



LEARNING-BASED FULL-ENVELOPE AUTOPILOT DESIGN BASED ON THE KOOPMAN OPERATOR THEORY

Youngjun Lee¹, Jinrae Kim¹ & Youdan Kim¹

¹Seoul National University

Abstract

A data-driven methodology based on Koopman operator theory is proposed for the design of a missile autopilot. The proposed approach uses artificial neural networks to approximate the Koopman embedding function, which constructs an approximate linear time-varying model of missile dynamics including nonlinear characteristics. The approximate dynamic equations are trained using collected trajectory data without any prior knowledge of the missile dynamics. The data-driven model makes it possible to apply the Robust Servomechanism Linear Quadratic Regulator (RSLQR) to determine the gains of the feedback controller for a given flight condition. This feature allows designers to generate a continuous gain table for various operating points by selecting the weighting matrices of the cost function. Numerical simulation results show that the proposed method captures a larger portion of the missile's state space compared to conventional linear models. The extended state space coverage allows the designed controller to outperform traditional controllers based on linearized models. Additionally, a comparative evaluation of the designed controller against a classical three-loop controller, which demonstrates its effective full-envelope coverage.

Keywords: Missile Autopilot Design, Koopman Operator Theory, Linear Optimal Control, Artificial Neural Network

1. Introduction

Modern tactical missiles are required to operate in a variety of operating conditions to obtain high maneuverability. Achieving a rapid response and ensuring robustness are critical aspects of autopilot design especially in the environment that uncertainties and disturbances exist. For the complex dynamic operational environments including threats, missile autopilot techniques have been continuously developed.

Traditionally, autopilot designs have relied on conventional model-based approaches, often including linearization of missile dynamics around nominal operating points. To extend the applicability of these controllers across a broad range of operating points, researchers have proposed various gain-scheduling techniques within the three-loop structure leveraging the advanced theories, such as the eigenstructure assignment [1], constrained optimization [2], and mixed H_2/H_∞ with linear parameter-varying model [3]. Recent advancements have observed a shift towards nonlinear control methodologies, capitalizing on their inherent ability to handle strong nonlinear characteristics. Penchal et al. used the feedback linearization approach and the extended state observer to track acceleration command [4, 5, 6]. Despite these improvements in control theory, gain scheduling remains a prevalent and practical approach for missile autopilot design, owing to its simplicity and interpretability. However, gain scheduling technique has certain drawbacks. The performance of the designed controller is significantly influenced by the granularity of the gain table, and the widely used linear interpolation technique may introduce unexpected instability. Moreover, the process of creating the gain table is accompanied by lots of tedious, time-consuming, and error-prone tasks. Consequently, there is a growing tendency to develop alternative methodologies that overcome the limitations inherent in conventional model-based approaches.

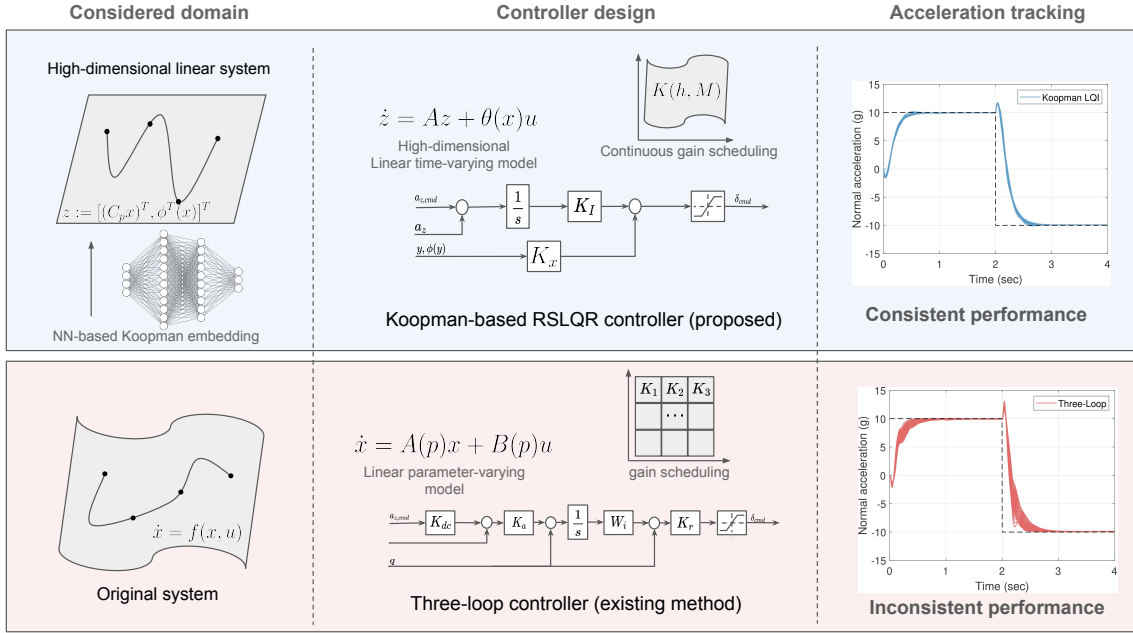


Figure 1 – The proposed Koopman-based controller vs traditional three-loop controller. The proposed method utilizes an NN-based Koopman embedding to find a high-dimensional linear system by *unfolding* nonlinearity. In the high-dimensional space, a Robust Servomechanism Linear Quadratic Regulator (RSLQR) controller can be designed with a single controller gain. The resulting control performance is consistent, while the existing method is not due to the tedious gain scheduling.

