where $Z_k = \{z_{T,1}, z_{T,2}, \cdots, z_{T,k}\}$ represents the measurement sequence up to time t_k, and the model probability satisfies

$$\sum_{j=1}^{M} \mu_{j,k} = 1, \mu_{j,k} \geq 0 \tag{58}$$

In traditional IMM models, the transition probability matrix is usually set as a constant matrix.

$$\Pi = [p_{ij}]_{M \times M} \tag{59}$$

where p_{ij} represents the probability that the target motion pattern transitions from model i to model j, and satisfies

$$p_{ij} \geq 0, \sum_{j=1}^{M} p_{ij} = 1 \tag{60}$$

IMM filtering mainly includes four steps: model interaction, model conditional filtering, model probability update, and fusion output.

3.3.1. Model Interaction

At time t_k, the predicted probability of model j can be calculated from the model probability and model transition probability of the previous time step:

$$c_{j,k} = \sum_{i=1}^{M} p_{ij,k-1} \mu_{i,k-1} \tag{61}$$

where $p_{ij,k-1}$ is the model transition probability from time t_{k-1} to t_k. The mixture probability from model i to model j is:

$$\mu_{i|j,k-1} = \frac{p_{ij,k-1} \mu_{i,k-1}}{c_{j,k}} \tag{62}$$

where $\mu_{i|j,k-1}$ represents the contribution of model i to the initial state mixture at the previous time step under the current condition of using model j.

Let the state estimate and covariance of model i at the previous time step be $\hat{x}_{i,k-1}$ and $P_{i,k-1}$, respectively. Then the initial state mixture of model j is:

$$\hat{x}_{0j,k-1} = \sum_{i=1}^{M} \mu_{i|j,k-1} \hat{x}_{i,k-1} \tag{63}$$

The initial covariance of the mixture is

$$P_{0j,k-1} = \sum_{i=1}^{M} \mu_{i|j,k-1} \left[P_{i,k-1} + \left(\hat{x}_{i,k-1} - \hat{x}_{0j,k-1} \right) \left(\hat{x}_{i,k-1} - \hat{x}_{0j,k-1} \right)^T \right] \tag{64}$$

Equations (63) and (64) enable information exchange between state estimation results of different models.

3.3.2. Model-Based Filtering

For the *j-th* model, time updates are performed using a mixed initial state.

$$\hat{x}_{j,k|k-1} = f_j \left(\hat{x}_{0j,k-1} \right) \tag{65}$$

$$P_{j,k|k-1} = F_{j,k} P_{0j,k-1} F_{j,k}^T + Q_{j,k-1} \tag{66}$$

where $F_{j,k}$ is the state transition matrix or Jacobian matrix of the nonlinear state function of model j at time t_k.

The measured predicted value is:

$$\hat{z}_{j,k|k-1} = h\left(\hat{x}_{j,k|k-1}\right) \tag{67}$$

Measurement residual is

$$e_{j,k} = z_{T,k} - \hat{z}_{j,k|k-1} \tag{68}$$

The angular component in the measurement residuals should be normalized to fall within the range of $(-\Pi, \Pi]$ to avoid abrupt changes in the residuals caused by angles crossing boundaries. The residual covariance is…

$$S_{j,k} = H_{j,k} P_{j,k|k-1} H_{j,k}^T + R_k \tag{69}$$

where $H_{j,k}$ is the Jacobian matrix of the measurement function $h(\cdot)$ with respect to the state variables. The filter gain is:

$$K_{j,k} = P_{j,k|k-1} H_{j,k}^T S_{j,k}^{-1} \tag{70}$$

Therefore, the state of model j is updated as follows:

$$\hat{x}_{j,k} = \hat{x}_{j,k|k-1} + K_{j,k} e_{j,k} \tag{71}$$

Covariance updated to

$$P_{j,k} = \left(I - K_{j,k} H_{j,k}\right) P_{j,k|k-1} \tag{72}$$

where I is the identity matrix, which has the same dimension as the state.

3.3.3. Model Probability Update

The likelihood function of model j with respect to the current measurement can be written as:

$$\Lambda_{j,k} = \frac{\exp\left[-\frac{1}{2} e_{j,k}^T S_{j,k}^{-1} e_{j,k}\right]}{\sqrt{\left(2\Pi\right)^{n_z} \mid S_{j,k} \mid}} \tag{73}$$

where $n_z = 6$ is the measurement dimension, and $\mid S_{j,k} \mid$ represents the determinant of matrix $S_{j,k}$.

According to Bayes' theorem, the model probability is updated as follows:

$$\mu_{j,k} = \frac{\Lambda_{j,k} c_{j,k}}{\sum_{l=1}^{M} \Lambda_{l,k} c_{l,k}} \tag{74}$$

Equation (74) shows that when a model can better explain the current measurement, its model probability will increase; conversely, its model probability will decrease.

3.3.4. State Fusion Output

After obtaining the condition estimates for each model, the overall state estimate of the target is:

$$\hat{x}_{T,k} = \sum_{j=1}^{M} \mu_{j,k} \hat{x}_{j,k} \tag{75}$$

The overall covariance is

$$P_{T,k} = \sum_{j=1}^{M} \mu_{j,k} \left[P_{j,k} + \left(\hat{x}_{j,k} - \hat{x}_{T,k} \right) \left(\hat{x}_{j,k} - \hat{x}_{T,k} \right)^T \right] \tag{76}$$

Equations (75) and (76) give the IMM's final estimate of the target state. This result can be used as the target current state input in a multi-missile guidance system.

3.4. Adaptive Correction of Model Transition Probabilities Based on LSTM

Traditional IMM usually uses a fixed transition matrix. This is simple but cannot adjust switching probabilities from recent maneuver history, so the response may lag during abrupt target motion. Here, an LSTM generates the transition matrix online from historical motion features.

$$X_k^L = \{ \xi_{k-L+1}, \xi_{k-L+2}, \cdots, \xi_k \} \tag{77}$$

where $\xi_k \in \mathbb{R}^{12}$ is the input feature vector at the *k-th* sampling time. The LSTM network extracts the evolution law of the target maneuvering pattern based on the historical feature sequence and outputs the time-varying model transition probability matrix Π_k.

3.4.1. LSTM Training Data Generation

The LSTM is trained on simulated trajectories containing CV, CA, and CT segments. Each sample records true states, noisy measurements, mode labels, and oracle transition matrices generated from the known maneuver sequence.

The dataset contains 5000 simulated target trajectories sampled at 50 Hz for LSTM training, validation, and testing.

Each training sample contains a historical feature window and an oracle transition-matrix label. Measurement noise is randomized to improve robustness across observation conditions.

3.4.2. LSTM Input Feature Construction

Based on Equation (28), the target measurement vector is:

$$z_{T,k} = \left[x_{T,k}^m,\ y_{T,k}^m,\ z_{T,k}^m,\ V_{T,k}^m,\ \theta_{T,k}^m,\ \psi_{T,k}^m \right]^T \in \mathbb{R}^6 \tag{78}$$

To fully characterize the changing trends during the target's maneuver, this paper uses the current measurement vector and its first-order difference as the input features of the LSTM, defining the 12-dimensional feature vector at the *k-th* time step as follows:

The operator $wrap(\cdot)$ denotes angle normalization that maps an angular difference to the interval $[-\Pi,\ \Pi]$.

$$\xi_k = \left[z_{T,k}^T,\ \Delta z_{T,k}^T \right]^T \in \mathbb{R}^{12} \tag{79}$$

where

$$\Delta z_{T,k} = \left[\Delta x_{T,k}^m,\ \Delta y_{T,k}^m,\ \Delta z_{T,k}^m,\ \Delta V_{T,k}^m,\ \Delta \theta_{T,k}^m,\ \Delta \psi_{T,k}^m \right]^T \tag{80}$$

And there are

$$\Delta x_{T,k}^m = x_{T,k}^m - x_{T,k-1}^m, \Delta y_{T,k}^m = y_{T,k}^m - y_{T,k-1}^m, \Delta z_{T,k}^m = z_{T,k}^m - z_{T,k-1}^m \tag{81}$$

$$\Delta V_{T,k}^m = V_{T,k}^m - V_{T,k-1}^m, \Delta \theta_{T,k}^m = \text{wrap}\left(\theta_{T,k}^m - \theta_{T,k-1}^m \right), \Delta \psi_{T,k}^m = \text{wrap}\left(\psi_{T,k}^m - \psi_{T,k-1}^m \right) \tag{82}$$

Here, $\text{wrap}(\cdot)$ represents the angle normalization operator, used to limit the angle difference within the range $(-\Pi, \Pi]$. To avoid the influence of different dimensional

features on network training, this paper uses the mean and standard deviation of the training set to standardize the feature vectors, i.e.,

$$\widetilde{\xi}_k = \frac{\xi_k - \mu_\xi}{\sigma_\xi + \varepsilon} \tag{83}$$

where μ_ξ and σ_ξ are the mean and standard deviation of the corresponding dimension of the training set, respectively, and ε is the numerical stability term. Thus, the LSTM input sequence of the *k-th* sample can be written as $X_k^L = \{\widetilde{\xi}_{k-L+1}, \widetilde{\xi}_{k-L+2}, \cdots, \widetilde{\xi}_k\}$.

3.4.3. Oracle Transition Probability Matrix Label Definition

Because the simulated mode sequence is known, the oracle transition matrix is built from local switching behavior around each sampling instant. The network is trained to approximate this matrix rather than classify only the dominant mode.

$$\Pi_k^{oracle} = \left[\Pi_{ij,k}^{oracle}\right]_{3\times 3} \tag{84}$$

where $\Pi_{ij,k}^{\text{oracle}}$ represents the empirical probability that the target transitions from model i to model j near time k. The number of mode transitions within the window is defined as

$$N_{ij,k} = \sum_{\updownarrow=k-L+2}^{k} 1\left(m_{\updownarrow-1} = i,\ m_{\updownarrow} = j\right) \tag{85}$$

where $1(\cdot)$ is the indicator function. To prevent transition events that do not occur within the window from yielding zero probability, a smoothing term $\lambda > 0$ is introduced. The elements of the oracle transition probability matrix are then defined as follows:

$$\Pi_{ij,k}^{oracle} = \frac{N_{ij,k} + \lambda p_{ij}^0}{\sum_{j=1}^{3}\left(N_{ij,k} + \lambda p_{ij}^0\right)} \tag{86}$$

where $\Pi^0 = \left[p_{ij}^0\right]_{3\times 3}$ is the prior transition probability matrix of the IMM. Equation (86) shows that each row of the matrix satisfies the non-negativity and normalization constraints, i.e.,

$$\Pi_{ij,k}^{oracle} \geq 0, \sum_{j=1}^{3} \Pi_{ij,k}^{oracle} = 1 \tag{87}$$

During training, the 3×3 oracle transition probability matrix is expanded into a 9-dimensional vector: $y_k = \text{vec}(\Pi_k^{\text{oracle}}) \in \mathbb{R}^9$, which is used as the network supervision label.

3.4.4. LSTM Network Architecture and Output Constraints

A two-layer LSTM maps a 12-dimensional historical feature sequence to a 9-dimensional vector, which is reshaped into a 3×3 transition-matrix output.

$$i_k = \sigma(W_i\xi_k + U_i h_{k-1} + b_i) \tag{88}$$

$$f_k = \sigma\left(W_f\xi_k + U_f h_{k-1} + b_f\right) \tag{89}$$

$$o_k = \sigma(W_o\xi_k + U_o h_{k-1} + b_o) \tag{90}$$

$$\widetilde{c}_k = \tanh(W_c\xi_k + U_c h_{k-1} + b_c) \tag{91}$$

$$c_k = f_k \odot c_{k-1} + i_k \odot \widetilde{c}_k \tag{92}$$

$$h_k = o_k \odot \tanh(c_k) \tag{93}$$

where i_k, f_k, and o_k are the input gate, forget gate, and output gate, respectively; c_k is the cell state; h_k is the hidden state; $\sigma(\cdot)$ is the Sigmoid activation function; $\odot$ represents the Hadamard product; and $W_{(\cdot)}$, $U_{(\cdot)}$, and $b_{(\cdot)}$ are the trainable parameters of the network.

Each LSTM layer has 128 hidden units, followed by normalization and 0.3 dropout. A row-wise Softmax enforces non-negativity and unit row sums.

$$S_k = \begin{bmatrix} s_{11,k} & s_{12,k} & s_{13,k} \\ s_{21,k} & s_{22,k} & s_{23,k} \\ s_{31,k} & s_{32,k} & s_{33,k} \end{bmatrix} \tag{94}$$

To ensure that the model transition probabilities satisfy non-negativity and row normalization constraints, this paper uses row-wise Softmax normalization to obtain the time-varying transition probability matrix:

$$\hat{\Pi}_{ij,k} = \frac{\exp\left(s_{ij,k}\right)}{\sum_{l=1}^{3} \exp(s_{il,k})}, i, j = 1, 2, 3 \tag{95}$$

Therefore, there is

$$\hat{\Pi}_{ij,k} \geq 0, \sum_{j=1}^{3} \hat{\Pi}_{ij,k} = 1 \tag{96}$$

The transition probability matrix of the time-varying model is denoted as

$$\hat{\Pi}_k = \begin{bmatrix} \hat{\Pi}_{11,k} & \hat{\Pi}_{12,k} & \hat{\Pi}_{13,k} \\ \hat{\Pi}_{21,k} & \hat{\Pi}_{22,k} & \hat{\Pi}_{23,k} \\ \hat{\Pi}_{31,k} & \hat{\Pi}_{32,k} & \hat{\Pi}_{33,k} \end{bmatrix} \tag{97}$$

Each row gives the next-step transition probabilities from the current model to the three candidate models.

