BILINGUAL PUBLISHING GROUP

Artificial Intelligence Advances

https://journals.bilpubgroup.com/index.php/aia

ARTICLE

DNN-Based AI-Driven H_2/H_∞ Filter Design of Nonlinear Stochastic Systems via Two-Coupled HJIEs-Supervised Adam Learning Algorithm

Bor-Sen Chen* , ***Jui-Ming Ma, Ruei-Syuan Wu***

Department of Electrical Engineering, National Tsing Hua University, Hsinchu 300044, Taiwan

ABSTRACT

This study introduces a new approach using supervised learning deep neural networks (DNNs) to develop an AI-driven filter for nonlinear stochastic signal systems with external disturbance and measurement noise. The filter aims to achieve a balanced design between and norm of the state estimation error to achieve both optimal and robust filtering design of nonlinear signal system simultaneously while considering environmental disturbance and measurement noise. Traditionally, this nonlinear H_2/H_∞ filter design involves solving complex two-coupled Hamilton-Jacobi-Issac Equations (HJIEs). To simplify this complicated design process, a novel two-coupled HJIEs-supervised Adam learning algorithm is proposed for DNN-based AI-driven filter. This algorithm trains a H_2/H_∞ DNN-based AI-driven filter offline using worst-case scenarios of environmental disturbance and measurement noise. This training phase generates state estimation errors that teach the DNN-based AI-driven filter how to coordinate nonlinear system model with worst-case external disturbance and measurement noise, Luenberger-type filter, estimation error dynamic model and two-coupled HJIEs-supervised deep Adam learning algorithm to achieve the mixed H_2/H_∞ filtering strategy effectively. The study demonstrates theoretically that this approach will achieve the desired mixed H_2/H_∞ filtering strategy once the Adam learning algorithm converges. Finally, the effectiveness of the proposed DNN-based AI-driven filter design method is validated through simulations, specifically involving trajectory estimation and prediction of an incoming ballistic missile detected by a radar system.

Keywords: DNN-based AI filter; Deep neural network (DNN); Mixed H_2/H_∞ filter; Hamilton-Jacobi-Isaacs Equation (HJIE); Nonlinear stochastic system and HJIE-supervised adam learning algorithm

*CORRESPONDING AUTHOR:

Bor-Sen Chen, Department of Electrical Engineering, National Tsing Hua University, Hsinchu 300044, Taiwan; Email: bschen@ee.nthu.edu.tw

ARTICLE INFO

Received: 2 September 2024 | Revised: 15 September 2024 | Accepted: 20 September 2024 | Published Online: 20 October 2024

DOI: https://doi.org/10.30564/aia.v6i1.7261

CITATION

Chen, B.-S., Ma, J.-M., Wu, R.-S., 2024. DNN-Based AI-Driven H_2/H_∞ Filter Design of Nonlinear Stochastic Systems via Two-Coupled HJIEs-Supervised Adam Learning Algorithm. Artificial Intelligence Advances. 6(1): 34–55. DOI: https://doi.org/10.30564/aia.v6i1.7261

1. Introduction

The estimation of state variables in stochastic signal systems, affected by environmental disturbance and measurement noise, presents a significant challenge in signal processing with broad engineering applications. The Kalman optimal H_2 filter design is widely used in fields such as control engineering, aerospace engineering, and signal processing, assuming known covariances of disturbance and noise[1]. However, when these covariances are uncertain or not well-defined, the robust H_∞ filter is employed to mitigate the impact of worst-case disturbances and noise on state estimation performance, ensuring attenuation below a specified level[2]. While the H_∞ filter enhances robustness compared to the optimal H_2 filters, its design tends to be overly conservative, resulting in high filter gains. Consequently, the mixed H_2/H_∞ filter design has been introduced to combine the advantages of both the optimal H_2 and robust H_∞ filters[3–6].

In linear stochastic signal systems, the mixed H_2/H_∞ filter design requires solving two-coupled Riccati-like equations for filter gain[3, 5, 6]. However, in nonlinear stochastic signal systems, the mixed H_2/H_∞ filter design involves solving two-coupled Hamilton-Jacobi-Isaacs Equations (HJIEs). These equations are complex nonlinear partial differential equations involving the state $x(t)$, state estimation $\widehat{x}(t)$ and estimation error $\widetilde{x}(t)$, and they cannot be easily solved using conventional analytic or numerical methods due to the unavailability of $x(t)$. Therefore, engineers avoid directly solving these coupled HJIEs in the mixed H_2/H_∞ nonlinear filter design process. Instead, interpolation techniques such as the fuzzy interpolation method[7], global linearization method[8], and gain scheduling method[9] are employed. These methods interpolate local linearized stochastic systems to approximate the behavior of the nonlinear stochastic system. The coupled H_2/H_∞ HJIEs are transformed into sets of Riccati-like Inequalities and then into sets of Linear Matrix Inequalities (LMIs). For example, if l local linearized stochastic systems at l different operating points are interpolated using l fuzzy smoothing functions to approximate a nonlinear stochastic system, a Takagi-Sugeno (T-S) fuzzy filter can be used to interpolate the corresponding l local linear filters for the mixed H_2/H_∞ nonlinear filter design[6]. However, fuzzy mixed H_2/H_∞ filter designs require solving l^2 set of coupled LMIs. This means that if l local linearized systems are used for nonlinear system approximation, l^2 LMIs must be solved for the nonlinear filter design[8, 10]. Furthermore, l^2 local linear filters need to be interpolated using l^2 interpolation functions to obtain the mixed H_2/H_∞ filter for the nonlinear stochastic system. Consequently, considerable computation time is necessary for filtering at each time step, currently making it impractical for real-time applications. Additionally, the use of quadratic Lyapunov function $V\left(\widetilde{x}^T(t)\right) = \widetilde{x}^T(t)P\widetilde{x}(t)$, where $P^T = P > 0$ is common. These functions treat the filtering error $\widetilde{x}(t)$ as the solution of the coupled HJIEs is transformed into coupled sets of l^2 LMIs. However, relying on such quadratic Lyapunov functions may lead to conservative results in the mixed H_2/H_∞ nonlinear filter design.

Recently, deep neural networks (DNNs) have emerged as highly effective models for information processing[11–17]. Trained by large datasets, DNNs excel in various applications such as speech recognition[13], language translation[11], and image classification[14], tasks traditionally associated with human cognition. However, integrating DNNs into man-made machines with dynamic equations remains challenging[15, 16]. To address this challenge, reinforcement learning schemes have been increasingly utilized to enhance learning algorithms for some specific objectives[15, 18]. Furthermore, with the rapid advancement of AI neural network chips, there has been a recent proposal to embed HJIEs within DNNs for H_∞ filtering designs in nonlinear stochastic systems[19]. Given that H_2 filtering schemes represent critical optimal filter designs, there is growing interest in developing DNN-based filter trained with H_2/H_∞ HJIEs-supervised learning algorithm. This approach aims to coordinate the nonlinear stochastic system, Luenberger filter, estimation error system and two-coupled H_2/H_∞ HJIEs-supervised learning algorithm to directly solve the coupled H_2/H_∞ HJIEs for achieving mixed H_2/H_∞ filtering schemes tailored for nonlinear stochastic systems with external disturbance and measurement noise.

In this study, a novel H_2/H_∞ HJIEs-supervised Adam learning algorithm for training DNN-based AI-filter is proposed for nonlinear stochastic systems to achieve mixed H_2/H_∞ filtering strategy. In scenarios where environmental disturbance and measurement noise are unavailable, the system substitutes them by the available worst-case environmental disturbance and measurement, denoted as $v^*(t)$

and $n^*(t)$, respectively, in the nonlinear system model and the estimation error dynamic. The resulting filtering error is used as input for training the DNN with a two-coupled H_2/H_∞ HJIEs-supervised deep learning algorithm. This training aims to solve two coupled H_2/H_∞ HJIEs to output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$, which is essential for constructing the mixed H_2/H_∞ filter gain $l^*(\widehat{x}(t))$, $v^*(t)$ and $n^*(t)$. The construction of the mixed H_2/H_∞ DNN-based AI-driven filter is divided into two phases: (i) off-line pre-training phase and (ii) on-line operation phase. In the off-line pre-training phase (illustrated in **Figure 1**), the DNN takes the estimation error $\widetilde{x}\ (t)$ as input and is trained using the two coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm to output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$. This output is then used to simultaneously generate the mixed H_2/H_∞ filter gain $l^*(\widehat{x}(t))$, as well as the worst-case measurement noise $n^*(t)$ and environmental disturbance $v^*(t)$ for further supervised deep learning iterations of the DNN-based AI-driven filter. In the off-line pre-training phase, the study employs available worst-case scenarios $v^*(t)$ and $n^*(t)$ to replace the unavailable environmental disturbance $v(t)$ and measurement noise $n(t)$ in the mixed H_2/H_∞ filter design. This substitution ensures that the state estimation performance of the mixed H_2/H_∞ filtering strategy remains for DNN-based AI-driven filter, as it is designed to handle worst-case conditions in the absence of $v(t)$ and $n(t)$ during the pre-training phase. Further, the use of available worst-case $v^*(t)$ and $n^*(t)$ can significantly compress the training data and training time of DNN-based AI.

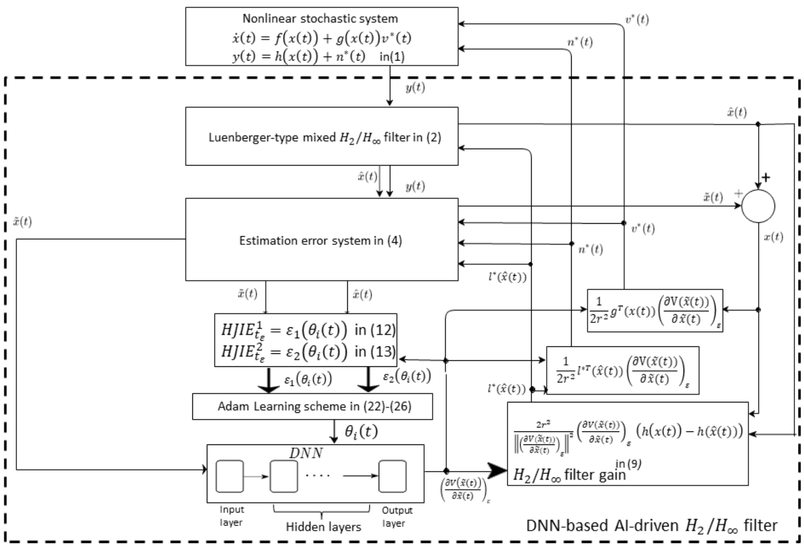

Figure 1. The DNN-based AI-driven mixed H_2/H_∞ nonlinear filter coordinates the nonlinear stochastic system with the worst-case disturbance and measurement noise, Luenberger-type filter in (2), estimation error system in (4) and a H_2/H_∞ HJIEs-supervised learning DNN to output $\left(\partial V\left(\widetilde{x}\ (t)\right)/\partial\ \widetilde{x}\ (t)\right)$ to generate $\left(\partial V\left(\widetilde{x}\ (t)\right)/\partial\ \widetilde{x}\ (t)\right)$ to produce H_2/H_∞ filter gain $l^*(\widehat{x}(t))$ for Luenberger-type filter to become a mixed H_2/H_∞ filter. The flowchart depicts a two-coupled learning scheme using H_2/H_∞ HJIEs-supervised Adam learning algorithm for training a DNN-based AI-driven mixed H_2/H_∞ filter for a nonlinear stochastic system described in (1). During the offline pre- training phase, the DNN is trained using the estimation error $\widetilde{x}\ (t)$ obtained from the estimation error system in (4), considering the worst-case input $v^*(t)$ and $n^*(t)$. The trained DNN-based AI outputs $\left(\partial V\left(\widetilde{x}\ (t)\right)/\partial\ \widetilde{x}\ (t)\right)$, which is then used to compute the worst-case input $v^*(t)$, $n^*(t)$, and the filter gain $l^*(\widehat{x}(t))$ based on (9)–(11). These are utilized by the nonlinear stochastic system in (1) to produce output y(t), estimated state $\widehat{x}(t)$ via a Luenberger-type filter in (2), and estimation error $\widetilde{x}\ (t)$ via the system in (3). Furthermore, $\widetilde{x}\ (t)$, $\widehat{x}(t)$, and $\left(\partial V\left(\widetilde{x}\ (t)\right)/\partial\ \widetilde{x}\ (t)\right)$ are inputs to two coupled H_2/H_∞ HJIEs in (12) and (13), yielding outputs $\left[HJIE^1_\epsilon(t), HJIE^2_\epsilon(t)\right]^T = [\epsilon_1(\theta_i(t)), \epsilon_2(\theta_i(t))]^T$ as described in (19) and (20). These outputs serve as inputs to a supervised Adam learning Algorithms (16)–(18). If $[\epsilon_1(\theta_i(t)), \epsilon_2(\theta_i(t))]^T \to [0,0]^T$, according to Theorem 2, the convergence of $\left[HJIE^1_\epsilon(t), HJIE^2_\epsilon(t)\right]^T \to [0,0]^T$ implies that the proposed two-coupled H_2/H_∞HJIEs-supervised DNN-based AI-driven filter scheme approaches the mixed H_2/H_∞ filter strategy in (7) and (8). This marks the transition to the online operational phase, where real inputs $v(t)$ and $n(t)$ in (1) are used to obtain the output $y(t)$. Typically, no further training of the DNN-based AI-driven filter using the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm is necessary. However, if $|\epsilon_1(\theta_k(t))| > \delta$ or $|\epsilon_2(\theta_k(t))| > \delta$ for a specified $\delta > 0$, the feedback of $[\epsilon_1(\theta_i(t)), \epsilon_2(\theta_i(t))]^T$ is required for the H_2/H_∞ HJIEs-supervised Adam learning algorithm of the DNN during the online phase to enhance the filtering performance of the DNN-based AI-driven mixed H_2/H_∞ filter.