In recent years, one methodology has gained attention, which is rooted in the concepts of the Koopman operator theory. Originating in the field of dynamic systems and data-driven modeling, the Koopman operator theory offers an alternative perspective on system analysis. One of the primary challenges lies in identifying the suitable observable functions capable of *lifting* the original nonlinear systems to a higher-dimensional linear system. Williams et al. introduced the extended dynamic mode decomposition [7], which is a data-driven approximation of dynamical systems. Furthermore, leveraging artificial neural networks to approximate the Koopman embedding function reduces the need for selecting effective a priori basis functions. Champion et al. [8] and Lucsh et al. [9] demonstrated the applicability of using deep neural networks by designing custom deep auto-encoders for the sparse identification of nonlinear dynamics, while still retaining the physical interpretability of Koopman embeddings. Numerous studies have also been conducted to extend the Koopman operator theory so that it could express controlled systems. It has been demonstrated that the general nonlinear control systems of interest can be effectively represented as bilinear systems [10]. This idea has led researchers to design control laws using the data-driven bilinear system, incorporating techniques such as optimal quadratic regulator [11] and model predictive control [12].

This paper focuses on integrating the Koopman operator theory into missile autopilot design. The proposed approach is summarized in Figure 1. Employing a data-driven approach, a linear high-dimensional dynamical system, distinct from locally linearized ones, is obtained. Subsequently, the Robust Servomechanism Linear Quadratic Regulator (RSLQR) controller is designed based on the identified model to achieve tracking ability and incorporate integral terms for expected robustness. Note that it is possible to schedule gains continuously over the operating region because the embedding state vector includes variables such as angle of attack, Mach number, and altitude, which represents essential parameters throughout the air-to-air missile's operational envelope. The performance of the designed controller is examined at various operating points, comparing with a traditional three-loop controller.

The remainder of this paper is structured as follows: In Section 2, the background of the Koopman operator theory is introduced, and the equations of motion for missile longitudinal model are provided.

Section 3 offers an overview of the proposed method, presenting sequential summaries of data collection methods, Koopman embedding training techniques, and controller design procedures. Section 4 presents the results of training, model excitation, and controller testing. Section 5 concludes this study with remarks and a discussion on future research.

2. Problem Formulation

This section presents the basic theoretical background and the missile dynamic equations to be controlled. First, the fundamental principles of Koopman operator theory and the Koopman Observer Form (KOF) are briefly summarized. Second, the governing equations of nonlinear longitudinal missile dynamics, along with the associated measurement models, are described in detail.

2.1 Koopman operator theory

Consider a general nonlinear autonomous system,

$$\dot{x}(t) = f(x(t)), \quad (1)$$

where $x(t) \in X$ is the state of the autonomous system, defined in the state space $X \subset \mathbb{R}^n$, at time $t \in \mathbb{R}_+$, and $f : X \rightarrow \mathbb{R}^n$ is a continuously differentiable vector field. Let $\Phi(t, x)$ be the solution of (1) at the initial condition $x_0 \in X$. The continuous-time Koopman operator $\mathcal{K}^t : \mathcal{F} \rightarrow \mathcal{F}$ associated with the flow map $\Phi(t, x)$ is defined by

$$(\mathcal{K}^t \phi)(x) := \phi \circ \Phi(t, x), \quad (2)$$

where $\mathcal{F}$ is the space of all complex-valued observable functions $\phi : X \rightarrow \mathbb{C}$, and $\circ$ represents function composition. An eigenfunction $\psi \in \mathcal{F}$ of the Koopman operator is an observable function that satisfies:

$$(\mathcal{K}^t \psi)(x) = e^{\lambda t} \psi(x), \quad (3)$$

where $\lambda \in \mathbb{C}$ denotes the corresponding eigenvalue [13].

In Koopman operator theory, a transformation $z = T(x)$ may show a new representation of the nonlinear dynamics in (1) such that the new dynamics impose linearity as follows [14],

$$\dot{z} = Kz, \quad (4)$$

which implies that finding such eigenfunctions (so called *Koopman embedding*) would result in high-dimensional linear dynamics by *unfolding* the nonlinearity of the original dynamic system (1) [14].

2.2 Transformation into the Koopman Observer Form

Consider a control-affine system represented as

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= c(x), \end{aligned} \quad (5)$$

where $u \in U \subset \mathbb{R}^m$ and $y \in Y \subset \mathbb{R}^o$ are the control input and output of the system, respectively. The $f : X \rightarrow \mathbb{R}^n$ and $g : X \rightarrow \mathbb{R}^{n \times m}$ are the continuously differentiable drift and control effective vector fields, and $c : X \rightarrow \mathbb{R}^o$ is the output function. Let us define a transformation function $T(x) := [\hat{\phi}_1(x), \dots, \hat{\phi}_N(x)]^\top \in \mathbb{R}^N$ such that the following two statements hold:

1. $\hat{\phi}_i$ is equal to the ϕ_i if ϕ_i is a real-valued function.
2. $\hat{\phi}_i = 2\text{Re}(\phi_i)$ and $\hat{\phi}_{i+1} = -2\text{Im}(\phi_i)$ if ϕ_i and ϕ_{i+1} are complex conjugate pairs.

where $i = 1, \dots, N$ and $\phi_i(x) \in \mathcal{F}$ are the N eigenfunctions of the drift vector field $f(x)$. Note that $z = T(x)$ can be represented as the linear combination of finite subsets of Koopman eigenfunctions. There exists a matrix Λ that satisfies the following relationship:

$$\mathcal{L}_f T(x) = \Lambda T(x) = \Lambda z, \quad (6)$$

where $\mathcal{L}_f$ is Lie derivative with respect to f and Λ is a real-valued block-diagonal matrix such that:

1. $\Lambda_{i,i} = \lambda_i$ if i -th eigenvalue is real.
2. Λ has a block diagonal entry Q_i if i -th and $(i+1)$ -th eigenvalues are complex conjugate pairs.
3. All the rest entries are zeros.