3.4.5. Loss Functions and Training Settings

This paper uses the Oracle transition probability matrix as the supervision label and KL divergence as the main loss function. For a training sequence of length T, its KL divergence loss is defined as follows:

$$\mathcal{L}_{KL} = \frac{1}{T} \sum_{k=1}^{T} \sum_{i=1}^{3} \sum_{j=1}^{3} \Pi_{ij,k}^{oracle} \log \frac{\Pi_{ij,k}^{oracle} + \varepsilon}{\hat{\Pi}_{ij,k} + \varepsilon} \tag{98}$$

Here, $\varepsilon = 10^{-7}$ is the numerically stable term. To avoid the network output being too sharp, an entropy regularization term is added in this paper.

$$\mathcal{H} = -\frac{1}{T} \sum_{k=1}^{T} \sum_{i=1}^{3} \sum_{j=1}^{3} \hat{\Pi}_{ij,k} \log(\hat{\Pi}_{ij,k} + \varepsilon) \tag{99}$$

The total loss is defined as

$$\mathcal{L} = \alpha_{KL} \mathcal{L}_{KL} - \beta_H \mathcal{H} + \lambda_{L2} \parallel \Theta \parallel_2^2 \tag{100}$$

where Θ represents the trainable parameters of the network, $\alpha_{\mathrm{KL}} = 1.0$, $\beta_{\mathrm{H}} = 0.01$, and $\lambda_{\mathrm{L2}} = 10^{-4}$.

Training uses Adam, 50 epochs, a mini-batch size of 32, learning-rate decay every 15 epochs, and a gradient threshold of 1.0.

3.4.6. LSTM-IMM Online Application Method

The LSTM transition matrix is generated from the latest feature history and used in the next IMM interaction step. Therefore, the model prediction probability in Equation (61) becomes

$$c_{j,k} = \sum_{i=1}^{3} p_{ij,k-1}\mu_{i,k-1} \tag{101}$$

where $p_{ij,k-1}$ is the element in the *i-th* row and j-th column of matrix $\hat{\Pi}_{k-1}$, representing the probability that the target mode transitions from model i at time t_{k-1} to model j at time t_k.

After each measurement update, the latest feature window is passed to the LSTM to generate the transition matrix used in the next IMM interaction and prediction step.

The default online history is $L_w = 50$ samples, or 1.0 s at 50 Hz. Section 5.1.3 evaluates this choice.

3.4.7. Computational Burden and Real-Time Implementation Considerations

The online cost of LSTM-IMM is dominated by transition-matrix inference. The implemented network uses 12 input features, two 128-unit BiLSTM layers, and a 9-element output for the 3×3 transition matrix, yielding about 5.41×10^5 trainable parameters and about 5.39×10^5 multiply–accumulate operations per sequence element.

For real-time interception, training is offline and only forward inference is executed online. A fixed historical window keeps the cost bounded; at the 50 Hz measurement rate, 20–50 samples require about 0.54–1.35 GMAC/s, which should be profiled on the target processor.

Because the LSTM output is only a 3×3 matrix, subsequent prediction and guidance remain low-dimensional analytical operations.

3.5. Target Future State Prediction Method

Future target states are predicted to generate intercept points and estimate remaining missile flight times.

Let the current time be t_k, the prediction step size be T_p, and the number of prediction steps be N_p; then, the prediction time at the *s-th* step is

$$t_{k+s} = t_k + sT_p, s = 1, 2, \cdots, N_p \tag{102}$$

Here, the prediction step and window length follow the simulation settings.

For the *j-th* model, its *s-th* step state prediction is:

$$\hat{x}_{j,k+s|k} = f_j\left(\hat{x}_{j,k+s-1|k}\right) \tag{103}$$

where the initial value is

$$\hat{x}_{j,k|k} = \hat{x}_{j,k} \tag{104}$$

Model probability prediction can be written as

$$\mu_{k+s|k} = \mu_{k+s-1|k}\Pi_{k+s-1} \tag{105}$$

where

$$\mu_{k+s|k} = \begin{bmatrix} \mu_{1,k+s|k} & \mu_{2,k+s|k} & \mu_{3,k+s|k} \end{bmatrix} \tag{106}$$

This represents the model probability vector at the *s-th* step in the future. If future measurements are unknown, the transition probability matrix generated by the current LSTM can be kept unchanged within the prediction window, i.e.,

$$\Pi_{k+s-1} = \Pi_k, s = 1, 2, \cdots, N_p \tag{107}$$

After obtaining the predicted states and probabilities of each model, the target's comprehensive predicted state at step s is:

$$\hat{x}_{T,k+s|k} = \sum_{j=1}^{M} \mu_{j,k+s|} \hat{x}_{j,k+s|k} \tag{108}$$

The predicted target location and prediction velocity are respectively taken as.

$$\hat{r}_{T,k+s|k} = \begin{bmatrix} \hat{x}_{T,k+s|k} \\ \hat{y}_{T,k+s|k} \\ \hat{z}_{T,k+s|k} \end{bmatrix} \tag{109}$$

$$\hat{v}_{T,k+s|k} = \begin{bmatrix} \hat{v}_{x\mathrm{T,k+s|}k} \\ \hat{v}_{y\mathrm{T,k+s|}k} \\ \hat{v}_{z\mathrm{T,k+s|}k} \end{bmatrix} \tag{110}$$

The guidance reference uses an intermediate prediction step rather than necessarily the end of the prediction window.

$$1 \leq N_u \leq N_p \tag{111}$$

The target prediction state used in the guidance is:

$$\hat{x}_T^{\,p}(t_k) = \hat{x}_{T,k+N_u|k} \tag{112}$$

The corresponding predicted position and velocity are

$$\hat{r}_T^{\,p}(t_k) = \hat{r}_{T,k+N_u|k} \tag{113}$$

$$\hat{v}_T^{\,p}(t_k) = \hat{v}_{T,k+N_u|k} \tag{114}$$

where $\hat{r}_T^{\,p}(t_k)$ can be used as the target prediction position input in subsequent multi-missile cooperative guidance.

To avoid the adverse effects of low-confidence prediction results on the guidance system, the prediction confidence is defined as the maximum value of the current model probability:

$$C_k = \max_{1 \leq j \leq M} \mu_{j,k} \tag{115}$$

Let the prediction confidence threshold be $C_{\min}$, when

$$C_k \geq C_{\min} \tag{116}$$

If the prediction confidence or physical consistency check fails, the current filtered state or a simple extrapolation is used instead.

Consider setting the prediction window length to $N_p = 120$, using the prediction result at step $N_u = 100$, and setting the prediction confidence threshold to $C_{\min} = 0.45$. This setting allows the guidance system to obtain short-term forward-looking target states while avoiding error accumulation caused by excessively long prediction times.

3.6. LSTM-IMM Tracking Prediction Algorithm Flow

Algorithm 1 summarizes historical feature construction, LSTM transition-matrix generation, IMM filtering, and recursive prediction.

Algorithm 1 LSTM-IMM Target Tracking and Prediction Algorithm

Input:

Target measurement sequence $z_{T,k}$; historical sequence length L; model set $\mathcal{M} = \{CV, CA, CT\}$; prediction step size T_p prediction horizon N_p; guidance-used; prediction step N_u; prediction confidence threshold $C_{\min}$; initial model-conditioned state estimate $\hat{x}_{j,0}$; initial covariance matrix $P_{j,0}$; initial model probability $\mu_{j,0}$, where $j = 1, 2, 3$.

Output:

Current fused target state estimate $\hat{x}_{T,k}$; predicted target state **for** guidance $\hat{x}_T^p(t_k)$; prediction validity flag $Flag_k$.

Procedure:

1: **Initialization**

Given $\hat{x}_{j,0}, P_{j,0}, \mu_{j,0}$, $j = 1, 2, 3$,

ensure $\sum_{j=1}^{3} \mu_{j,0} = 1$;

set the initial transition probability matrix as $\hat{\Pi}_0 = \Pi^0$.

2 : **for** $k = 1, 2, \cdots$ **do**

3 : **IMM model interaction using the previous transition probability matrix:**

$\hat{\Pi}_{k-1} = [p_{ij,k-1}]_{3\times 3}$;

for $j = 1, 2, 3$ **do**

$c_{j,k} = \sum_{i=1}^{3} p_{ij,k-1}\mu_{i,k-1}$;

$\mu_{i|j,k-1} = \frac{p_{ij,k-1}\mu_{i,k-1}}{c_{j,k}}$;

$\hat{x}_{0j,k-1} = \sum_{i=1}^{3} \mu_{i|j,k-1}\hat{x}_{i,k-1}$;

$P_{0j,k-1}$

$= \sum_{i=1}^{3} \mu_{i|j,k-1}\left[P_{i,k-1} + \left(\hat{x}_{i,k-1} - \hat{x}_{0j,k-1}\right)\left(\hat{x}_{i,k-1} - \hat{x}_{0j,k-1}\right)^T\right]$;

end for

4 : **Model-conditioned filtering:**

for $j = 1, 2, 3$ **do**

Perform one-step state prediction under model j :

$\hat{x}_{j,k|k-1} = f_j\left(\hat{x}_{0j,k-1}\right)$;

$P_{j,k|k-1} = F_{j,k}P_{0j,k-1}F_{j,k}^T + Q_{j,k-1}$;

Update with the current measurement $z_{T,k}$ to obtain $\hat{x}_{j,k}, P_{j,k}, \Lambda_{j,k}$;

end for

5 : **Update model probabilities:**

for $j = 1, 2, 3$ **do**

$\mu_{j,k} = \frac{\Lambda_{j,k}c_{j,k}}{\sum_{l=1}^{3} \Lambda_{l,k}c_{l,k}}$;

end for

ensure

$$\sum_{j=1}^{3} \mu_{j,k} = 1$$

6 : **Fuse model-conditioned estimates:**

$\hat{x}_{T,k} = \sum_{j=1}^{3} \mu_{j,k}\hat{x}_{j,k}$;

$P_{T,k} = \sum_{j=1}^{3} \mu_{j,k}\left[P_{j,k} + \left(\hat{x}_{j,k} - \hat{x}_{T,k}\right)\left(\hat{x}_{j,k} - \hat{x}_{T,k}\right)^T\right]$;

Algorithm 1 *Cont.*

7 : **Construct historical feature sequence:**

Construct the 12-dimensional LSTM input feature using the latest measurement $z_{T,k}$ and its first-order difference: e:

$$\xi_k = \left[z_{T,k}^T, \Delta z_{T,k}^T \right]^T \in \mathbb{R}^{12};$$

where

$$\Delta z_{T,k} = z_{T,k} - z_{T,k-1}.$$

For angular components, perform angle wrapping:

$$\Delta\theta_{T,k}^m = \text{wrap}\left(\theta_{T,k}^m - \theta_{T,k-1}^m\right);$$

$$\Delta\psi_{T,k}^m = \text{wrap}\left(\psi_{T,k}^m - \psi_{T,k-1}^m\right).$$

Standardize the feature vector:

$$\widetilde{\xi}_k = \frac{\xi_k - \mu_\xi}{\sigma_\xi + \varepsilon};$$

construct the historical feature sequence:

$$X_k^L = \left\{ \widetilde{\xi}_{k-L+1}, \widetilde{\xi}_{k-L+2}, \cdots, \widetilde{\xi}_k \right\}.$$

If $k < L$, use zero padding, repeated initial feature, or the prior matrix Π^0 for initialization.

8 : **Generate time-varying model transition probability matrix:**

Feed X_k^L into the LSTM network to obtain the unnormalized output matrix S_k :

$$S_k = \begin{bmatrix} s_{11,k} & s_{12,k} & s_{13,k} \\ s_{21,k} & s_{22,k} & s_{23,k} \\ s_{31,k} & s_{32,k} & s_{33,k} \end{bmatrix};$$

compute the row-wise Softmax normalization:

$$p_{ij,k} = \frac{\exp\left(s_{ij,k}\right)}{\sum_{l=1}^3 \exp\left(s_{il,k}\right)}, i, j = 1, 2, 3;$$

obtain the time-varying transition probability matrix:

$$\hat{\Pi}_k = [p_{ij,k}]_{3\times 3};$$

where

$$p_{ij,k} \geq 0, \sum_{j=1}^{3} p_{ij,k} = 1.$$

9 : **Predict future target states:**

Set $\hat{x}_{j,k|k} = \hat{x}_{j,k}, \mu_{k|k} = [\mu_{1,k}, \mu_{2,k}, \mu_{3,k}]$;

for $s = 1, 2, \cdots, N_p$ **do**

for $j = 1, 2, 3$ **do**

$\hat{x}_{j,k+s|k} = f_j\left(\hat{x}_{j,k+s-1|k}\right)$;

end for

$\mu_{k+s|k} = \mu_{k+s-1|k}\Pi_k$;

$\hat{x}_{T,k+s|k} = \sum_{j=1}^3 \mu_{j,k+s|k}\hat{x}_{j,k+s|k}$;

end for

10 : **Select the predicted state for guidance:**

$\hat{x}_T^p(t_k) = \hat{x}_{T,k+N_u|k}$;

$\hat{r}_T^p(t_k) = \left[\hat{x}_{T,k+N_u|k}, \hat{y}_{T,k+N_u|k}, \hat{z}_{T,k+N_u|k}\right]^T$;

$\hat{v}_T^p(t_k) = \left[\hat{v}_{xT,k+N_u|k}, \hat{v}_{yT,k+N_u|k}, \hat{v}_{zT,k+N_u|k}\right]^T$;

Algorithm 1 *Cont.*

11 : **Evaluate prediction validity:**

$C_k = \max_{1 \le j \le 3} \mu_{j,k}$;

if $C_k \ge C_{\min}$ and $\hat{x}_T^p(t_k)$ satisfies physical constraints **then**

$Flag_k = 1$;

output $\hat{x}_T^p(t_k)$ as the guidance reference state;

else

$Flag_k = 0$;

use the current filtered state or a simple extrapolated state as the substitute target state;

end if

12 : **Save the current transition probability matrix**

Save $\hat{\Pi}_k$ for the next filtering cycle :

$\hat{\Pi}_k \to \hat{\Pi}_{(k+1)-1}$;

end for

The validity flag determines whether LSTM-IMM prediction or the fallback filtered/extrapolated state is used by guidance.

