Physics-Informed Neural Models for Uncertain Trajectory Prediction in Urban Air Mobility

Alice Inbaraj* and Jun Chen†
San Diego State University, San Diego, California 92182

https://doi.org/10.2514/1.I011720

This study proposes a physics-informed neural network (PINN) framework for trajectory prediction and uncertainty quantification in urban air mobility (UAM), with a particular focus on electric vertical takeoff and landing (eVTOL) vehicles. UAM operations are subject to complex and variable urban environments, requiring accurate and reliable real-time trajectory forecasting. By embedding governing physical laws into the learning architecture, the proposed PINN model adheres to eVTOL flight dynamics, thereby improving prediction fidelity and offering more consistent uncertainty estimates. Comparative evaluations using NASA's UAM simulation dataset demonstrate that the PINN approach outperforms conventional neural networks and Gaussian mixture regression in both accuracy and robustness. Additionally, the model enhances interpretability, making it particularly suitable for safety-critical applications. These results underscore the potential of physics-informed learning to support the development of more dependable and efficient trajectory planning tools for emerging UAM systems.

Nomenclature

F_X, F_Y, F_Z	=	aeropropulsive forces, lbf
g	=	gravitational acceleration, ft/s^2
I_x, I_y, I_z, I_{xz}	=	moments of inertia, slug $\cdot$ ft^2
L, M, N	=	body-axis moments, ft $\cdot$ lb
m	=	aircraft mass, slugs
p, q, r	=	body-axis angular velocity components, rad/s or deg /s
u, v, w	=	body-axis translational velocity components, ft/s
X, Y, Z	=	positions
θ, ϕ, ψ	=	pitch, roll, yaw angles, rad/s or deg /s
λ	=	hyperparameter

I. Introduction

URBAN air mobility (UAM) has emerged as a promising approach to alleviate surface congestion and improve transport efficiency in densely populated areas. Leveraging electric vertical takeoff and landing (eVTOL) aircraft, UAM envisions rapid, on-demand aerial transportation within urban and suburban airspaces. However, the realization of safe and scalable UAM systems requires accurate, real-time trajectory prediction, particularly in the face of uncertainties inherent to urban environments, such as variable wind patterns, turbulence, and complex obstacles [1–3].

Traditional trajectory prediction methods, including standard neural networks (NNs), Gaussian mixture regression (GMR), and model predictive control (MPC), often struggle to meet these requirements. NNs and GMR offer flexibility and adaptability but lack direct incorporation of physical constraints, which limits their generalizability under dynamic and uncertain flight conditions [4]. While MPC provides a principled framework for dynamic feasibility, it requires explicit models and does not natively address uncertainty unless extended through stochastic formulations [5]. Moreover, many uncertainty quantification (UQ) techniques rely on probabilistic sampling methods that are computationally intensive and impractical for real-time applications with sparse or noisy data [6].

To address these challenges, we propose a physics-informed neural network (PINN) approach that embeds governing equations of motion into the learning process, producing physically consistent, data-efficient, and uncertainty-aware trajectory predictions. Our work builds on recent advancements that demonstrate the promise of PINNs in aerospace and fluid dynamics applications, particularly where nonlinear behavior and data scarcity are dominant concerns [7–10].

Although the concept of personal air travel dates back over a century, recent technological advances in electric propulsion, autonomous navigation, and lightweight materials have accelerated the development of eVTOL aircraft tailored for UAM. Compared to traditional helicopters, eVTOLs offer benefits including reduced noise, lower emissions, and the ability to operate in congested urban environments without runways. These vehicles are integral to the UAM vision of decentralized, point-to-point transport and require sophisticated flight planning systems that can handle complex, uncertain environments in real time.

PINNs, introduced by Raissi et al. [7], offer a novel paradigm for integrating physical laws directly into machine learning models. By embedding governing equations, such as conservation of momentum or energy, into the training loss function, PINNs ensure that predictions not only fit the data but also adhere to known physics. This makes them particularly effective in scenarios with limited training data or where extrapolation beyond the training domain is necessary. In aerospace applications, PINNs have been applied to constrained trajectory modeling, fluid–structure interaction, and parameter estimation. For example, the FlyNet architecture by Stachiw et al. [11] demonstrated improved generalization in control tasks by incorporating physical constraints. Recent efforts, including those by Michek et al. [9] and Kunz et al. [10], have extended the use of PINNs to uncertainty-aware modeling of nonlinear flight dynamics, showing that ensemble-based PINNs can provide confidence bounds on predicted trajectories, an essential feature for risk-aware path planning.

Uncertainty quantification (UQ) plays a critical role in aerospace systems where safety and robustness are paramount. In trajectory prediction, uncertainties may arise from incomplete knowledge of vehicle dynamics (epistemic) or from inherent environmental variability (aleatoric), such as wind gusts or sensor noise [6]. Properly accounting for both types is crucial for designing reliable navigation and control systems. Common UQ methods [12] include Monte Carlo simulation, Bayesian inference, and ensemble modeling. Monte Carlo simulation estimates statistical distributions using repeated random sampling, which is computationally expensive for real-time UAM operations. Bayesian inference updates the probability of model states

Received 26 June 2025; accepted for publication 14 January 2026; published online 16 February 2026.

*Master's Student, Department of Aerospace Engineering. Student Member AIAA.

†Associate Professor, Department of Aerospace Engineering; jun.chen@sdsu.edu. Senior Member AIAA.

as new data become available, requiring strong prior assumptions. Ensemble methods leverage variability across multiple model realizations to estimate prediction confidence, offering a practical balance.

In this study, we adopt an ensemble-based PINN framework to capture both epistemic and aleatoric uncertainties. Epistemic uncertainty represents the uncertainty in the model that arises from a lack of knowledge or data about the system. Aleatoric uncertainty, on the other hand, represents the randomness in the environment or system that cannot be reduced by collecting more data. In our framework, epistemic uncertainty is addressed through multiple network instantiations trained on subsets of the data, while aleatoric uncertainty is modeled via explicit inclusion of stochastic perturbations in the training process. This dual-level approach enables the system to produce more reliable and interpretable trajectory predictions under uncertain operating conditions, supporting safer and more resilient UAM operations.

This paper aims to develop a PINN framework for predicting eVTOL trajectories under uncertain environmental conditions. By integrating physical constraints into a data-driven architecture, the proposed model enhances prediction accuracy while maintaining robustness in complex, real-world urban settings. The key contributions of this work are as follows:

1) We formulate a PINN-based trajectory prediction model for eVTOL aircraft that explicitly incorporates flight dynamics, enabling physically consistent predictions with limited data.

2) We introduce an ensemble-based uncertainty quantification approach within the PINN framework, capturing both epistemic and aleatoric uncertainties relevant to UAM operations.

3) We conduct a comparative evaluation of the proposed model against conventional NNs and GMR, demonstrating the superiority of PINNs in accuracy, reliability, and generalizability under uncertain conditions.

4) We validate the model using NASA's UAM simulation dataset, illustrating its potential for enhancing safety and operational resilience in future UAM systems.