After completing the off-line pre-training phase, the mixed H_2/H_∞ DNN-based AI-driven filter scheme is transmitted to the on-line operation phase. During this phase, since $v(t)$ and $n(t)$ are available in the nonlinear stochastic system, the worst-case $v^*(t)$ and $n^*(t)$ are unnecessary for generating $\widetilde{x}\ (t)$ as DNN input. It is demonstrated that as the error $[\epsilon_1(\theta(t)),\ \epsilon_2(\theta(t))]^T$ in solving the two coupled H_2/H_∞ HJIEs using the trained DNN approaches $[0,\ 0]^T$ via the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm, the proposed DNN-based AI-driven filtering scheme achieves the mixed H_2/H_∞ filter design for the nonlinear stochastic signal system. However, in practical applications, the H_2/H_∞ HJIEs-supervised Adam deep learning algorithm is stopped after the pre-training phase and is transmitted to the operation phase when both $|\epsilon_1(\theta(t))| \leq \epsilon$ and$|\epsilon_2(\theta(t))| \leq \epsilon$simultaneously, where ϵ is a small prescribed positive value.

The contributions of this paper are described as follows:

1) A novel approach is introduced: a deep learning-based DNN AI-driven filter scheme supervised by embedded two-coupled H_2/H_∞ HJIEs coordinates with nonlinear system model, Luenberger filter and estimation error model with the worst-case $v^*(t)$ and $n^*(t)$ to achieve mixed H_2/H_∞filter performance for nonlinear stochastic systems under uncertain environmental disturbances and measurement noises. During the off-line pre-training phase, the worst-case scenarios $v^*(t)$ and $n^*(t)$ replace unavailable $v(t)$ and $n(t)$, respectively, to train the DNN-based AI-driven filter using the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm. This approach ensures the DNN-based AI-driven filter can achieve the desired mixed H_2/H_∞ filtering performance. It has been demonstrated that the proposed DNN-based AI-driven filter, trained with the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm, effectively achieves the mixed H_2/H_∞ filter performance as the convergence of the supervised Adam learning algorithm for solving two-coupled H_2/H_∞ HJIEs is ensured.
2) The proposed DNN-based AI-driven filter scheme, trained by two coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm, can directly compute $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ for the filter gain $l^*(\widehat{x}(t))$ in the mixed H_2/H_∞ filter design. This method bypasses the conventional approach of solving Lyapunov function $V\left(\widetilde{x}\ (t)\right)$ from two coupled H_2/H_∞ HJIEs, thereby avoiding the complex computation of partial differentials $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ based on numerical data of $\widetilde{x}\ (t)$ and $V\left(\widetilde{x}\ (t)\right)$ in traditional nonlinear filter design methods for more practical applications.
3) The proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm for DNN-based AI-driven filter scheme represents a significant advancement in mixed H_2/H_∞ nonlinear filter design. It integrates HJIEs-supervised Adam learning algorithm and DNN-based AI with mixed H_2/H_∞ filter co-design, bridging the gap between recent advances in supervised machine learning and the complex requirements of optimal H_2 and robust H_∞ nonlinear filtering design in nonlinear stochastic signaling systems. This approach leverages the H_2/H_∞ HJIEs-supervised Adam learning algorithm for DNN-based AI-driven filter to tackle the intricate mixed H_2/H_∞ filter design challenges of nonlinear stochastic signal systems with uncertain external disturbance and measurement noise, significantly compressing training time and data requirements compared to conventional big data-driven deep learning methods that do not utilize system dynamic models and theoretical derivatives.

The rest of this paper is structured as follows: Section 2 presents the problem formulation, focusing on the mixed H_2/H_∞ filter design for nonlinear stochastic systems. In this section, we introduce a novel two-coupled H_2/H_∞ HJIEs-supervised learning DNN-based AI-driven filter scheme tailored for achieving the mixed H_2/H_∞ filter strategy of nonlinear stochastic signal systems. In the Section 3, we introduced how we combine our proposed method and the deep Adam learning algorithm, and a pseudocode is given to illustrate the proposed H_2/H_∞ DNN-based AI-driven filter scheme. Section 4 includes a simulation example demonstrating the application of the proposed mixed H_2/H_∞ DNN-based AI-driven filter using the two H_2/H_∞ HJIEs-supervised Adam learning algorithm for trajectory estimation of an incoming ballistic missile by a radar system. Finally, Section 5 concludes the paper.

Notation: $\mathbb{R}^n$: The space real vectors of dimension. $\mathbb{R}^{n\times m}$: The space of real $n \times m$ matrices. I_a the identity matrix of dimension $a \times a$; $0_{n\times n}$: the zero matrix of dimension $n\times n$. If $x(t) \in \mathbb{R}^n$, then $\| x(t) \|_2 = \left(\sum_{i=1}^{n} x_i^2(t)\right)^{1/2} = \left(x^T(t)x(t)\right)^{1/2}$, where $x^T(t)$ denotes the transpose of $x(t)$. $\mathcal{L}_F^2[0,\infty)$: The set of n-tuple stochastic functions with finite energy i.e., $x(t) \in \mathcal{L}_F^2[0,\infty)$, if $E\left(\int_0^\infty \left(x^T(t)x(t)\right)^{1/2} dt\right) = E\left(\int_0^\infty \| x(t) \|_2 dt\right) < \infty$.

2. Problem description

Consider the following nonlinear stochastic system with environmental disturbance and measurement noise

$$\begin{gathered} x(t) = f(x(t)) + g(x(t))v(t) \\ y(t) = h(x(t)) + n(t) \end{gathered} \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $v(t) \in \mathbb{R}^l$ denotes the random environmental disturbance, $n(t) \in \mathbb{R}^m$ denotes the measurement noise, and $y(t) \in \mathbb{R}^m$ is the output measurement. $f(x(t)) \in \mathbb{R}^n$, $g(x(t)) \in \mathbb{R}^{n\times l}$, and $h(x(t)) \in \mathbb{R}^m$ are the nonlinear system and measurement functions of state vector and satisfy with the Lipschitz condition.

We assume that the nonlinear stochastic system described in (1) is observable. Therefore, to achieve a specific filtering objective, we can utilize the following nonlinear Luenberger filter to estimate the state vector of the system from its output measurement $y(t)$.

$$\dot{\hat{x}}(t) = f(\hat{x}(t)) + l(\hat{x}(t))(y(t) - h(\hat{x}(t)) \tag{2}$$

where $\hat{x}(t)$ denotes the state estimation of $x(t)$.

For the nonlinear Luenberger filter design in (2), we need to design filter gain $l(\hat{x}(t))$ to estimate x(t) from the measurement output $y(t)$ in (1) as good as possible despite unavailable $v(t), n(t) \in L_F^2[0,\ \infty)$.

$$\tilde{x}(t) = x(t) - \hat{x}(t) \tag{3}$$

From (1) and (2), we get the estimation error dynamic as follows:

$$\begin{gathered} \dot{\tilde{x}}(t) = f(x(t)) - f(\hat{x}(t)) - l(\hat{x}(t))(h(x(t)) - h(\hat{x}(t))) \\ +g(x(t))v(t) - l(\hat{x}(t))n(t) \end{gathered} \tag{4}$$

Due to the absence of statistical information regarding $v(t)$ and $n(t)$, the effectiveness of H_2 optimal state estimation is compromised. Hence, in this study, the design of the filter gain $l(\hat{x}(t))$ for the Luenberger-type filter in (2) aims to achieve the H_2/H_∞ filtering performance. This approach seeks to leverage the advantages of both optimal H_2 filtering and robust H_∞ filtering concurrently. Specifically, the goal is to implement an H_2 optimal filter while considering the worst-case scenaros of $v^*(t)$ and $n^*(t)$ based on robust H_∞ filter design principles for the state estimation process of the nonlinear stochastic system described in (1). The performance indices $J_\infty(l(\hat{x}(t)),\ v(t),\ n(t))$ for the H_∞ filter design and $J_2(l(\hat{x}(t)),\ v(t),\ n(t))$ for the H_2 filter design in (2) for the nonlinear stochastic system (1), are given in the following[3, 5–8]:

$$\begin{gathered} J_\infty(l(\hat{x}(t)),\ v(t),\ n(t)) = E\left(V\left(\tilde{x}(t_f)\right)\right) + \\ E\left(\int_0^{t_f} [\tilde{x}^T(t) Q \tilde{x}(t) - r^2(v^T(t)v(t) + n^T(t)n(t))]dt\right) \end{gathered} \tag{5}$$

$$\begin{gathered} J_2(l(\hat{x}(t)),\ v(t),\ n(t)) = E\left(V\left(\tilde{x}(t_f)\right)\right) \\ +E\left(\int_0^{t_f} \tilde{x}^T(t) Q \tilde{x}(t) dt\right) \end{gathered} \tag{6}$$

where $Q = Q^T \geq 0$ represents the weighting matrix applied to the state estimation error. The Lyapunov function $V\left(\tilde{x}(t)\right)$ serves as the energy function for the state estimation error at time t. The parameter $r > 0$ indicates the desired robustness level for H_∞ filtering against disturbances $v(t)$ and measurement noise $n(t)$ affecting the state estimation error, viewed in terms of energy. The function $E(\cdot)$ denotes the expectation operation, and t_f denotes the terminal time.

The mixed H_2/H_∞ filtering design of nonlinear stochastic system in (1) needs to solve the following two-person Nash non-zero sum games in which $l(\hat{x}(t))$ is one player and $(v(t),\ n(t))$ is considered as another player[8, 10, 20],

$$J_\infty(l^*(\hat{x}(t)),\ v^*(t),\ n^*(t)) \leq J_\infty(l^*(\hat{x}(t)),\ v(t),\ n(t)) \tag{7}$$

$$J_2(l^*(\hat{x}(t)),\ v^*(t),\ n^*(t)) \leq J_2(l(\hat{x}(t)),\ v^*(t),\ n^*(t)) \tag{8}$$

where $J_\infty(l^*(\hat{x}(t)), v^*(t), n^*(t)) \triangleq \max_{v(t),n(t)\in\mathcal{L}_F^2[0,\infty)} J_\infty (l^*(\hat{x}(t)),\ v(t),\ n(t))$, $J_2(l^*(\hat{x}(t)), v^*(t), n^*(t)) \triangleq \min_{l(\hat{x}(t))} J_2(l(\hat{x}(t)),\ v^*(t), n^*(t))$ with $v^*(t)$ and $n^*(t)$ being the solution of (7) and $(l^*(\hat{x}(t)), v^*(t), n^*(t))$ denotes the Nash equilibrium point of two Nash non-zero sum games in (7) and (8) for the mixed H_2/H_∞ filtering strategy.

The concept conveyed by the two-person Nash non-zero sum games in (7) and (8) for the mixed H_2/H_∞ filter design is that, when confronted with the worst-case external disturbances $v^*(t)$ and measurement noise $n^*(t)$ as per the robust H_∞ filtering strategy in (7), the filter gain $l^*(\widehat{x}(t))$ in the Luenberger-type filter in (2) can achieve the optimal H_2 filtering performance in (8) simultaneously, as discussed in[7, 8].

From the two-person Nash non-zero sum games in (7) and (8), we get the mixed H_2/H_∞ filter design of nonlinear stochastic system as follows:

Theorem 1. *(i) The filter gain $l^*(\widehat{x}(t))$, along with the worst-case scenarios of $v^*(t)$ and $n^*(t)$, are specified for the mixed H_2/H_∞ filter strategy in (7) and (8) of the nonlinear stochastic system described in (1):*

$$l^*(\widehat{x}(t)) = \frac{2r^2}{\left\|\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right\|^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\times (h(x(t)) - h(\widehat{x}(t)))^T \tag{9}$$

$$v^*(t) = \frac{1}{2r^2} g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) \tag{10}$$

$$n^*(t) = \frac{1}{2r^2} l^{*T}(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) \tag{11}$$

where $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ is the solution of the following two-coupled H_2/H_∞ HJIEs,

$$HJIE^1(t) = \widetilde{x}^T(t) Q \widetilde{x}(t) + \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T (f(x(t)) - f(\widehat{x}(t))) + \frac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t)) \times\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) - r^2 (h(x(t)) - h(\widehat{x}(t)))^T (h(x(t)) - h(\widehat{x}(t)))^T = 0 \tag{12}$$

$$HJIE^2(t) = \widetilde{x}^T(t) Q \widetilde{x}(t) + \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T (f(x(t)) - f(\widehat{x}(t))) + \frac{1}{2r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t)) \times\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) - 2r^2 (h(x(t)) - h(\widehat{x}(t)))^T (h(x(t)) - h(\widehat{x}(t)))^T = 0 \tag{13}$$

(ii) If $v(t)$ and $n(t)$ belong to $\mathcal{L}_F^2[0,\infty)$, meaning they possess finite energy, then the mixed H_2/H_∞ filter strategy can achieve the mean square asymptotic filtering capability. This is expressed as $E\left(\widetilde{x}^T(t)\,\widetilde{x}(t)\right) \rightarrow 0$ as $t \rightarrow \infty$.

