Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 56-2 (2023) 9436–9441

Safety Monitoring and Alert for Neural Network-Enabled Control Systems ⋆

Jianglin Lan

James Watt School of Engineering, University of Glasgow, Glasgow G12 8QQ, UK (e-mail: Jianglin.Lan@glasgow.ac.uk).

Abstract: This paper considers the safety monitoring and enhancement for neural network-enabled control systems with disturbance and measurement noise. A robustly stable interval observer is designed to generate sound lower and upper bounds of the system state. The obtained interval is used to monitor the runtime system state and predict the one-step ahead future system trajectory, providing system safety monitoring and alert. The simulation results of a numerical example and an adaptive cruise control system demonstrate efficacy of the observer in runtime system monitoring and its potentials in detecting sensor faults and enhancing system safety.

Keywords: Safety, neural network, observer, fault detection, intelligent autonomous vehicles.

1. INTRODUCTION

The recent rapid development of machine learning techniques, with neural networks (NNs) as the dominant type of models, have revealed their power in modelling, perception, and control for dynamic systems such as robots and autonomous vehicles (Moe et al., 2018; Hewing et al., 2020; Tang et al., 2022). This paper focuses on NN-enabled control systems, where NNs are used to model the nonlinear system dynamics (Zhou et al., 2022) or generate control actions (Dai et al., 2021). One challenge of applying NNs to control systems is lacking performance guarantee in the continuous operation space as NNs are trained on discrete samples of system trajectories. Another challenge is the vulnerability of NNs to perturbations, noise and adversarial attacks (Huang et al., 2020). This is even more problematic for NN-enabled control systems such as aircraft (Julian and Kochenderfer, 2021), because uncertainties in the NN components will be propagated and accumulated through the closed loop. It is thus critical to provide safety assurance of NN-enabled control systems via developing safety monitoring techniques that can provide real-time and predictive information of system safety.

The past decade has seen the development of many formal methods for verifying safety of NNs in the open-loop settings such as image classification and natural language processing (Liu et al., 2021). These methods are usually based on solving optimisation problems like mixed-integer linear programming (MILP) (Lomuscio and Maganti, 2017), semidefinite programming (SDP) (Lan et al., 2022), or linear programming (LP) combined with the branch-and-bound technique (Bunel et al., 2020), requiring huge computational resources. For example, verification of certain safety properties in the next-generation airborne collision avoidance system for unmanned aircraft (ACAS Xu) takes more than 100 hours (Katz et al., 2017). Hence, the above methods are unsuitable for runtime safety verification of NN-enabled control systems with limited onboard computing resources.

Safety of NN-enabled control systems can be verified via computing the reachable set that contain all the possible future system trajectories for a given initial state set and examining inclusion of the reachable set within the safe region. Reachability analyses have been performed based on MILP (Xiang et al., 2019; Karg and Lucia, 2020), SDP (Hu et al., 2020), LP with input partition (Everett et al., 2021), or constrained zonotopes (Zhang and Xu, 2022). However, these reachability analysis is performed in the offline setting and their reliance on solving computationally expensive optimisation problems or set operations make them unappealing for runtime safety verification. Runtime safety monitoring for NN-based aircraft taxiing is achieved by a rule-based approach, where a set of monitors continuously measure the aircraft positions relative to the runway and behaviour of the NN relative to its training data (Cofer et al., 2020). Inspired by the observer design techniques (Efimov et al., 2013; Tang et al., 2019), an interval observer is proposed in Xiang (2021) for monitoring the state of continuous-time systems. The interval observer consists of two dynamic systems which estimate the runtime lower and upper bounds of the system state. The observer gains are computed from linear matrix inequality (LMI) problems that are much lighter than MILP and SDP. However, the work (Xiang, 2021) does not consider system disturbance and measurement noise.

This paper advances the current state of the art with a new interval observer for discrete-time NN-enabled control systems subject to disturbance and measurement noise. The main contributions of this paper are as follows:

1) A robustly stable interval observer is proposed for NN-enabled control systems with disturbance and measurement noise. The proposed design is simpler than Xiang (2021) as it requires computing a single gain rather than two gains. Also, estimation robustness is not considered in Xiang (2021).

⋆ This work was supported by a Leverhulme Trust Early Career Fellowship under Award ECF-2021-517.

10.1016/j.ifacol.2023.10.237

2) The proposed observer generates sound and robust intervals for the system state and output, useful for runtime safety monitoring and predictive fault detection and safety alerting.
3) The design is applied to monitor and enhance vehicle safety of an adaptive cruise control (ACC) system.

Notations. The symbol $\mathbb{R}^n$ denotes the n-dimensional Euclidean space, $\mathbb{R}^{n\times n}$ denotes the set of $n\times n$ matrices, and $\mathbb{R}^{n\times n}_+$ denotes the $\mathbb{R}^{n\times n}$ matrices with non-negative elements. $\mathcal{L}^n_\infty$ denotes the set of all n-dimensional ∞-norm bounded functions. $\text{diag}(a_1,\cdots,a_n)$ represents a diagonal matrix with main diagonals $a_1,\cdots,a_n$ and zero elsewhere. $\mathbf{0}$ and I denote respectively a zero matrix and an identity matrix with the dimensions known from the context. $\star$ indicates symmetry in a matrix. For a matrix X, $X\prec$ (or $\succ$)0 indicates that it is symmetric positive (or negative) definite, $X^+=\max\{0,X\}$, $X^-=X^+-X$, and $|X|=X^++X^-$. For two matrices $X_1,X_2\in\mathbb{R}^{n\times n}$, $X_1\le X_2$ is defined component-wise.