Here,

$$Q_i = |\lambda_i| \begin{bmatrix} \cos \angle \lambda_i & \sin \angle \lambda_i \\ -\sin \angle \lambda_i & \cos \angle \lambda_i \end{bmatrix}, \quad (7)$$

where $|\lambda_i|$ and $\angle \lambda_i$ are the modulus and argument of the i -th complex eigenvalue λ_i .

Assumption 1 ([15]). *There exist matrices $H \in \mathbb{R}^{n \times N}$ and $C \in \mathbb{R}^{o \times N}$ such that H and C reconstruct the original state x and output y , respectively, from the transformation $z = T(x)$,*

$$\begin{aligned} x &= Hz, \\ y &= Cz. \end{aligned} \quad (8)$$

From (5), (6), and (8), the coordinate transformation from x to z is developed as

$$\dot{z} = \frac{\partial T(x)}{\partial x} (f(x) + g(x)u) \quad (9)$$

$$= \mathcal{L}_f T(x) + \sum_{i=1}^m (\mathcal{L}_{g_i} T(x)) u_i \quad (10)$$

$$= \Lambda z + \sum_{i=1}^l \tilde{g}_i(z) u_i \quad (11)$$

where $\tilde{g}_i(z) = \mathcal{L}_{g_i} T(x)|_{x=Hz}$, which is called Koopman Canonical Transform (KCT) [15]. Thus, the system (5) can be transformed into the following equivalent form called the Koopman observer form:

$$\begin{aligned} \dot{z} &= \Lambda z + \sum_{i=1}^m \tilde{g}_i(z) u_i, \\ x &= Hz, \\ y &= Cz. \end{aligned} \quad (12)$$

Note that (12) is a structure of the approximate model, and its parameters can be determined by learning with given trajectory data.

2.3 Missile longitudinal dynamics

The dynamic equations of a missile to be controlled are given in this subsection. In this study, the missile longitudinal dynamics model is considered on the flat Earth assumption [16]. The equations of motion for the missile longitudinal dynamics can be written as

$$\dot{u} = -qw + \sum F_{B_x}/m, \quad (13)$$

$$\dot{w} = qu + \sum F_{B_z}/m, \quad (14)$$

$$\dot{q} = \sum M_Y/I_Y, \quad (15)$$

$$\dot{\theta} = q, \quad (16)$$

$$\dot{h} = u \sin \theta - w \cos \theta, \quad (17)$$

where $u \in \mathbb{R}$ and $w \in \mathbb{R}$ are velocity with respect to x - and z - axes of the body-fixed frame, respectively, $\theta \in [-\pi/2, \pi/2]$ is the pitch angle, and $q \in \mathbb{R}$ is the pitch rate, and $h \in \mathbb{R}$ denotes the altitude. Figure 2 shows the longitudinal missile model considered in this study. The forces and moments applied to

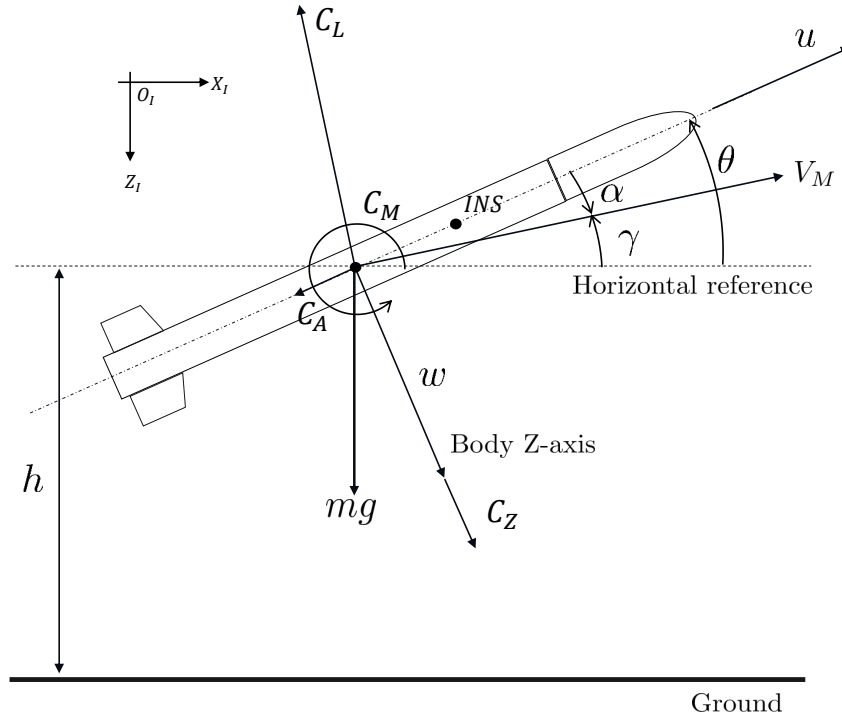


Figure 2 – Missile longitudinal dynamics

the center of gravity are expressed as

$$\sum F_{B_x} = -F_A - mg \sin \theta, \quad (18)$$

$$\sum F_{B_z} = F_Z + mg \cos \theta, \quad (19)$$

$$\sum M_Y = M, \quad (20)$$

where the axial force F_A , body z -axis force F_Z , and pitching moment M are defined as follows,