In the simulations, the prediction horizon is set to $N_p = 120$, the $N_u = 100$-th predicted state is used by the guidance module, and the prediction confidence threshold is set as $C_{\min} = 0.45$.

The resulting prediction state is used as the target reference for the cooperative guidance law.

4. A Multi-Missile Time Coordination Guidance Method Based on Active Curvature Planning

The prediction module supplies future target states, but prediction alone does not synchronize missile arrivals. This section converts prediction information into fixed-speed cooperative guidance commands.

The idea of guidance is to regulate remaining flight time by changing path curvature instead of missile speed. Each missile receives a normal-acceleration command that adjusts path length while respecting overload constraints.

The method first generates a predicted intercept point for each missile, estimates feasible time-to-go intervals, and then allocates curvature commands according to common-arrival-time feasibility.

4.1. Predictive Interception Point Generation Method

The LSTM-IMM output provides the filtered and predicted target states used to generate each missile's predicted intercept point.

Let the current position and velocity of the *i-th* missile be $r_i(t)$ and $v_i(t)$, respectively, and its velocity magnitude satisfies a fixed velocity constraint:

$$\| v_i(t) \| = V_{i0} \tag{117}$$

Based on the target prediction results in Section 3, the sequence of target positions within the future prediction window can be represented as:

$$\left\{ \hat{r}_T\left(t + sT_p \mid t\right) \right\}_{s=1}^{N_p} \tag{118}$$

where T_p is the prediction step size, and N_p is the prediction window length.

If the target is intercepted at the *s-th* prediction time in the future, the corresponding prediction time is:

$$\tau_s = sT_p. \tag{119}$$

The target's position at the predicted time is:

$$\hat{r}_{T,s}(t) = \hat{r}_T\left(t + sT_p \mid t\right). \tag{120}$$

For the *i-th* missile, if it flies to the predicted position along the approximate shortest path, the required time can be approximated as:

$$\overline{\tau}_{i,s}(t) = \frac{\parallel \hat{r}_{T,s}(t) - r_i(t) \parallel}{V_{i0}}. \tag{121}$$

To ensure that the missile's arrival time is as close as possible to the target's predicted time, the time matching error is defined as:

$$E_{i,s}(t) = \mid \overline{\tau}_{i,s}(t) - sT_p \mid . \tag{122}$$

The number of steps for predicting and intercepting the *i-th* missile is selected as follows:

$$s_i^\star(t) = \arg\min_{1 \le s \le N_p} E_{i,s}(t). \tag{123}$$

Therefore, the predicted intercept point corresponding to the *i-th* missile is defined as:

$$p_i(t) = \hat{r}_T\left(t + s_i^\star T_p \mid t\right). \tag{124}$$

From this, the geometric distance from the missile to the predicted interception point can be obtained:

$$D_i(t) = \parallel p_i(t) - r_i(t) \parallel . \tag{125}$$

Under a fixed speed, the corresponding shortest remaining flight time is approximately:

$$\tau_{i,\min}(t) = \frac{D_i(t)}{V_{i0}}. \tag{126}$$

Equations (118)–(126) search the predicted trajectory and select the intercept point with the smallest time-matching error.

For lower online cost, a constant-velocity analytical approximation can also be used.

$$\rho_i^{ref}(t) = \hat{r}_T^{ref}(t) - r_i(t). \tag{127}$$

If the target moves at approximately a constant velocity over a short period of time, then the predicted interception time $\tau_i^0(t)$ satisfies:

$$\parallel \rho_i^{ref}(t) + \hat{v}_T^{ref}(t)\tau_i^0 \parallel = V_{i0}\tau_i^0. \tag{128}$$

Squaring both sides of Equation (128), we obtain a quadratic equation:

$$A_i\left(\tau_i^0\right)^2 + B_i\tau_i^0 + C_i = 0. \tag{129}$$

where

$$\begin{aligned} A_i &= \| \hat{v}_T^{ref}(t) \|^2 - V_{i0}^2, \\ B_i &= 2\left(\rho_i^{ref}(t)\right)^T \hat{v}_T^{ref}(t), \\ C_i &= \| \rho_i^{ref}(t) \|^2 . \end{aligned} \tag{130}$$

If Equation (129) has a positive real root, then the smallest positive real root is taken as the predicted interception time:

$$\tau_i^0(t) = \min\left\{ \tau > 0 \mid A_i\tau^2 + B_i\tau + C_i = 0 \right\}. \tag{131}$$

The corresponding predicted interception point is:

$$p_i(t) = \hat{r}_T^{ref}(t) + \hat{v}_T^{ref}(t)\tau_i^0(t). \tag{132}$$

If no positive root exists, the method falls back to the predicted-trajectory search in Equations (118)–(124).

4.2. The Relationship Between Curvature and Path Length

Under fixed speed, arrival time is adjusted by changing path length. The remaining flight time is

$$\tau_i(t) = \frac{L_i(t)}{V_{i0}}. \tag{133}$$

Since the straight-line distance between two points is the shortest, the actual flight path length satisfies:

$$L_i(t) \geq D_i(t). \tag{134}$$

When a missile flies towards the predicted interception point using an approximately constant curvature trajectory, its local trajectory can be approximated as a circular arc. Let the curvature of the arc be $\kappa_i(t)$, and the corresponding radius of curvature be:

$$\kappa_i(t) = \frac{1}{R_{c,i}(t)} \tag{135}$$

Let $\phi_i(t)$ be the central angle corresponding to the arc. Then the arc length and the chord length are respectively:

$$L_i(t) = R_{c,i}(t)\phi_i(t) = \frac{\phi_i(t)}{\kappa_i(t)} \tag{136}$$

$$D_i(t) = 2R_{c,i}(t)\sin\frac{\phi_i(t)}{2} = \frac{2}{\kappa_i(t)}\sin\frac{\phi_i(t)}{2} \tag{137}$$

From Equations (136) and (137), we obtain:

$$\frac{L_i(t)}{D_i(t)} = \frac{\phi_i(t)}{2sin(\phi_i(t)/2)} \tag{138}$$

Equation (138) shows that as long as the trajectory has non-zero curvature and the endpoints remain unchanged, the actual flight path length must be greater than the straight-line distance between the two points.

When $\phi_i(t)$ is small, using the small-angle approximation, we can obtain:

The approximation in Equation (139) is obtained by expanding the sine term in Equation (138) to third order and retaining the leading non-zero curvature contribution.

This is a local small-curvature approximation used for online correction, not an assumption that the full trajectory is globally circular.

$$L_i(t) \approx D_i(t) + \frac{\kappa_i^2(t) D_i^3(t)}{24} \tag{139}$$

Therefore, the path length increment caused by curvature is approximately:

$$\Delta L_i(t) = L_i(t) - D_i(t) \approx \frac{\kappa_i^2(t) D_i^3(t)}{24} \tag{140}$$

Combining this with Equation (140), the curvature-induced increase in remaining flight time is

$$\Delta \tau_i(t) = \frac{\Delta L_i(t)}{V_{i0}} \approx \frac{\kappa_i^2(t) D_i^3(t)}{24 V_{i0}} \tag{141}$$

Equation (141) establishes an approximate mapping relationship between curvature and remaining flight time. Here, the velocity term in the denominator is the constant flight velocity of the *i-th* missile. With a fixed missile velocity, increasing the trajectory curvature increases the path length, thereby extending the remaining flight time.

Equation (140) can also be used to inversely solve for the desired curvature required to satisfy a given path increment $\Delta L_i(t)$:

$$\kappa_{i,d}(t) = \sqrt{\frac{24 \Delta L_i(t)}{D_i^3(t)}} \tag{142}$$

The corresponding expected normal acceleration amplitude is

$$a_{i,d}(t) = V_{i0}^2 \kappa_{i,d}(t) \tag{143}$$

Equations (139)–(143) are local constant-curvature and small-angle approximations. In guidance, curvature is updated online from the predicted target state, missile position, and remaining time error; the equations generate corrections rather than a fixed circular arc.

4.3. Geometric Mechanism of Lateral Curvature Adjustment Increasing Path Length

The normal acceleration is decomposed into a hit guidance term and a lateral curvature term. The first reduces line-of-sight error, and the second adjusts path length.

The missile's unit velocity vector is defined as:

$$\hat{v}_i(t) = \frac{v_i(t)}{\| v_i(t) \|}. \tag{144}$$

Define the unit vector of the missile pointing towards the predicted intercept point as:

$$\hat{q}_i(t) = \frac{p_i(t) - r_i(t)}{\| p_i(t) - r_i(t) \|}. \tag{145}$$

The velocity-normal plane projection matrix is defined as follows:

$$P_i(t) = I_3 - \hat{v}_i(t) \hat{v}_i^T(t). \tag{146}$$

Projecting the predicted interception direction onto the velocity-normal plane yields the hit correction direction:

$$n_{gi}(t) = P_i(t) \hat{q}_i(t). \tag{147}$$

When $\| n_{gi}(t) \| > \varepsilon_n$, the normalization hit correction direction is:

$$\hat{n}_{gi}(t) = \frac{n_{gi}(t)}{\| n_{gi}(t) \|}. \tag{148}$$

In the velocity-normal plane, a direction orthogonal to $\hat{n}_{gi}(t)$ is chosen as the lateral curvature direction:

$$\hat{n}_{ci}(t) = \frac{\hat{v}_i(t) \times \hat{n}_{gi}(t)}{\| \hat{v}_i(t) \times \hat{n}_{gi}(t) \|}. \tag{149}$$

From Equation (149), we obtain:

$$\hat{n}_{ci}^T(t)\hat{v}_i(t) = 0, \hat{n}_{ci}^T(t)\hat{n}_{gi}(t) = 0. \tag{150}$$

Equation (150) shows that the lateral curvature direction neither changes the missile's velocity nor is orthogonal to the main hit correction direction. Therefore, it can reduce interference with the main hit guidance direction in a first-order sense.

The following further explains why the lateral curvature command can increase the path length. The actual flight path length of the missile is:

$$L_i = \int_t^{t_f} \|\hat{v}_i(s)\| \, ds = \int_t^{t_f} V_{i0} ds. \tag{151}$$

The effective advance distance along the predicted interception direction can be expressed as:

$$D_i \approx \int_t^{t_f} v_i^T(s)\hat{q}_i(s) ds. \tag{152}$$

When the missile has a non-zero lateral motion component, we have:

$$v_i^T(s)\hat{q}_i(s) < \| v_i(s) \| = V_{i0}. \tag{153}$$

Thus, lateral curvature creates a deviation-return path whose length exceeds the straight-line path.

From a curve geometry perspective, for any trajectory curve Γ_i connecting $r_i(t)$ and $p_i(t)$, its length satisfies:

$$L(\Gamma_i) = \int_{\Gamma_i} ds \geq \| p_i(t) - r_i(t) \| = D_i(t). \tag{154}$$

When the trajectory is not a straight line between two points, we have:

$$L(\Gamma_i) > D_i(t). \tag{155}$$

Therefore, the essence of lateral curvature adjustment is not to change the final aiming point, but to introduce controllable lateral curvature while maintaining the main direction of hit guidance, thereby increasing the actual flight path length.

4.4. Expected Public Arrival Time and Feasibility Assessment

The common expected remaining time must be no smaller than the largest shortest-time estimate among the missiles.

On the other hand, the path length that a missile can increase through curvature adjustment is limited by a maximum overload constraint. Let the maximum lateral acceleration

available for curvature coordination of the *i-th* missile be $a_{ci,\max}(t)$; then, the corresponding maximum available curvature is approximately:

$$\kappa_{ci,\max}(t) = \frac{a_{ci,\max}(t)}{V_{i0}^2}. \tag{156}$$

From Equation (140), the maximum time delay that the *i-th* missile can produce can be approximated as:

$$\Delta\tau_{i,\max}(t) \approx \frac{\kappa_{ci,\max}^2(t) D_i^3(t)}{24 V_{i0}}. \tag{157}$$

Therefore, the achievable remaining flight time interval for the *i-th* missile is:

$$\mathcal{I}_i(t) = [\tau_{i,\min}(t),\ \tau_{i,\min}(t) + \Delta\tau_{i,\max}(t)]. \tag{158}$$

Define the lower and upper bounds of the common feasible interval as follows:

$$\tau_{d,\min}(t) = \max_{1\le i\le N} \tau_{i,\min}(t), \tau_{d,\max}(t) = \min_{1\le i\le N} [\tau_{i,\min}(t) + \Delta\tau_{i,\max}(t)]. \tag{159}$$

When the following conditions are met:

$$\tau_{d,\min}(t) \le \tau_{d,\max}(t) \tag{160}$$

This indicates that there is a common intersection in the achievable time intervals of all missiles, and at this time there is a common expected remaining flight time.

To allow for target maneuvers, prediction errors, and overload variations, the expected remaining flight time is set using the saturation operator $sat(x, a, b) = \min(\max(x, a), b)$, which limits x to the interval $[a, b]$:

$$\tau_d(t) = \mathrm{sat}(\tau_{d,\min}(t) + \Delta T_s,\ \tau_{d,\min}(t),\ \tau_{d,\max}(t)), \tag{161}$$

where $\Delta T_s \ge 0$ represents the time co-occurrence margin, and $\mathrm{sat}(\cdot)$ is the saturation function.