Proof. See **Appendix A**. □

According to Theorem 1, solving $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ to determine the filter gain $l^*(\widehat{x}(t))$ for the mixed H_2/H_∞ filter involves addressing the coupled HJIEs equations $\left[HJIE^1(t), HJIE^2(t)\right]^T = [0,\ 0]^T$ in (12) and (13). This task remains particularly challenging due to the complex nature of solving $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ from the coupled nonlinear partial differential equations $HJIE^1(t) = 0$ and $HJIE^2(t) = 0$ in (12) and (13), whether through numerical or analytical methods[21]. Additionally, $HJIE^1(t)$ and $HJIE^2(t)$ are functions involving $x(t)$, $\widehat{x}(t)$, $\widetilde{x}(t)$. Since the state $x(t)$ of the stochastic system in (1) cannot be directly accessed, and only the output measurement $y(t)$ is available, solving $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ directly from the coupled $HJIE^1(t) = 0$ and $HJIE^2(t) = 0$ equations is nearly impossible due to the unavailability of $f(x(t))$, $g(x(t))$, and $h(x(t))$ in (12) and (13).

Remark 1. *In recent decades, various interpolation techniques such as the fuzzy method[10], global linearization method[8], and gain scheduling method[9] have emerged to address the challenge of solving two coupled H_2/H_∞ HJIEs in (12) and (13). These methods involve interpolating several local linearized stochastic models across different operating conditions to approximate the behavior of the nonlinear stochastic system described in (1). For instance, by interpolating N local linear stochastic systems, one can effectively approximate the dynamics of (1).*

$$\begin{aligned} \dot{x}(t) &= \sum_{i=1}^{N} I_i(\widehat{x}(t))(F_i x(t) + G_i v(t)) \\ y(t) &= \sum_{i=1}^{N} I_i(\widehat{x}(t))(H_i x(t) + n(t)) \end{aligned} \tag{14}$$

where $I_i(\widehat{x}(t))$, $i = 1, \ldots, N$ are the local interpolation functions in T-S fuzzy method[7, 10], smoothing functions in the global linearization method[8] or gain scheduling method[9], and (F_i, G_i, H_i) are system matrices of the ith local linearized system in these interpolation methods, respectively.

Based on the interpolatory methods in (14), the nonlinear Luenberger-type filter in (2) can be represented by the following interpolation filter.

$$\begin{aligned}\dot{\widehat{x}}(t) = \sum_{i=1}^{N}\sum_{j=1}^{N} I_i(\widehat{x}(t))I_j(\widehat{x}(t)) \\ (F_i\widehat{x}(t) + L_i(y(t) - H_j\widehat{x}(t)))\end{aligned} \quad (15)$$

where $I_i(\widehat{x}(t))$ denotes the ith interpolation functions of nonlinear Luenberger-type filter, and L_i denotes the ith local filter gain of $l(\widehat{x}(t))$ in (2).

Given the interpolation methods discussed above, designing a mixed H_2/H_∞ filter for the nonlinear stochastic system described by (1) and (2) involves solving N^2 coupled H_2/H_∞ Riccati-like algebraic equations, under the assumption that the coupled HJIEs in (12) and (13) admit a Lyapunov solution $V\left(\widetilde{x}(t)\right) = \widetilde{x}^T(t)P\widetilde{x}(t)$ for some positive definite matrix $P = P^T > 0$. For highly nonlinear systems characterized by (1), solving N^2 coupled Riccati-like equations for L_i, *i = 1, ..., N* in (15) requires significant computational effort and time[7, 8, 10]. Moreover, if the number N of local linear systems is large, computing $\widehat{x}(t)$ using the interpolation filter defined in (15) at each time step demands substantial computational resources, highlighting the current challenges of nonlinear filter design in this area.

3. Mixed H_2/H_∞ DNN-based AI-driven nonlinear filter via coupled H_2/H_∞ HJIEs-supervised adam learning algorithm

To circumvent the challenge of concurrently solving $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial\widetilde{x}(t)}\right)$ for the filter gain $l^*(\widehat{x}(t))$ in (9) from two coupled H_2/H_∞ HJIEs (12) and (13) using conventional methods for the mixed H_2/H_∞ filter (2), an attractive alternative is to employ a robust DNN-based AI. This DNN-based AI can be trained to simultaneously address $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial\widetilde{x}(t)}\right)$ for $HJIE^1(t) = 0$ in (12) and $HJIE^2(t) = 0$ in (13) in order to determine $l^*(\widehat{x}(t))$ in (9), $v^*(t)$ in (10), and $n^*(t)$ in (11) simultaneously. This approach leverages a supervised deep learning algorithm specifically tailored for handling two-coupled H_2/H_∞ HJIEs in practical applications.

This study proposes a two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm for DNN-based AI-driven filter scheme to achieve the mixed H_2/H_∞ filter strategy of nonlinear stochastic signal systems, as illustrated in **Figure 1**. Specifically, it coordinates with nonlinear system model in (1), Luenberger filter in (2) and estimation error system in (4) with the worst-case $v^*(t)$ and $n^*(t)$, to develop a H_2/H_∞ HJIEs-supervised Adam learning algorithm for DNN-based AI-driven filtering scheme capable of achieving simultaneous optimal H_2 and robust H_2 state estimation for the nonlinear stochastic system (1). The design process consists of two distinct phases: an off-line pre-training phase and an on-line operation phase.

In the off-line pre-training phase, due to the unavailability of $f(x(t))$, $g(x(t))$, and $h(x(t))$ required by the two-coupled H_2/H_∞ HJIEs in (12) and (13), we approximate them using the synthesized state $x(t) = \widehat{x}(t) + \widetilde{x}(t)$. Here, $\widehat{x}(t)$ is obtained from the output of the Luenberger-type filter in (2), and $\widetilde{x}(t)$ represents the output of the estimation error system in (4). In this phase, after computing $f(x(t))$, $g(x(t))$, and $h(x(t))$ and obtaining $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial\widetilde{x}(t)}\right)_\in$ from the DNN output, we calculate $\left[HJIE_\in^1(t),\ HJIE_\in^2(t)\right]^T = \left[\in_1(\theta_i(t)),\ \in_2(\theta_i(t))\right]^T$, where $\in_1(\theta_i(t))$ and $\in_1(\theta_i(t))$ represent the errors with respect to the ideal values 0. During pre-training, the error vector between $\left[\in_1(\theta_i(t)),\ \in_2(\theta_i(t))\right]^T$ and $[0,\ 0]^T$ is fed back to the DNN, facilitating training using a two-coupled H_2/H_∞ HJIEs-supervised deep learning-based Adam algorithm in the sequel. This process aims to train the DNN to accurately predict $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial\widetilde{x}(t)}\right)$ as shown in **Figure 2**. Moreover, as depicted in **Figure 1**, the output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial\widetilde{x}(t)}\right)$ from the DNN is utilized to compute $l^*(\widehat{x}(t))$, $v^*(t)$, and $n^*(t)$. These values contribute to generating $y(t)$ through the nonlinear stochastic system in (1), $\widehat{x}(t)$ via the Luenberger filter in (2), and $\widetilde{x}(t)$ through the estimation error system in (4) for the $(i+1)$th step of the training process using the two-coupled H_2/H_∞ HJIEs -supervised deep learning-based Adam algorithm for DNN-based AI-driven filter scheme in the off-line training phase in the sequel.

Remark 2. *In this study, by substituting the unavailable $v(t)$ and $n(t)$ with the accessible worst-case $v^*(t)$ from (10) and $n^*(t)$ from (11) in the nonlinear stochastic system described by (1), the outputs $y(t)$ (from (1)), estimated state $\widehat{x}(t)$ (from (2)), and state estimation error $\widetilde{x}(t)$ (from (3)) are generated during the off-line pre-training phase simultaneously. Importantly, this substitution has no impact on the performance of the mixed H_2/H_∞ DNN-based AI-driven filter. This is because the design of the mixed H_2/H_∞ filter inherently* *considers the worst-case external disturbance $v^*(t)$ (from (10)) and worst-case measurement noise $n^*(t)$ (from (11)) as specified in Theorem 1.*

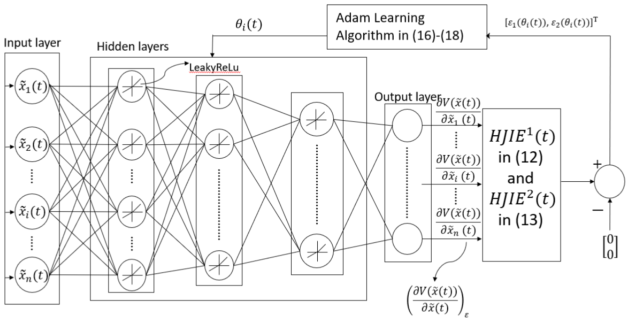

Figure 2. The core structure of proposed approach involves a two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm DNN with $\tilde{x}(t)$ as input and $\left(\partial V\left(\tilde{x}(t)\right)/\partial \tilde{x}(t)\right)$ as output. This DNN aims to simultaneously solve $HJIE^1(t)$ in (12) and $HJIE^2(t)$ in (13) following pre-training using the Adam learning algorithm (16), (17), and (18). The pre-training focuses on minimizing the errors $HJIE^1_\in(t) = \in_1(\theta_i(t))$ and $HJIE^2_\in(t) = \in_2(\theta_i(t))$, where $\theta_i(t)$ denotes the weighting vector of neurons in the hidden layers of the DNN. The objective is to adjust $\theta_i(t)$ iteratively to minimize the combined squared error $\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$ by the H_2/H_∞ HJIEs-supervised Adam learning algorithm as specified in (16)–(18).

As the coupled error $[\in_1(\theta_i(t)),\ \in_2(\theta_i(t))]^T \rightarrow [0,\ 0]^T$ during the pre-training phase, the behavior of $[HJIE^1_\in(t),\ HJIE^2_\in(t)]^T$ approaches $\left[HJIE^1(t),\ \ HJIE^2(t)\right]$. This transition signifies that the DNN-based AI-driven H_2/H_∞ filter scheme depicted in **Figure 1** achieves towards the mixed H_2/H_∞ filter strategy outlined in (7) and (8). Consequently, the off-line pre-training phase depicted in **Figure 1** shifts into the online operational phase.

During the online operation phase, the availability of external disturbance $v(t)$ and measurement noise $n(t)$ renders the substitution of $v^*(t)$ and $n^*(t)$ unnecessary to generate outputs $y(t)$, $\hat{x}(t)$, and $\tilde{x}(t)$ as defined in (1), (2), and (3), respectively. Therefore, there is no requirement to restrain the DNN using the two-coupled HJIEs-supervised Adam deep learning algorithm during the online phase. However, if $|\in_1(\theta_i(t))| > \in$ or $|\in_2(\theta_i(t))| > \in$ for a specified threshold $\in > 0$, it indicates that $[\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T$ should be fed back into the two-coupled H_2/H_∞ HJIEs-supervised Adam deep learning algorithm to train DNN. This feedback mechanism aims to enhance the filtering performance of the proposed mixed H_2/H_∞ DNN-based AI-driven filter scheme in operation phase as illustrated in **Figure 1**.

The core structure of the proposed mixed H_2/H_∞ DNN-based AI-driven filter scheme, supervised by two coupled H_2/H_∞ HJIEs, is depicted in **Figure 2**. It comprises an input layer, multiple hidden layers, and an output layer. Each hidden layer is equipped with neurons employing the LeakyReLU activation function, as described in[16].

$$a\left(\tilde{x}(t)\right) = \begin{cases} \alpha_1\ \tilde{x}(t) & if\ \ \tilde{x}(t) > 0 \\ \alpha_2\ \tilde{x}(t) & if\ \ \tilde{x}(t) \leq 0 \end{cases}$$

where α_1 and α_2 are of constant within (0,1).

The following two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm is proposed to minimize the objective function $[\in_1(\theta_i(t)), \in_2(\theta_i(t))][\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T = \in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$ [10, 15, 18, 22, 23]:

$$\theta_i(t) = \theta_{i-1}(t) - \frac{l}{\sqrt{\hat{v}_i(t)+\tau}}\hat{m}_i(t), \quad i = 1, \ldots, I \tag{16}$$

$$\hat{m}_i(t) = \frac{m_i(t)}{1-\lambda_1^i},\ \hat{v}_i(t) = \frac{v_i(t)}{1-\lambda_2^i} \tag{17}$$

$$\begin{aligned} m_i(t) &= \lambda_1 m_{i-1}(t) + (1-\lambda_1)g_i(t) \\ v_i(t) &= \lambda_2 v_{i-1}(t) + (1-\lambda_2)g_i^2(t) \end{aligned} \tag{18}$$

$$\begin{aligned} g_i(t) &= \frac{\partial}{\partial\theta_i(t)}\sqrt{\frac{1}{N}\sum_{i=1}^{N}(\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t)))} \\ &= \frac{\partial}{\partial\theta_i(t)}\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(HJIE^1_{\in(\theta_i(t))}(t)^2 + HJIE^2_{\in(\theta_i(t))}(t)^2\right)} \end{aligned}$$

where $\theta_i(t)$ represents the vector of weighting coefficients in the hidden layers of the DNN, which are trained to produce $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial\tilde{x}(t)}\right)$, as depicted in **Figure 2**. Here, l denotes the learning rate and I denotes the number of training steps at time *t*. $\hat{m}_i(t)$ and $\hat{v}_i(t)$ are bias-corrected estimators. $g_i(t)$ denotes the gradient vector of the objective function, which is the root mean square error (RMSE) of a batch of $\frac{1}{N}\sum_{i=1}^{N}\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$. $\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$ represents the quadratic error of the two-coupled $[HJIE^1_\in(t),\ HJIE^2_\in(t)]$ in (12) and (13), which serves as the supervisor for the Adam learning algorithm in (16)–(18) to minimize $\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$, aiming for $\in_1^2(\theta_i(t)) = \in_2^2(\theta_i(t)) = 0$ or $HJIE^1_\in(t), = HJIE^2_\in(t) = 0$. N represents the batch size. In (18), λ_1, $\lambda_2 \in (0,\ 1)$ denote coefficients specifying the previous impact on the current gradient direction, designed to prevent the algorithm from getting trapped in local minima, incorporating the momentum concept to accelerate learning. τ in (16) is a small value used to prevent division by zero in the denominator. $m_i(t)$ and $v_i(t)$ in (18) are moving averages of the gradient and square gradient at time t, respectively. $v_i(t)$ adjusts adaptively, starting large and decreasing near the minimum to benefit from adaptive learning rates. Currently, the Adam learning algorithm in (16)–(18) is recognized as a potent deep learning technique for stochastic optimization with global linear convergence [23].