$$F_A = \frac{1}{2} \rho V^2 S C_A, \quad (21)$$

$$F_Z = \frac{1}{2} \rho V^2 S C_Z, \quad (22)$$

$$M = \frac{1}{2} \rho V^2 S C_M. \quad (23)$$

The aerodynamic coefficients C_A , C_N , and C_M are modelled as nonlinear functions of the angle of attack α and control surface deflection δ as follows,

$$C_A = a_a, \quad (24)$$

$$C_Z = a_z \alpha^3 + b_z \alpha |\alpha| + c_z \left(2 - \frac{M_a}{3} \right) \alpha + d_z \delta, \quad (25)$$

$$C_M = a_m \alpha^3 + b_m \alpha |\alpha| + c_m \left(-7 + \frac{8M_a}{3} \right) \alpha + d_m \delta + e_m q. \quad (26)$$

where $M_a \in \mathbb{R}$ and $\delta \in \mathbb{R}$ denote the Mach number and tail-fin deflection angle, respectively. The aerodynamic coefficients are summarized in Table 1. The second-order actuator dynamics is considered for the missile system as follows,

$$\ddot{\delta} + 2\zeta \omega_n \dot{\delta} + \omega_n^2 \delta = \omega_n^2 \delta_c, \quad (27)$$

Symbols	Values	Symbols	Values
a_z	19.373	a_m	40.44
b_z	-31.023	b_m	-64.015
c_z	-9.717	c_m	2.922
d_z	-1.948	d_m	-11.803
a_a	-0.3	e_m	-1.719

Table 1 – Aerodynamic coefficients [16]

where $\delta_c \in \mathbb{R}$ is the actuator command. The damping ratio ζ is 0.7, and the natural frequency ω_n is 150 rad/s. For more details on the missile dynamics, the readers are referred to [16].

The overall missile longitudinal dynamics can be written as

$$\dot{s} = f_s(s, u) = f(s) + g(s)u, \quad (28)$$

$$y = c(s), \quad (29)$$

where $s = [u, w, q, \theta, h, \delta, \dot{\delta}]^\top \in \mathbb{R}^n$, $u = \delta_c \in \mathbb{R}^m$, and $y_s = [a_z, q, \delta]^\top \in \mathbb{R}^o$ are the state, control input, and output, respectively, at time t . The $f(s)$, $g(s)$, and $c(s)$ are represented as

$$f(s) = \begin{bmatrix} -qw - \frac{\rho V^2 S}{2m} C_A - g \sin \theta \\ qu + \frac{\rho V^2 S}{2m} C_Z + g \cos \theta \\ \frac{\rho V^2 S d}{2I_y} C_M \\ q \\ u \sin \theta - w \cos \theta \\ \dot{\delta} \\ -2\zeta \omega_n \dot{\delta} - \omega_n^2 \delta \end{bmatrix}, \quad g(s) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \omega_n^2 \end{bmatrix}, \quad c(s) = \begin{bmatrix} \frac{\rho V^2 S C_Z}{2m} & q & \delta \end{bmatrix}^\top, \quad (30)$$

Note that the accelerometer cannot measure acceleration excited by gravity, and therefore the gravitational acceleration does not appear in the output function $c(s)$ in (30). The missile longitudinal system (28) is constructed to simulate the behavior of actual missile motion as closely as possible, a flight simulator for collecting data and testing the designed controllers.

3. Autopilot Design

In this section, the design procedure for a normal acceleration tracking controller is proposed based on Koopman operator theory. The primary objective is to achieve precise tracking control based only on the sequence of states and measurements. The proposed approach seeks to establish a high-dimensional linear dynamic model capable of encompassing a broader range of state regions than the conventional linearization-based models.

To address this, artificial neural networks are utilized to approximate the Koopman embedding function, effectively yielding a reconstructed linear model, thereby, optimal linear control theory can be readily applied to the reconstructed model. Section 3.1 provides the details on dataset generation for training. Section 3.2 investigates the training specifics including model structure, the loss function, and the training algorithm. Section 3.3 outlines the procedure for designing the controller using the identified model. Figure 3 shows the illustration of the proposed procedure.

3.1 Data collection for Koopman identification

Obtaining the analytical solution for the embedding dynamical model is not an easy problem. In this study, to address this problem, a data-driven approach is adopted, which involves the collection of a sequence of input and output data for training a neural network that approximates the model. To construct the training data for Koopman identification, the longitudinal dynamic equations (30), are utilized. Considering the operating regions of a conventional air-to-air missile, state and control input are uniformly randomly sampled from the intervals: $\alpha \in [-20, 20]$ deg, $q \in [-2, 2]$ rad/s, $\theta \in$

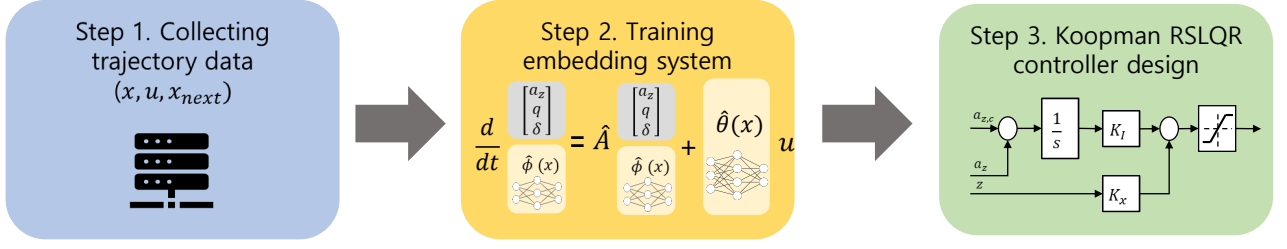


Figure 3 – Flow chart of the proposed method

$[-40, 40]$ deg, $M_a \in [2.5, 4.5]$, $h \in [8, 11]$ km, and $\delta_{cmd} \in [-30, 30]$ deg. Here, $u = V \cos \alpha$, $w = V \sin \alpha$ where $V = V_s M_a$, and V_s denotes the speed of sound at a given altitude. The δ and $\dot{\delta}$ are set to be zero. Hence, the corresponding state $s = [u, w, q, \theta, h, \delta, \dot{\delta}]^T$ follows from the sampled point. Detailed formulas can be found in [16]. The next state s_{next} can be obtained by the following equation over a fixed time interval $T_s > 0$,