2. PROBLEM DESCRIPTION AND PRELIMINARIES

2.1 Problem Description

Consider a discrete-time system represented by

$$x_{t+1}=Ax_t+f_{\text{NN}}(y_t,u_t)+w_t, \tag{1a}$$
$$y_t=Cx_t+v_t, \tag{1b}$$

where $x_t\in\mathbb{R}^n$, $u_t\in\mathbb{R}^m$, $w_t\in\mathbb{R}^n$, $y_t\in\mathbb{R}^p$, and $v_t\in\mathbb{R}^p$ are the vectors of system state, inputs (reference/control input), disturbances, measured outputs, and noise, respectively. $A\in\mathbb{R}^{n\times n}$ and $C\in\mathbb{R}^{p\times n}$ are known constant matrices. t is the sampling step. f_{NN} is a L-layer feedforward NN capturing the nonlinear system dynamics or as a stabilising control policy and it is defined as:

$$z_0=[y_t^\top,u_t^\top]^\top, \tag{2a}$$
$$z_\imath=\phi_\imath(W_\imath z_{\imath-1}+b_\imath),\ \imath\in[1,L], \tag{2b}$$
$$f_{\text{NN}}(z_0)=z_L, \tag{2c}$$

where $W_\imath$ and $b_\imath$ are the $\imath$-th layer weight matrix and bias vector, respectively. $\phi_\imath(\cdot)$, $\imath\in[1,L]$, are the activation functions for the hidden layers $\imath\in[1,L-1]$ and the last layer $\imath=L$. Without loss of generality, this paper considers that all the activation functions in the hidden layers are of the same type such as a `ReLU` (Rectified Linear Unit) or `tanh` functions (Dubey et al., 2022), and the last layer is a `purelin` function $z_L=W_Lz_{L-1}+b_L$. All the above activation functions are monotonically non-decreasing (Dubey et al., 2022).

Assumption 1. The pair (A,C) is observable. $x_0\in[\underline{x}_0,\bar{x}_0]$ for some known $\underline{x}_0,\bar{x}_0\in\mathbb{R}^n$. $u_t\in[\underline{u}_t,\bar{u}_t]$, $w_t\in[\underline{w}_t,\bar{w}_t]$ and $v_t\in[\underline{v}_t,\bar{v}_t]$ for all $t\ge0$, where $\underline{u}_t,\bar{u}_t\in\mathcal{L}^m_\infty$, $\underline{w}_t,\bar{w}_t\in\mathcal{L}^n_\infty$ and $\underline{v}_t,\bar{v}_t\in\mathcal{L}^p_\infty$ are known signals.

This paper aims to design a robustly stable observer (in the form of an interval observer) that can: (i) provide a real-time estimated lower and upper bounds of the state and (ii) be used to monitor the runtime value of state and predict future state trajectory. The second capability will be valuable for checking whether the state trajectory is within a safe region, detecting possible sensor faults or attacks and providing safety alert in advance.

2.2 Preliminaries

We recall Lemmas 2 and 3 that are to be used in the design.

Lemma 2. (Efimov et al., 2013) Given a vector $x\in\mathbb{R}^n$ satisfying $x\in[\underline{x},\bar{x}]$ and a constant matrix $M\in\mathbb{R}^{m\times n}$, it holds that $M^+\underline{x}-M^-\bar{x}\le Mx\le M^+\bar{x}-M^-$.

Lemma 3. (Farina and Rinaldi, 2000) A matrix $X\in\mathbb{R}^{n\times n}_+$ is Schur stable if and only if there exists a diagonal matrix $P\succ0$ such that $X^\top PX-P\prec0$.

The design of an interval observer for the system (1) needs the intervals of x_0, w_t, u_t, v_t and $f_{\text{NN}}(y_t,u_t)$. The intervals of x_0, w_t, u_t and v_t are known from Assumption 1, while the interval of $f_{\text{NN}}(y_t,u_t)$ needs to be derived. It follows from Assumption 1 and Lemma 2 that $C^+\underline{x}_t-C^-\bar{x}_t+\underline{v}_t\le y_t\le C^+\bar{x}_t-C^-\underline{x}_t+\bar{v}_t$. Let $z_0=[y_t^\top,u_t^\top]^\top$, then $\underline{z}_0\le z_0\le\bar{z}_0$, where $\underline{z}_0=[\underline{y}_t^\top,\underline{u}_t^\top]^\top$ and $\bar{z}_0=[\bar{y}_t^\top,\bar{u}_t^\top]^\top$. Similar to Theorem 1 in Xiang (2021), this paper considers two auxiliary matrices $\underline{W}_\imath$ and $\overline{W}_\imath$ for each weight matrix $W_\imath$. The (i,j) elements $\underline{W}_\imath^{i,j}$ and $\overline{W}_\imath^{i,j}$ are defined as

$$\underline{W}_\imath^{i,j}=\begin{cases}W_\imath^{i,j},\ W_\imath^{i,j}<0\\ 0,\quad W_\imath^{i,j}\ge0\end{cases},\ \overline{W}_\imath^{i,j}=\begin{cases}W_\imath^{i,j},\ W_\imath^{i,j}\ge0\\ 0,\quad W_\imath^{i,j}<0\end{cases}, \tag{3}$$

where $W_\imath^{i,j}$ is the (i,j) element of $W_\imath$. Then $f_{\text{NN}}(z_0)$ satisfies $\underline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)\le f_{\text{NN}}(z_0)\le\overline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)$, with $\underline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)$ and $\overline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)$ being given by

$$\begin{cases}\underline{z}_\imath=\phi_\imath(\underline{W}_\imath\underline{z}_{\imath-1}+\overline{W}_\imath\underline{z}_{\imath-1}+b_\imath),\ \imath\in[1,L]\\ \underline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)=\underline{z}_L\end{cases}, \tag{4a}$$

$$\begin{cases}\bar{z}_\imath=\phi_\imath(\underline{W}_\imath\bar{z}_{\imath-1}+\overline{W}_\imath\bar{z}_{\imath-1}+b_\imath),\ \imath\in[1,L]\\ \overline{f}_{\text{NN}}(\underline{z}_0,\bar{z}_0)=\bar{z}_L\end{cases}. \tag{4b}$$

3. INTERVAL OBSERVER DESIGN

The system (1) can be rewritten as

$$x_{t+1}=(A-L_oC)x_t+f_{\text{NN}}(y_t,u_t)+L_oy_t-L_ov_t+w_t \tag{5}$$

for any matrix $L_o\in\mathbb{R}^{n\times p}$.