The corresponding expected hit time is:

$$t_d(t) = t + \tau_d(t). \tag{162}$$

To avoid drastic changes in guidance commands due to sudden changes in the common expected hit time, a first-order smoothing is applied:

$$t_d^f(t) = (1 - \lambda_t) t_d^f(t - \Delta t) + \lambda_t t_d(t), \tag{163}$$

where $\lambda_t \in (0, 1]$ is the smoothing coefficient, and Δt is the guidance update period. Unless otherwise specified, the smoothed $t_d^f(t)$ is used as the common expected hit time.

4.5. Minimum-Variance Allocation Strategies When Public Expected Time Is Unfeasible

When Equation (161) does not hold, that is:

$$\tau_{d,\min}(t) > \tau_{d,\max}(t), \tag{164}$$

If feasible intervals do not intersect, exact simultaneity is infeasible under the current geometry, speed, and overload constraints.

Therefore, this paper adopts a minimum time distribution strategy in the infeasible case. Let:

$$\mathcal{I}_i(t) = [\tau_{i,L}(t), \tau_{i,U}(t)], \tag{165}$$

where

$$\begin{aligned} \tau_{i,L}(t) &= \tau_{i,\min}(t), \\ \tau_{i,U}(t) &= \tau_{i,\min}(t) + \Delta\tau_{i,\max}(t). \end{aligned} \tag{166}$$

Introducing a common reference remaining time $\tau_d^\star(t)$, it is determined through the following optimization problem:

The interval-distance operator $dist(x, I)$ denotes the shortest scalar distance from x to interval I.

$$\tau_d^\star(t) = \arg\min_{\tau} \sum_{i=1}^{N} w_i(t) \mathrm{dist}^2(\tau, \mathcal{I}_i(t)), \tag{167}$$

where $w_i(t) > 0$ is the weight, and $\mathrm{dist}(\tau, \mathcal{I}_i)$ represents the distance from point τ to interval $\mathcal{I}_i$:

$$\mathrm{dist}(\tau, \mathcal{I}_i) = \begin{cases} \tau_{i,L}(t) - \tau, & \tau < \tau_{i,L}(t), \\ 0, & \tau_{i,L}(t) \le \tau \le \tau_{i,U}(t), \\ \tau - \tau_{i,U}(t), & \tau > \tau_{i,U}(t). \end{cases} \tag{168}$$

The weights can be selected based on the target prediction confidence, the remaining flight time, and the missile's remaining overload capacity. For example:

$$w_i(t) = \frac{C_k}{\tau_{i,\min}(t) + \varepsilon_\tau}, \tag{169}$$

where C_k is the target prediction confidence level given in Section 3, and $\varepsilon_\tau > 0$ is a small positive number to prevent the denominator from being zero.

After obtaining the common reference remaining time $\tau_d^\star(t)$, it is no longer required that all missiles strictly use the same remaining time, but rather it is projected onto the feasible interval of each missile itself:

The projection operator $proj_{I(x)} = \min(\max(x, l), u)$, with $I = [l, u]$, maps x onto the feasible interval.

$$\tau_{i,d}(t) = \mathrm{proj}_{\mathcal{I}_i(t)}[\tau_d^\star(t)]. \tag{170}$$

That is:

$$\tau_{i,d}(t) = \mathrm{sat}(\tau_d^\star(t), \tau_{i,L}(t), \tau_{i,U}(t)). \tag{171}$$

Generally, the expected remaining flight time can be expressed as:

$$\tau_{i,d}(t) = \begin{cases} \tau_d(t), & \bigcap_{i=1}^{N} \mathcal{I}_i(t) \neq \varnothing, \\ \mathrm{proj}_{\mathcal{I}_i(t)}\left[\tau_d^\star(t)\right], & \bigcap_{i=1}^{N} \mathcal{I}_i(t) = \varnothing. \end{cases} \tag{172}$$

The expected path length for the *i-th* missile is:

$$L_{i,d}(t) = V_{i0}\tau_{i,d}(t). \tag{173}$$

The path length that needs to be increased through curvature planning is:

$$\Delta L_{i,d}(t) = L_{i,d}(t) - D_i(t). \tag{174}$$

Considering numerical errors and saturation in infeasible cases, the non-negative path length increment is defined as follows:

$$\Delta L_i^+(t) = \max(0, \Delta L_{i,d}(t)). \tag{175}$$

From Equation (141), the desired curvature command can be obtained:

$$\kappa_{i,d}(t) = \sqrt{\frac{24\Delta L_i^+(t)}{D_i^3(t) + \varepsilon_D}}. \tag{176}$$

The corresponding expected curvature acceleration amplitude is:

$$a_{\kappa i,d}(t) = V_{i0}^2 \kappa_{i,d}(t). \tag{177}$$

4.6. Design of a Curvature-Cooperative Guidance Law Prioritizing Hit Probability

This section designs a multi-missile curvature coordinated guidance law. To avoid conflicts between curvature adjustment and hit guidance, this paper adopts a hierarchical structure of "hit guidance main channel + lateral curvature coordinated channel" and prioritizes hit guidance under overload-limited conditions.

4.6.1. Basic Hit Guidance Parameters

The relative position between the *i-th* missile and the target reference point is:

$$\rho_i(t) = r_T^{ref}(t) - r_i(t), R_i(t) = \| \rho_i(t) \| . \tag{178}$$

where $r_T^{ref}(t)$ is the target reference position used for guidance, which can be obtained by fusing the predicted state from Section 3 or the current filtered state.

The relative velocity is:

$$\dot{\rho}_i(t) = v_T^{ref}(t) - v_i(t). \tag{179}$$

The line-of-sight angular velocity and closing velocity are respectively:

$$\begin{aligned} \omega_{Li}(t) &= \frac{\rho_i(t) \times \dot{\rho}_i(t)}{R_i^2(t)}, \\ V_{ci}(t) &= -\frac{\rho_i^T(t)\dot{\rho}_i(t)}{R_i(t)}. \end{aligned} \tag{180}$$

The three-dimensional proportional guidance acceleration is defined as:

$$a_{PN,i}(t) = N_i V_{ci}(t)[\omega_{Li}(t) \times \hat{v}_i(t)], \tag{181}$$

where $N_i > 0$ is the proportional guidance coefficient.

To ensure that the guidance acceleration does not change the missile's velocity, it is projected onto the velocity-normal plane:

$$a_{gi}(t) = P_i(t) a_{PN,i}(t). \tag{182}$$

where $a_{gi}(t)$ is the acceleration of the hit guidance main channel.

4.6.2. Lateral Curvature Interaction Term

According to Equations (176) and (177), the acceleration amplitude required for curvature coordination is $a_{\kappa i,d}(t)$. To minimize interference between the curvature coordination term and the hit guidance term, the lateral curvature direction is preferably chosen as the velocity-normal plane direction orthogonal to the hit guidance direction.

When $\| a_{gi}(t) \| > \varepsilon_a$, the unit direction for hit guidance is defined as:

$$\hat{n}_{gi}(t) = \frac{a_{gi}(t)}{\| a_{gi}(t) \|}. \tag{183}$$

The lateral curvature direction is taken as:

$$\hat{n}_{ci}(t) = \frac{\hat{v}_i(t) \times \hat{n}_{gi}(t)}{\| \hat{v}_i(t) \times \hat{n}_{gi}(t) \|}. \tag{184}$$

At this point:

$$\hat{n}_{ci}^T(t)\hat{v}_i(t) = 0, \hat{n}_{ci}^T(t)\hat{n}_{gi}(t) = 0. \tag{185}$$

When the hit correction is small, the lateral curvature direction is used; otherwise, the previous lateral direction is held, or the coordination term is set to zero.

The unlimited form of the curvature coordination acceleration is:

$$a_{ci}^{raw}(t) = a_{\kappa i,d}(t)\hat{n}_{ci}(t). \tag{186}$$

To adjust the curvature coordination strength based on time error, the current predicted remaining flight time is defined as follows:

$$\hat{\tau}_i(t) = \frac{\hat{L}_i(t)}{V_{i0}}, \tag{187}$$

where $\hat{L}_i(t)$ is the estimated remaining path length under the current plan. In engineering implementation, it can be approximated by $D_i(t) + \Delta L_i^+(t)$.

The time coordination error is defined as:

$$e_{\tau i}(t) = \tau_{i,d}(t) - \hat{\tau}_i(t). \tag{188}$$

A positive time error increases curvature to lengthen the path; otherwise, curvature correction is reduced to prioritize interception.

Curvature co-activation coefficients are defined as:

$$\sigma_i(t) = \text{sat}(k_t \max(0, e_{\tau i}(t)), 0, 1), \tag{189}$$

where $k_t > 0$ represents the time error adjustment gain.

Therefore, the curvature coordination term is:

$$a_{ci}(t) = \sigma_i(t)a_{ci}^{raw}(t). \tag{190}$$

4.6.3. Hit-Priority Allocation Under Overload Saturation

To avoid the curvature coordination term using overload needed for hit guidance, this paper adopts a hit-priority overload allocation strategy. First, the hit guidance term is limited:

$$a_{gi}^{\lim}(t) = \begin{cases} a_{gi}(t), & \| a_{gi}(t) \| \leq a_{i,\max}, \\ a_{i,\max}\frac{a_{gi}(t)}{\|a_{gi}(t)\|}, & \| a_{gi}(t) \| > a_{i,\max}. \end{cases} \tag{191}$$

Equation (191) shows that when the overload required for hit guidance exceeds the missile's maximum overload, all available overload is prioritized for hit guidance, and the curvature coordination term should be automatically turned off.

Since the lateral curvature direction is designed to be approximately orthogonal to the hit guidance direction, the remaining overload available for curvature coordination is approximately:

$$a_{rem,i}(t) = \sqrt{\max\left(0, a_{i,\max}^2 - \| a_{gi}^{\lim}(t) \|^2\right)}. \tag{192}$$

Limit the curvature covariance term:

$$a_{ci}^{\lim}(t) = \begin{cases} a_{ci}(t), & \| a_{ci}(t) \| \leq a_{rem,i}(t), \\ a_{rem,i}(t)\frac{a_{ci}(t)}{\|a_{ci}(t)\|}, & \| a_{ci}(t) \| > a_{rem,i}(t). \end{cases} \tag{193}$$

The final guidance-unswitched command during the mid-course guidance phase is:

$$a_i^{cmd}(t) = a_{gi}^{\lim}(t) + a_{ci}^{\lim}(t). \tag{194}$$

Let $L_a\{\cdot\}$ denote the overload limiting operator that clips an acceleration vector's norm to the allowable overload while preserving its direction.

To strictly ensure that the acceleration lies within the velocity-normal plane and satisfies the maximum overload constraint, the final guidance acceleration is defined as:

$$a_i(t) = \mathcal{L}_a\left\{P_i(t)a_i^{cmd}(t)\right\}. \tag{195}$$

The overload limiting operator $\mathcal{L}_a\{\cdot\}$ is defined as follows:

$$\mathcal{L}_a\{u\} = \begin{cases} u, & \| u \| \leq a_{i,\max}, \\ a_{i,\max}\frac{u}{\|u\|}, & \| u \| > a_{i,\max}. \end{cases} \tag{196}$$

This design enforces the following priorities:

1. When the target maneuverability is strong and the hit guidance requirement is high, overload is used first to ensure a hit;
2. When the hit guidance overload requirement is low, the remaining overload is used for time coordination;
3. When the overload is saturated, the curvature coordination term is automatically compressed or turned off, so as not to compete with the hit guidance for control capability.

4.7. End-Stage Switching and Reference-State Fusion Strategy

Curvature planning is used mainly in the mid-course phase. Near the target, terminal accuracy takes priority, so the coordination term is gradually reduced.

Let the terminal-switching distance be $R_{ter} > 0$, and the fully effective coordinated guidance distance be $R_{coop} > R_{ter}$. Define the range-dependent coordinated weighting function:

$$\chi_i(t) = \text{sat}\left(\frac{R_i(t) - R_{ter}}{R_{coop} - R_{ter}}, 0, 1\right). \tag{197}$$

When $R_i(t) \geq R_{coop}$, $\chi_i(t) = 1$, and the curvature coordination term is fully effective; when $R_i(t) \leq R_{ter}$, $\chi_i(t) = 0$, the curvature coordination term is turned off, and the guidance law reduces to terminal proportional guidance.

Considering the guidance commands after terminal switching:

$$a_i^{cmd}(t) = a_{gi}^{\lim}(t) + \chi_i(t)a_{ci}^{\lim}(t). \tag{198}$$

The final guidance acceleration is:

$$a_i(t) = \mathcal{L}_a\left\{P_i(t)a_i^{cmd}(t)\right\}. \tag{199}$$

Because terminal accuracy is sensitive to prediction error, the target reference state is fused from the predicted and filtered states.

$$\alpha_i(t) = \mathrm{sat}\left(\frac{R_i(t) - R_{ter}}{R_{coop} - R_{ter}}, 0, 1\right)\mathrm{sat}\left(\frac{C_k - C_{\min}}{C_{\max} - C_{\min}}, 0, 1\right). \tag{200}$$

The target reference state is set as follows:

$$\hat{x}_T^{ref}(t) = \alpha_i(t)\hat{x}_T^p(t) + [1 - \alpha_i(t)]\hat{x}_T(t). \tag{201}$$

The corresponding target reference position and velocity are as follows:

$$\begin{array}{ll} \hat{r}_T^{ref}(t) & = \alpha_i(t)\hat{r}_T^p(t) + [1 - \alpha_i(t)]\hat{r}_T(t), \\ \hat{v}_T^{ref}(t) & = \alpha_i(t)\hat{v}_T^p(t) + [1 - \alpha_i(t)]\hat{v}_T(t). \end{array} \tag{202}$$

Far from the target and under high confidence, the predicted state dominates; near terminal engagement or under low confidence, the filtered state dominates.