$$s_{next} = s + \frac{T_s}{6}(k_1 + 2k_2 + 2k_3 + k_4) =: f_{RK4}(s, u|f), \quad (31)$$

where coefficients are given as

$$\begin{aligned} k_1 &= f_s(s, u), \\ k_2 &= f_s(s + \frac{T_s}{2}k_1, u), \\ k_3 &= f_s(s + \frac{T_s}{2}k_2, u), \\ k_4 &= f_s(s + T_s k_3, u). \end{aligned} \quad (32)$$

In this context, the $f_{RK4}(\cdot, \cdot|f)$ represents the Runge-Kutta 4th order (RK4) prediction at each state s and control input u for a given dynamical system f . Note that the control input remains constant throughout the RK4 prediction.

In this study, the *state of interest* $x = [a_z, q, \delta, \alpha, \theta, M_a, h]^T$ is used for the Koopman embedding representation, while the state s is only utilized for modeling actual missile maneuvers and collecting dataset. This distinction is made because the variables crucial for the controller are found in the elements of x , not those of s . Therefore, the unknown dynamic equations of x are the functions to be approximated, not those of s .

More specifically, the dynamic equations of the state s are propagated and all of the elements of x can be obtained from the state s . The normal acceleration a_z can be calculated using (29). The angle of attack and the Mach number are derived from the equations: $\alpha = \tan^{-1}(w/u)$, $M_a = \sqrt{u^2 + w^2}/V_s$. Then, x and x_{next} are obtained from s and s_{next} , respectively. Finally, the paired data $\{x, u, x_{next}\}$, collected from this process, constitutes the dataset for training Koopman approximate system.

3.2 Training for model identification

Let us define the structure of the embedding dynamical model as

$$\frac{dz}{dt} = Az + \theta(x)u = AT(x) + \theta(x)u := A \begin{bmatrix} C_p x \\ \phi(x) \end{bmatrix} + \theta(x)u, \quad (33)$$

where $z = [(C_p x)^T, \phi(x)^T]^T \in \mathbb{R}^{n_e}$ is the Koopman embedding, and $C_p \in \mathbb{R}^{3 \times n}$ is the output matrix for remaining some of the original states, i.e., $C_p x = [a_z, q, \delta]^T$. The dimension of the embedding system state is denoted by $n_e := N + 3$. The explicit embedding of some physically meaningful original states, $C_p x$, helps to implement a tracking controller easily after training. However, note that the inclusion of original states is possible only when the drift term of the original dynamical system has a single fixed point [17]. The approximation method cannot be globally valid; instead, it is expected that the method may extend the represented region near the equilibrium point of our interest beyond what is possible with the linearization approach by unfolding nonlinear terms.

The functions $\phi : X \rightarrow \mathbb{R}^N$ and $\theta : X \rightarrow \mathbb{R}^{n_e \times m}$, and the matrix $A \in \mathbb{R}^{n_e \times n_e}$ are the two functions and parameters to be determined through training. In this study, a matrix $\hat{A} \approx A$ and two artificial neural networks $\hat{\phi}$ and $\hat{\theta}$ are trained to approximate the corresponding functions as follows,

$$\phi(x) \approx \hat{\phi}(x) = W_2^1 \sigma(W_1^1 x + b_1^1) + b_2^1, \quad (34)$$

$$\theta(x) \approx \hat{\theta}(x) = W_2^2 \sigma(W_1^2 x + b_1^2) + b_2^2. \quad (35)$$

The neural networks are two-layer structures with the state x as the input, and W_i^j and b_i^j , $\forall i = 1, 2$ and $j = 1, 2$, are the network parameters, i.e., the weights and biases of each layer in the network. The σ represents an activation function. The loss function, denoted as the mean-squared error between prediction and next value, is given by

$$L(\hat{\phi}, \hat{\theta}, \hat{A}) = \frac{1}{|\mathcal{D}|} \sum_{(x, u, x_{next}) \in \mathcal{D}} \|\hat{T}(x_{next}) - f_{\text{RK4}}(\hat{T}(x), u) \|^2, \quad (36)$$

where $\hat{T}(x) := [(C_p x)^T, \hat{\phi}(x)^T]^T$, and $\mathcal{D}$ denotes the stacked dataset according to Section 3.1. The training objective indicated by this loss function is to acquire a linearly embedded continuous dynamic equation that accurately predicts the dynamic characteristics of the missile.