By replacing w_t, L_ov_t and $f_{\text{NN}}(y_t,u_t)$ in (5) with their intervals, we propose the interval observer:

$$\begin{aligned}\bar{x}_{t+1}=&A_o\bar{x}_t+\overline{f}_{\text{NN}}(\underline{y}_t,\bar{y}_t,\underline{u}_t,\bar{u}_t)+L_oy_t\\&-L_o^+\underline{v}_t+L_o^-\bar{v}_t+\bar{w}_t,\end{aligned} \tag{6a}$$

$$\begin{aligned}\underline{x}_{t+1}=&A_o\underline{x}_t+\underline{f}_{\text{NN}}(\underline{y}_t,\bar{y}_t,\underline{u}_t,\bar{u}_t)+L_oy_t\\&-L_o^+\bar{v}_t+L_o^-\underline{v}_t+\underline{w}_t,\end{aligned} \tag{6b}$$

where $A_o=A-L_oC$ and L_o is the design gain matrix.

Precision of the observer (6) can be evaluated by measuring the width of its interval $\delta x_t=\bar{x}_t-\underline{x}_t$. The dynamics of the interval width are derived as

$$\delta x_{t+1}=A_o\delta x_t+\delta f_{\text{NN}}+|L_o|\delta v_t+\delta w_t, \tag{7}$$

where $\delta v_t=\bar{v}_t-\underline{v}_t$, $\delta w_t=\bar{w}_t-\underline{w}_t$, $\delta f_{\text{NN}}=\overline{f}_{\text{NN}}-\underline{f}_{\text{NN}}$ and $|L_o|=L_o^++L_o^-$.

The time response of δx_t is derived as

$$\delta x_t=A_o^t\delta x_0+\sum_{i=0}^{t-1}A_o^{t-1-i}\left(\delta f_{\text{NN}}+|L_o|\delta v_i+\delta w_i\right). \tag{8}$$

Hence, the value of δx_t is determined by the choice of L_o and the uncertainty levels of w_t and v_t. For given uncertainty bounds, L_o needs to be designed to ensure

$\overline{x}_t \le x_t \le \underline{x}_t, \forall t \ge 0$ and Schur stability of A_o. This then guarantees soundness and robust stability of δx_t (i.e., the interval observer). Theorem 4 describes the design of L_o.

Theorem 4. Under Assumption 1, if there is a scalar ρ, a diagonal matrix $P \in \mathbb{R}^{n\times n}$, and matrices $G^+, G^- \in \mathbb{R}^{n\times p}$ such that the following LMI problem is feasible:

$$\begin{aligned} \min_{\rho,P,G^+,G^-} \quad & \rho \\ \text{s.t.} \quad & \rho > 0,\ P \succ \mathbf{0}, && \text{(9a)} \\ & PA - (G^+ - G^-)C \ge \mathbf{0}, && \text{(9b)} \\ & \begin{bmatrix} P-I & \mathbf{0} & [PA-(G^+-G^-)C]^\top \\ \star & \rho I & \hat{D}^\top \\ \star & \star & P \end{bmatrix} \succ \mathbf{0}, && \text{(9c)} \end{aligned}$$

where $\hat{D} = [P,\ G^+ + G^-,\ P]$, then (6) with the gains $L_o^+ = P^{-1}G^+$, $L_o^- = P^{-1}G^-$ and $L_o = L_o^+ - L_o^-$ is an interval observer for the system (1), i.e., $\underline{x}_t \le x_t \le \overline{x}_t$, $\forall t \ge 0$ and the interval width δx_t is robust to the disturbance $\xi_t = [\delta f_{\text{NN}}^\top, \delta w_t^\top, \delta v_t^\top]^\top$ with the H_∞ performance gain $\sqrt{\rho}$.

Proof. Define the estimation errors as $\overline{e}_t = \overline{x}_t - x_t$ and $\underline{e}_t = x_t - \underline{x}_t$. Their dynamics are derived as

$$\begin{aligned} \overline{e}_{t+1} &= A_o\overline{e}_t + (\overline{f}_{\text{NN}} - f_{\text{NN}}) + (\overline{w}_t - w_t) \\ &\quad + L_o v_t - L_o^+\underline{v}_t + L_o^-\overline{v}_t, && \text{(10a)} \\ \underline{e}_{t+1} &= A_o\underline{e}_t + (f_{\text{NN}} - \underline{f}_{\text{NN}}) + (w_t - \underline{w}_t) \\ &\quad - L_o v_t + L_o^+\overline{v}_t - L_o^-\underline{v}_t. && \text{(10b)} \end{aligned}$$