This continuous fusion strategy avoids abrupt changes in the reference state caused by hard switching between the predicted and filtered states, which helps improve the smoothness of guidance commands. The specific proof of necessity is presented in Appendices A–E.

4.8. Communication Assumption and Non-Ideal Information Exchange

The baseline guidance model assumes consistent target states, virtual targets, and desired impact times among missiles. This isolates the LSTM-IMM and curvature-time mechanism before communication imperfections are introduced.

With non-ideal communication, a missile may receive delayed, missing, or biased cooperative variables. Delay and packet loss produce stale virtual targets and desired impact times; clock error directly biases time-to-go estimation.

These errors enter mainly through the time-coordination channel. Small induced time errors cause smooth degradation, while large errors activate the hit-priority and terminal-switching logic so that local interception accuracy is preserved.

Perfect communication is therefore used only as the baseline assumption. Communication-aware guidance with delay compensation, event-triggered updates, and distributed re-synchronization is left for future work.

5. Simulation Experiment

The simulations evaluate tracking, prediction, closed-loop guidance, computational cost, Monte Carlo reliability, and additional interception geometries. The experiments compare LSTM-IMM with observer, Kalman-filter, IMM, GRU-IMM, Transformer-IMM, and adaptive Bayesian baselines and then test cooperative interception through 200 Monte Carlo trials and five encounter scenarios.

The simulation was conducted in a three-dimensional inertial coordinate system with a simulation step size of 0.001 s and a maximum simulation time of 180 s. The gravitational acceleration was taken as $g_0 = 9.81\ \mathrm{m/s^2}$. The hit criterion was set to a distance of less than 5 m between the missile and the target. The initial position of the target was:

$$r_T(0) = \left[15{,}000,\ 25{,}000,\ 0\right]^T m \tag{203}$$

The initial velocity was 1200 m/s, and the initial trajectory inclination and heading angles were respectively:

$$\gamma_T(0) = -15^\circ, \psi_T(0) = -90^\circ. \tag{204}$$

The initial states of the four missiles are shown in Table 2.

Table 2. Missile initial state.

Missile	x_0/m	y_0/m	z_0/m	V_0/(m/s)	γ_0	ψ_0
M1	0	60,000	8000	2120	20°	−30°
M2	5000	60,000	0	2220	40°	−30°
M3	20,000	60,000	0	2100	40°	−120°
M4	30,000	60,000	8000	2200	20°	−160°

To make the experimental settings reproducible, the main simulation, tracking, training, and robustness-test parameters are summarized in Table 3.

Table 3. Simulation parameters used in the experiments.

Category	Parameter	Value
Numerical simulation	Integration step size	0.001 s
Numerical simulation	Maximum engagement time	180 s
Interception scenario	Number of missiles/target	4/1
Interception scenario	Target initial state	position (15,000, 25,000, 0) m; speed 1200 m/s; theta = −15 deg; psi = −90 deg
Interception scenario	Hit criterion	miss distance < 5 m
Guidance module	Cooperative activation/time-coordination activation	2.0 s/3.0 s
Guidance module	Encirclement radius/overload limit	500 m/20 g
Prediction module	Prediction step and horizon	0.001 s, 120 steps (0.12 s); guidance uses the 100th step (0.10 s)
Prediction module	Prediction confidence threshold/terminal-switching range	0.45/1500 m
LSTM-IMM model	IMM motion models	CV, CA, and three-dimensional CT
LSTM-IMM model	Feature/output dimension	12 input features/9 transition probability outputs
LSTM-IMM model	BiLSTM architecture	2 layers, 128 hidden units, dropout 0.30
LSTM-IMM model	Online historical window	50 samples (1.0 s at 50 Hz)
Training dataset	Generated trajectories and split	5000 trajectories; train/validation/test = 70%/15%/15%
Training dataset	Trajectory length and sampling	20–40 s; sampling step 0.02 s (50 Hz)
Training dataset	Maneuver settings	1–4 segments; 3–12 s each; acceleration 1–8 g; maximum turn rate 0.8 rad/s
Training dataset	Measurement-noise variance range	position 1–16 m^2; speed 4–16 $(m/s)^2$; angle $[1\ deg, 5\ deg]^2$
Network training	Optimizer and schedule	Adam, 50 epochs, mini-batch 32, initial learning rate 1×10^{-3}
Robustness test	Monte Carlo trials	200 independent trials
Robustness test	Monte Carlo perturbation range	missile position +/−(2000, 2000, 1000) m; speed +/−150 m/s; angles +/−(10, 15) deg

Unless otherwise stated, the values in Table 3 are used consistently in the state estimation, prediction, cooperative-interception, and Monte Carlo experiments.

The following metrics are used to evaluate algorithm performance. The target position estimation error is defined as

$$e_p(k) = \| \hat{r}_T(k) - r_T(k) \| \tag{205}$$

Velocity estimation error is defined as

$$e_v(k) = \| \hat{v}_T(k) - v_T(k) \| \tag{206}$$

Forward prediction error is defined as

$$e_{\text{pred}}(k) = \| \hat{r}_T(k + \Delta K \mid k) - r_T(k + \Delta K) \| \tag{207}$$

where ΔK is the number of prediction steps. The miss distance is defined as

$$d_i = \min_t \| r_{M_i}(t) - r_T(t) \| \tag{208}$$

Time coordination error is defined as

$$\Delta t_h = \max_i(t_{h,i}) - \min_i(t_{h,i}) \tag{209}$$

where $t_{h,i}$ is the hit time of the i-th missile.

5.1. Experiment on State Estimation and Prediction of Highly Maneuverable Targets

This section evaluates LSTM-IMM tracking and 0.30 s prediction for a maneuvering target. The comparison covers position error, prediction error, time lag, chatter, and maneuver-mode recognition.

5.1.1. Comparison of Multi-Sensor State Estimation and Forward Prediction Performance

The first comparison uses the same trajectory, measurements, and initial conditions for Hybrid, LSTM, FTDO, ESO, EKF, UKF, CV, and LSTM-IMM.

During maneuver switching, single-model filters lag and disturbance observers may chatter. LSTM-IMM updates IMM model probabilities from recent motion features, which improves adaptation during maneuver phases.

Figure 2 shows that LSTM-IMM keeps the position error low during maneuvers. FTDO produces large transient peaks, while Hybrid and ESO show stronger fluctuations than the Kalman and IMM-based methods.

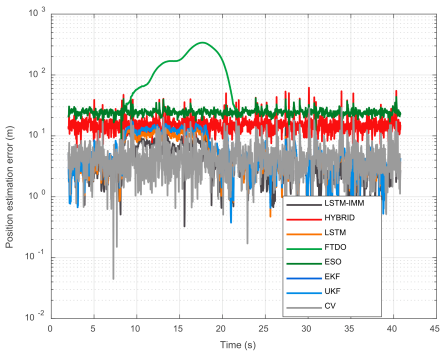


Figure 2. Time series of position estimation errors.

Figure 3 reports the lowest position RMSE for LSTM-IMM, 4.958 m. The reductions relative to Hybrid, LSTM, FTDO, ESO, EKF, UKF, and CV are 69.2%, 23.4%, 95.6%, 79.9%, 34.4%, 34.4%, and 12.0%, respectively.

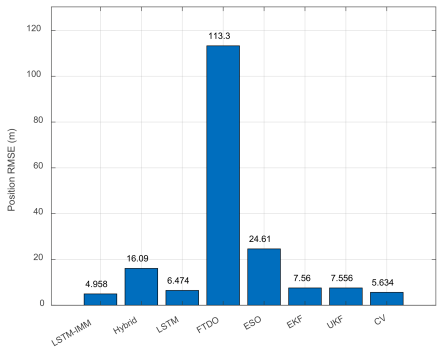


Figure 3. Position error.

For 0.30 s prediction (Figure 4), LSTM-IMM obtains the lowest RMSE, 18.8 m. The corresponding reductions over Hybrid, LSTM, FTDO, ESO, EKF, UKF, and CV are 7.4%, 33.1%, 89.5%, 34.1%, 34.2%, 34.2%, and 27.6%.

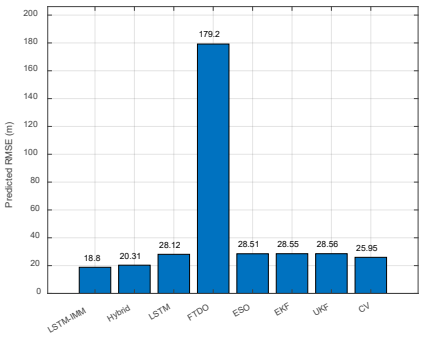


Figure 4. Forward prediction: 0.30 s.

Figure 5 gives average lags of 15.59 ms, 31.64 ms, and 19.99 ms for Hybrid, FTDO, and ESO. LSTM-IMM, LSTM, EKF, and UKF are close to zero lag, but LSTM-IMM also keeps lower estimation and prediction errors.

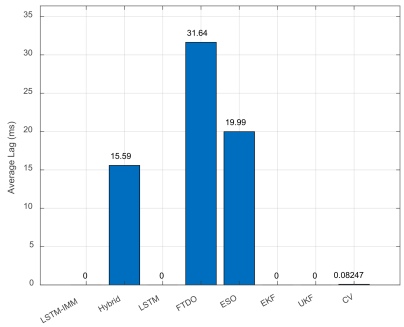


Figure 5. Time lag.

Figure 6 gives a chatter index of 1.367 m for LSTM-IMM, close to EKF (1.322 m) and UKF (1.324 m), and below Hybrid (10.29 m), ESO (4.75 m), and CV (4.767 m).

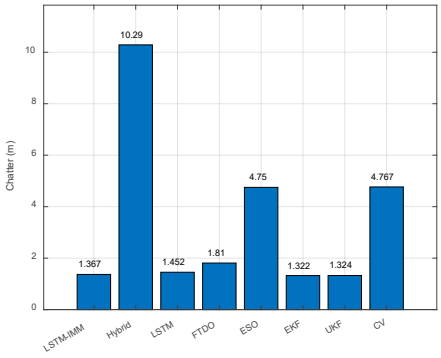


Figure 6. Estimation chatter.

These results show that LSTM-IMM improves estimation and short-term prediction while keeping lag and chatter low.

Algorithmically, this behavior occurs because the learned transition matrix smooths maneuver evidence over the 1.0 s history window. Isolated measurement spikes are less likely to switch the IMM mode, whereas sustained acceleration or turn-rate patterns shift probability toward the matching motion model.

5.1.2. Comparison of Estimation Performance Among Different IMM Model Switching Strategies

The second comparison isolates the transition probability mechanism by testing classic IMM, adaptive IMM, and LSTM-IMM with the same models, measurements, and initial conditions.

Figure 7 shows position RMSEs of 4.93 +/− 5.77 m, 7.48 +/− 7.81 m, and 4.97 +/− 5.61 m for classic IMM, adaptive IMM, and LSTM-IMM. Adaptive IMM shows larger dispersion because residual-driven switching is less stable under noise.

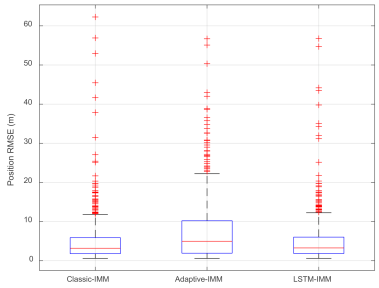


Figure 7. Position estimation accuracy comparison.

Figure 8 gives velocity RMSEs of 1.75 +/− 1.90 m/s for classic IMM, 1.62 +/− 1.23 m/s for adaptive IMM, and 1.79 +/− 2.19 m/s for LSTM-IMM. Velocity accuracy is therefore comparable among the three methods.

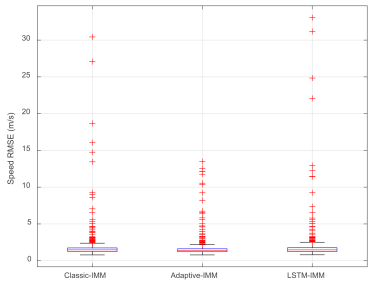


Figure 8. Velocity estimation accuracy.

Figure 9 shows the main gain of LSTM-IMM: mode accuracy reaches 79.47%, compared with 52.62% for classic IMM and 34.45% for adaptive IMM.

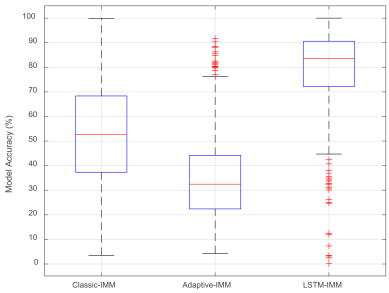


Figure 9. Comparison of pattern recognition methods.

Figure 10 shows continuous CV, CA, and CT probability evolution in LSTM-IMM. The probabilities follow maneuver trends instead of remaining fixed or jumping with residual spikes.

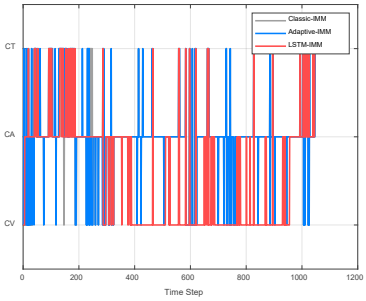


Figure 10. Comparison of pattern recognition methods.

Thus, LSTM-IMM mainly improves maneuver-mode recognition without sacrificing position or velocity estimation accuracy.

The numerical pattern is consistent with this mechanism: the 79.47% mode accuracy of LSTM-IMM is 26.85 and 45.02 percentage points higher than those of the classic IMM and adaptive IMM, while velocity RMSE remains within 0.17 m/s of the best value because the three methods use the same velocity measurements and motion-model set.

5.1.3. Sensitivity Analysis of the LSTM Historical Sequence Length

A sensitivity analysis was conducted for $L_w = 10,\ 20,\ 50,\ 100$, and 200 samples, corresponding to 0.2, 0.4, 1.0, 2.0, and 4.0 s at 50 Hz.