3.3 RSLQR with Koopman embedding

The model acquired through training can be used to design a controller based on a model-based control theory. A primary advantage of learning Koopman embedding lies in its structural attributes. The linear-quadratic optimal control problem is formulated to regulate the tracking error. Notably, time-integrated acceleration tracking error is augmented for zero steady-state error. The time derivative of the augmented state is denoted by $\xi \in \mathbb{R}^{n_a}$, where $n_a := n_e + 1$. For the controller design, in this study, the state-dependent control effectiveness $\hat{\theta}(x)$ is replaced by $\hat{\theta}(x_0)$. It should be noted that x_0 includes the angle of attack α , Mach number M_a , and altitude h , which are values denoting the operating points typically used for gain-scheduling. Therefore, a suitable gain at each operating point can be determined. Let us set the infinite-time quadratic cost to be minimized as

$$J = \frac{1}{2} \int_0^\infty (\xi^T Q \xi + \mu^T R \mu) dt, \quad (37)$$

subject to

$$\dot{\xi} = A_a \xi + B_a \mu, \quad (38)$$

where

$$A_a := \begin{bmatrix} 0 & C_a \\ 0 & \hat{A} \end{bmatrix}, B_a := \begin{bmatrix} 0 \\ \hat{\theta}(x_0) \end{bmatrix}, C_a = [1 \quad 0_{1 \times (N-1)}], \quad (39)$$

$\xi := [a_z - a_{z,cmd}, \dot{z}]^T$, and $\mu = \delta_c$. The symmetric positive semi-definite matrix $Q \in \mathbb{R}^{n_a \times n_a}$ and symmetric positive-definite matrix $R \in \mathbb{R}^m$ are the two weighting matrices to be tuned.

The feedback control law that minimizes the cost function (37) is:

$$u = \int \mu dt = - \int K \xi dt = -K x_a, \quad (40)$$

where

$$K = R^{-1} B_a^T P. \quad (41)$$

The P is the solution to the following continuous-time algebraic Riccati equation.

$$A_a^T P + P A_a - P B_a R^{-1} B_a^T P + Q = 0. \quad (42)$$

Training parameters	Values	Network architecture	Values
Number of data	1,000,000	Number of hidden layers	2
Number of epoch	100	Number of nodes	64
Batch size	32	Activations of hidden layers	Leaky ReLU
Learning rate	0.001	Activation of the output layer	Identity
Train/test data split ratio	85:15	Output dimensions	$\phi:9, \theta:12$

Table 2 – Training parameters and network architectures

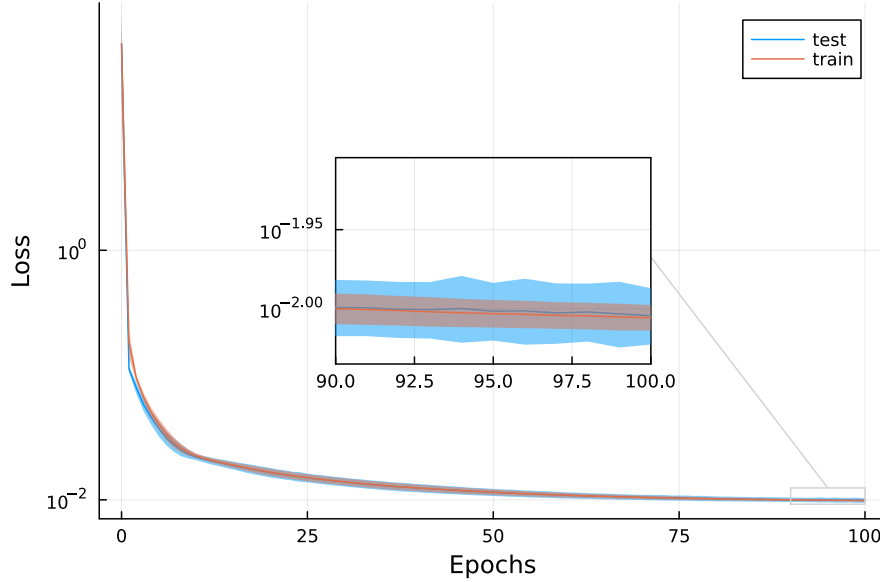


Figure 4 – Training loss

4. Numerical Simulation

This section shows the training and test results, offering a comprehensive examination of both the training process and the performance of the designed controller through repeated testing and comparative analysis.

4.1 Training results

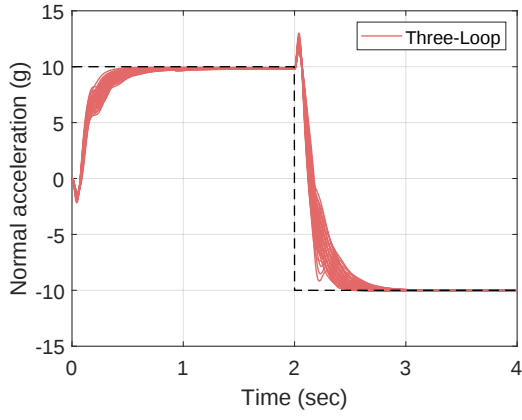
The training procedure was repeated with various initial network parameters, employing Xavier initialization [18] for the random assignment of each network's parameters. Detailed information on all training parameters and the architectures of the two networks is summarized in Table 2. In Figure 4, losses are depicted from minimum to maximum values across different initializations. The reduction in loss, as shown in Figure 4, consistently occurs across epochs, irrespective of the network's initialization. The zero epoch indicates the loss value at the initialization. Therefore, the training loss histories show that the current training settings can produce consistent results