II. Methodology

This section presents the comprehensive methodological framework developed for trajectory prediction and uncertainty quantification of eVTOL vehicles using PINNs. The workflow includes simulation-based data generation, integration of aircraft dynamics into the learning model, architectural design of the PINNs, uncertainty modeling through ensemble and probabilistic techniques, and evaluation of prediction performance using standard error metrics. Each stage is structured to ensure the model's alignment with both the physical laws of motion and the operational complexity of UAM scenarios.

A. Data Collection and Preparation

We leverage simulation data from the NASA Generic UAM (GUAM) model [13], a high-fidelity MATLAB Simulink environment built to study UAM systems. The model simulates Lift + Cruise eVTOL configurations and includes modules for full six-degree-of-freedom (6-DOF) rigid-body dynamics, environmental condition modeling, and aircraft control systems.

Key simulation features include the following:

1) **Reference frames:** Both north-east-down (NED) and Earth-centered, Earth-fixed (ECEF) coordinate systems

2) **Subsystem modularity:** Capability to incorporate varied actuator dynamics, sensor configurations, and flight controllers

3) **Trajectory inputs:** Include user-defined profiles such as ramps, sinusoidal waves, and piecewise Bézier curves

The simulation scenarios vary in environmental conditions (e.g., wind gusts, turbulence levels), flight paths, and operational parameters such as airspeed and climb rate. The dataset spans three key flight regimes: i) hover, ii) climb/transition from hover to forward flight, and iii) steady cruise. All of these represent the most critical dynamic segments. Each GUAM reference simulation lasted approximately 80 s. From these trajectories, 20 s segments (4001 samples each) were extracted such that the data collectively covered all flight phases and were normalized to train and validate the PINN model. A total of 1000 trajectories were collected, each corresponding to a 20 s segment under varying flight and environmental conditions, each comprising sequences of positional (x, y, z), velocity (u, v, w), acceleration, and angular state variables (p, q, r, ϕ, θ, ψ), forming the input–output state vector used in the model training. These high-fidelity outputs serve as the foundational training and validation data for the PINN model.

The NASA GUAM simulation model setup is illustrated in Fig. 1, showcasing the modular Lift + Cruise configuration used to generate diverse flight scenarios. Figure 2 further depicts key state variables from the simulation output, including vehicle position, velocity, and acceleration in the X, Y, and Z axes. These time-series plots help visualize the range of trajectory responses captured by the dataset.

Preprocessing steps involved several stages to prepare the raw simulation output for model training:

1) **Normalization and feature scaling:** All input variables, including state and control vectors, were scaled to the range $[-1,1]$ using min–max normalization. This step ensures numerical stability and faster convergence during training.

2) **Outlier filtering and smoothing:** Trajectory samples exhibiting abrupt anomalies due to simulation artifacts were filtered using a moving average filter and statistical bounds based on interquartile ranges.

3) **State augmentation:** In addition to primary kinematic states, first derivatives such as acceleration and angular acceleration were computed and appended. These derivatives were used in the physics-based loss for better dynamic constraint representation.

4) **Data partitioning:** The final dataset was split into training and testing subsets with an 80/20 ratio. Stratified sampling ensured representation across varying flight conditions (e.g., hover, climb, cruise).

B. Equations of Motion and Physical Constraints

A central aspect of integrating physics into the learning model is the incorporation of the equations of motion that govern the flight dynamics of the eVTOL vehicle. These equations form the basis for deriving the physics-based loss used in the PINN framework. By embedding these physical laws into the network training, the model inherently respects conservation laws and dynamic feasibility, improving generalizability and robustness under variable conditions. This section outlines the translational and rotational dynamics that are used to formulate the constraints imposed on the network outputs.

1. *Translational Dynamics*

The translational motion equations [14] model the aircraft's linear accelerations influenced by gravity and external forces:

$$\dot{u} = rv - qw - g\sin\theta + \frac{X}{m} \tag{1}$$

$$\dot{v} = pw - ru + g\sin\phi\cos\theta + \frac{Y}{m} \tag{2}$$

$$\dot{w} = qu - pv + g\cos\theta\cos\phi + \frac{Z}{m} \tag{3}$$

where

1) u, v, w represent the body-axis translational velocity components (in ft/s);

2) p, q, r are the body-axis angular velocity components (in rad/s);

3) ϕ, θ are body-axis roll and pitch angles (in rad/s or deg /s);

4) X, Y, Z denote the aeropropulsive forces (in lbf); and

5) m is the aircraft mass (in slugs), and g is the gravitational acceleration (in ft/s^2).

2. *Rotational Dynamics*

The rotational equations describe the evolution of angular motion under external moments:

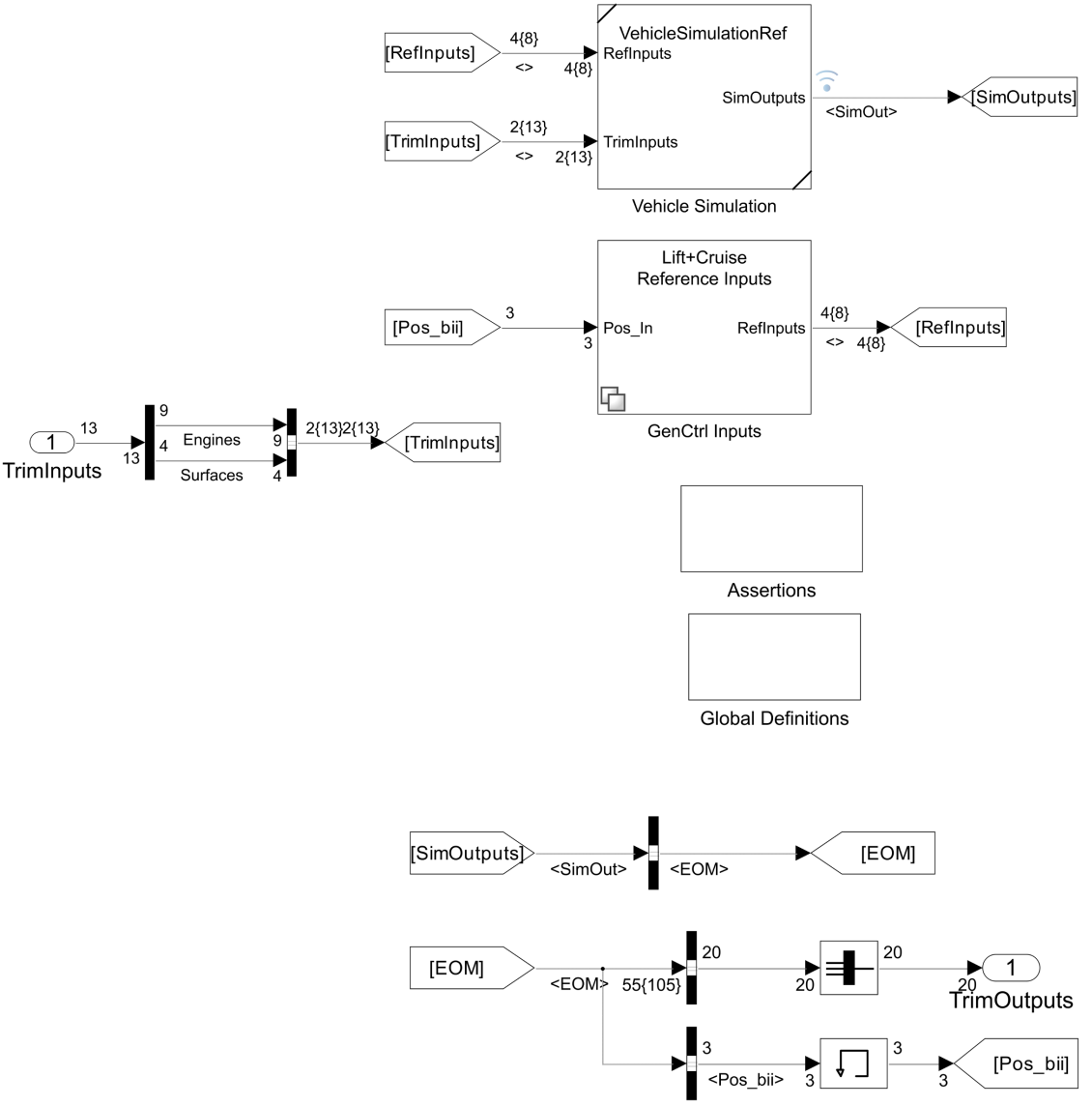

Fig. 1 NASA GUAM Simulation Model (Lift + Cruise configuration) used to generate trajectory data under variable flight and environmental conditions.

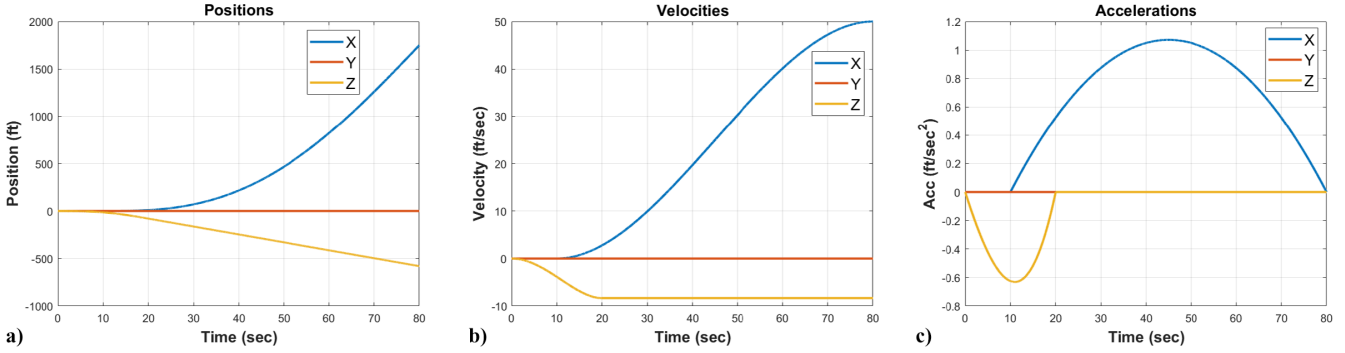

Fig. 2 Representative outputs from GUAM: a) position profiles, b) velocity profiles, and c) acceleration profiles in *X*, *Y*, and *Z* directions. These outputs were used to train and validate the PINN model.

$$L = I_x\dot{p} + I_{xz}(\dot{r} + pq) + (I_z - I_y)qr \tag{4}$$

$$M = I_y\dot{q} + I_{xz}(p^2 - r^2) + (I_x - I_z)pr \tag{5}$$

$$N = I_z\dot{r} + I_{xz}(\dot{p} + qr) + (I_y - I_x)pq \tag{6}$$

where

1) L, M, N represent the body-axis moments (in ft · lb);
2) I_x, I_y, I_z, I_{xz} are the moments of inertia (in slug · ft^2); and
3) p, q, r denote the angular velocity components (in rad/s or deg /s).

These formulations allow the model to account for complex coupled dynamics that are typical in 6-DOF flight models. Integrating these dynamics into the neural network architecture ensures that the learned trajectory predictions adhere to real-world flight physics.

Furthermore, these equations serve as the core of the physics-based loss function, where residuals are computed by comparing the model-predicted derivatives with those inferred from the equations of motion using automatic differentiation. This process ensures that even when direct supervision is limited, the network predictions are still guided by physical principles. As a result, the PINN model can achieve greater fidelity and generalizability in complex, data-sparse UAM environments.

C. PINN Model Architecture

The PINN architecture serves as a hybrid learning framework that integrates data-driven approximations with domain-specific physical knowledge. The architecture is specifically designed to model the dynamic behavior of eVTOL systems by learning to satisfy both data observations and governing physical laws. This dual-objective structure enables improved extrapolation and stability when operating in variable and uncertain flight regimes. Furthermore, the design of the architecture enables natural integration with uncertainty quantification methods. By leveraging ensemble training and probabilistic loss functions, the same network framework can be extended to quantify both epistemic and aleatoric uncertainties in model predictions.

1. *Neural Network Design*

The PINN architecture was selected to balance model complexity with interpretability. It comprises the following:

1) **Input:** Twelve features, including translational and angular state variables

2) **Hidden layers:** Two fully connected layers, each with 128 neurons and ReLU activation

3) **Output:** Twelve predicted states for the next time step

2. *Physics-Based Loss Formulation*

PINNs incorporate physical laws into the learning objective using a hybrid loss function. The total loss combines a data-driven component with a physics-informed component:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{Data}} + \lambda(\mathcal{L}_{\text{Translational}} + \mathcal{L}_{\text{Rotational}}) \tag{7}$$

The data loss is computed using mean-squared error (MSE):

$$\mathcal{L}_{\text{Data}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{8}$$

where y_i is the ground truth and $\hat{y}_i$ is the predicted value.

The translational physics loss enforces consistency with the translational equations of motion:

$$\mathcal{L}_{\text{Translational}} = \frac{1}{N}\sum_{i=1}^{N}\left(\left(\dot{u}_{\text{EoM}}^{(i)} - \dot{u}_{\text{pred}}^{(i)}\right)^2 + \left(\dot{v}_{\text{EoM}}^{(i)} - \dot{v}_{\text{pred}}^{(i)}\right)^2 + \left(\dot{w}_{\text{EoM}}^{(i)} - \dot{w}_{\text{pred}}^{(i)}\right)^2\right) \tag{9}$$

Here, $\dot{u}_{\text{EoM}}, \dot{v}_{\text{EoM}}, \dot{w}_{\text{EoM}}$ are accelerations computed from the equations of motion using the predicted states, while $\dot{u}_{\text{pred}}, \dot{v}_{\text{pred}}, \dot{w}_{\text{pred}}$ are obtained through automatic differentiation of the neural network output.