In view of the relations

$$\begin{aligned} L_o v_t - L_o^+\underline{v}_t + L_o^-\overline{v}_t &= L_o^+(v_t - \underline{v}_t) + L_o^-(\overline{v}_t - v_t) \ge \mathbf{0}, \\ -L_o v_t + L_o^+\overline{v}_t - L_o^-\underline{v}_t &= L_o^-(v_t - \underline{v}_t) + L_o^+(\overline{v}_t - v_t) \ge \mathbf{0}, \end{aligned}$$

the following inequalities hold:

$$\begin{aligned} (\overline{f}_{\text{NN}} - f_{\text{NN}}) + (\overline{w}_t - w_t) + L_o v_t - L_o^+\underline{v}_t + L_o^-\overline{v}_t \ge \mathbf{0}, \\ (f_{\text{NN}} - \underline{f}_{\text{NN}}) + (w_t - \underline{w}_t) - L_o v_t + L_o^+\overline{v}_t - L_o^-\underline{v}_t \ge \mathbf{0}. \end{aligned}$$

Since $\overline{e}_0, \underline{e}_0 \ge \mathbf{0}$, if A_o is non-negative, then it follows from (10) that $\overline{e}_t, \underline{e}_t \ge \mathbf{0}$, $\forall t \ge 0$. Let $P \in \mathbb{R}^{n\times n} \succ \mathbf{0}$ be a diagonal matrix, then it follows from Lemma 3 that A_o is non-negative if

$$PA_o \ge \mathbf{0}. \tag{11}$$

Submitting $A_o = A - L_oC$ with $L_o = L^+ - L^-$ into (11) and introducing $G^+ = PL^+$ and $G^- = PL^-$ gives (9b).

The next step is to ensure Schur stability of A_o. Consider a Lyapunov function $V_t = \delta x_t^\top P \delta x_t$. By using (7), the difference $\Delta V_t = V_{t+1} - V_t$ is derived as

$$\Delta V_t = \begin{bmatrix} \delta x_t \\ \xi_t \end{bmatrix}^\top \underbrace{\begin{bmatrix} A_o^\top P A_o - P & A_o^\top P D \\ \star & D^\top P D \end{bmatrix}}_{\Pi} \begin{bmatrix} \delta x_t \\ \xi_t \end{bmatrix}, \tag{12}$$

where $\xi_t = [\delta f_{\text{NN}}^\top, \delta w_t^\top, \delta v_t^\top]^\top$ and $D = [I\ |L_o|\ I]$.

A sufficient condition for Schur stability of A_o is $\Pi \prec \mathbf{0}$. To enhance the robustness of δx_t against the disturbance term $D\xi_t$, we use the performance index $J = \sum_{t=0}^{\infty}(\delta x_t^\top \delta x_t - \gamma^2 \xi_t^\top \xi_t + \Delta V_t)$ for a scalar $\gamma > 0$. If $J < 0$, the interval width dynamics (7) satisfy the H_∞ performance $\sum_{t=0}^{\infty} \delta x_t^\top \delta x_t \le \gamma^2 \sum_{t=0}^{\infty} \xi_t^\top \xi_t$. By using (12), $J < 0$ is equivalent to

$$\begin{bmatrix} A_o^\top P A_o - P + I & A_o^\top P D \\ \star & D^\top P D - \gamma^2 I \end{bmatrix} \prec \mathbf{0}. \tag{13}$$

The inequality (13) can be re-arranged into

$$\begin{bmatrix} A_o^\top P \\ D^\top P \end{bmatrix} P^{-1} [PA_o\ PD] + \begin{bmatrix} -P + I & \mathbf{0} \\ \star & -\gamma^2 I \end{bmatrix} \prec \mathbf{0}. \tag{14}$$

By applying Schur complement to (14) and introducing the variables $G^+ = PL^+$, $G^- = PL^-$ and $\rho = \gamma^2$, it leads to the constraint (9c). □

According to Theorem 4, if the LMI problem (9) is feasible, then (6) always provides a sound interval (outer-approximation) for the system state, i.e., $x_t \in [\underline{x}_t, \overline{x}_t]$, $\forall t \ge 0$, given that $x_0 \in [\underline{x}_0, \overline{x}_0]$. We will show in Section 4 that the proposed interval observer can be used for safety monitoring and alert for the NN-enable control system (1).

4. SAFETY MONITORING AND ALERT

This section describes the use of the interval observer (6) for monitoring system safety, alerting potential unsafe operations and detecting potential faults.

Runtime safety monitoring: The interval generated by (6) is always an outer-approximation of the true state value. Let the safe state interval be $\mathcal{X}_t$, then the i-th state x_t^i is safe if $[\underline{x}_t^i, \overline{x}_t^i] \subseteq \mathcal{X}_t^i$, $i \in [1, n]$. When this condition is false, safety of x_t^i is undefined: it could be either that the interval generated by (6) gives a too coarse outer-approximation of x_t^i or the state x_t^i is indeed unsafe.

Predictive safety alerts: The intervals generated by (6) can also be used to raise alerts of potential unsafe operations in the future and detect possible faults. These features are based on the one-step ahead predicted intervals of the system state and output offered by (6). To illustrate this, we first rewrite (6) as

$$\begin{aligned} \overline{x}_{t+1} &= A_o\overline{x}_t + \overline{f}_{\text{NN}}(\underline{y}_t, \overline{y}_t, \underline{u}_t, \overline{u}_t) + L_o y_t \\ &\quad - L_o^+\underline{v}_t + L_o^-\overline{v}_t + \overline{w}_t, && \text{(15a)} \\ \underline{x}_{t+1} &= A_o\underline{x}_t + \underline{f}_{\text{NN}}(\underline{y}_t, \overline{y}_t, \underline{u}_t, \overline{u}_t) + L_o y_t \\ &\quad - L_o^+\overline{v}_t + L_o^-\underline{v}_t + \underline{w}_t, && \text{(15b)} \\ \overline{y}_{t+1} &= C^+\overline{x}_{t+1} - C^-\underline{x}_{t+1} + \overline{v}_{t+1}, && \text{(15c)} \\ \underline{y}_{t+1} &= C^+\underline{x}_{t+1} - C^-\overline{x}_{t+1} + \underline{v}_{t+1}, && \text{(15d)} \end{aligned}$$

where $\underline{y}_{t+1}$ and $\overline{y}_{t+1}$ are the estimated lower and upper bounds of the measured output y_{t+1}.