The Adam learning algorithm, supervised by two coupled H_2/H_∞ HJIEs as detailed in (16)–(18), leverages the benefits of RMSProp and momentum. This adaptive learning method efficiently addresses the task of solving $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ when $\left[HJIE^1(t),\ HJIE^2(t)\right]^T = [0,\ 0]^T$ for $l^*(\hat{x}(t))$, $v^*(t)$, and $n^*(t)$ in (9)–(11). It demonstrates robust convergence performance and offers straightforward implementation [20], building on the widely adopted Adam algorithm used extensively in optimizing neural networks. Hence, it is selected for application in the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in (16)–(18) for DNN-based AI-driven filter scheme in this study.

Remark 3. *(i) The proposed mixed H_2/H_∞ DNN-based AI-driven filter differs from conventional big data-driven DNN-based AIs used in applications such as speech recognition [11] and image classification [14]. In this study, we employ the Luenberger filter described in (2), the estimation error system in (4), and the theoretical results (9)–(12) from the mixed H_2/H_∞ filter design strategy, where two-coupled conditions $HJIE^1(t) = 0$ in (12) and $HJIE^2(t) = 0$ in (13) are utilized. These two conditions supervise the Adam learning algorithm in (16)–(18), ensuring convergence to $HJIE^1_\in(t) = 0$ and $HJIE^2_\in(t) = 0$. This convergence ensures that the output $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_\in$ of the pre-trained DNN converges to $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$. The supervised Adam learning algorithm described in (16)–(18) guarantees that the weighting parameter vector $\theta_i(t)$ of the DNN converges to the stochastic optimal solution [23]. Specifically, with a sufficiently large number of neurons in the hidden layers and training steps I in (16), $\theta_i(t)$ can converge linearly to a globally optimal parameter vector $\theta_i^*(t)$ for minimizing $\in_1^2(\theta_i(t)) + \in_2^2(\theta_i(t))$ [23]. (ii) A robust H_∞ DNN-based filter of nonlinear stochastic system was introduced in [17]. Since the robust H_∞ filter is more conservative with high filter gain and the optimal H_2 filter is more appealing in the conventional designs, the mixed H_2/H_∞ filter design of nonlinear stochastic systems is proposed in this study. Further, it is more difficult and challenging to solve $HJIE^1(t) = 0$ in (12) and $HJIE^2(t) = 0$ in (13) simultaneously in mixed H_2/H_∞ nonlinear filter design than to solve $HJIE^1(t)$ for H_∞ nonlinear filter in [19].*

During the off-line pre-training process of two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in **Figure 1**, the DNN output $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_\in$ is sent to two-coupled H_2/H_∞ HJIEs in (12) and (13) as follows:

$$
\begin{aligned}
HJIE^1_\in(t) &= \tilde{x}^T(t)Q\,\tilde{x}(t) + \left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in (f(x(t)) - f(\hat{x}(t))) + \tfrac{1}{4r^2}\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in g(x(t))g^T(x(t)) \\
&\quad \times \left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in - r^2(h(x(t)) - h(\hat{x}(t)))^T(h(x(t)) - h(\hat{x}(t)))^T \\
&= \in_1(\theta_i(t)) \qquad (19) \\
HJIE^2_\in(t) &= \tilde{x}^T(t)Q\,\tilde{x}(t) + \left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in (f(x(t)) - f(\hat{x}(t))) + \tfrac{1}{2r^2}\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in g(x(t))g^T(x(t)) \\
&\quad \times \left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)^T_\in - 2r^2(h(x(t)) - h(\hat{x}(t)))^T(h(x(t)) - h(\hat{x}(t)))^T = \in_2(\theta_i(t)) \qquad (20)
\end{aligned}
$$

where the terms $\in_1(\theta_i(t)) = HJIE^1_\in(t) - HJIE^1(t)$ and $\in_2(\theta_i(t)) = HJIE^2_\in(t) - HJIE^2(t)$ represent the discrepancies between the two coupled H_2/H_∞ HJIEs defined in (12) and (13), and their counterparts predicted by the DNN during the pre-training phase, as described in (19) and (20). The state $x(t) = \hat{x}(t) + \tilde{x}(t)$ in (19) and (20) is accessible via $\hat{x}(t)$ through the Luenberger filter in (2), and via $\tilde{x}(t)$ through the state estimation error dynamics in (4) as illustrated in **Figure 1**.

The discrepancy $\left[\in_1(\theta_i(t)), \in_2(\theta_i(t))\right]^T$ as defined in (19) and (20) serves as feedback to train the DNN using the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in (16)–(18) during the pre-training phase. Consequently, the DNN is expected to output $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ to derive the mixed H_2/H_∞ filter gain $l^*(\hat{x}(t))$ in (9), the worst-case $v^*(t)$ in (10), and $n^*(t)$ in (11) following the completion of the offline pre-training phase, as illustrated in **Figure 1**.

As the approximation error $\left[HJIE_{\in}^1(t),\ HJIE_{\in}^2(t)\right]^T = \left[\in_1(\theta_i(t)), \in_2(\theta_i(t))\right]^T$ approaches to $[0,\ 0]^T$ by the two coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm (16)–(18), we can prove that the output $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_{\in}$ of DNN will approach $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ of two-coupled H_2/H_∞ HJIEs in (12) and (13), i.e., $\left[HJIE_{\in}^1(t),\ HJIE_{\in}^2(t)\right]^T \to \left[HJIE^1(t),\ HJIE^2(t)\right]^T = [0,\ 0]^T$ in the following theorem.

Theorem 2. *If* $\left[\in_1(\theta_i(t)), \in_2(\theta_i(t))\right]^T = \left[HJIE_{\in}^1(t),\ HJIE_{\in}^2(t)\right]^T$ *in (19) and (20) approaches* $[0,\ 0]^T$ *through the two-coupled* H_2/H_∞ *HJIEs-supervised Adam learning algorithm in (16)–(18), then the output* $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_{\in}$ *of the DNN in (19) and (20) will approximate* $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ *in (12) and (13). This means* $\left[HJIE_{\in}^1(t),\ HJIE_{\in}^2(t)\right]^T$ *in (19) and (20) will approach* $\left[HJIE^1(t),\ HJIE^2(t)\right]^T$ *in (12) and (13), respectively. The HJIEs-supervised DNN-based filter gain* $l^*(\hat{x}(t)) = \frac{2r^2}{\left\|\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_{\in}\right\|}\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)_{\in}\left(H(x(t)) - H(\hat{x}(t))\right)$ *in* ***Figure 1*** *will then achieve the mixed* H_2/H_∞ *filter gain* $l^*(\hat{x}(t))$ *in (9).*

Proof. See **Appendix B**. □

Remark 4. *(i) According to Theorem 2 and Theorem 1, the proposed two-coupled* H_2/H_∞ *HJIEs-supervised Adam learning algorithm for DNN-based AI-driven filtering scheme in* ***Figure 1*** *is capable of generating* $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ *to derive the mixed* H_2/H_∞ *filter gain* $l^*(\hat{x}(t))$ *in (9) as* $\left[\in_1(\theta_i(t)), \in_2(\theta_i(t))\right]^T \to [0,\ 0]^T$*. In practical applications, however, the off-line pre-training phase transitions to the operational phase once* $|\in_1(\theta_i(t))|$ *and* $|\in_2(\theta_i(t))| \leq \delta$ *for a small prescribed* δ*, or when the pre-training steps reach a specified number* I *in (16)–(18). (ii) Unlike the conventional big data-driven supervised DNN-based AI learning schemes used for binary classification tasks like image classification or speech recognition (i.e., yes or no outcomes), the proposed two-coupled* H_2/H_∞ *HJIEs-supervised learning algorithm for DNN-based AI-driven filtering scheme is applied to achieve mixed* H_2/H_∞ *state estimation of the nonlinear stochastic system in (1). In this study, the nonlinear Luenberger-type filter in (2) and the two-coupled* H_2/H_∞ *HJIEs act as supervisors for the Adam learning algorithm in (16)–(18) to train the DNN-based AI to solve* $\left(\frac{\partial V\left(\tilde{x}(t)\right)}{\partial \tilde{x}(t)}\right)$ *for the two coupled* H_2/H_∞ *HJIEs in (19) and (20) of the mixed* H_2/H_∞ *filter scheme. Thus, this approach allows substantial savings in training data and time compared to conventional big data-driven DNN-based AI learning schemes, leveraging the system model and theoretical mixed* H_2/H_∞ *filtering results (9)–(13) to achieve complex mixed* H_2/H_∞ *DNN-based AI-driven filter designs for state estimation of nonlinear stochastic systems of man-made machines for more practical applications.*

To facilitate the training of the DNN-based AI-driven filter using the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm for practical applications, the continuous nonlinear stochastic system in (1) must be transformed into the corresponding sampled-data nonlinear stochastic system.

$$\begin{aligned} x(t+h) &\approx (x(t) + hf(x(t))) + hg(x(t))v(t) \\ y(t) &= h(x(t)) + n(t) \end{aligned} \tag{21}$$

where h is the sampling period. Consequently, the Luenberger-type filter in (2) and the estimation error system in (4) need to be changed, respectively, as follows:

$$\hat{x}(t+h) = (\hat{x}(t) + hf(\hat{x}(t))) + hl(\hat{x}(t))(y(t) - h(\hat{x}(t))) \tag{22}$$

and

$$\begin{aligned} \tilde{x}(t+h) = &\left(\tilde{x}(t) + hf(x(t)) - hf(\hat{x}(t))\right) \\ &+ hl(\hat{x}(t))(h(x(t)) - h(\hat{x}(t))) \\ &+ hg(x(t))v(t) - hl(\hat{x}(t))n(t) \end{aligned} \tag{23}$$

Accordingly, the flow chart of two-coupled H_2/H_∞ HJIEs-supervised DNN-based AI-driven filter scheme of nonlinear stochastic sample data system in (21)–(23) is modified in **Figure 3**. For the convenience of design, the off-line pre-training process is given as Algorithm 1.

Upon completing the off-line pre-training of the mixed H_2/H_∞ DNN-based AI-driven filter using the aforementioned two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in Algorithm 1, the process transitions to the on-line operational phase of the mixed H_2/H_∞ DNN-based AI-driven filter. During this phase, since $v(t)$ and $n(t)$ are directly available, there is no need for the information of $v^*(t)$ and $n^*(t)$. Thus, at each time step, only $\widetilde{x}\ (t)$ is required as input for the DNN to produce $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$, which is used to compute the filter gain $l^*(\widehat{x}(t))$ via (9) for the Luenberger-type mixed H_2/H_∞ filter in (22) and the estimation error system in (23). However, if the absolute error $|\in_1(\theta_i(t))|$ or $|\in_2(\theta_i(t))| \ \geq \ \delta$ for a prescribed error threshold δ, it necessitates retraining the DNN using the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in (16)–(18), ensuring it does not disrupt the operational phase process of the mixed H_2/H_∞ DNN-based AI-driven filter.