The rotational physics loss is similarly defined:

$$\mathcal{L}_{\text{Rotational}} = \frac{1}{N}\sum_{i=1}^{N}\left(\left(\dot{p}_{\text{EoM}}^{(i)} - \dot{p}_{\text{pred}}^{(i)}\right)^2 + \left(\dot{q}_{\text{EoM}}^{(i)} - \dot{q}_{\text{pred}}^{(i)}\right)^2 + \left(\dot{r}_{\text{EoM}}^{(i)} - \dot{r}_{\text{pred}}^{(i)}\right)^2\right) \tag{10}$$

Again, $\dot{p}_{\text{EoM}}, \dot{q}_{\text{EoM}}, \dot{r}_{\text{EoM}}$ represent angular accelerations calculated from physical dynamics, and $\dot{p}_{\text{pred}}, \dot{q}_{\text{pred}}, \dot{r}_{\text{pred}}$ are those predicted by the network.

The physics-based loss thus acts as a regularization term, ensuring that the learned trajectories conform to established flight physics, enhancing both interpretability and generalization performance.

3. *Training Configuration*

The training is performed using the Adam optimizer with a learning rate of 1×10^{-5}. Each model is trained for 100 epochs, and loss metrics are recorded every 10 epochs. External forces and moments are initialized from simulation data and updated iteratively through backpropagation.

D. Uncertainty Quantification

Building on the modular PINN architecture, this section outlines how uncertainty quantification is seamlessly incorporated into the framework. To achieve this, the network is extended to capture both epistemic and aleatoric uncertainties through ensemble training and probabilistic output modeling. Epistemic uncertainty refers to the uncertainty in the model parameters that arises from incomplete knowledge or limited training data; it is reducible and can be mitigated by collecting additional data or improving model capacity. In contrast, aleatoric uncertainty represents the inherent randomness and noise present in the system or measurement process; it is irreducible, as it persists even with more data since it reflects intrinsic variability in the environment. In this work, epistemic uncertainty is quantified using an ensemble of independently trained PINNs, while aleatoric uncertainty is modeled through probabilistic output layers that estimate predictive variance. This dual-level uncertainty modeling enables the system to generate accurate trajectory predictions along with meaningful confidence bounds, which are critical for safety-critical and risk-aware UAM decision-making.

1. *Ensemble-Based Epistemic Uncertainty*

To capture epistemic uncertainty, 50 independent PINNs are trained with different random initializations. For each input sample, the prediction variance across the ensemble is calculated as

$$\sigma^2_{\text{epistemic}} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - \bar{y})^2 \tag{11}$$

where

1) N is the number of ensemble members (here, $N = 50$);

2) $\hat{y}_i$ is the predicted output from the ith PINN model in the ensemble; and

3) $\bar{y}$ is the mean of predictions across the ensemble:

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N}\hat{y}_i \tag{12}$$

This variance represents the model's uncertainty due to limited knowledge or data. A higher variance indicates lower confidence in the prediction, which typically arises in regions of the input space where data are sparse or model generalization is weak. By analyzing the spread of outputs across the ensemble, the framework provides a measure of how consistent or uncertain the model is when confronted with new inputs.

2. *Aleatoric Uncertainty via NLL Loss*

Each PINN predicts both the mean and variance of outputs. The training loss incorporates Gaussian negative log-likelihood (NLL) [15], which allows the network to model prediction confidence directly:

$$\mathcal{L}_{\text{NLL}} = \frac{1}{2}\log(\sigma^2) + \frac{(y-\mu)^2}{2\sigma^2} \tag{13}$$

where

1) y is the ground truth value for a given state variable;

2) μ is the predicted mean output of the network; and

3) σ^2 is the predicted variance representing aleatoric uncertainty.

The first term penalizes predictions with underestimated uncertainty (i.e., small σ^2), while the second term penalizes large deviations between predicted means and actual values. This formulation encourages the model to output meaningful uncertainty estimates that reflect observed noise in the data.

The predicted variance σ^2 is modeled as a separate output head of the PINN architecture, enabling the network to learn noise characteristics specific to different parts of the trajectory. Aleatoric uncertainty is particularly important in environments with inherent randomness (e.g., wind turbulence) that cannot be reduced through additional training data.

3. *Combined Uncertainty*

The total uncertainty is given by

$$\sigma_{\text{combined}} = \sqrt{\sigma^2_{\text{epistemic}} + \sigma^2_{\text{aleatoric}}} \tag{14}$$

This enables a more comprehensive risk assessment under real-world variability.

E. Evaluation Metrics

Model performance is assessed using three commonly adopted metrics:

1) **Mean-squared error (MSE):** Penalizes large prediction errors more severely

2) **Root-mean-square error (RMSE):** Provides an interpretable measure of prediction error in original units

3) **Mean absolute error (MAE):** Averages absolute deviations, offering robustness to outliers

Each metric is applied across predicted trajectory sequences and compared against the ground truth.

F. Summary

The proposed methodology tightly integrates physics-based modeling and data-driven learning using PINNs. By embedding physical constraints, implementing a custom hybrid loss function, and employing ensemble learning for uncertainty quantification, the approach addresses both accuracy and reliability requirements for safety-critical UAM applications. It is validated on a high-fidelity simulation dataset and designed to extend to real-time UAM trajectory prediction in future work.

III. Results

This section presents the results of the PINN framework across several key dimensions, including trajectory prediction accuracy, uncertainty quantification, training dynamics, and model comparison. Each subsection addresses specific aspects of model performance to highlight the benefits of physics-informed learning in UAM trajectory forecasting.

A. PINN Performance on Trajectory Prediction

The PINN framework exhibits strong predictive accuracy in modeling the flight trajectories of eVTOL vehicles. By integrating the governing physical equations directly into the neural network architecture, the model is able to capture not only the observable patterns from training data but also the underlying dynamics of flight motion.

Figure 3 shows the predicted and actual flight trajectories in three-dimensional space. The close alignment of these paths highlights the model's ability to accurately learn and generalize trajectory dynamics across a variety of simulated urban air mobility conditions. Minimal deviation is observed throughout, even during complex flight segments.

To better understand the model's temporal accuracy, Fig. 4 presents time-series predictions of the *X, Y,* and *Z* position components compared to their ground truth values. The results show that the model maintains high fidelity in all three dimensions over time. In particular, the *Z*-axis prediction, often more susceptible to noise and turbulence, remains tightly coupled with the true values.

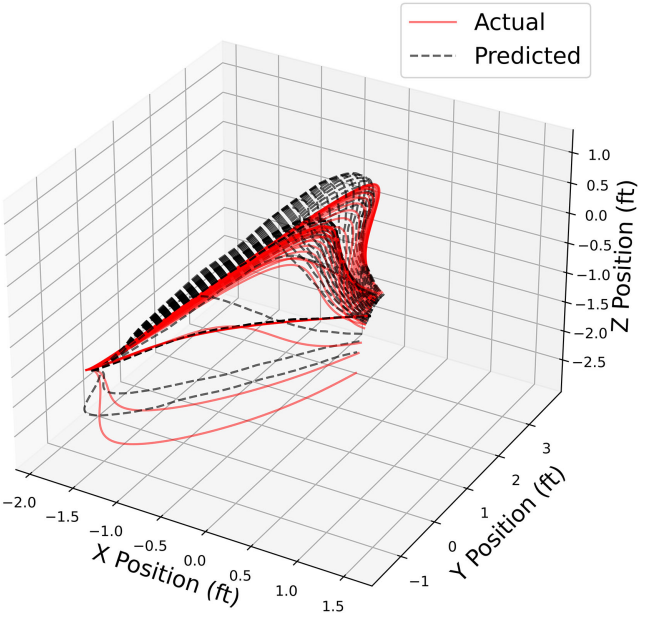

Fig. 3 Three-dimensional trajectory comparison: actual vs predicted flight path using the PINN model.