4.2 Comparison of the proposed controller with the gain-scheduled three-loop controller

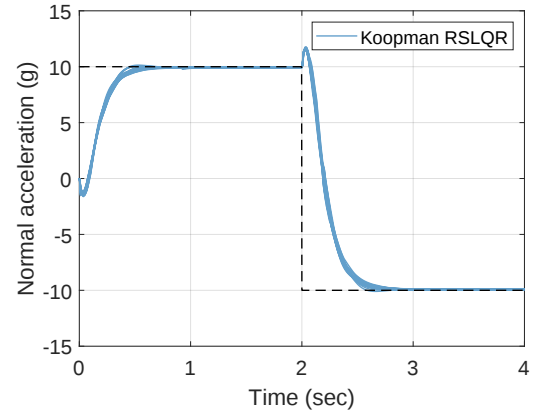
Figure 5 shows the tracking results and the recorded histories of dominant state variables tested within the operational envelope of a conventional air-to-air missile. The objective is to assess the capability of the controller designed by the proposed methodology to effectively operate across a wide range of operational conditions. To facilitate the assessment, the tracking results of a classical gain-

Q	$\text{diag}([10, 0.1, 1, 0_{1 \times 10}])$
R	1

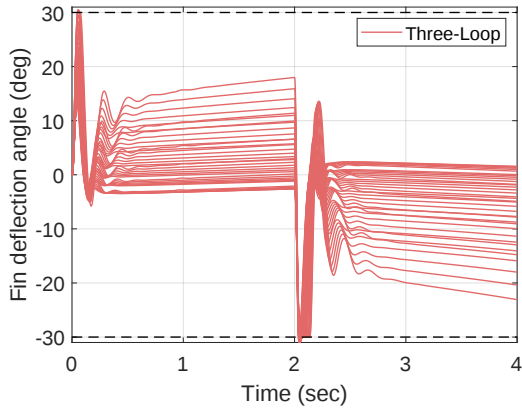
Table 3 – RSLQR design parameters



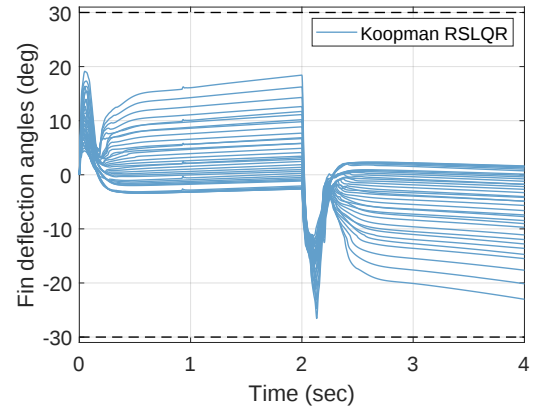
(a) Tracking results



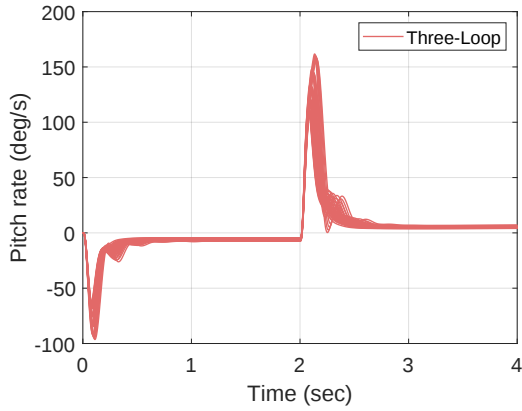
(b) Tracking results



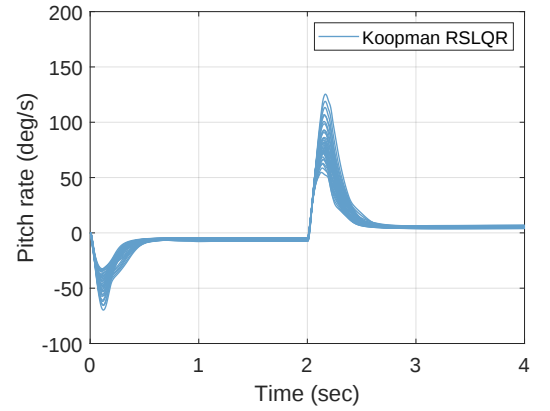
(c) Control Inputs



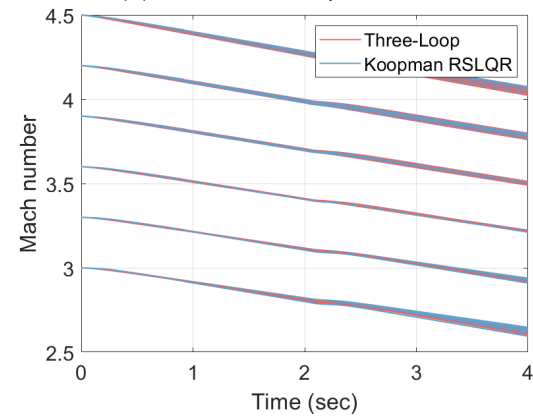
(d) Control Inputs



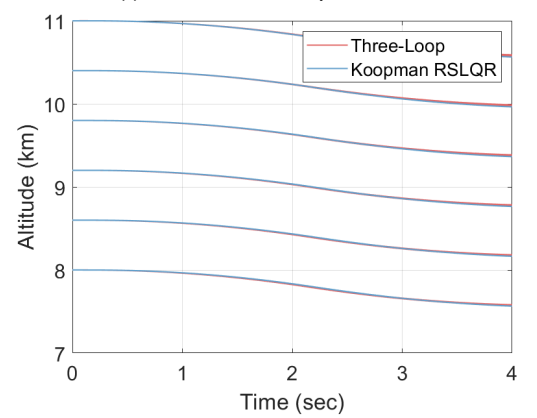
(e) Pitch rates responses



(f) Pitch rates responses



(g) Mach number histories



(h) Altitude histories

Figure 5 – Comparative results between three-loop controller and the Koopman RSLQR controller

scheduled three-loop controller are also shown. The scheduled gain table is provided by the MATLAB Aerospace Blockset [19]. Note that the proposed methodology relies on the Koopman embedding to derive the feedback control law obtained by solving the linear quadratic problem. Notably, the tracking capability is enhanced by augmenting the acceleration tracking error. First of all, the Koopman-based RSLQR controller consistently demonstrates precise tracking of step commands across all test case, as illustrated in Figure 5b). Additionally, Figure 5c) and Figure 5d) show that the proposed controller uses less energy while producing similar tracking performance with the three-loop controller. Furthermore, as shown in Figure 5g) and Figure 5h), the Koopman RSLQR controller performs well within the Mach number range of [2.5, 4.5] and the altitude range of [8, 11] km, affirming its effectiveness over a broad operational spectrum.