The KL divergences for $L_w = 10,\ 20,\ 50,\ 100$, and 200 are 0.4875, 0.3367, 0.2935, 0.3028, and 0.3061. The inference times are 8.10, 8.05, 11.44, 16.09, and 25.42 ms.

The minimum KL divergence occurs at $L_w = 50$. Longer windows raise inference time by 40.6% to 122.2% relative to $L_w = 50$ without reducing transition-matrix error, so $L_w = 50$ is used by default.

5.2. Simulation Analysis of Curvature Simultaneous Impact Guidance Assisted by LSTM-IMM Predictions

The closed-loop interception test combines LSTM-IMM prediction with curvature-time guidance for a maneuvering target.

Figure 11 shows that the common remaining flight time is 36.171 s and the expected impact time is 38.171 s. The time-coordination mode starts at t = 3.00 s, with missiles 1–4 assigned to encirclement points 4, 3, 1, and 2.

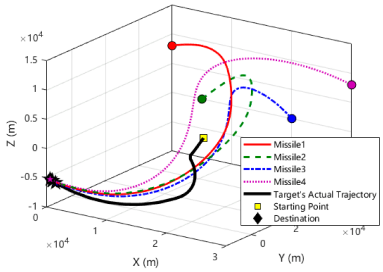


Figure 11. Four-directional coordinated interception trajectory.

All four missiles hit the target. The hit times are 40.031, 40.635, 40.700, and 41.083 s, giving a 1.052 s interval and a 0.435 s standard deviation. The four miss distances are 4.627, 4.709, 4.110, and 4.820 m.

Figures 12 and 13 show that the combined overload reaches the imposed 20.00 g limit without exceeding it. The peak Y-direction overloads are 15.93, 18.00, 14.75, and 17.49 g; the peak Z-direction overloads are 19.26, 19.03, 18.38, and 18.80 g.

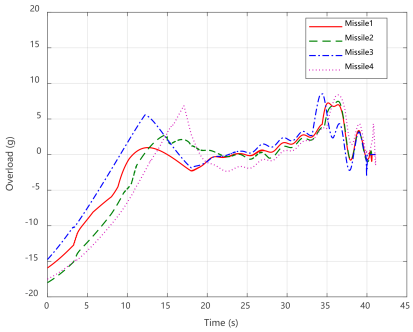


Figure 12. Comparison of normal overload.

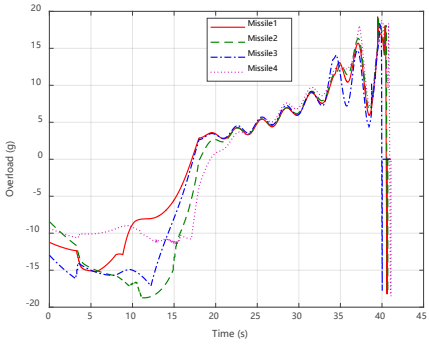


Figure 13. Z-direction overload comparison.

The LSTM-IMM + curvature simultaneous impact method completes the four-way encirclement task by combining target prediction with fixed-speed path-length regulation. Curvature increases effective path length without changing missile speed, and hit-priority overload allocation keeps all four miss distances below 5 m when timing correction competes with terminal interception accuracy.

5.3. Computational Complexity and Real-Time Feasibility Evaluation

The online computation consists of LSTM-IMM tracking and prediction, cooperative-time update, curvature-guidance calculation, and state propagation. The dominant terms are LSTM inference $O(L_w h^2)$, IMM update $O(M\, n_x^3)$, and H-step prediction $O(H\, M\, n_x^2)$.

In the tested setting, $M = 3$, $n_x = 9$, $H = 120$, and $N_m = 4$. Prediction dominates the online load; assignment, common-time update, and guidance evaluation remain low-dimensional.

Profiling gives 0.545 ms for prediction, 0.060 ms for cooperative-time update, 0.123 ms for four-missile guidance, and 0.607 ms for one online cycle.

The measured 0.607 ms online cycle is below the tested update budget, supporting real-time feasibility in the simulated 50 Hz tracking setting.

5.4. Transition Probability Matrix Comparison with Adaptive IMM

The LSTM-generated transition matrix was compared with fixed-transition IMM and NIS-adaptive IMM on 50 test trajectories using the oracle matrix as reference.

In Figure 14 LSTM-IMM is closest to the oracle matrix. Mean KL divergence is 0.8207 for fixed-transition IMM, 1.0026 for NIS-adaptive IMM, and 0.0512 for LSTM-IMM; mean Frobenius error is 0.4045, 0.4540, and 0.0629.

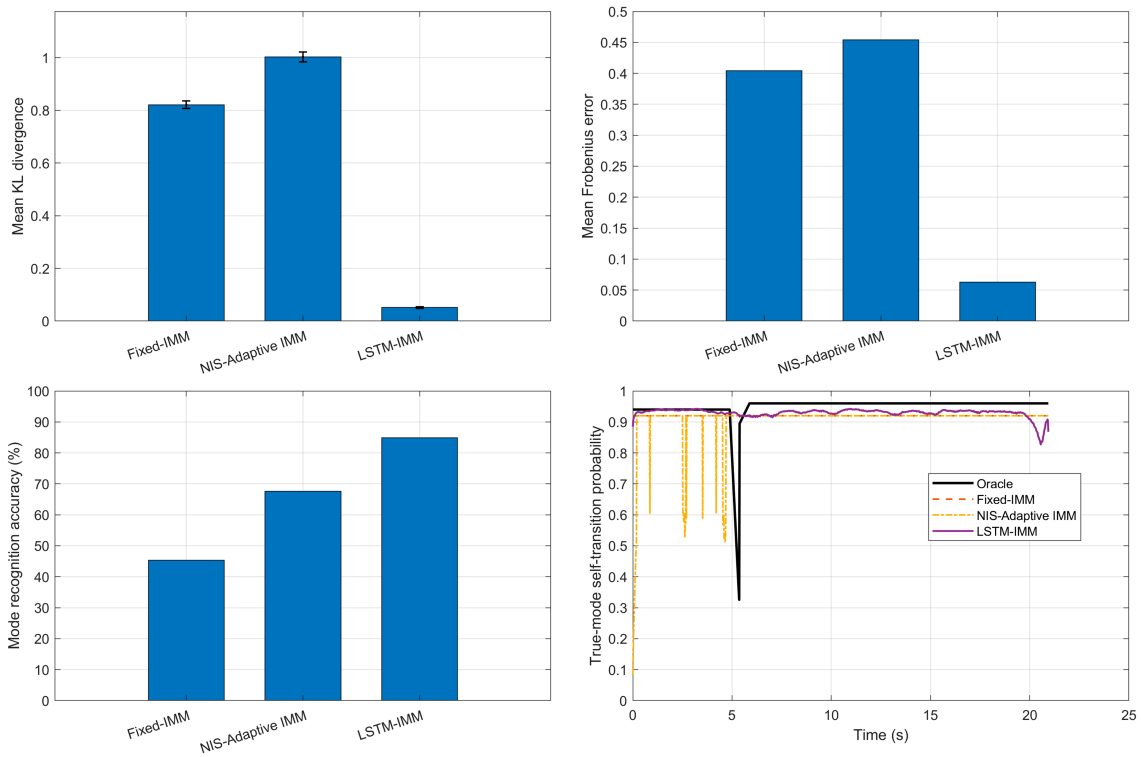


Figure 14. Transition probability matrix comparison among fixed-transition IMM, NIS-adaptive IMM, and LSTM-IMM.

Element-wise errors follow the same trend. Compared with NIS-adaptive IMM, LSTM-IMM reduces diagonal and off-diagonal probability errors by 85.98% and 88.03%, respectively.

Mode-recognition accuracy is 45.36% for fixed-transition IMM, 67.59% for NIS-adaptive IMM, and 84.85% for LSTM-IMM. The gains over the two baselines are 39.49 and 17.26 percentage points.

5.5. Comparison with Modern Target Tracking Algorithms

LSTM-IMM was compared with fixed-transition IMM, NIS-adaptive IMM, GRU-IMM, Transformer-IMM, and adaptive Bayesian-style updating.

In Figure 15 Position RMSEs for fixed-transition IMM, NIS-adaptive IMM, GRU-IMM, Transformer-IMM, and LSTM-IMM are 3.930, 6.102, 3.930, 3.954, and 3.965 m. The 0.30 s prediction RMSEs are 21.315, 26.018, 21.300, 21.350, and 21.308 m.

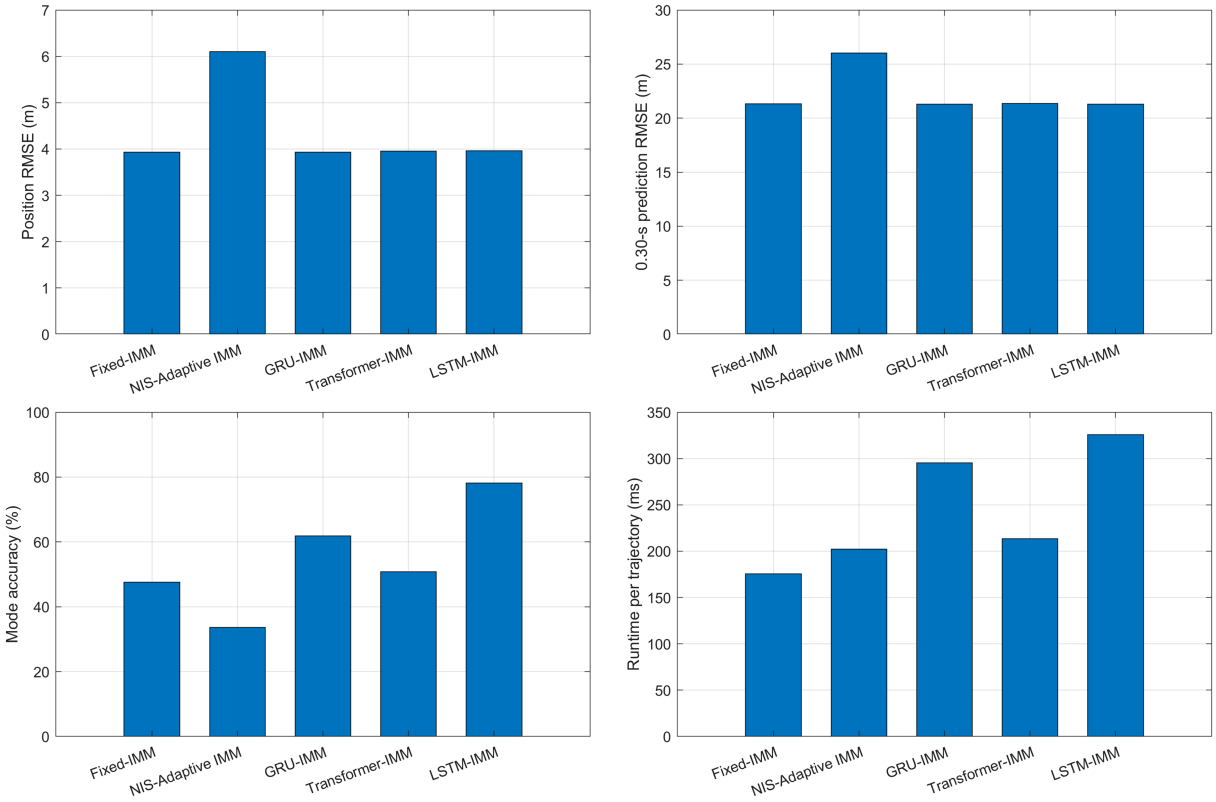


Figure 15. Comparison with modern target tracking algorithms, including fixed-transition IMM, NIS-adaptive IMM, GRU-IMM, Transformer-IMM, and LSTM-IMM.

Mode-recognition accuracies are 47.54%, 33.63%, 61.84%, 50.76%, and 78.12% for fixed-transition IMM, NIS-adaptive IMM, GRU-IMM, Transformer-IMM, and LSTM-IMM. LSTM-IMM improves over GRU-IMM and Transformer-IMM by 16.28 and 27.36 percentage points.

Average runtimes per trajectory are 175.62, 202.11, 295.44, 213.55, and 325.70 ms. LSTM-IMM is slower than GRU-IMM and Transformer-IMM but gives the highest mode-recognition accuracy.

5.6. Monte Carlo Reliability and Statistical Robustness Analysis

The Monte Carlo reliability analysis uses 200 independent trials and reports the full-hit rate, at-least-three-hit rate, miss distance, hit-time interval, hit-time standard deviation, and completion time.

In Figure 16 Method A reaches an 80.0% full-hit rate with a 95% confidence interval of 73.91% to 84.96%, and a 99.0% at-least-three-hit rate with a 95% confidence interval of 96.43% to 99.73%. Method B reaches a 100.0% full-hit rate with a 95% confidence interval of 98.12% to 100.0%.

In full-hit cases, Method A has a mean miss distance of 3.974 m and a 95th percentile value of 4.481 m; Method B gives 4.692 m and 4.830 m. Method A also reduces mean hit-time interval, hit-time standard deviation, and completion time by 30.07%, 30.04%, and 48.66%.

The Monte Carlo results define the trade-off: Method B has higher full-hit reliability, while Method A gives tighter timing and faster completion in full-hit cases.