These results underscore the model's robustness in real-time tracking scenarios. Unlike purely data-driven models, PINNs preserve physically feasible trajectories, offering not only prediction accuracy but also physical plausibility, a critical requirement for trajectory management in UAM systems.

B. Uncertainty Quantification and Interpretation

A key advantage of the PINN framework is its ability to quantify uncertainty in trajectory predictions. This section presents three forms of uncertainty assessment: epistemic, aleatoric, and their combined effect. Visualizations of these uncertainties across the *X, Y,* and *Z* trajectory components reveal how the model captures prediction confidence under different conditions.

1. *Epistemic Uncertainty*

Epistemic uncertainty reflects the model's confidence due to limited data or insufficient generalization. In this study, it is estimated through an ensemble of 50 PINN models, each trained with different initializations. The randomization was applied to the network weights with different random seeds, leading to diverse convergence paths and estimates. These ensembles are independently trained networks, which allows quantification of epistemic uncertainty by capturing variability in model predictions that arises from limited data and model parameter uncertainty. Figure 5 illustrates the predicted trajectories with epistemic uncertainty bounds for the *X*, *Y,* and *Z* components.

Wider confidence intervals appear during maneuvering and high-dynamic transitions, such as turns or rapid altitude changes. Narrower bands are observed during steady flight segments, indicating that the model becomes more confident when conditions are predictable and well represented in the training data.

2. *Aleatoric Uncertainty*

Aleatoric uncertainty arises from inherent noise in the system, such as wind variability, turbulence, and sensor limitations, and cannot be reduced by adding more data. It is captured by enabling the PINN to predict both the mean and variance of each state using the NLL loss. Figure 6 shows the aleatoric uncertainty visualized along each axis.

These uncertainty bands appear more uniform than their epistemic counterparts and increase slightly during rapid changes,

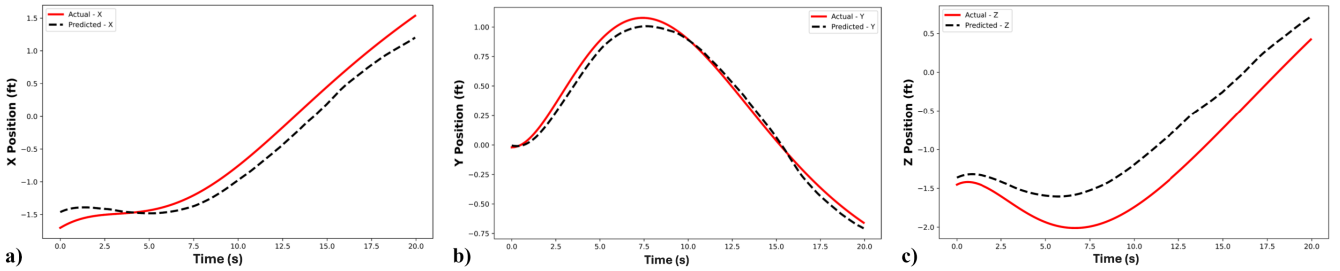

Fig. 4 Actual vs predicted positions over time: a) *X*-axis, b) *Y*-axis, and c) *Z*-axis.

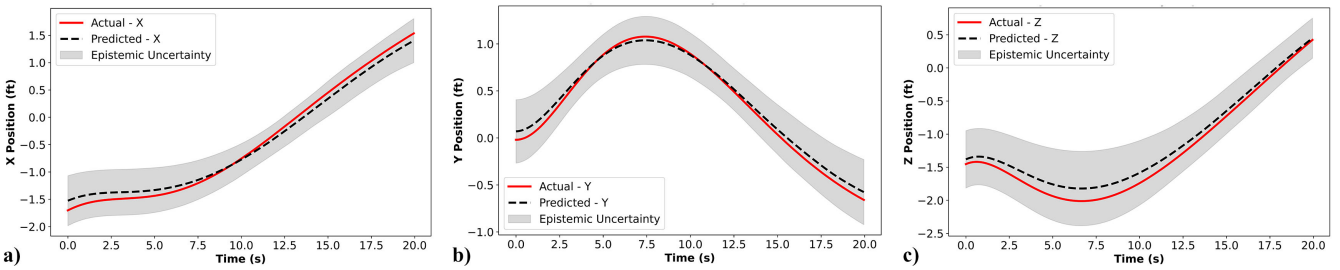

Fig. 5 Epistemic uncertainty across a) *X*, b) *Y*, and c) *Z* positions with 95% confidence intervals.

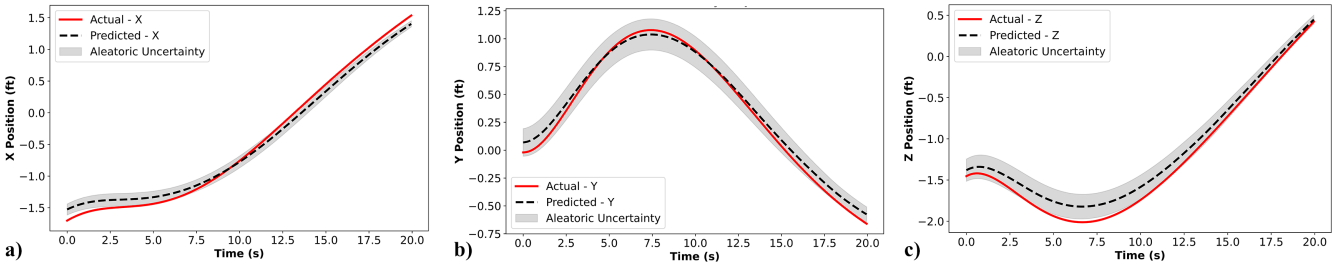

Fig. 6 Aleatoric uncertainty for a) *X*, b) *Y*, and c) *Z* positions with predicted variance bands.

reflecting the model's estimation of inherent environmental unpredictability.

3. *Combined Uncertainty*

The final uncertainty estimate combines both epistemic and aleatoric components, yielding an integrated confidence interval. This provides a more holistic view of the model's prediction reliability. Figure 7 shows these total uncertainty bounds for each spatial axis.

The combined uncertainty intervals widen and contract dynamically based on the flight phase. These adaptive bounds can be used by flight controllers and UAM systems to make proactive decisions, such as increasing safety margins during critical phases or executing corrective actions during unexpected deviations.

This capability to quantify uncertainty is critical for building trust in AI-driven flight systems. It provides both engineers and operators with interpretable metrics to assess prediction reliability and risk in real time.