5. Conclusion

This study introduced a data-driven modeling method based on the Koopman operator theory for missile longitudinal autopilot design. Utilizing trajectory data, the nonlinear missile model was identified without relying on prior knowledge of the underlying dynamics by leveraging the power of artificial neural networks. Using the identified linear model with Koopman embedding, the model-based robust servomechanism linear quadratic regulator can readily be designed for acceleration tracking. Numerical simulation results demonstrated versatility across a broad spectrum of operating points, contributing to the favorable performance of the proposed controller in various operational scenarios. A notable feature is the ability of the proposed controller to replace time-consuming gain-scheduled controllers by adjusting only a small number of parameters. Future research will encompass various directions, including the application of the proposed approach to model reference adaptive control and extended state observer-based control. Additionally, the stability-guaranteed controller design scheme and its robustness analysis remain to be further studied.

Acknowledgement

This work was supported by Artificial Intelligence Research Laboratory for Flight Control funded by Agency for Defense Development and Defense Acquisition Program Administration under Grant UD230014SD.

References

- [1] Ali R. M. and Jafar R. Skid-to-turn missile autopilot design using scheduled eigenstructure assignment technique. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 220(3):225–239, 2006.
- [2] Abd-elatif M.A., Qian L., and Bo Y. Optimization of three-loop missile autopilot gain under crossover frequency constraint. *Defence Technology*, 12(1):32–38, 2016.
- [3] Behrouz H. F., Mohammadzaman I., and Mohammadi A. H_2 / H_∞ autopilot design with regional pole placement constraints: An lmi-based approach. *Scientia Iranica B*, 28(2):757–772, 2021.
- [4] ÇLimen T. A generic approach to missile autopilot design using state-dependent nonlinear control. In *18th IFAC World Congress*, Milano, Italy, Aug. 28-Sep. 2 2011.
- [5] Peter F., Leitão M., and Holzapfel F. Adaptive augmentation of a new baseline control architecture for tail-controlled missiles using a nonlinear reference model. In *AIAA Guidance, Navigation, and Control Conference*, Minneapolis, MN, Aug. 2012.
- [6] Panchal B., Subramanian K., and Talole S. E. Robust missile autopilot design using two time-scale separation. *IEEE Transactions on Aerospace and Electronic Systems*, 54(3):1499–1510, 2018.
- [7] Williams M. O., Kevrekidis I. G., and Rowley C. W. A data-driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25:1307–1346, 2015.
- [8] Champion K., Lusch B., Kutz J. N., and Brunton S. L. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, 2019.
- [9] Lusch B., Kutz J. N., and Brunton S. L. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(4950), 2018.
- [10] Bevanda P., Sosnowski S., and Hirche S. Koopman operator dynamical models: Learning, analysis and control. *Annual Reviews in Control*, 52:197–212, 2021.
- [11] Ma X., Huang B., and U. Vaidya. Optimal quadratic regulation of nonlinear system using koopman operator. In *2019 American Control Conference (ACC)*, Philadelphia, PA, Jul. 2019.

- [12] Narasingam A. and Kwon J. S.-I. Closed-loop stabilization of nonlinear systems using koopman lyapunov-based model predictive control. In *59th IEEE Conference on Decision and Control (CDC)*, Jeju Island, Republic of Korea, Dec. 2020.
- [13] Bevanda P., Sosnowski S., and Hirche S. Koopman operator dynamical models: Learning, analysis and control. *Annual Reviews in Control*, 52:197–212, 2021.
- [14] Brunton S. L., Budišić M., Kaiser E., and Kutz J. N. Modern Koopman Theory for Dynamical Systems. 64(2):229–340, October 2022.
- [15] Surana A. Koopman operator based observer synthesis for control-affine nonlinear systems. In *IEEE 55th Conference on Decision and Control (CDC)*, Las Vegas, NV, Dec. 2016.
- [16] Mracek C. P. and Cloutier J. R. Full envelope missile longitudinal autopilot design using the state-dependent Riccati equation method. In *AIAA Guidance, Navigation, and Control Conference*, New Orleans, LA, Aug. 1997.
- [17] Brunton S. L., Brunton B. W., Proctor J. L., and Kutz J.N. Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLoS ONE*, 10(2):e0150171, 2016.
- [18] Xavier G. and Yoshua B. Understanding the difficulty of training deep feedforward neural networks. In *13th International Conference on Artificial Intelligence and Statistics*, Sardinia, Italy, May 2010.
- [19] The MathWorks Inc. Design a guidance system in matlab and simulink. Natick, MA, 2024.

Copyright Statement

The authors confirm that they, and/or their company or organization, hold copyright on all of the original material included in this paper. The authors also confirm that they have obtained permission, from the copyright holder of any third party material included in this paper, to publish it as part of their paper. The authors confirm that they give permission, or have obtained permission from the copyright holder of this paper, for the publication and distribution of this paper as part of the ICAS proceedings or as individual off-prints from the proceedings.