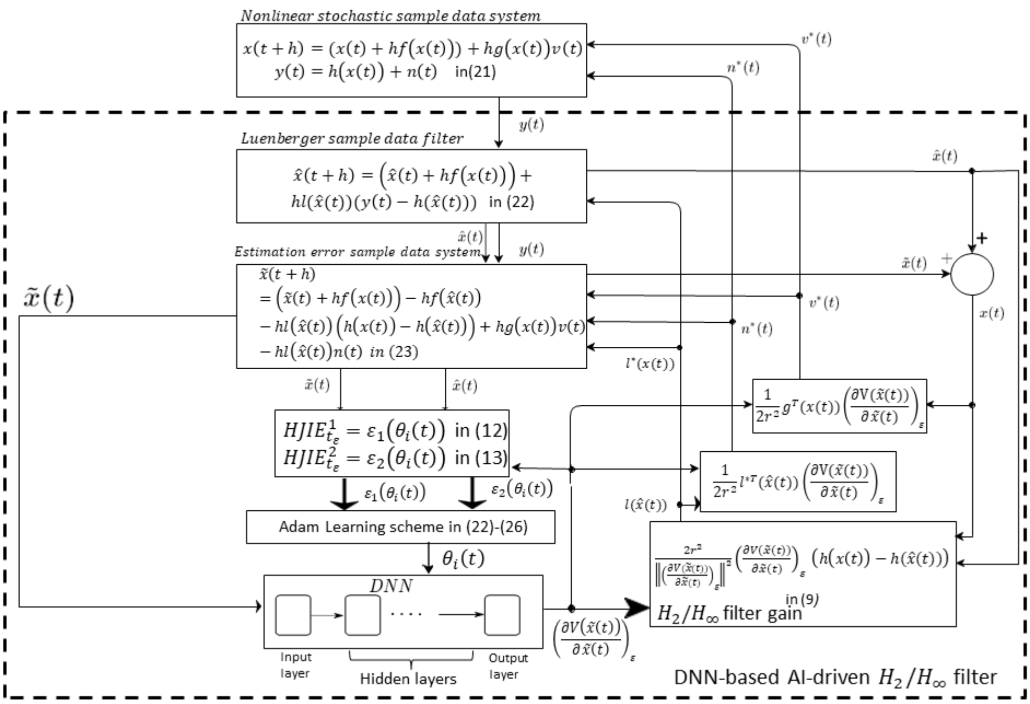

Figure 3. The flow chart of mixed H_2/H_∞ DNN-based AI-driven filtering scheme of nonlinear stochastic sample-data systems in (21) and (22) with sampling period h. The pre-training process of the nonlinear sample-data system is the same as nonlinear stochastic system in **Figure 1** and is given in Algorithm 1.

Remark 5. *Before applying the proposed H_2/H_∞ HJIEs-supervised Adam learning scheme in the off-line pre-training phase to train the DNN, it is essential to preprocess the data. This preprocessing includes normalization and standardization to mitigate the impact of potentially large estimation errors $\widetilde{x}\ (t)$ on the training efficiency of the deep learning methodologies proposed. Therefore, it is necessary to standardize these sample data before feeding them into the DNN. Standardization serves the dual purpose of enhancing the estimation accuracy of the mixed H_2/H_∞ DNN-based AI-driven filter and accelerating its training speed.*

Remark 6. *Currently, there is no efficient deep learning approach available to implement the mixed H_2/H_∞ DNN-based AI-driven filter for nonlinear signal systems in (1), Luenberger filter in (2) and mixed H_2/H_∞ filter in Theorem 1 and H_2/H_∞ HJIEs-supervised DNN-based AI that integrates theoretical insights into nonlinear H_2/H_∞ filtering strategy with the deep Adam learning algorithm. Conventional deep learning algorithms are primarily designed for training DNNs as classifiers or recognizers. The challenge lies in the fact that conventional big data-driven DNN-based AIs require extensive empirical data, making it challenging to realize a cohesive design of the nonlinear mixed H_2/H_∞ filtering scheme and the supervised Adam algorithm learning DNN-based AI for robust H_∞ and optimal H_2 state estimation of nonlinear stochastic signal systems simultaneously.*

Algorithm 1. Two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm for DNN-based AI-driven H_2/H_∞ filter scheme of nonlinear stochastic system

Input: Time step t, sampling time h, terminal time t_f, amplitude A, state vector $x(t)$, state estimation $\hat{x}(t)$ of $x(t)$, system functions $f(x(t))$, $g(x(t))$ and $h(x(t))$, weighting matrix Q, attenuation level r^2, a small prescribed value δ, missile coefficient β, gravity constant g, external disturbance $v(t)$, measurement noise $n(t)$

for $t \leftarrow 0$ to t_f ($+h$, total t) **do**
    $\tilde{x}(t) \leftarrow x(t) - \hat{x}(t)$
    Calculate $f(x(t))$, $f(\hat{x}(t))$, $h(x(t))$, $h(\hat{x}(t))$, $g(x(t))$ and $g(\hat{x}(t))$
    Inputs in the Input Layer of DNN $\leftarrow \tilde{x}(t)$
    $\frac{\partial V(\tilde{x}(t))}{\partial \tilde{x}(t)}\big|_\epsilon \leftarrow$ Obtain the partial differential of a Lyapunov function from the Output Layer of DNN by using DNN
    $HJIE_\epsilon^1, HJIE_\epsilon^2 \leftarrow$ Equation (19) and Equation (20) with $\frac{\partial V(\tilde{x}(t))}{\partial \tilde{x}(t)}\big|_\epsilon$, $f(\tilde{x}(t))$ and $h(\tilde{x}(t))$, respectively
    if $|\epsilon_1(\theta_i(t))| \geq \delta$ or $|\epsilon_2(\theta_i(t))| \geq \delta$ or $t < t_f$ **then**
        $l^*(\tilde{x}(t))$, $v^*(t)$, $n^*(t) \leftarrow$ Calculate from Equation (9)–Equation (11), respectively
        $x(t+h)$, $\tilde{x}(t+h)$, $v(t+h) \leftarrow$ Equation (21)–Equation (23), respectively
        Train the DNN with $\tilde{x}(t+h)$ and calculate $HJIE_\epsilon^1(t+h)$, $HJIE_\epsilon^2(t+h)$ by using the output of DNN
    Else
        Calculate H_2/H_∞ attenuation level $\leftarrow$ by Equation (30)
    end if
end for

4. Simulation example

In this section, after building upon the flow chart of the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm for mixed H_2/H_∞ DNN-based AI-driven filtering scheme depicted in **Figure 3** for the nonlinear stochastic sample-data system described by (21), including the sample-data Luenberger-type filter presented in (22) and estimation error sample data system, we provide a design example with computer simulation. This example focuses on the mixed H_2/H_∞ trajectory estimation problem of an incoming ballistic missile following trajectories akin to Eugen Sanger or Tsien Hsueshen[24], observed via sensor measurements within a radar system. The objective is to validate the trajectory estimation performance of the incoming ballistic missile using the proposed two-coupled H_2/H_∞ HJIEs-supervised mixed H_2/H_∞ DNN-based AI-driven filtering scheme, as illustrated in **Figure 4**.

In **Figure 4**, an incoming ballistic missile, maneuvering with either Eugen Sanger or Tsien Hsueshen trajectory[24], is targeting an origin-bound target and detected by a radar sensor located at $(x,\ y,\ z)\ =\ (300,\ 300,\ 0)$. The ballistic missile's flight trajectory is governed by the guidance control law $u(t)$ associated with Eugen Sanger or Tsien Hsueshen trajectory[24], aimed at evading precise estimation and subsequent interception by anti-missile defenses. Considering external disturbances $v(t)$ affecting the missile and measurement noise $n(t)$ from the radar sensor, the dynamics of both the incoming ballistic missile and the radar detection system can be described by the following nonlinear stochastic system[9, 25]:

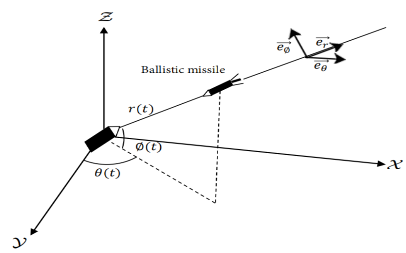

Figure 4. The incoming ballistic missile followed a trajectory known as the Tsien Hsueshen trajectory[24]. The x axis, y axis, and z axis represent the downrange, offrange, and attitude of the missile, respectively. The missile's target is situated at the origin, and a detection radar is positioned at $(x, y, z) = (300, 300, 0)$ to estimate the missile's trajectory using the proposed mixed DNN-based AI-driven H_2/H_∞ filter with sensor measurements $Y(t)$.

$$\begin{aligned} \dot{X}(t) &= f(X(t)) + u(t) + g(X(t))v(t) \\ Y(t) &= h(X(t)) + n(t) \end{aligned} \tag{24}$$

where $X(t)\ = [x(t),\ y(t),\ z(t),\ \dot{x}(t),\ \dot{y}(t),\ \dot{z}(t)]^T$.

$$f(X(t)) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \\ \frac{-\rho(z(t))g\sqrt{\dot{x}^2(t)+\dot{y}^2(t)+\dot{z}^2(t)}}{2\beta}\dot{x}(t) \\ \frac{-\rho(z(t))g\sqrt{\dot{x}^2(t)+\dot{y}^2(t)+\dot{z}^2(t)}}{2\beta}\dot{y}(t) \\ \frac{-\rho(z(t))g\sqrt{\dot{x}^2(t)+\dot{y}^2(t)+\dot{z}^2(t)}}{2\beta}\dot{z}(t) - g \end{bmatrix} \quad (25)$$

$$g(X(t)) = diag[0_{1\times 3},\ 1,\ 1,\ 1] \quad (26)$$

$$h(X(t)) = [x(t) - 300,\ y(t) - 300,\ z(t), \dot{x}(t),\ \dot{y}(t),\ \dot{z}(t)]^T \quad (27)$$

where $x(t)$, $y(t)$ and $z(t)$ denote the target-centered Cartesian coordinates of the incoming ballistic missile, g is the gravity constant, $\rho(z(t))$ denotes the density of the atmosphere at the position of the incoming ballistic missile defined in the following [25, 26]:

$$\rho(z(t)) = \begin{cases} \rho_h e^{-\alpha_h z(t)},\ \rho_h = 1.75,\ \alpha_h = 1.49 \times 10^{-4}, if\ z(t) \geq 9144\, meters \\ \rho_l e^{-\alpha_l z(t)},\ \rho_l = 1.227,\ \alpha_l = 1.093 \times 10^{-4}, if\ z(t) < 9144\, meters \end{cases}$$

For the convenience of representation, let us denote the state vector of the ballistic missile detection system in **Figure 4** as follows:

$$X(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix}$$

To steer the ballistic missile along a Tsien Hsueshen trajectory [24] and prevent precise detection followed by interception from anti-missile systems, the following guidance strategy is proposed for the ballistic missile:

$$\begin{aligned} u(t) = kX(t) = [&-0.1x_1(t) - x_4(t) + a(t), \\ &-0.1x_2(t) - x_5(t), -0.08x_3(t) - x_6(t), \\ &-e^{-0.1t}\sin(10t)x_4(t), -0.01x_5(t), -0.01x_6(t)]^T \end{aligned} \quad (28)$$

where $a(t)$ is the following wave function:

$a(t) = Ae^{-0.1t}\frac{4}{\pi}sin(t)$ where A represents the amplitude value, set to 15000 in the simulation, and $e^{-0.1t}$ acts as a decreasing function over time. In our simulation, as time progresses, the missile's wavelike trajectory transitions towards a straight path to approach the target. As the maneuvered missile approaches the target, its trajectory needs to be more direct to ensure accurate impact. Therefore, the aforementioned decreasing function is multiplied with $a(t)$ to attenuate the influence of the wave function.

Then the nonlinear ballistic stochastic missile system in (24) under the maneuvering flight by the guidance control strategy in (28) can be given as follows:

$$\begin{aligned} \dot{X}(t) &= f(X(t)) + g(X(t))v(t) \\ Y(t) &= h(X(t)) + n(t) \end{aligned} \quad (29)$$

where $f(X(t)) = \begin{bmatrix} -0.1x_1(t) + a(t) \\ -0.1x_2(t) \\ -0.08x_3(t) \\ \frac{-\rho(x_3(t))g\sqrt{x_4^2(t)+x_5^2(t)+x_6^2(t)}}{2\beta}x_4(t) - e^{-0.1t}\sin(10t)x_4(t) \\ \frac{-\rho(x_3(t))g\sqrt{x_4^2(t)+x_5^2(t)+x_6^2(t)}}{2\beta}x_5(t) - 0.01x_5(t) \\ \frac{-\rho(x_3(t))g\sqrt{x_4^2(t)+x_5^2(t)+x_6^2(t)}}{2\beta}x_6(t) - 0.01x_6(t) - g \end{bmatrix}$,
$h(X(t)) = [x_1(t) - 300,\ x_2(t) - 300,\ x_3(t), x_4(t),\ x_5(t),\ x_6(t)]^T$ and $g(X(t)) = diag[0_{1\times 3},\ 1,\ 1,\ 1]$.

In **Figure 3**, the radar detection system employs a two-coupled HJIEs-supervised Adam learning algorithm for mixed H_2/H_∞ DNN-based AI-driven filtering scheme. The weighting matrix is set to $Q = 10^{-5}$ I and the attenuation level $r^2 = 0.03$. This scheme is utilized to estimate the trajectory of an incoming ballistic missile following a Tsien trajectory. For ease of design, the nonlinear stochastic missile detection system described in (29) is transformed into a sample-data nonlinear stochastic system as given in (21), with a sampling interval of $h = 0.01$ seconds. Under the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in **Figure 3**, DNN-based AI-driven mixed H_2/H_∞

filter scheme is applied to the nonlinear stochastic sample-data missile system (21)–(23). The DNN architecture in **Figure 2** includes an input layer, four hidden layers, and an output layer. The input layer has 6 inputs, while the four hidden layers sequentially 256, 128, 32, and 6 neurons as output, following a data compression concept.