C. Training and Loss Analysis

To evaluate the training dynamics of the PINN framework, we conducted a series of experiments with varying dataset sizes, learning rates, network depths, and ensemble sizes. The goal was to understand how these hyperparameters influence the convergence of both data-driven and physics-based loss components.

Table 1 summarizes the results of selected initial runs. Smaller datasets (e.g., 50 samples) with limited model capacity (64 nodes)

Table 1 Initial training runs and observations

Dataset size	Epochs	LR	Nodes	Models	ITL	FTL
50	50	0.001	64	5	188.73	7.29
50	50	0.005	64	5	97.72	1.98
1000	50	0.001	64	10	189.38	9.21
1000	50	0.001	128	50	175.01	4.02

LR = learning rate, ITL = initial total loss, FTL = final total loss.

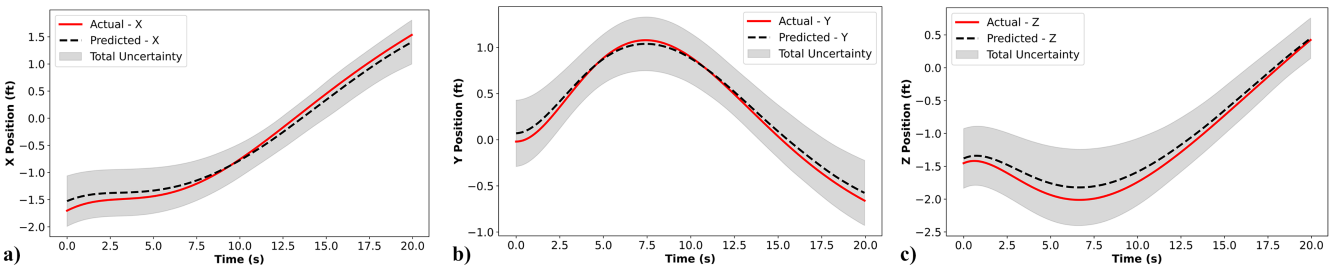

Fig. 7 Combined uncertainty in *X*, *Y*, and *Z* positions. Confidence bounds reflect total predictive uncertainty.

led to poor generalization and high total loss. Although the MSE loss remained relatively low, the physics-based loss stayed elevated, indicating difficulty in satisfying physical constraints under limited data conditions.

The learning rate (LR) controls how quickly the network updates its weights during training. Initial total loss (ITL) refers to the total loss at epoch 0, while final total loss (FTL) reflects the total loss after completing all training epochs. These values help quantify both the starting point and the convergence behavior of the model across configurations. Adjusting the learning rate from 0.001 to 0.005 significantly improved convergence speed. However, higher learning rates also introduced instability in the early training stages, which was mitigated by increasing model depth and training epochs.

Figure 8 compares the training loss trajectories of an initial run and the final optimized configuration. In the initial setup, the physics-based loss exhibited sharp fluctuations, indicating limited learning of physical dynamics. In contrast, the optimized configuration shows smooth and consistent convergence of both the physics and data loss terms.

The best-performing configuration (summarized in Table 2) used a dataset of 1000 samples, 128 hidden units, a learning rate of 0.005, and an ensemble of 50 models. This setup yielded the lowest total loss (1.4851) and offered a stable learning curve across training epochs.

Notably, the physics-based loss played a crucial role in model regularization. It prevented overfitting to noisy data and guided the model to learn feasible state transitions, particularly in high-velocity and nonlinear segments of the trajectory. These observations underscore the importance of properly balancing data and physics loss weights and optimizing the model's architecture to capture both learning performance and physical interpretability.

D. Error Gradient and Local Performance Insights

To better understand how the model performs over time and across different spatial regions, we conducted an error gradient analysis on the predicted trajectories. This technique highlights the magnitude of absolute error between predicted and true values at each time step and visualizes it as a color gradient.

The error is computed as

$$\text{Error}(t) = |\text{True Position}(t) - \text{Predicted Position}(t)| \tag{15}$$

For each spatial axis (X, Y, Z), the absolute error was calculated at every time step. These errors were then mapped onto the trajectory using a color scale, ranging from blue (low error) to red (high error), allowing for an intuitive assessment of local prediction performance.

Figure 9 shows the error gradient visualizations for the X, Y, and Z components. Overall, the model exhibits low error across most of the trajectory, especially in steady-state flight phases. However, higher errors are observed during dynamic transitions, such as sharp turns, rapid ascents/descents, and hover-to-cruise transitions.

Table 2 Optimal training configuration

Parameter	Value
Dataset size	1000
Epochs	100
Learning rate	0.005
Nodes	128
Models	50
Total loss	1.4851

This fine-grained analysis helps identify where the model may need additional support, either through training on more diverse data, refining physical constraints, or using adaptive loss weighting. For example, consistently elevated error zones suggest that those phases of flight (e.g., transitions) are underrepresented in the training data or involve dynamics that are more difficult to capture with the current model configuration.

In practical deployment, this kind of spatial error insight can inform strategies for selective model refinement or real-time alerting, particularly in mission-critical phases such as landing approach or dense urban navigation.

E. Baseline Model Comparison

To assess the advantages of the proposed PINN framework, we benchmarked its performance against two baseline methods:

1) **Gaussian mixture regression:** A probabilistic regression model that fits a Gaussian mixture model (GMM) to the training data and infers predictions based on weighted conditional expectations. While computationally efficient and capable of capturing smooth trends, GMR does not incorporate physical knowledge and struggles with generalization under dynamic conditions.

2) **Standard neural networks:** A conventional feedforward neural network trained purely on data using a mean-squared error (MSE) loss. Although flexible and expressive, NNs are sensitive to data distribution and lack built-in mechanisms for enforcing physical consistency or quantifying uncertainty.

3) **Physics-informed neural networks:** The proposed model, which embeds physical equations of motion directly into the training objective, allowing it to produce physically feasible predictions with interpretable confidence estimates.

All models were trained on the same dataset of 1000 trajectories with identical preprocessing and input–output structures. Performance was evaluated using standard regression metrics.

As shown in Table 3, the PINN model achieved the lowest error values across all metrics, with an MSE that is an order of magnitude lower than the NN model and nearly two orders of magnitude lower than GMR. This substantial improvement demonstrates the value of incorporating physical laws into the learning process, particularly in complex and uncertain environments where purely data-driven models tend to fail.