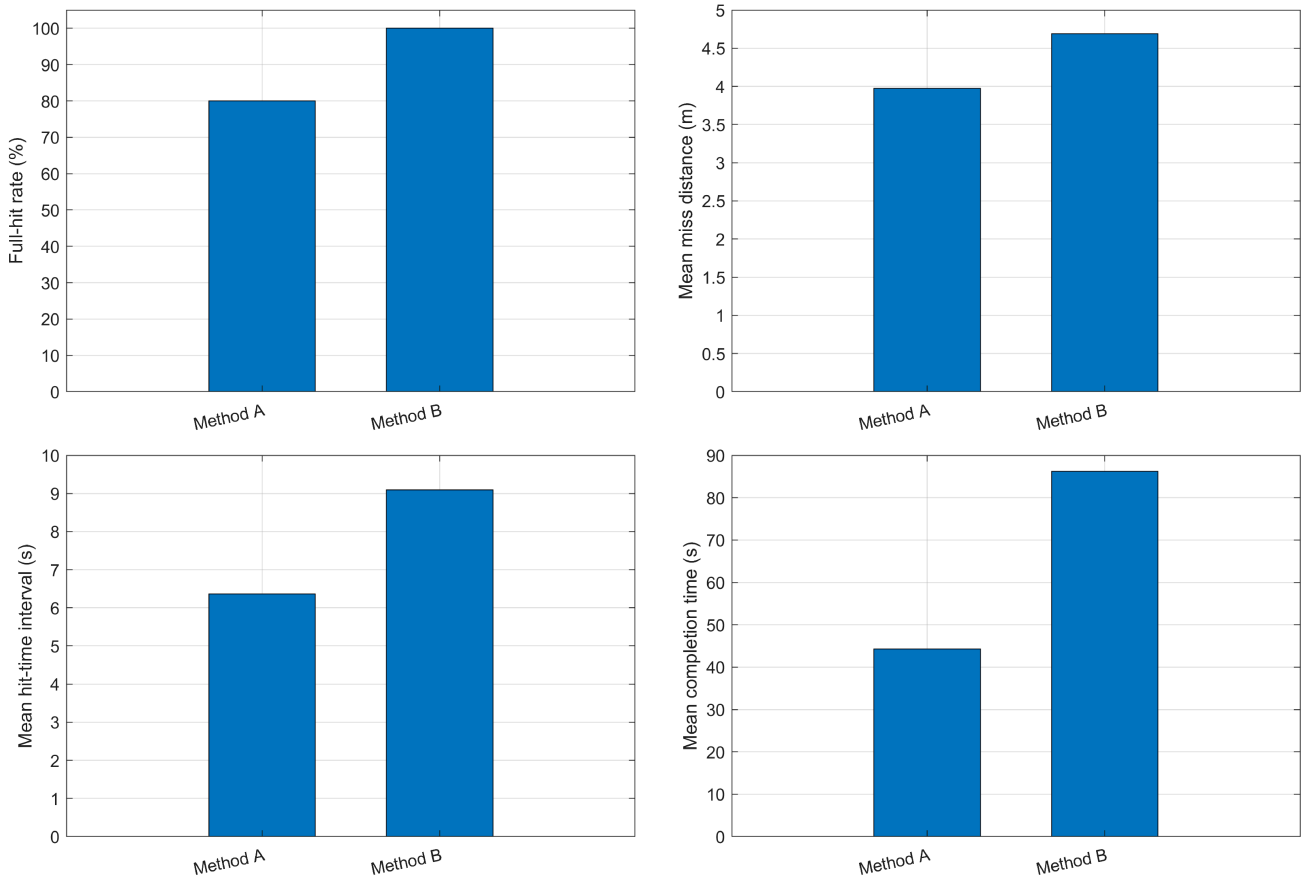


Figure 16. Monte Carlo reliability summary of Method A and Method B over 200 independent trials.

5.7. Additional Interception Scenario Verification

To broaden the validation beyond the original interception case, an additional scenario study was conducted. The test set includes the original baseline scenario and four additional geometries: a left-crossing target, a high-altitude descending target, a dispersed four-missile salvo, and a fast oblique target. Only the initial engagement geometry was changed; the proposed guidance law and its main parameters were kept unchanged.

In Table 4 and Figure 17 All five scenarios achieve 4/4 hits. Mean miss distance remains within 4.271–4.708 m, and the maximum miss distance stays below the 5 m threshold in every case.

Table 4. Additional interception scenarios and proposed-method results.

Scenario	Interception Geometry	Hits	Mean/Max Miss (m)	Interval/Completion (s)
S1	Baseline head-on engagement	4/4	4.567/4.820	1.052/41.083
S2	Left-crossing target with lateral offset	4/4	4.271/4.367	2.667/30.521
S3	High-altitude descending target	4/4	4.514/4.755	2.511/31.583
S4	Dispersed four-missile salvo geometry	4/4	4.619/4.978	7.787/40.509
S5	Fast oblique target engagement	4/4	4.708/4.926	5.688/42.556

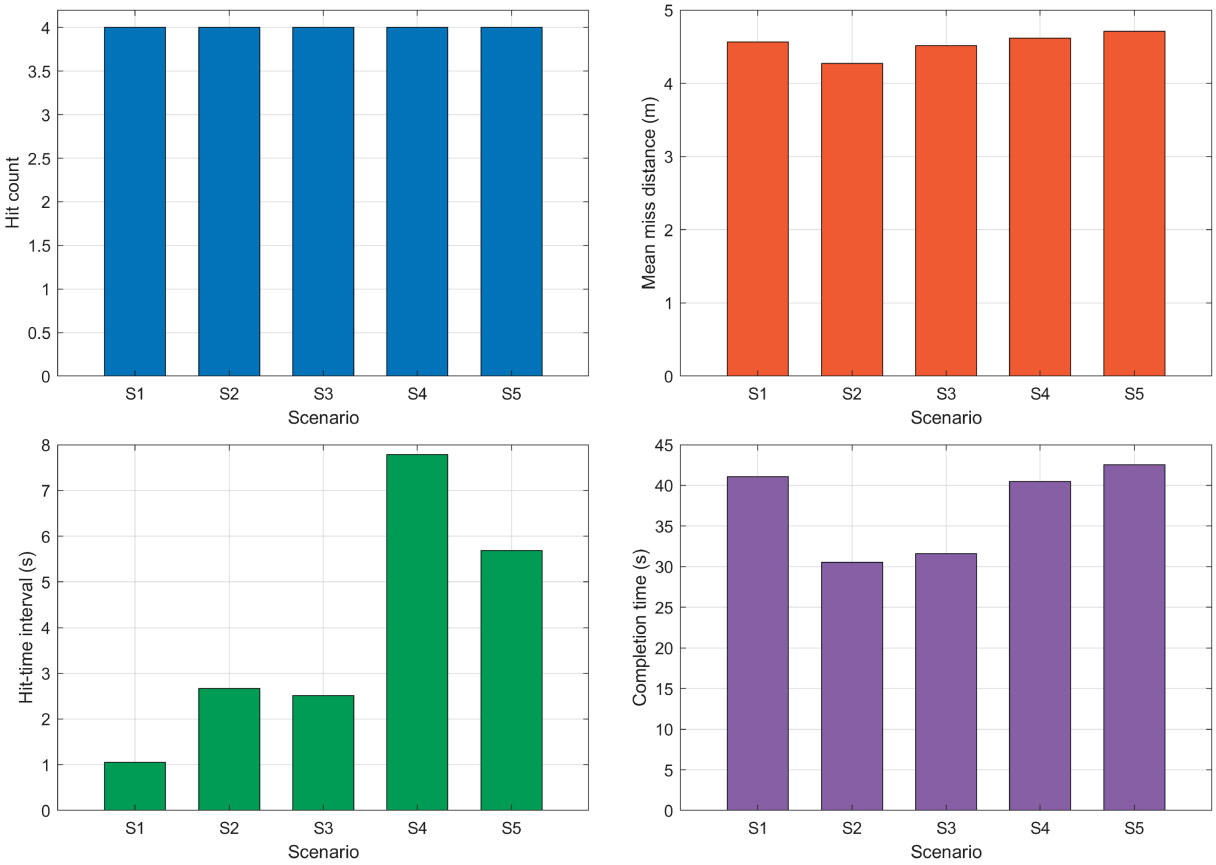


Figure 17. Additional interception scenario verification of the proposed fixed-speed curvature-time guidance method.

S2 and S3 keep compact timing, with hit-time intervals of 2.667 s and 2.511 s. S4 is the hardest timing case because of the dispersed salvo geometry; its interval reaches 7.787 s, but the maximum miss distance remains 4.978 m. S5 completes in 42.556 s with a 4.708 m mean miss distance.

5.8. Technical Discussion of the Results

The results are explained by the coupling of maneuver-mode learning, short-horizon prediction, and fixed-speed coordination. LSTM-IMM supplies a history-dependent transition matrix, so IMM model weights follow acceleration and turning trends instead of reacting only to instantaneous residuals.

This mechanism explains the transition-matrix and mode-recognition gains. Residual-adaptive IMM can respond quickly to large innovations, but it can also amplify noise; the LSTM window suppresses these spurious probability jumps.

For guidance, fixed speed means remaining time cannot be adjusted by tangential acceleration. Curvature commands change effective path length; when the feasible common-time interval narrows, hit-priority allocation limits timing correction and protects miss distance.

The Monte Carlo and scenario results follow this mechanism. Compact geometries produce small timing dispersion, whereas the dispersed S4 case raises the interval to 7.787 s but still keeps all misses below 5 m.

This also explains the reliability trade-off. A variable-speed baseline can reduce time-to-go dispersion directly through tangential speed changes, whereas the proposed fixed-speed method realizes timing correction indirectly through lateral curvature under overload limits. When the curvature margin is exhausted, hit-priority allocation preserves interception accuracy but limits further timing synchronization.

The method is therefore a prediction-assisted fixed-speed coordination scheme with hit-priority protection, not an unconstrained exact simultaneous-arrival controller.

5.9. Practical Engineering Limitations

Several engineering limits remain. The point-mass model omits six-degree-of-freedom attitude motion, aerodynamic uncertainty, actuator bandwidth, control-surface saturation, autopilot lag, and structural load limits.

Time coordination depends on available overload margin. Dispersed salvos, stronger target maneuvers, or low missile energy can shrink or remove the common impact-time window; in that case, the guidance law preserves hit accuracy rather than forcing simultaneity.

LSTM-IMM depends on training coverage and measurement quality. Out-of-distribution maneuvers, unfamiliar noise, or sensor dropout can make the learned transition matrix overconfident or delayed; deployment therefore needs broader data, confidence monitoring, and a fallback IMM or observer mode.

Communication constraints also remain. Delay, packet loss, clock bias, bandwidth limits, and inconsistent target estimates perturb the shared virtual target and desired impact time; deployment requires delay compensation and distributed re-synchronization.

The validation remains simulation-based. Hardware-in-the-loop or processor-in-the-loop tests are needed for scheduling, memory, numerical robustness, embedded inference, and autopilot interaction.

6. Conclusions

This study developed a fixed-speed cooperative interception framework that couples LSTM-IMM maneuver tracking with active curvature planning. The conclusions are supported by three groups of simulation evidence.

For tracking and prediction, LSTM-IMM achieved a position estimation RMSE of 4.958 m and a 0.30 s prediction RMSE of 18.8 m. Its maneuver-mode recognition accuracy reached 79.47%, which is 26.85 and 45.02 percentage points higher than those of classic IMM and adaptive IMM, respectively. These results support the conclusion that the learned, row-normalized transition matrix improves maneuver-mode adaptation while retaining the IMM filtering structure.

For closed-loop interception, all four missiles hit the target in the baseline case. The hit times were 40.031, 40.635, 40.700, and 41.083 s, with a 1.052 s interval and a 0.435 s standard deviation, and the four miss distances were 4.627, 4.709, 4.110, and 4.820 m. The combined overload reached but did not exceed the imposed 20.00 g limit. These results support the conclusion that fixed-speed impact-time coordination can be achieved by curvature-based path-length regulation when sufficient overload margin is available.

The added comparison and robustness tests further define the usable range of the conclusion. Relative to NIS-adaptive IMM, the LSTM-generated transition matrix reduced mean KL divergence by 94.90% and mean Frobenius error by 86.16%. The measured online cycle time was 0.607 ms in the tested 50 Hz setting. In 200 Monte Carlo trials, the proposed fixed-speed Method A achieved an 80.0% full-hit rate and a 99.0% at-least-three-hit rate; in full-hit cases, it reduced mean hit-time interval, hit-time standard deviation, and completion time by 30.07%, 30.04%, and 48.66%, respectively. Across five additional engagement geometries, all cases achieved 4/4 hits, with mean miss distances of 4.271–4.708 m and maximum miss distances below 5 m.

These results indicate that the proposed architecture links maneuver-mode learning, prediction confidence, feasible arrival-time planning, and hit-priority terminal guidance in a single closed-loop scheme. The conclusion is nevertheless limited to the simulated point-mass model, ideal communication assumption, generated maneuvering target cases,

and tested processor/software environment. Future work should evaluate six-degree-of-freedom dynamics, non-ideal communication, online uncertainty calibration, and processor-in-the-loop or hardware-in-the-loop implementation.

Author Contributions: Conceptualization, H.Z.; methodology, H.Z.; software, H.Z.; validation, H.Z. and W.Y.; formal analysis, H.Z.; investigation, H.Z.; resources, W.Y.; data curation, H.Z.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z.; visualization, W.Y.; supervision, W.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article. The data presented in this study can be requested from the authors.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Equations (195) and (199) project the actual guidance acceleration onto the velocity-normal plane. Since

$$P_i(t) = I_3 - \hat{v}_i(t)\hat{v}_i^T(t), \hat{v}_i^T(t)P_i(t) = 0, \tag{A1}$$

Therefore:

$$\hat{v}_i^T(t)a_i(t) = 0. \tag{A2}$$

Since $v_i(t) = V_{i0}\hat{v}_i(t)$, therefore:

$$v_i^T(t)a_i(t) = 0. \tag{A3}$$

The missile's kinematics satisfy $\dot{v}_i(t) = a_i(t)$; therefore, the rate of change of velocity modulus is:

$$\frac{d}{dt}\left[\frac{1}{2} \parallel v_i(t) \parallel^2\right] = v_i^T(t)\dot{v}_i(t) = v_i^T(t)a_i(t) = 0. \tag{A4}$$

It follows that:

$$\parallel v_i(t) \parallel = \parallel v_i(0) \parallel = V_{i0}. \tag{A5}$$

Equation (A5) shows that the proposed guidance law can keep the missile velocity constant under the mass kinematic model; that is, it satisfies the fixed velocity constraint.