In **Figure 3**, during the off-line pre-training phase, the DNN-based AI undergoes training using the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm described in (16)–(18). This training is guided by minimizing the errors $HJIE^1_\in(t) = \in_1(\theta_i(t))$ from (19) and $HJIE^2_\in(t) = \in_2(\theta_i(t))$ from (20) to achieve $HJIE^1(t) = 0$ in (12) and$HJIE^2(t) = 0$ in (13). The objective is for the DNN to output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$. Consequently, the filter gain $l^*(\widehat{x}(t))$, the external disturbance $v^*(t)$, and the measurement noise $n^*(t)$ are determined, which replace $l(\widehat{x}(t))$, $v(t)$, and $n(t)$ in (21)–(23).These variables are utilized to compute $Y(t)$, $\widehat{X}(t+h)$, $\widetilde{X}(t+h)$, and $X(t+h) = \widehat{X}(t+h) + \widetilde{X}(t+h)$ for the subsequent training step at $t+h$.

Remark 7. *(i) Since $HJIE^1(t)$ in (12) and $HJIE^2(t)$ in (13) depend on the state $X(t)$ of the stochastic nonlinear system described by the ballistic missile equations in (29), which is not directly available and needs to be estimated from the output measurements $Y(t)$, DNN-based AI-driven filtering scheme cannot directly train the DNN using the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm in (16)-(18). Therefore, $\widehat{X}(t)$ is generated by the filter model in (22), and $\widetilde{X}(t)$ represents the estimation error modeled in (23). This allows AI-driven filter to obtain $X(t)$ as $X(t) = \widehat{X}(t) + \widetilde{X}(t)$ to satisfy $HJIE^1(t) = 0$ in (12) and $HJIE^2(t) = 0$ in (13), enabling the training of the DNN-based AI to achieve the mixed H_2/H_∞ DNN-based AI-driven filter scheme. (ii) Since the external disturbance $v(t)$ and measurement noise $n(t)$ are not directly available for the stochastic missile dynamics in (29), it is challenging to generate the output $Y(t)$ using the sample-data stochastic system model in (21) and to model $\widetilde{X}(t)$ accurately in (23). Therefore, $v(t)$ and $n(t)$ are replaced with worst-case estimates $v^*(t)$ from (10) and $n^*(t)$ from (11). This substitution enables the generation of $Y(t)$ in (21), $\widehat{X}(t)$ in (22), and $\widetilde{X}(t)$ in (23). Subsequently, $X(t)$ is computed as $X(t) = \widehat{X}(t) + \widetilde{X}(t)$ to satisfy $HJIE^1(t) = 0$ in (12) and $HJIE^2(t) = 0$ in (13), facilitating the training of the DNN using the two-coupled H_2/H_∞ supervised Adam learning algorithm in (16)–(18) during each time step t in the off-line pre-training phase.(iii) Additionally, to solve $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ required for obtaining the filter gain $l^*\left(\widehat{X}(t)\right)$ in (9), $v^*(t)$ in (10), and $n^*(t)$ in (11), a calculation block $\left[HJIE^1(t),\ HJIE^2(t)\right]^T = [\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T$ is integrated into the final step of the DNN pre-training using the Adam learning algorithm (16)-(18). This adjustment allows the DNN to approach the desired output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$. Finally, the output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ of the DNN is utilized to compute the filter gain $l^*\left(\widehat{X}(t)\right)$ in (9), $v^*(t)$ in (10), and $n^*(t)$ in (11) for the subsequent training phase at $t+h$, as illustrated in **Figure 3**.*

In this simulation example, the goal of DNN-based AI-driven filter is to train a DNN using the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm (16)–(18) to output the nonlinear function $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ and thereby obtain the mixed H_2/H_∞ filter gain $l^*\left(\widehat{X}(t)\right)$ with fewer neurons in each hidden layer of the DNN. Generally, if the learning rate in (16) is initially too large, the optimization process might become stuck in a local minimum or result in slow training speed. The parameters λ_1 and λ_2 in (17) influence the current direction based on previous directions and should not be set too small due to their dependence on historical information. In this simulation, the parameters l, λ_1, λ_2, and τ of the Adam learning algorithm are set to 0.005, 0.9, 0.999, and 10^{-7}, respectively. The training steps I and batch size N are chosen as 40 and 50 respectively, with a sampling time of 0.01 seconds. Both the environmental disturbance $v(t)$ and measurement noise $n(t)$ are modeled as $20 \times N(0,\ 1)$. Using the proposed two-coupled H_2/H_∞-supervised Adam learning algorithm, the trajectory $X(t) = [x_1(t),\ ...,\ x_6(t)]^T$ of the incoming ballistic missile in (29) is simulated starting from the initial condition $X(0) = [150000, 210000, 120000, -2500, -2500, -2500]^T$ and $\widehat{X}(0) = [140000, 204000, 111000, -2200, -2800, -2100]^T$. Additionally, 2000 random inputs are selected around these initial conditions as training inputs.

Remark 8. *In practical applications of the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm to DNN-based AI-driven filter, it is essential to gather a large number of training points around the state estimator error*

$\tilde{X}(t)$. *During the off-line training phase, random data points are selected, which means that during online operation, the state* $X(t) = \hat{X}(t) + \tilde{X}(t)$ *may initially be far from the training data. However, over time, it gradually converges towards* $\left(\frac{\partial V(\tilde{x}(t))}{\partial \tilde{x}(t)}\right)$ *to approach the actual solution of two coupled HJIEs.*

The robustness (attenuation) level r in (5) and its real attenuation level of external disturbance and measurement noise of the proposed DNN-based AI-driven filter scheme in this simulation example are given as follows:

$$\frac{\int_0^{70}\tilde{X}^T(t)Q\tilde{X}(t)dt - V\left(\tilde{x}(0)\right)}{\int_0^{70}\left(v^T(t)v(t)+n^T(t)n(t)\right)dt} \approx 0.004 \leq 0.03 = r^2 \quad (30)$$

The improved H_∞ attenuation level of 0.004, compared to the specified level of 0.03, for mitigating the impact of external disturbance and measurement noise on the state estimation of a ballistic missile can be attributed to two main reasons: (i) The inclusion of an optimal H_2 filter scheme alongside the robust H_∞ filter scheme provides enhanced performance. (ii) The prescribed robust H_∞ filtering performance with $r^2 = 0.03$ is designed to handle worst-case scenarios involving external disturbance $v^*(t)$ (from (10)) and measurement noise $n^*(t)$ (from (11)). In our simulation scenario, both $v(t)$ and $n(t)$ are modeled as white noise with $20 * N(0,\ 1)$, which do not represent the worst-case $v^*(t)$ and $n^*(t)$.

Using the proposed H_2/H_∞ DNN-based AI-driven filter, **Figure 5** illustrates the trajectory and its estimation of an incoming ballistic missile maneuvering with a Tsien Hsueshen trajectory. **Figure 6** displays the position, and **Figure 7** shows the velocity along with their respective estimations obtained through the proposed H_2/H_∞ DNN-based AI-driven filter scheme. From the simulation results depicted in these figures, it is evident that the proposed DNN-based AI-driven H_2/H_∞ filter scheme for the radar system can effectively estimate the Tsien Hsueshen trajectory of the incoming ballistic missile. This is achieved despite the presence of unknown environmental disturbances $v(t)$, measurement noise $n(t)$, and maneuvering effects $u(t)$. The actual H∞ state estimation performance of the incoming ballistic missile system by the proposed two-coupled H_2/H_∞ HJIEs-supervised learning DNN-based AIdriven filter for the radar system is evaluated with comparison as follows:

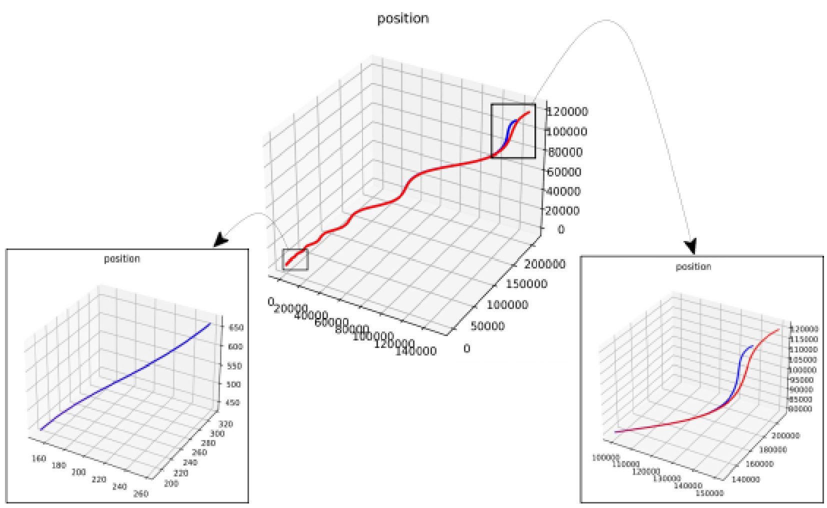

Figure 5. The maneuvered Tsien Hsueshen trajectory of the ballistic missile and the estimated trajectory by the proposed two-coupled H_2/H_∞ HJIEs-supervised DNN-based AI-driven filter. It can be seen that the trajectory $X(t)$ of ballistic missile (the red line) and its estimation $\hat{X}(t)$ (the blue line) show that the asymptotic filtering ability is achieved.

The proposed two-coupled HJIEs-supervised DNN-based AI-driven filtering scheme depicted in **Figure 1** successfully achieves the specified H_2/H_∞ filtering performance, effectively reducing the impact of random external disturbance $v(t)$ and measurement noise $n(t)$ on the estimation accuracy of the Tsien Hsueshen trajectory of maneuvered ballistic missiles under the guidance control strategy (28). Notably, the filtering performance of this approach surpasses both the prescribed value $\rho = 0.03$ and the performance of the DNN H_∞ DNN-based AI-driven filter with $\rho = 0.012$

as reported in[17]. This improvement is attributed to the inclusion of an optimal H_2 filtering criterion in the design process. The mixed H_2/H_∞ filter is specifically designed to minimize H_2 filtering performance while considering worst-case scenarios of external disturbance $v^*(t)$ (from (10)) and measurement noise $n^*(t)$ (from (11)), which are typically more severe than those encountered in real-world situations. In our simulation example, $v(t)$ and $n(t)$ are modeled as $20 * N(0, 1)$, representing nominal disturbances and noise levels, which contribute to achieving a better performance than the $\rho = 0.03$ attenuation level for H_∞ filters. From the simulation results shown in **Figures 6** and **8**, it is evident that the proposed H_2/H_∞ DNN-based filter outperforms the results depicted in **Figures 7** and **9** of the mixed H_2/H_∞ global linearization filter with $N = 8$ in (15) as described in[8], especially in accurately estimating missile velocities (**Figures 8** and **9**).

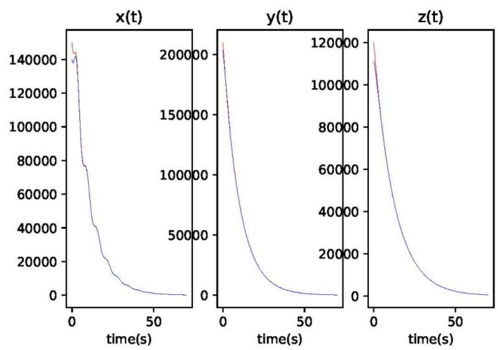

Figure 6. The positions in downrange, off range, and vertical directions $(x(t), y(t), z(t))$ along with their respective estimations $(\widehat{x}(t), \widehat{y}(t), \widehat{z}(t))$of the incoming ballistic missile following a Tsien Hsueshen trajectory are depicted using the proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning DNN-based AI-driven filter. In the figure, the red line represents the actual position of the incoming ballistic missile $x(t)$, while the blue line illustrates the estimated position generated by the proposed DNN-based AI-driven filter using two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm.

In the proposed H_2/H_∞ DNN-based AI-driven filter scheme, following the off-line pre-training of the DNN, the filter achieves the asymptotic trajectory estimation of an incoming ballistic missile maneuvered with a Tsien Hsueshen trajectory, despite finite energy external disturbances $v(t)$ and measurement noise $n(t)$. The enhanced state estimation performance can be attributed to several factors: (i) The filter design is based on achieving the global nonlinear solution $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ of $\left[HJIE^1(t), HJIE^2(t)\right]^T = [0, 0]^T$ in (12) and (13). (ii) The mixed H_2/H_∞ filter effectively attenuates the impact of random external disturbances and measurement noise to a prescribed level $r = (0.03)^{\frac{1}{2}}$ of robust H_∞ filtering performance while achieving optimal H_2 filtering simultaneously. (iii) The DNN is capable of universally approximating any nonlinear function, including $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$, enabling computation of the H_2/H_∞ filter gain $l^*(\widehat{x}(t))$ in (9) through an efficient two-coupled H_2/H_∞ HJIEs-supervised Adam deep learning algorithm described in (16)–(18).

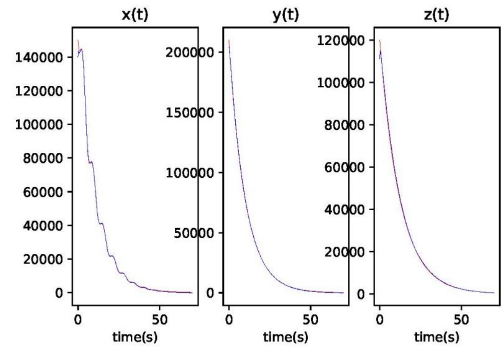

Figure 7. The downrange, off range, and vertical positions $(x(t), y(t), z(t))$ and their corresponding estimations $(\widehat{x}(t), \widehat{y}(t), \widehat{z}(t))$ are estimated using the mixed H_2/H_∞ global linearization filter method described in[8]. Comparing with **Figure 6**, the performance of the mixed H_2/H_∞ global linearization filtering method in[8] is slightly inferior to that of the proposed DNN-based AI-driven filter in **Figure 6**. However, the former method requires interpolating 64 local filters using 64 complex interpolation functions $I_i(\widehat{x}(t)), i = 1, \ldots, 64$ to compute $\widehat{x}(t)$ as given in (15), with $N = 8$ applied at each time sampling.