1) Alerting unsafe operations: Denote the one-step ahead predicted intervals of x_t and y_t as $X_{\text{pred},t+1} := [\underline{x}_{t+1}, \overline{x}_{t+1}]$ and $Y_{\text{pred},t+1} := [\underline{y}_{t+1}, \overline{y}_{t+1}]$, respectively. Let the safe intervals of x_{t+1} and y_{t+1} be $\mathcal{X}_{t+1}$ and $\mathcal{Y}_{t+1}$, respectively. At time step t, the i-th state, $i \in [1, n]$, is safe at $t+1$ if

$$X_{\text{pred},t+1}^i \subseteq \mathcal{X}_{t+1}^i. \tag{16}$$

The j-th output, $j \in [1, n]$, is safe at $t+1$ if

$$Y_{\text{pred},t+1}^j \subseteq \mathcal{Y}_{t+1}^j. \tag{17}$$

When (16) or (17) is violated, safety of the i-th state or j-th output is undefined. It could be either that their predicted intervals are too coarse or the next step state or output are indeed unsafe. Nevertheless, the information of violation is still practically valuable for alerting the potential unsafe system operations and introducing appropriate preventions. For the example of adaptive cruise control (ACC), when the next step inter-vehicular distance is

predicted to be less than the desired safe distance, the ego vehicle can perform emergency breaking to avoid collisions with the lead vehicle. This will be demonstrated through simulation in Case 3 of Section 5.2.

2) Fault detection: Consider the case when there are sensor faults f_t^{s} acting on the measured output y_t as follows:

$$y_t = Cx_t + F_{\mathrm{s}} f_t^{\mathrm{s}} + v_t, \tag{18}$$

where the matrix $F_{\mathrm{s}} \in \mathbb{R}^{p\times s}$, with $s \leq p$, specifying the channels (outputs) that the faults influence. The faults f_t^{s} can be regarded as "actuator faults" to the interval observer (15) because y_t is acting as an input steering the estimation dynamics. As seen in (15), for all $t \geq 0$, the proposed observer can provide a one-step ahead predicted output interval $[\underline{y}_{t+1}, \overline{y}_{t+1}]$. According to Theorem 4, the inclusion $y_t \in [\underline{y}_t, \overline{y}_t]$ holds for all $t \geq 0$. At time $t+1$, there is a fault in the i-th output channel, $i \in [1, p]$, if

$$y_{t+1}^i \notin [\underline{y}_{t+1}^i, \overline{y}_{t+1}^i]. \tag{19}$$

Simulation examples of detecting sensor faults will be provided in Case 2 of Section 5.2.

When the effect of a sensor fault is relatively small, y_{t+1} may remain within the predicted interval. In such case, checking (19) alone is unable to detect the fault occurrence. To address this, we could combine the interval observer with the advanced fault estimation technique (Lan and Patton, 2020) to estimate the shape of the fault. This aspect will be explored in the future research.

5. SIMULATION RESULTS

5.1 Neural Network based Dynamic System

We consider the discrete-time form of Example 1 in Xiang (2021) with additional disturbance and noise:

$$x_{t+1} = Ax_t + f_{\mathrm{NN}}(y_t, u_t) + w_t, \tag{20a}$$
$$y_t = Cx_t + v_t, \tag{20b}$$

where $A = [0.9, 0.05; 0.15, 0.75]$, $C = I_2$, $f_{\mathrm{NN}}(x_t, u_t) = t_s\Phi$ and $t_s = 0.05$ s is the sampling time. Φ is a three-input two-output 2-layer NN capturing the nonlinear system dynamics. The hidden layer has 5 neurons with `tanh` activations and the output layer has `purelin` activations.

We consider $u_t = 10\sin(0.25t)$, $w_t = 0.1\cos(0.1\pi t)$ and a zero-mean white noise v_t satisfying $|v_t| \leq 0.01$. The initial state is $x_0 = [0.5\ \ 0.5]^\top$. The signal bounds are set as $\underline{x}_0 = [-0.6, -0.6]^\top$, $\overline{x}_0 = [0.6, 0.6]^\top$, $\underline{u}_t = -10$, $\overline{u}_t = 10$, $\underline{v}_t = -0.01$, and $\overline{v}_t = 0.01$. Solving (9) gives the gains $L_o^- = [-0.4500, -0.025; -0.0275, -0.375]$ and $L_o^+ = [0.4500, 0.025; 0.0275, 0.375]$. Fig. 1 shows that the proposed observer provides real-time monitoring (sound intervals) of x and it obtains tighter intervals than the existing method (Xiang, 2021).

5.2 Neural Network Control System

We consider the following ACC system:

$$\dot{x} = A_c x + B_c u_{\mathrm{e}} + w, \tag{21a}$$
$$y = x + v, \tag{21b}$$

where $x = [p_{\mathrm{l}}\ v_{\mathrm{l}}\ a_{\mathrm{l}}\ p_{\mathrm{e}}\ v_{\mathrm{e}}\ a_{\mathrm{e}}]^\top$, y is the measured output with noise v, and