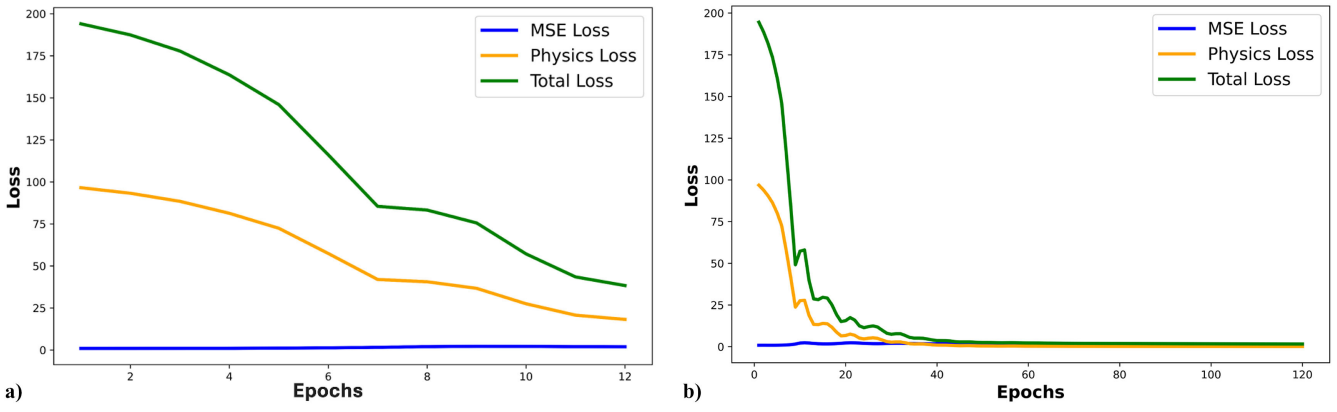

Fig. 8 Loss evolution: a) initial training run and b) final optimized configuration.

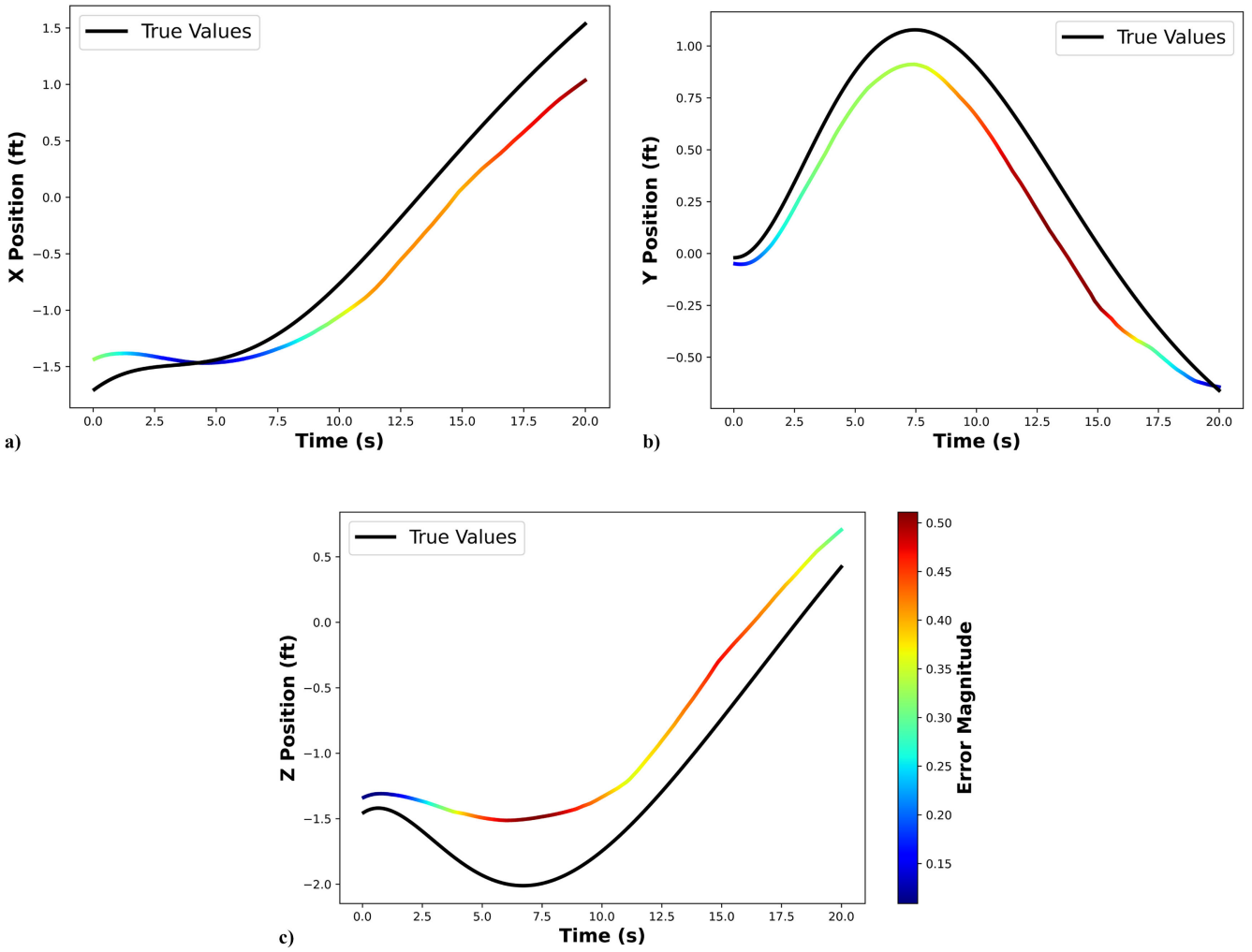

Fig. 9 Error gradient visualization of position predictions: a) *X*-axis, b) *Y*-axis, and c) *Z*-axis. Color intensity indicates magnitude of absolute error (blue = low; red = high).

Table 3 Model comparison metrics[a]

Metric	GMR	NNs	PINNs
MSE	0.1525	0.0330	**0.0005**
RMSE	0.3905	0.1816	**0.0216**
MAE	0.3282	0.1536	**0.0153**

[a]Lower values indicate better performance.

To better understand the model's prediction behavior, we visualized the predicted trajectories along the *X*, *Y*, and *Z* axes. Figure 10 shows predictions from NNs, GMR, and PINNs compared to ground truth trajectories. While both baseline models (NNs and GMR) capture overall trends, they exhibit visible deviations during high-dynamic phases such as rapid ascent or turning.

In contrast, PINN predictions (as shown in Fig. 10) closely follow the actual trajectories across all flight phases. The model's ability to retain physical realism, especially during nonlinear transitions, results in smoother, more accurate paths with fewer deviations or spikes. Unlike NNs, which do not quantify uncertainty, and GMR, which offers probabilistic estimates but lacks physical grounding, the PINN framework delivers uncertainty-aware predictions anchored in physical principles. This capability is particularly critical in urban air mobility scenarios, where prediction reliability, safety, and operational trust are paramount.