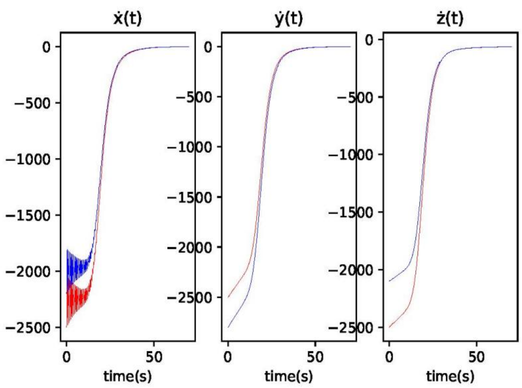

Figure 8. The velocities in downrange, off range, and vertical directions $(\dot{x}(t), \dot{y}(t), \dot{z}(t))$ and their corresponding estimations $(\hat{\dot{x}}(t), \hat{\dot{y}}(t), \hat{\dot{z}}(t))$ are captured by the proposed mixed H_2/H_∞ DNN-based AI-driven filter. In the figure, the red lines represent the actual velocities $(\dot{x}(t), \dot{y}(t), \dot{z}(t))$ of the incoming ballistic missile, while the blue lines depict the estimated velocities $(\hat{\dot{x}}(t), \hat{\dot{y}}(t), \hat{\dot{z}}(t))$ generated by the proposed mixed H_2/H_∞ HJIEs-supervised DNN-based AI-driven filter.

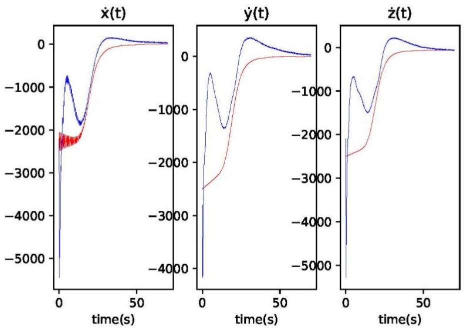

Figure 9. The velocities in downrange, off range, and vertical directions $(\dot{x}(t), \dot{y}(t), \dot{z}(t))$ and their respective estimations $(\dot{\widehat{x}}(t), \dot{\widehat{y}}(t), \dot{\widehat{z}}(t))$ by the mixed H_2/H_∞ global linearization filter[8] are illustrated in this figure. In comparison with **Figure 8**, the filtering performance of the proposed mixed H_2/H_∞ DNN-based AI-driven filter exhibits a significantly superior estimation accuracy. During the simulation, the mixed H_2/H_∞ global linearization filter initially provides reasonably accurate velocity estimations. However, as the simulation progresses, there is noticeable estimation error, particularly when the real velocities gradually approach zero. In contrast, the proposed mixed H_2/H_∞ DNN-based AI-driven filter consistently maintains better performance throughout the simulation period.

5. Conclusions

In this study, we address the challenging design problem of mixed H_2/H_∞ filter for nonlinear stochastic signal systems affected by unknown environmental disturbance and measurement noise. To tackle this, we embed a two-coupled H_2/H_∞ HJIEs-supervised scheme within the Adam learning algorithm to coordinate nonlinear signal system model, Luenberger filter and estimation error model with worst-case external disturbance and measurement noise to generate $y(t)$, $\widehat{x}(t)$, $\widetilde{x}\ (t)$ to train a DNN-based AI-driven filter to achieve the mixed H_2/H_∞ filter strategy shown in **Figure 1** by directly solving the two coupled HJIEs (12) and (13) of the filtering problem. The proposed method leverages DNNs to simplify the otherwise complex process of designing robust nonlinear mixed H_2/H_∞ DNN-based AI-driven filters. Unlike conventional approaches that rely on interpolatory local linearized models of several stochastic systems, our method uses the nonlinear system model, Luenberger filter, and estimation error model to generate $\widehat{x}(t)$ and $\widetilde{x}\ (t)$ under the worst-case scenarios of external disturbances and measurement noise. Supervised by the two-coupled H_2/H_∞ HJIEs theoretical results via Adam learning algorithm, the mixed H_2/H_∞ DNN-based AI-driven filter scheme can be achieved for nonlinear stochastic signal systems. Our co-design of the mixed H_2/H_∞ filter and the two-coupled H_2/H_∞ HJIEs-supervised Adam learning algorithm not only reduces the need for extensive training data and time, typical in big data-driven DNN methods, but also overcomes the analytical and numerical challenges in solving $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ directly from $HJIE^1(t) = 0$ and $HJIE^2(t) = 0$. This approach demonstrates superior performance compared to conventional mixed H_2/H_∞ interpolation filter designs (such as (14)–(15) in[8, 10]) that assume $V(x(t)) = x^T(t)P\, x(t)$. In the simulation example, the proposed mixed H_2/H_∞ DNN-based AI-driven filter achieves better filtering performance in estimating the Tsien Hsueshen trajectory of a ballistic missile detected by a radar system sensor. As AI (DNN) technologies become more accessible and affordable, training these DNNs as sophisticated nonlinear mixed H_2/H_∞ AI-driven filters as shown in **Figure 3** using our proposed two-coupled H_2/H_∞ HJIEs-supervised Adam learning method will become increasingly practical for state estimation of man made machine in signal processing area. Looking ahead, we envision extending the proposed mixed H_2/H_∞ DNN-based AI-driven filter design to applications such as state estimator-based output feedback target tracking control (i.e., the guidance and control) for nonlinear stochastic anti-missile systems under external disturbances, measurement noise, and potential malicious attacks.

Appendix A. Proof of Theorem 1

(i) From the H_∞ filtering strategy in (7), we get

$$\begin{aligned} J_\infty(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) &= \max_{v(t),n(t)} J_\infty(l^*(\widehat{x}(t)), v(t),\ n(t)) \\ &= \max_{v(t),n(t)} E(V(\ \widetilde{x}\ (t_f))) + E\Big(\textstyle\int_0^{t_f} [\widetilde{x}^T(t) Q\ \widetilde{x}\ (t) - r^2\big(v^T(t)v(t) + n^T(t)n(t)\big)]dt\Big) \\ &= \max_{v(t),n(t)} E(V(\ \widetilde{x}\ (0))) + E(\textstyle\int_0^{t_f} [\widetilde{x}^T(t) Q\ \widetilde{x}\ (t) - r^2\big(v^T(t)v(t) + n^T(t)n(t)\big) \qquad \text{(A1)} \\ &\quad + \tfrac{dV(\widetilde{x}(t))}{dt}](dt) \end{aligned}$$

By the chain rule and the estimation error system in (4)

$$\begin{aligned}\frac{dV\left(\widetilde{x}\left(t\right)\right)}{dt} = &\left(\frac{\partial V\left(\widetilde{x}\left(t\right)\right)}{\partial \widetilde{x}(t)}\right)^T \frac{d\widetilde{x}(t)}{dt}\\ = &\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T [f(x(t)) - f(\widehat{x}(t)) - l(\widehat{x}(t))(h(x(t)) - h(\widehat{x}(t))) + g(x(t))v(t) - l(\widehat{x}(t)) \\ &+g(x(t))v(t) - l(\widehat{x}(t))n(t)]\end{aligned} \quad \text{(A2)}$$

Substituting (A2) into (A1), we get

$$\begin{aligned}J_\infty(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = &\max_{v(t),n(t)} E(V(\ \widetilde{x}\ (0))) + E(\textstyle\int_0^{t_f}[\widetilde{x}^T(t)Q\ \widetilde{x}\ (t) + \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T((f(x(t)) - f(\widehat{x}(t))))\\ &-\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T(h(x(t)) - h(\widehat{x}(t))) + \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(\widehat{x}(t))v(t)\\ &-\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T l^*(\widehat{x}(t))n(t) - r^2v^T(t)v(t) - r^2n^T(t)n(t)dt)\end{aligned} \quad \text{(A3)}$$

By the completing square technique of $l^*(\widehat{x}(t))$, $v(t$ and $n(t)$

$$\begin{aligned}J_\infty(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = &\max_{v(t),n(t)} E(V(\ \widetilde{x}\ (0))) + E(\textstyle\int_0^{t_f}[\widetilde{x}^T(t)Q\ \widetilde{x}\ (t) + \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T((f(x(t)) - f(\widehat{x}(t))))\\ &-r^2(h(x(t)) - h(\widehat{x}(t)))^T(h(x(t)) - h(\widehat{x}(t)))\\ &-\left(rv(t) - \frac{1}{2r}g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)^T\left(rv(t) - \frac{1}{2r}g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)\\ &+\frac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\\ &-\left(rn(t) - \frac{1}{2r}l^*(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)^T\left(rn(t) - \frac{1}{2r}l^*(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)\\ &+(\frac{1}{2r}l^{*T}(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) - r(h(x(t)) - h(\widehat{x}(t))))^T(\frac{1}{2r}l^{*T}(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\\ &-r(h(x(t)) - h(\widehat{x}(t))))\end{aligned} \quad \text{(A4)}$$

By $HJIE^1(t) = 0$ in (12) and $l^*(\widehat{x}(t))$ in (9), we get

$$\begin{aligned}J_\infty(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = &\max_{v(t),n(t)} E\{V(\ \widetilde{x}\ (0)) + \textstyle\int_0^{t_f}[-\frac{1}{r}(rv(t) - \frac{1}{2}g^T(x(t)\Big)\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\Big)\Big)^T\\ &\times\left(rv(t) - \frac{1}{2}g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right) - \frac{1}{r}\left(rn(t) - \frac{1}{2}l^*(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)^T\\ &\times\left(rn(t) - \frac{1}{2}l^*(\widehat{x}(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\right)]dt\}\\ = &E\Big\{V\left(\widetilde{x}\ (0)\right)\Big\}\end{aligned} \quad \text{(A5)}$$

And we obtain the worst-case external disturbance $v^*(t)$ in (10) and measurement noise $n^*(t)$ in (11). Similarly, from the H_2 filtering strategy in (8), we get

$$\begin{aligned}J_2(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = &\min_{l(\widehat{x}(t))} E\{V(\ \widetilde{x}\ (t_f)) + \textstyle\int_0^{t_f}\left(\widetilde{x}^T\left(t\right)Q\ \widetilde{x}\left(t\right)\right)dt\}\\ = &\min_{l(\widehat{x}(t))} E\{V(\ \widetilde{x}\ (0)) + \textstyle\int_0^{t_f}(\widetilde{x}^T(t)Q\ \widetilde{x}\ (t) + \frac{dV\left(\widetilde{x}\left(t\right)\right)}{dt})dt\}\end{aligned} \quad \text{(A6)}$$

From (A2), we get

$$\begin{aligned}
J_2(l^*(\widehat{x}(t)),v^*(t),\ n^*(t)) = &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}\widetilde{x}^T(t)Q\widetilde{x}(t)+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T[(f(x(t))-f(\widehat{x}(t)))\\
&-l(\widehat{x}(t))(h(x(t))-h(\widehat{x}(t)))+g(x(t))v^*(t)-l(\widehat{x}(t))n^*(t)]dt\}\\
= &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}[\widetilde{x}^T(t)Q\widetilde{x}(t)+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T(f(x(t))-f(\widehat{x}(t)))\\
&+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T l(\widehat{x}(t))(h(x(t))-h(\widehat{x}(t)))+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T g(x(t))v^*(t)\\
&-\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T l(\widehat{x}(t))n^*(t)]dt\} \qquad \text{(A7)}
\end{aligned}$$

Substituting the worst-case $v^*(t)$ in (10) and $n^*(t)$ in (11) into (A7), we get

$$\begin{aligned}
J_2(l^*(\widehat{x}(t)),v^*(t),\ n^*(t)) = &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}[\widetilde{x}^T(t)Q\widetilde{x}(t)+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T(f(x(t))-f(\widehat{x}(t)))\\
&+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T l(\widehat{x}(t))(h(x(t))-h(\widehat{x}(t)))+\frac{1}{2r^2}\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T\\
&\times g(x(t))g^T(x(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)+\frac{1}{2r^2}\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T l(\widehat{x}(t))l^T(\widehat{x}(t))\\
&\times\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)]dt\}\\
= &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}[\widetilde{x}^T(t)Q\widetilde{x}(t)+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T(f(x(t))-f(\widehat{x}(t)))\\
&+\frac{1}{2r^2}\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)\\
&+\left(\frac{1}{\sqrt{2}r}l^{*T}(\widehat{x}(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)^T\left(\frac{1}{\sqrt{2}r}l^{*T}(\widehat{x}(t))\right.\\
&\left.\times\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)\\
&+2r^2(h(x(t))-h(\widehat{x}(t)))^T(h(x(t))-h(\widehat{x}(t)))]dt\}\\
= &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}[\widetilde{x}^T(t)Q\widetilde{x}(t)+\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T(f(x(t))-f(\widehat{x}(t)))+2r^2\\
&\times(h(x(t))-h(\widehat{x}(t)))^T(h(x(t))-h(\widehat{x}(t)))+\frac{1}{2r^2}\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)^T g(x(t))\\
&\times g^T(x(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)+\left(\frac{1}{\sqrt{2}r}l^T(\widehat{x}(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)^T\\
&\times\left(\frac{1}{\sqrt{2}r}l^T(\widehat{x}(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)]dt\} \qquad \text{(A8)}
\end{aligned}$$

By $HJIE^2(t)=0$ in (13), we get

$$\begin{aligned}
J_2(l^*(\widehat{x}(t)),v^*(t),\ n^*(t)) = &\min_{l(\widehat{x}(t))} E\{V(\widetilde{x}(0))+\int_0^{t_f}[\left(\frac{1}{\sqrt{2}r}l^T(\widehat{x}(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)^T\\
&\times\left(\frac{1}{\sqrt{2}r}l^T(\widehat{x}(t))\left(\frac{\partial V(\widetilde{x}(t))}{\partial\widetilde{x}(t)}\right)-\sqrt{2}r(h(x(t))-h(\widehat{x}(t)))\right)]dt\} \qquad \text{(A9)}
\end{aligned}$$

Then we get the H_2 optimal case $l^*(\widehat{x}(t))$ in (9), and (A9) becomes

$$J_2(l^*(\widehat{x}(t)),v^*(t),\ n^*(t)) = E\{V(\widetilde{x}(0))\} \qquad \text{(A10)}$$

i.e., with the worst-case $v^*(t)$ in (10) and $n^*(t)$ in (11) to replace $v(t)$ in and $n(t)$, we can obtain the optimal H_2 filtering gain $l^*(\widehat{x}(t))$ in (9).