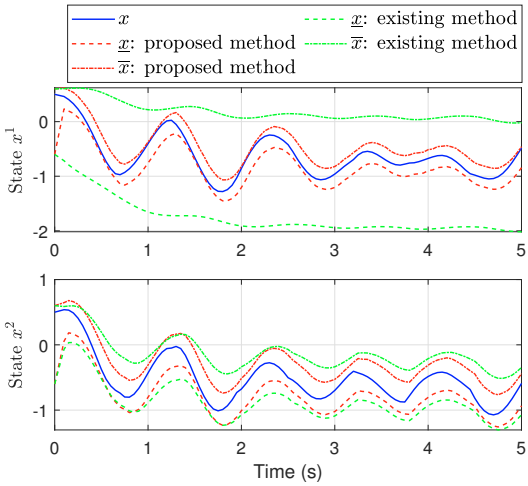

Fig. 1. System state $x = [x^1\ x^2]^\top$ and their intervals

$$A_c = \begin{bmatrix} 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&-2&0&0&0\\ 0&0&0&0&1&0\\ 0&0&0&0&0&1\\ 0&0&0&0&0&-2 \end{bmatrix}, B_c = \begin{bmatrix} 0\\0\\0\\0\\0\\2 \end{bmatrix}, w = \begin{bmatrix} 0\\0\\2u_{\mathrm{l}} - \mu v_{\mathrm{l}}^2\\0\\0\\-\mu v_{\mathrm{e}}^2 \end{bmatrix}.$$

The variables p, v, a and u are the position, velocity, acceleration and control command of the lead (with subscript l) and ego vehicles (with subscript e), respectively. μ is the friction parameter. Discretising (21) with a sampling time of $t_s = 0.05$ s gives a system in the form of (1).

Define the inter-vehicular distance as $h = p_{\mathrm{l}} - p_{\mathrm{e}}$, relative velocity as $\tilde{v} = v_{\mathrm{l}} - v_{\mathrm{e}}$, and the safe inter-vehicular distance as $d_{\mathrm{safe}} = v_{\mathrm{e}} t_{\mathrm{gap}} + d_{\mathrm{still}}$, where t_{gap} is the time headway and d_{still} is the standstill inter-vehicular distance. When $h \geq d_{\mathrm{safe}}$, the ACC controller u_{e} is in the speed control mode which maintains the ego vehicle at the driver-set speed v_{set}. When $h < d_{\mathrm{safe}}$, u_{e} is in the spacing control mode which ensures $h = d_{\mathrm{safe}}$ to avoid collisions. We borrow the 2-layer feedforward NN $u_e = f_{\mathrm{NN}}(u, z)$ from Xiang (2021) as the ACC controller, where $u = [v_{\mathrm{set}}\ t_{\mathrm{gap}}]^\top$, $z = [h\ \tilde{v}\ v_{\mathrm{e}}]^\top = C_z y$ and $C_z = [1,0,0,-1,0,0;0,1,0,0,-1,0;0,0,0,0,1,0]$. The hidden layer has 20 neurons using `tanh` activations and the output layer uses `purelin` activations.

The simulation parameters are: $\mu = 0.0001$, $t_{\mathrm{gap}} = 1.4$ s, $d_{\mathrm{still}} = 10$ m, $v_{\mathrm{set}} = 30$ m/s, $a_{\min} = -3$ m/s^2, $a_{\max} = 2$ m/s^2. v is a zero-mean white noise with $|v| \leq 0.001$. The initial state is $x_0 = [50, 20, 0, 10, 20, 0]^\top$. The signal bounds are set as $\underline{x}_0 = [49, 19, -1, 9, 19, -1]^\top$, $\overline{x}_0 = [51, 21, 1, 11, 21, 1]^\top$, $\underline{u}_t = \overline{u}_t = [v_{\mathrm{set}}\ t_{\mathrm{gap}}]^\top$, $\underline{v}_t = -0.001$, and $\overline{v}_t = 0.001$. Note that the interval for f_{NN} is constructed using $C_z y$ instead of y. Solving (9) gives the gains $L_o^- = \mathrm{diag}(-0.5, -0.5, -0.45, -0.5, -0.5, -0.45)$ and $L_o^+ = \mathrm{diag}(0.5, 0.5, 0.45, 0.5, 0.5, 0.45)$.

Case 1: The results in Figs. 2, 3 and 4 demonstrate efficacy of the proposed interval observer in terms of generating sound and robust runtime intervals of the vehicle positions, velocities and accelerations.

Case 2: Suppose the lead vehicle sends its real-time position, velocity and acceleration to the ego vehicle through a vehicle-to-vehicle (V2V) wireless communication network.