The PINN model also outperformed GMR and NNs in both accuracy and robustness, particularly in scenarios involving complex aerodynamic behaviors or environmental variability. Its advantage lies in the integration of physics-based constraints, which steer learning toward dynamically feasible solutions and help reduce overfitting to noise or limited data. A key strength of PINNs is their

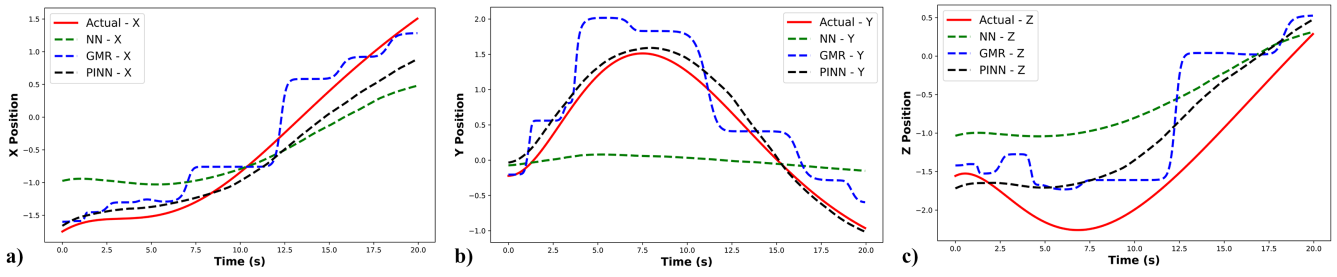

Fig. 10 Actual versus predicted (NN, GMR, PINN) across *X*, *Y*, and *Z* axes. Ground truth is shown in solid lines; predictions are shown in dashed lines.

ability to generate interpretable uncertainty estimates. By modeling both aleatoric and epistemic uncertainty within a physically consistent framework, they enable confidence-aware decision-making for UAM mission planning and risk mitigation.

Furthermore, PINNs demonstrated strong adaptability in high-variability conditions, such as fluctuating wind speeds or abrupt maneuvering. While NN and GMR models struggled to generalize under such dynamics, the physics-informed nature of PINNs allowed them to maintain reliable performance and smooth trajectory tracking. Despite their added complexity, the computational efficiency of PINNs remained competitive, offering a balanced tradeoff between accuracy, interpretability, and runtime feasibility.

IV. Conclusions

This work presents a PINN framework for trajectory prediction and uncertainty quantification in UAM. By embedding physical laws into the learning process, the model achieves superior accuracy and robustness compared to traditional methods such as Gaussian mixture regression and standard neural networks. The integration of epistemic and aleatoric uncertainty modeling further enhances the framework's reliability in dynamic and uncertain environments. The PINN approach not only ensures physically consistent predictions but also enables interpretable confidence estimation, making it well suited for safety-critical UAM applications. Future extensions include incorporating more diverse datasets, enabling multi-agent coordination, and integrating with control systems.

Acknowledgment

The authors would like to acknowledge the support from the National Science Foundation under Grants CCF-2402689 and CCF-2523829. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors and do not reflect the views of the National Science Foundation.

References

[1] Liu, Y., and Li, X. R., "Intent Based Trajectory Prediction by Multiple Model Prediction and Smoothing," *AIAA Guidance, Navigation, and Control Conference*, AIAA Paper 2015-1324, 2015. https://doi.org/10.2514/6.2015-1324

[2] Pang, Y., Yao, H., Hu, J., and Liu, Y., "A Recurrent Neural Network Approach for Aircraft Trajectory Prediction with Weather Features," *AIAA SciTech Forum*, AIAA Paper 2019-3413, 2019. https://doi.org/10.2514/6.2019-3413

[3] Corbetta, M., Banerjee, P., Okolo, W. A., Gorospe, G. E., and Luchinsky, D. G., "Real-Time UAV Trajectory Prediction for Safety Monitoring in Low-Altitude Airspace," *AIAA Aviation Forum 2019*, AIAA Paper 2019-3514, 2019. https://doi.org/10.2514/6.2019-3514

[4] Barratt, S. T., Kochenderfer, M. J., and Boyd, S. P., "Learning Probabilistic Trajectory Models of Aircraft in Terminal Airspace from Position Data," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, No. 9, 2021, pp. 3536–3545. https://doi.org/10.1109/TITS.2018.2877572

[5] Eren, U., Prach, A., Koucer, B. B., Rakovic, S. V., Kayacan, E., and Accikmecse, B., "Model Predictive Control in Aerospace Systems: Current State and Opportunities," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 7, 2017, pp. 1541–1566. https://doi.org/10.2514/1.G002507

[6] Zheng, X., Li, X., Zhang, D., and Yang, N., "Multi-Objective UAV Trajectory Planning in Uncertain Environment," *Symmetry*, Vol. 13, No. 12, 2021, p. 2160. https://doi.org/10.3390/sym13112160

[7] Raissi, M., Perdikaris, P., and Karniadakis, G., "Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations," *Journal of Computational Physics*, Vol. 378, Feb. 2019, pp. 686–707. https://doi.org/10.1016/j.jcp.2018.10.045

[8] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics Informed Deep Learning (Part II): Data-Driven Discovery of Nonlinear Partial Differential Equations," arXiv preprint, 2017. https://doi.org/10.48550/arXiv.1711.10566

[9] Michek, N. E., Mehta, P., and Huebsch, W. W., "Flight Dynamic Uncertainty Quantification Modeling Using Physics-Informed Neural Network," *AIAA Journal*, Vol. 62, No. 11, 2024, pp. 4234–4246. https://doi.org/10.2514/1.J063992

[10] Kunz, J., Layer-Reiss, S., Holzapfel, F., and Bittner, M., "Development of eVTOL Scenarios for Trajectory Generation," *AIAA Aviation Forum 2024*, AIAA Paper 2024-3855, 2024. https://doi.org/10.2514/6.2024-3855

[11] Stachiw, T., Crain, A., and Ricciardi, J., "A Physics-Based Neural Network for Flight Dynamics Modelling and Simulation," *Advanced Modeling and Simulation in Engineering Sciences*, Vol. 9, No. 13, 2022. https://doi.org/10.1186/s40323-022-00227-7

[12] Abdar, M., Pourpanah, F., Hussain, S., Rezazadegan, D., Liu, L., Ghavamzadeh, M., Fieguth, P., Cao, X., Khosravi, A., Acharya, U. R., et al., "A Review of Uncertainty Quantification in Deep Learning: Techniques, Applications and Challenges," Vol. 9, Inst. of Electrical and Electronics Engineers, New York, 2021, pp. 243–297. https://doi.org/10.1016/j.inffus.2021.05.008

[13] Gregory, I. M., Acheson, M. J., Bacon, B. J., and Britton, T. C., "Intelligent Contingency Management for Urban Air Mobility," AIAA SciTech Forum 2021, AIAA Paper 2021-1000, 2021. https://doi.org/10.2514/6.2021-1000

[14] Oo, W. K. K., Tun, H. M., Naing, Z. M., and Moe, W. K., "Design of Vertical Take-Off and Landing (VTOL) Aircraft System," *International Journal of Scientific & Technology Research*, Vol. 6, No. 4, 2017, https://www.ijstr.org/final-print/apr2017/Design-Of-Vertical-Take-off-And-Landing-vtol-Aircraft-System.pdf.

[15] Nix, D. A., and Weigend, A. S., "Estimating the Mean and Variance of the Target Probability Distribution," *Proceedings of the 1994 IEEE International Conference on Neural Networks (ICNN'94)*, Inst. of Electrical and Electronics Engineers, New York, 1994, pp. 55–60. https://doi.org/10.1109/ICNN.1994.374138

P. Wei
Associate Editor