Appendix B

As defined by the overload limiting operator, for any input u, we have:

$$\parallel \mathcal{L}_a\{u\} \parallel \leq a_{i,max}. \tag{A6}$$

Therefore, the actual guidance acceleration satisfies:

$$\parallel a_i(t) \parallel \leq a_{i,max}. \tag{A7}$$

Equation (B6) demonstrates that the proposed guidance law always satisfies the missile's maximum normal overload constraint.

Furthermore, from missile kinematics:

$$\dot{r}_i(t) = v_i(t), \dot{v}_i(t) = a_i(t), \tag{A8}$$

Combining Equations (A6) and (A8), we obtain:

$$\parallel \dot{r}_i(t) \parallel = V_{i0}, \parallel \dot{v}_i(t) \parallel \leq a_{i,max}. \tag{A9}$$

Therefore, within any finite time interval $\left[0, t_f\right]$:

$$\| r_i(t) - r_i(0) \| \leq V_{i0}t, \| v_i(t) - v_i(0) \| \leq a_{i,\max}t. \tag{A10}$$

Equation (A10) shows that within a finite time interval, the missile's position, velocity, and guidance commands remain bounded, and there will be no finite time divergence.

Appendix C

Define the time coordination error of the *i-th* missile as:

$$e_{\tau i}(t) = \tau_{i,d}(t) - \hat{\tau}_i(t). \tag{A11}$$

If the feasible time intervals of all missiles have a non-empty intersection, that is:

$$\bigcap_{i=1}^{N} \mathcal{I}_i(t) \neq \varnothing, \tag{A12}$$

There exists a common expected remaining time, allowing each missile to achieve a near-expected remaining flight time within its own overload constraints through curvature adjustment.

Considering that target prediction error, curvature approximation error, discrete update error, and guidance execution error are all bounded, their combined influence is denoted as $\varepsilon_\tau(t)$. The time error then satisfies a conditional bounded relation:

$$| e_{\tau i}(t) | \leq \varepsilon_\tau(t), i = 1, 2, \cdots, \mathrm{N}. \tag{A13}$$

If a common feasible time interval does not exist, that is:

$$\bigcap_{i=1}^{N} \mathcal{I}_i(t) = \varnothing, \tag{A14}$$

Then both sides must reach infeasibility simultaneously. Define the width of the infeasibility interval:

$$\Delta\tau_{inf}(t) = \max\left(0, \max_i \tau_{i,L}(t) - \min_i \tau_{i,U}(t)\right). \tag{A15}$$

After adopting the minimum distribution strategy from Section 4.5, the time error can be expressed in a bounded form:

$$| e_{\tau i}(t) | \leq \Delta\tau_{inf}(t) + \varepsilon_\tau(t). \tag{A16}$$

Equation (A16) shows that when strictly simultaneous arrival is infeasible, the degree of hit-time dispersion is jointly limited by geometric conditions, velocity differences, available overload, and prediction errors. The method in this paper cannot eliminate this infeasibility, but it can minimize the time dispersion within the achievable range.

Appendix D

When the missile enters the terminal phase, that is:

$$R_i(t) \leq R_{ter}, \tag{A17}$$

From Equation (198), we can see that:

$$\chi_i(t) = 0. \tag{A18}$$

At this point, the guidance command degenerates into:

$$a_i^{cmd}(t) = a_{gi}^{lim}(t), \tag{A19}$$

This means the missile primarily uses proportional guidance for terminal interception.

Under the conditions that the target's maneuvering acceleration is bounded, the closing velocity remains positive, the missile has sufficient overload margin, and the target state estimation error is bounded, proportional guidance can keep the line-of-sight angular velocity bounded and tend towards a smaller neighborhood. Specifically, if:

$$\| a_T(t) \| \leq a_{T,max}, V_{ci}(t) \geq V_{c,min} > 0, \tag{A20}$$

Furthermore, if the available overload in the missile's terminal phase can cover the main guidance requirements, then the terminal miss distance satisfies a bounded relationship:

$$d_i \leq \varepsilon_{PN}(a_{T,\max}, a_{i,\max}, N_i, V_{ci}, \bar{e}_T), \tag{A21}$$

where $\bar{e}_T$ represents the upper bound of the target state estimation error, and $\varepsilon_{PN}(\cdot)$ is a bounded function determined by target maneuvering, missile overload, proportional guidance coefficient, closure velocity, and estimation error.

If:

$$\varepsilon_{PN}(a_{T,max}, a_{i,max}, N_i, V_{ci}, \bar{e}_T) \leq R_{hit}, \tag{A22}$$

Therefore, the missile can be considered to meet the effective hit criterion.

Thus, under bounded target maneuver, positive closing velocity, sufficient overload margin, and bounded estimation error, terminal proportional guidance yields a bounded miss distance that can satisfy the hit radius.

Appendix E

Equation (197) defines the cooperative weighting function as a continuous saturation function of relative distance, and

$$0 \leq \chi_i(t) \leq 1. \tag{A23}$$

When the missile transitions from the cooperative guidance phase to the terminal guidance phase, $\chi_i(t)$ gradually decreases from 1 to 0. Therefore, the curvature cooperative term $\chi_i(t)a_{ci}^{lim}(t)$ decays continuously and does not suddenly disappear at the terminal boundary.

Simultaneously, the target reference state adopts the continuous fusion form shown in Equation (201). Because:

$$0 \leq \alpha_i(t) \leq 1, \tag{A24}$$

Since both $\hat{x}_T^p(t)$ and $\hat{x}_T(t)$ are bounded, $\hat{x}_T^{ref}(t)$ is also bounded.

Furthermore, the actual guidance command undergoes projection onto the normal plane and overload limiting. Because the projection matrix $P_i(t)$ is bounded, $\chi_i(t)$ is bounded, $a_{gi}^{lim}(t)$ and $a_{ci}^{lim}(t)$ are bounded, and the limiting operator $\mathcal{L}_a\{\cdot\}$ is a bounded mapping; therefore:

$$\| a_i(t) \| \leq a_{i,max}. \tag{A25}$$

Equation (A25) shows that terminal handover does not introduce unbounded guidance commands. After adopting continuous weight decay and reference-state fusion, the guidance commands have good smoothness and boundedness in engineering implementation.

https://doi.org/10.3390/aerospace13080677

References

1. Yu, J.; Shi, Z.; Dong, X.; Li, Q.; Ren, Z. Secure Cooperative Guidance Strategy for Multimissile System with Collision Avoidance. *IEEE Trans. Aerosp. Electron. Syst.* **2025**, *61*, 1034–1047. [CrossRef]
2. Cho, N.; Kim, Y. Modified Pure Proportional Navigation Guidance Law for Impact Time Control. *J. Guid. Control Dyn.* **2016**, *39*, 852–872. [CrossRef]
3. Zhou, J.; Yang, J.; Li, Z. Simultaneous attack of a stationary target using multiple missiles: A consensus-based approach. *Sci. China Inf. Sci.* **2017**, *60*, 070205. [CrossRef]
4. Song, L.; Zhang, Y.; Huang, D.; Fu, S. Cooperative simultaneous attack of multi-missiles under unreliable and noisy communication network: A consensus scheme of impact time. *Aerosp. Sci. Technol.* **2015**, *47*, 31–41. [CrossRef]
5. Khalkhali, M.B.; Vahedian, A.; Yazdi, H.S. Multi-Target State Estimation Using Interactive Kalman Filter for Multi-Vehicle Tracking. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1131–1144. [CrossRef]
6. Wang, Y.; Li, X. Impulsive observer and impulsive control for time-delay systems. *J. Frankl. Inst.* **2020**, *357*, 8529–8542. [CrossRef]
7. Gao, Z.; Yi, W. Trajectory prediction method for incoming guided projectiles based on the fusion of Temporal Convolutional Network and dual attention mechanisms. *Measurement* **2025**, *257*, 118906. [CrossRef]
8. Tordesillas, J.; How, J.P. MADER: Trajectory Planner in Multiagent and Dynamic Environments. *IEEE Trans. Robot.* **2022**, *38*, 463–476. [CrossRef]
9. Ding, H.; Wang, D.; Li, C.; Liang, X.; Li, X. Design of Convergent and Accurate Guidance Law with Finite Time in Complex Adversarial Scenarios. *Aerospace* **2024**, *11*, 56. [CrossRef]
10. Zhang, Y.; Jin, J.; Huang, L. Model-Free Predictive Current Control of PMSM Drives Based on Extended State Observer Using Ultralocal Model. *IEEE Trans. Ind. Electron.* **2021**, *68*, 993–1003. [CrossRef]
11. Wang, F.; Gao, H.; Wang, K.; Zhou, C.; Zong, Q.; Hua, C. Disturbance Observer-Based Finite-Time Control Design for a Quadrotor UAV with External Disturbance. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 834–847. [CrossRef]
12. Hao, A.; Xia, H.; Wang, C. Finite-time output tracking control for air-breathing hypersonic vehicles with actuator constraints. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2016**, *231*, 2578–2593. [CrossRef]
13. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef] [PubMed]
14. Murray-Smith, R.; Johansen, T. *Multiple Model Approaches to Nonlinear Modelling and Control*; CRC Press: Boca Raton, FL, USA, 2020.
15. Chen, Z.; Zhang, Y.; Gu, J.; Kong, L.; Yang, X.; Yu, F. Dual Aggregation Transformer for Image Super-Resolution. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*; IEEE: Piscataway, NJ, USA, 2023; pp. 12278–12287.
16. Xu, H.; Zhang, J. AANet: Adaptive Aggregation Network for Efficient Stereo Matching. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; IEEE: Piscataway, NJ, USA, 2020; pp. 1956–1965.
17. Khodarahmi, M.; Maihami, V. A Review on Kalman Filter Models. *Arch. Comput. Methods Eng.* **2023**, *30*, 727–747. [CrossRef]
18. Xie, G.; Sun, L.; Wen, T.; Hei, X.; Qian, F. Adaptive Transition Probability Matrix-Based Parallel IMM Algorithm. *IEEE Trans. Syst. Man Cybern. Syst.* **2021**, *51*, 2980–2989. [CrossRef]
19. Han, B.; Huang, H.; Lei, L.; Huang, C.; Zhang, Z. An Improved IMM Algorithm Based on STSRCKF for Maneuvering Target Tracking. *IEEE Access* **2019**, *7*, 57795–57804. [CrossRef]
20. Zhang, Z.; Ma, K.; Zhang, G.; Yan, L. Virtual target approach-based optimal guidance law with both impact time and terminal angle constraints. *Nonlinear Dyn.* **2022**, *107*, 3521–3541. [CrossRef]
21. Gorbunov, A.S.; Dyachkovskii, A.S. Methods of velocity registration on ballistic. In *Proceedings of the XII All Russian Scientific Conference on Current Issues of Continuum Mechanics and Celestial Mechanics: XII CICMCM, Tomsk, Russia, 15–17 November 2023*; Springer Nature: Singapore, 2024; p. 101.
22. Min-Jea, T.; Chang-Kyung, R.; Hangju, C. Recursive time-to-go estimation for homing guidance missiles. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 13–24. [CrossRef]
23. Rigdon, E.E.; Schumacker, R.E.; Wothke, W. A comparative review of interaction and nonlinear modeling. In *Interaction and Nonlinear Effects in Structural Equation Modeling*; Lawrence Erlbaum Associates (LEA): Mahwah, NJ, USA, 2017; pp. 1–16.
24. Rumelhart, D.E. Toward an interactive model of reading. In *Attention and Performance VI*; Routledge: New York, NY, USA, 2022; pp. 573–603.
25. Lui, D.G.; Tartaglione, G.; Conti, F.; Tommasi, G.D.; Santini, S. Long Short-Term Memory-Based Neural Networks for Missile Maneuvers Trajectories Prediction. *IEEE Access* **2023**, *11*, 30819–30831. [CrossRef]
26. Di, Y.; Li, R.; Tian, H.; Guo, J.; Shi, B.; Wang, Z.; Yan, K.; Liu, Y. A maneuvering target tracking based on fastIMM-extended Viterbi algorithm. *Neural Comput. Appl.* **2025**, *37*, 7925–7934. [CrossRef]
27. Gao, H.; Qin, Y.; Hu, C.; Liu, Y.; Li, K. An Interacting Multiple Model for Trajectory Prediction of Intelligent Vehicles in Typical Road Traffic Scenario. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 6468–6479. [CrossRef] [PubMed]
28. Li, B.; Wu, Y. LSTM Model for Trajectory Design of Missile-Borne BFSAR Imaging Guidance. *J. Syst. Simul.* **2019**, *31*, 2696–2701.

29. Lee, I.H.; Park, C.G. An Improved Interacting Multiple Model Algorithm with Adaptive Transition Probability Matrix Based on the Situation. *Int. J. Control Autom. Syst.* **2023**, *21*, 3299–3312. [CrossRef]
30. Kuo, C.Y.; Soetanto, D.; Ying-Chwan, C. Geometric analysis of flight control command for tactical missile guidance. *IEEE Trans. Control Syst. Technol.* **2001**, *9*, 234–243. [CrossRef]
31. Mracek, C.; Ridgely, D. Missile longitudinal autopilots: Connections between optimal control and classical topologies. In *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*; AIAA: Reston, VA, USA, 2005; p. 6381.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.