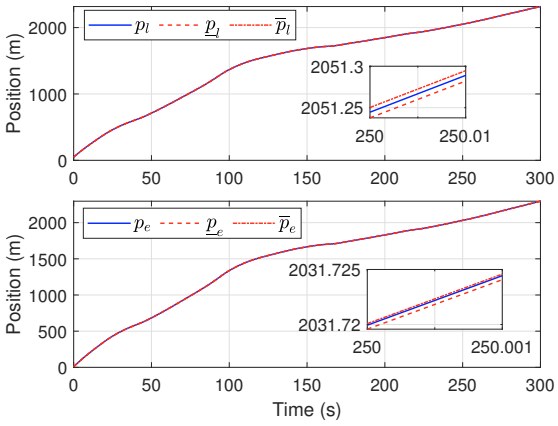

Fig. 2. Vehicle positions (p_l and p_e) and their intervals

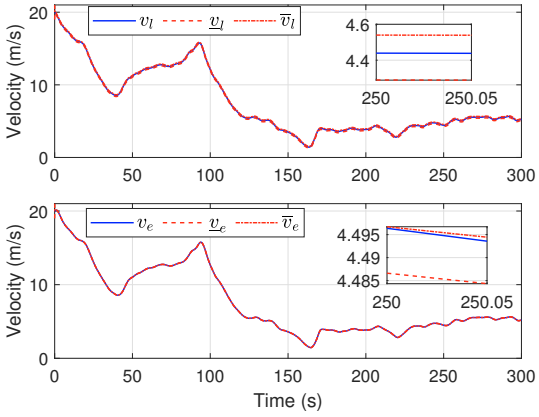

Fig. 3. Vehicle velocities (v_l and v_e) and their intervals

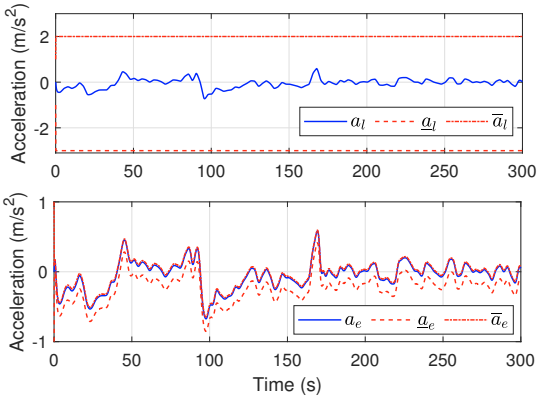

Fig. 4. Vehicle accelerations (a_l and a_e) and their intervals

We consider constant faults of different magnitudes adding on one of the data sent by the lead vehicle (i.e., p_l, v_l or a_l). These faults may be due to offsets in the sensors mounted on the lead vehicle (Lan et al., 2020) or cyber attacks during V2V transmission (Petit and Shladover, 2014).

Fig. 5 shows the results by applying three faults $f^s = 0.05, 0.1, 0.5$ m to the lead vehicle position p_l at the time of $200, 210, 220$ s, respectively. It is seen that all the three faults are detected by using the simple condition in (19). Fig. 6 shows the results by applying three faults $f^s = 0.09, 0.12, 0.2$ m/s to the lead vehicle velocity v_l at the time of $200, 210, 220$ s, respectively. The two larger faults are detected, while the fault $f^s = 0.09$ m/s is not detectable using (19), because the measured output at $t = 200$ s

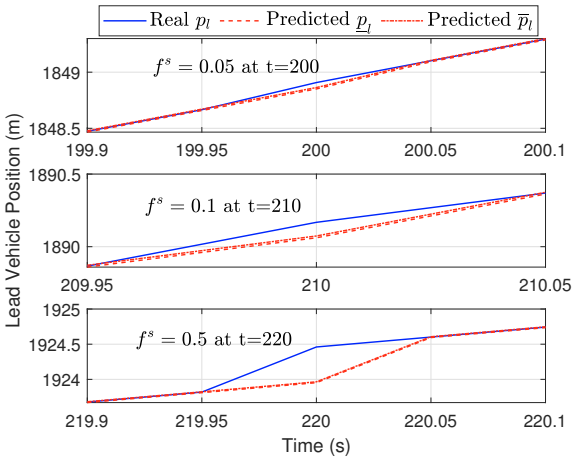

Fig. 5. Detection of faults on the lead vehicle position

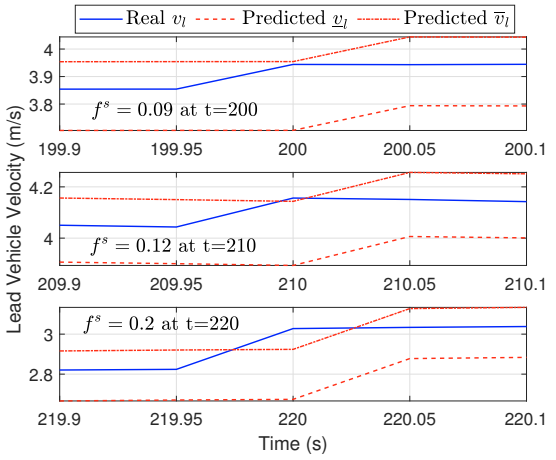

Fig. 6. Detection of faults on the lead vehicle velocity

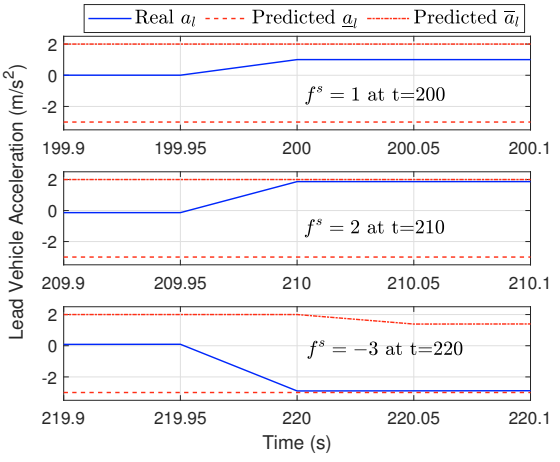

Fig. 7. Detection of faults on the lead vehicle acceleration

remains within its predicted interval. Fig. 7 shows the results by applying three faults $f^s = 1, 2, -3$ m/s^2 to the lead vehicle acceleration a_l at the time of $200, 210, 220$ s, respectively. In this case, we are unable to detect any of the faults, even though $f^s = -3$ m/s^2 and $f^s = 2$ m/s^2 have reached the minimal and maximal accelerations, respectively. This is because the interval of a_l generated by the proposed observer is $[-3, 2]$ m/s^2 for all $t > 0$, as shown in the top subplot of Fig. 4.