(ii) Due to the fact that $E\{V(x(0))\}$ is finite, from (A5) and (A10), $J_\infty(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = E\left\{V\left(\widetilde{x}\ (0)\right)\right\} < \infty$, $J_2(l^*(\widehat{x}(t)), v^*(t),\ n^*(t)) = E\left\{V\left(\widetilde{x}\ (0)\right)\right\} < \infty$ and $v(t)$ and $n(t) \in \mathcal{L}_F^2[0,\ \infty)$ i.e. $E\{\int_0^\infty v^T(t)v(t)dt\} < \infty$ and $E\{\int_0^\infty n^T(t)\, n(t)dt\} < \infty$. From (A1) and (A6), they imply $E\left\{\int_0^\infty \widetilde{x}^T(t) Q\ \widetilde{x}\ (t)dt\right\} < \infty$. It implies $E[\widetilde{x}^T(t)\ \widetilde{x}\ (t)] \to 0$ as $t \to \infty$. i.e., the mean square asymptotical filtering ability can be obtained by the mixed H_2/H_∞ filter. Q.E.D.

Appendix B. Proof of Theorem 2

Suppose $[\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T \to [0,\ 0]^T$ and there still exists an error function $e(\widetilde{x}\ (t))$, between $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in$ of $[HJIE_\in^1(t),\ HJIE_\in^2(t)]^T$ in (19), (20) and $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ of $[HJIE^1(t),\ HJIE^2(t)]$ in (12), (13) in the following.

$$\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in = \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) + e\left(\widetilde{x}\ (t)\right) \tag{B1}$$

In the following, we will prove that as $[\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T \to [0,\ 0]^T$, $e\left(\widetilde{x}\left(t\right)\right) \to 0$, i.e., $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in \to \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$. From (12) and (19), we get

$$\begin{aligned}\in_1(\theta_i(t)) = {} & HJIE_\in^1(t) - HJIE^1(t) \\ = {} & (\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in^T - \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T)(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in^T g(x(t))g^T(x(t)) \\ & \times\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in - \tfrac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)\end{aligned} \tag{B2}$$

Substituting $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_\in$ in (B1) into (B2), we can get

$$\begin{aligned}\in_1(\theta_i(t)) = {} & e^T(\widetilde{x}\ (t))(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{4r^2}e^T(\widetilde{x}\ (t))g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) \\ & + \tfrac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))e^T(\widetilde{x}\ (t)) + \tfrac{1}{4r^2}e^T(\widetilde{x}\ (t))g(x(t))g^T(x(t))e^T(\widetilde{x}\ (t))\end{aligned} \tag{B3}$$

By the following symmetric fact of the terms in (B3)

$$e^T\left(\widetilde{x}\ (t)\right)g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) = \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))e^T\left(\widetilde{x}\left(t\right)\right) \tag{B4}$$

Then (B3) becomes

$$\begin{aligned}\in_1(\theta_i(t)) = {} & e^T(\widetilde{x}\ (t))(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{2r^2}e^T(\widetilde{x}\ (t))g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) + \tfrac{1}{2r^2}e^T(\widetilde{x}\ (t)) \\ & \times g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) + \tfrac{1}{4r^2}e^T\left(\widetilde{x}\ (t)\right)g(x(t))g^T(x(t))e^T\left(\widetilde{x}\ (t)\right) \\ = {} & e^T\left(\widetilde{x}\ (t)\right)[(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{2r^2}g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) + \tfrac{1}{4r^2}g(x(t)) \\ & \times g^T(x(t))e^T(\widetilde{x}\ (t))]\end{aligned} \tag{B5}$$

Similarly, we get

$$\begin{aligned}\in_2(\theta_i(t)) = \;& HJIE^2_{\in}(t) - HJIE^2(t) \\ = \;& (\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T_{\in} - \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T)(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{4r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T_{\in} g(x(t))g^T(x(t)) \\ & \times\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_{\in} - \tfrac{1}{2r^2}\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) \qquad \text{(B6)}\end{aligned}$$

By substituting $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)^T$ in (B1) into (B6) and by similar procedure in (B4) and (B5), we get

$$\begin{aligned}\in_2(\theta_i(t)) = \;& e^T\left(\widetilde{x}\ (t)\right)[(f(x(t)) - f(\widehat{x}(t))) + \tfrac{1}{2r^2}g(x(t))g^T(x(t))\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right) + \tfrac{1}{2r^2}g(x(t))g^T(x(t)) \\ & \times e^T(\widetilde{x}\ (t))] \qquad \text{(B7)}\end{aligned}$$

Since the term in $[\cdot]$ in (B5) and (B7) are not equal to 0 for all $x(t)$, $\widetilde{x}\ (t)$ and $\widehat{x}(t)$, as $[\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T \to [0,\ 0]^T$ as $t \to \infty$, it implies the error function $e^T\left(\widetilde{x}\ (t)\right) \to 0$. From (B1), $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_{\in} \to \left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)$ as $[\in_1(\theta_i(t)), \in_2(\theta_i(t))]^T \to [0,\ 0]^T$. Therefore, the supervised learning DNN-based filter design $l^{\in}(\widehat{x}(t)) = \frac{2r^2}{\left\|\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_{\in}\right\|}$ $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_{\in}(h(x(t)) - h(\widehat{x}(t)))^T$ based on the output $\left(\frac{\partial V\left(\widetilde{x}(t)\right)}{\partial \widetilde{x}(t)}\right)_{\in}$ of DNN in **Figure 1** will approach the mixed H_2/H_∞ filter gain $l^*(\widehat{x}(t))$ in (9) for the Luenberger-type filter in (2) for the nonlinear stochastic system in (1). Q.E.D.

References

[1] Anderson, B.D.O., Moore, J.B., 1979. Optimal Filtering. Prentice Hall: Englewood Cliffs, NJ, USA.

[2] Khargonekar P.P., Rotea M.A., Baegens E., 1996. Mixed H_∞ filtering. International Journal of Robust and Nonlinear Control. 6(6), 313–330. DOI: https://doi.org/10.1002/(SICI)1099-1239(199605)6:4<313::AID-RNC235>3.0.CO;2-8

[3] Gao H., Lam J., Xie L., et al., 2005. New approach to mixed H_2/H_∞ filtering for polytopic discrete-time systems. IEEE Transactions on Signal Processing. 53(8), 3183–3192. DOI: https://doi.org/10.1109/TSP.2005.851116

[4] Li, H., Fu, M., 1997. A linear matrix inequality approach to robust H_∞ filtering. IEEE Transactions on Signal Processing. 45(9), 2338–2350. DOI: https://doi.org/10.1109/78.622956

[5] Palhares, R.M., Peres, P.L.D., 2001. LMI approach to the mixed H_2/H_∞ filtering design for discrete-time systems. IEEE Transactions on Aerospace and Electronic Systems. 37(1), 292–296. DOI: https://doi.org/10.1109/7.913689

[6] Zhang, W., Chen, B.S., Tseng, C.S., 2005. Robust filtering for nonlinear stochastic systems. IEEE Transactions on Signal Processing. 53(2), 589–599. DOI: https://doi.org/10.1109/WCICA.2008.4593266

[7] Chen, B.S., Tsai, C.L., Chen, D.S., 2003. Robust H_∞ and mixed H_2/H_∞ filter for equalization designs of nonlinear communication systems: Fuzzy interpolation approach. Fuzzy interpolation approach. IEEE Transactions on Fuzzy Systems. 11(3), 384–398. DOI: https://doi.org/10.1109/TFUZZ.2003.812698

[8] Chen, B.S., Chen, W.H., Wu, H.L., 2009. Robust mixed H_2/H_∞ global linearization filter design for nonlinear stochastic systems. IEEE Transactions on Circuits and Systems I. 56(7), 1441–1454. DOI: https://doi.org/10.1109/TCSI.2008.2007059

[9] Chen, B.S., Wu, C.F., 2015. Robust H_∞ scheduling filter design for a class of nonlinear stochastic system. IEEE Transactions on Signal Processing. 63(23), 6245–6257. DOI: https://doi.org/10.1109/TSP.2015.2465305

[10] Chen, B.S., Lee, H.C., Wu, C.F., 2015. Pareto optimal filter design for nonlinear stochastic systems via multiobjective H_2/H_∞ optimization. IEEE Transactions on Fuzzy Systems. 23(2), 387–399. DOI: https://doi.org/10.1109/TFUZZ.2014.2312985

[11] Dahl, G.E., Yu, D., Dang, L., Acero, A., 2012. Contex-dependent pretrained deep neural networks for large-vocabulary speech recognition. IEEE Transactions on Audio, Speech, and Language Processing. 20(1), 30–42. DOI: https://doi.org/10.1109/TASL.2011.2134090

[12] Eshratifar, A.E., Esmaili, A., Pedram, M., 2019. Bot-

tleNet: A deep learning architecture for intelligent mobile cloud computing service. ISLPED 2019, Lausanne, Switzerland. pp. 1–6.

[13] Maas, A.L., Qi, P., Xie, Z., et al., 2017. Building DNN acoustic models for large vocabulary speech recognition. Computer Speech and Language. 41, 195–213. DOI: https://doi.org/10.1016/j.csl.2016.06.007

[14] Shin H.-C., Roth, H.-R., Gao, M., et al, 2016. Deep convolutional neural networks for computer aided detection: CNN architectures dataset characteristics and transfer learning. IEEE Transactions on Medical Imaging. 35(5), 1285–1298. DOI: https://doi.org/10.1109/TMI.2016.2528162

[15] Sutton, R.S., Barto, A.G., 2015. Reinforcement Learning: An Introduction, 2nd ed., in progress, Bradford Books.

[16] Wooldridge M., 2022. What is missing from contemporary AI? The world Intelligent Computing. 2022, 9847630.

[17] Yin, J., Su, S., Xun, J., et al., 2020. Data-driven approaches for modeling train control: Comparison and case studies. ISA Transactions. 98(5), 349–363. DOI: https://doi.org/10.1016/j.isatra.2019.08.024

[18] Bertsekas Dimitri, P., 2019. Reinforcement Learning and Optimal Control. Athena Scientific.

[19] Chen, B.S., Wu, P.S., Lee, M.Y., 2021. Robust H_∞ deep neural network based filter design of nonlinear stochastic systems. IEEE Access. 9, 165103–1651199. DOI: https://doi.org/10.1109/ACCESS.2021.3133899

[20] Zhang, W., Xie, L., Chen, B.S., 2015. Stochastic H_2/H_∞ Control. CRC Press.

[21] Sakamoto, S., Vander Schaft, A.J., 2008. Analytical approximation methods for the stabilization solution of Hamilton-Jacobi equation. IEEE Transactions on Automatic Control. 53(10), 2335–2350. DOI: https://doi.org/10.1109/TAC.2008.2006113

[22] Dauphin, Y.N., Vries, H.D., Chung, J., et al., 2015. RMSProp and equilibrated adaptive learning rates for nonconvex optimization. Neural Networks. arxiv preprint. arxiv:1502.04390. pp. 1–9.

[23] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. ICLR. San Diego. arxiv preprint. arxiv:1412.6980. pp. 1–15.

[24] Cui, H., Liu, W., Zhu, S., et al., 2016. A hypersonic vehicle tracking algorithm based on the UKF Generalized labeled multi bernoulli filter. Proceedings of the 2016 35th Chinese Control Conference (CCC); Chengdu, China; 27–29 July 2016.. Volume 35, pp. 4911–4916.

[25] Siouris, G.M., Chen, G., Wang, J., 1997. Tracking an incoming ballistic missile using and extended internal kalman filter. IEEE Transactions on Aerospace and Electronic Systems. 33(1), 232–240. DOI: https://doi.org/10.1109/7.570753

[26] Chen, B.-S., Lee, M.-Y., Chen X.-H., 2020. Security-enhanced filter design for stochastic systems under malicious attack via smoothed signal model and multiobjective H_2/H_∞ estimation method. IEEE Transactions on Signal Processing. 68, 4971–4986. DOI: https://doi.org/10.1109/TSP.2020.3019136