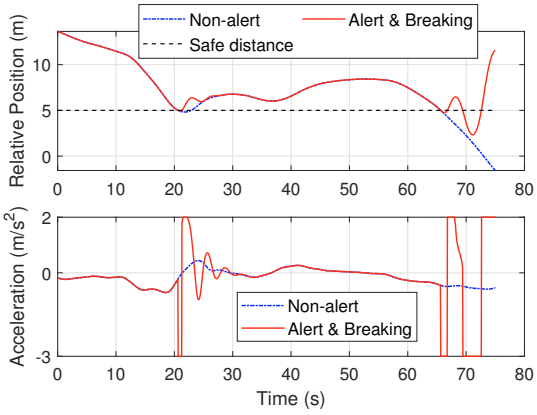

Fig. 8. Relative vehicle positions and ego vehicle accelerations with/without safety alert

Case 3: We use the interval observer to raise an unsafe alert whenever the predicted one-step ahead lower bound of the relative vehicle position $\underline{h}_{t+1} \leq h_{\text{safe}}$, where $h_{\text{safe}} = 5$ m is the desired safe distance. Upon receiving the alert, the ego vehicle applies an emergency braking with $u_{\text{e}} = -3$ m/s^2. It is seen from Fig. 8 that the ego vehicle receives alert and applies emergency braking at around 21 s, 66 s and 69 s when $\underline{h}_{t+1} \leq 5$ m and avoids collisions. Without alerting and breaking, the two vehicles crashes.

6. CONCLUSION

A robustly stable observer is proposed to generate sound state intervals of control systems with a NN modelling the nonlinear dynamics or being the controller. The observer is applied to monitoring the runtime system safety, detecting potential sensor faults and alerting unsafe system operations. The simulation results of a numerical example and an ACC system demonstrate effectiveness of the proposed observer and its advantage over the existing design. Future work will consider tightening the interval generated by the observer to enhance its capability in safety monitoring and alert. A systematic design of appropriate remedial actions to ensure safe operation of the NN-enabled control system is also of interest.

REFERENCES

Bunel, R. et al. (2020). Branch and bound for piecewise linear neural network verification. *JMLR*, 21(2020).

Cofer, D. et al. (2020). Run-time assurance for learning-enabled systems. In *Proc. NFM*, 361–368. Springer.

Dai, H. et al. (2021). Lyapunov-stable neural-network control. *arXiv preprint arXiv:2109.14152*.

Dubey, S.R., Singh, S.K., and Chaudhuri, B.B. (2022). Activation functions in deep learning: a comprehensive survey and benchmark. *Neurocomputing*, 503, 92–108.

Efimov, D. et al. (2013). Interval observers for time-varying discrete-time systems. *IEEE Trans. Automat. Contr.*, 58(12), 3218–3224.

Everett, M. et al. (2021). Reachability analysis of neural feedback loops. *IEEE Access*, 9, 163938–163953.

Farina, L. and Rinaldi, S. (2000). *Positive linear systems: theory and applications*, volume 50. John Wiley & Sons.

Hewing, L. et al. (2020). Learning-based model predictive control: Toward safe learning in control. *Annu. rev. control robot.*, 3, 269–296.

Hu, H. et al. (2020). Reach-SDP: Reachability analysis of closed-loop systems with neural network controllers via semidefinite programming. In *Proc. IEEE Conf. Decis. Control*, 5929–5934. IEEE.

Huang, X. et al. (2020). A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Comput. Sci. Rev.*, 37, 100270.

Julian, K.D. and Kochenderfer, M.J. (2021). Reachability analysis for neural network aircraft collision avoidance systems. *J. Guid. Control Dyn.*, 44(6), 1132–1142.

Karg, B. and Lucia, S. (2020). Stability and feasibility of neural network-based controllers via output range analysis. In *Proc. IEEE Conf. Decis. Control*, 4947–4954. IEEE.

Katz, G. et al. (2017). Reluplex: An efficient SMT solver for verifying deep neural networks. In *Proc. CAV*, 97–117. Springer.

Lan, J. and Patton, R.J. (2020). *Robust integration of model-based fault estimation and fault-tolerant control*. Springer.

Lan, J., Zhao, D., and Tian, D. (2020). Robust cooperative adaptive cruise control of vehicles on banked and curved roads with sensor bias. In *Proc. ACC*, 2276–2281. IEEE.

Lan, J., Zheng, Y., and Lomuscio, A. (2022). Tight neural network verification via semidefinite relaxations and linear reformulations. In *Proc. AAAI*, volume 36, 7272–7280.

Liu, C. et al. (2021). Algorithms for verifying deep neural networks. *Found. Trends. Optim.*, 4(3-4), 244–404.

Lomuscio, A. and Maganti, L. (2017). An approach to reachability analysis for feed-forward ReLU neural networks. *arXiv preprint arXiv:1706.07351*.

Moe, S., Rustad, A.M., and Hanssen, K.G. (2018). Machine learning in control systems: An overview of the state of the art. In *Proc. Innov. Appl. Artif. Intell. Conf.*, 250–265. Springer.

Petit, J. and Shladover, S.E. (2014). Potential cyberattacks on automated vehicles. *IEEE Trans. Intell. Transp. Syst.*, 16(2), 546–556.

Tang, W. et al. (2019). Interval estimation methods for discrete-time linear time-invariant systems. *IEEE Trans. Automat. Contr.*, 64(11), 4717–4724.

Tang, Y. et al. (2022). Perception and navigation in autonomous systems in the era of learning: A survey. *IEEE Trans. Neural Netw. Learn. Syst.* doi: 10.1109/TNNLS.2022.3167688.

Xiang, W. (2021). Runtime safety monitoring of neural-network-enabled dynamical systems. *IEEE Trans. Cybern.*, 52(9), 9587–9596.

Xiang, W. et al. (2019). Reachable set estimation and verification for neural network models of nonlinear dynamic systems. In *Safe, Autonomous and Intelligent Vehicles*, 123–144. Springer.

Zhang, Y. and Xu, X. (2022). Safety verification of neural feedback systems based on constrained zonotopes. *arXiv preprint arXiv:2204.00903*.

Zhou, R. et al. (2022). Neural Lyapunov control of unknown nonlinear systems with stability guarantees. *arXiv preprint arXiv:2206.01913*.