Préparée à Université de Lorraine

Development of new advanced estimation algorithms for improving the resilience of the autonomous and connected vehicle

Soutenue par

Quang Huy NGUYEN

Le xx mois 202x

École doctorale n°xxx

Titre de l'école doctorale

Spécialité

Automatique

Préparée au

Centre de Recherche Nancy

Composition du jury :

Prénom NOM 1 Affiliation	*Président du jury* *Examinateur*
Prenom NOM 2 Affiliation	*Rapporteur*
Prenom NOM 3 Affiliation	*Rapporteur*
Prenom NOM 4 Affiliation	*Rapporteur*
Prenom NOM 5 Affiliation	*Rapporteur*
Prenom NOM 6 DR Affiliation	*Directeur de thèse*

Chapitre 1

Resilient Trust–Aware Distributed Observer Design for Connected Vehicle Platoons

Objectifs

This chapter proposes a trust-aware distributed observer for vehicle platoons that maintains resilient state estimation under cyberattacks. A behavioral divergence metric evaluates the reliability of shared data, forming a dynamic neighbor set used to adapt observer's weighting gains. Stability conditions are derived via Lyapunov analysis. Simulations under bogus, replay, and DoS attacks demonstrate robust performance and stable platoon behavior.

1 Introduction

1.1 General introduction

The rapid development of autonomous and connected vehicles has introduced new opportunities to improve road safety, traffic flow, and fuel efficiency. In vehicle platoons-where multiple vehicles coordinate through vehicle-to-vehicle (V2V) communication-accurate distributed state estimation is essential for maintaining stability and efficiency. Each vehicle must reconstruct both its own and others' states using locally sensed data and information exchanged with neighbors. However, this distributed structure also exposes the system to cyber and communication attacks that can disrupt coordination or compromise state estimation [**Ali_book**].
Traditional estimation and control methods generally assume all vehicles are cooperative and trustworthy, which is unrealistic in adversarial environments [**Shengya_HG**]. Malicious or faulty agents can transmit falsified information, causing estimation errors that propagate through the network. Consequently, resilient estimation strategies capable of identifying and isolating untrustworthy data have become a key research direction in cyber-physical systems (CPS) and distributed multi-agent systems (DMAS).
Existing work on secure estimation and fault detection can be categorized into three main families :

- *Observer-based detection :* compares model-predicted states with sensor measurements to identify anomalies [**Huy_LCSS** ; **2_obs_Angelo**]. These methods are effective for detecting deviations but may propagate errors if compromised neighbors provide false data.
- *Consensus-based detection :* relies on cross-validation among agents to identify inconsistent information [**Guitao_multi_mesure**]. While robust under the assumption that most agents are trustworthy, their performance degrades under large-scale coordinated attacks.
- *Trust-based detection :* assigns reliability scores to neighboring agents based on behavioral or statistical consistency. Physical-signal approaches use device fingerprints for authentication [**trust_physic_connecte**], while statistical approaches compare communicated data against locally observed behaviors [**Wang_2021**]. These methods improve resilience by filtering unreliable inputs before consensus or estimation steps, but they typically require significant computational resources or long-term historical data-making real-time adaptation challenging for dynamic platoons.

Mitigation strategies in the literature often depend on attack detection results. Some switch to fallback modes (e.g., CACC to ACC), while others incorporate trust scores directly into control to down-weight suspicious data [**self_belive** ; **Enhancing_control_trust**]. The latter provides finer control adaptation but relies heavily on accurate and timely trust evaluation. To overcome these limitations, we propose a trust-aware distributed observer framework for vehicle platoons. The framework continuously evaluates the trustworthiness of communicated data by comparing reported and predicted behaviors, dynamically adjusting observer weights according to each neighbor's trust score. This mechanism ensures reliable estimation even under cyberattacks or communication faults.

The main contributions of the paper can be summarized in the following items :

- A formal definition of the resilient distributed state estimation problem for connected vehicles under adversarial conditions.
- A behavioral trust model that quantifies each agent's reliability and constructs a dynamic trusted neighbor set.
- A trust-informed distributed observer with provable stability guarantees, validated through simulations under multiple attack scenarios.

By integrating detection, mitigation, and estimation within a unified trust-based layer, the proposed method improves the resilience and stability of cooperative vehicle systems.

1.2 Graph theory

An undirected graph with a nonempty finite set of N nodes can be described by $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ where :

- $\mathcal{V} = \{1, 2, \ldots, N\}$ is the set of nodes.
- $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges.
- $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$ is the weighted adjacency matrix of $\mathcal{G}$ defined by

$$[a_{ij}] = \begin{cases} a_{ij}, & \text{if } (i,j) \in \mathcal{E}, \\ 0, & \text{otherwise.} \end{cases} \tag{1.1}$$

- The *Laplacian* matrix of $\mathcal{G}$ is denoted as $\mathcal{L} = [l_{ij}] \in \mathbb{R}^{N\times N}$ with

$$l_{ij} = \begin{cases} \sum_{k=1}^{N} a_{ik}, & \text{if } i = j, \\ -a_{ij}, & \text{otherwise.} \end{cases} \tag{1.2}$$

- The set of the neighbors of the node i is defined by $\mathcal{N}_i = \{j \in \mathcal{V} \mid (j,i) \in \mathcal{E}\}$.

1.3 Vehicle mathematical model

The platoon of N vehicles considered in this paper can be modeled as a network of N agents, represented by an undirected communication graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$. The discrete-time longitudinal dynamics of each vehicle $i \in \mathcal{V}$ are

$$\begin{aligned} x_i(t+1) &= A_{\mathrm{b}} x_i(t) + B_{\mathrm{b}} u_i(t) + \Delta_i(t), \\ y_i(t) &= C_i x_i(t), \end{aligned} \tag{1.3}$$

where $x = \begin{bmatrix} s_i & v_i & a_i \end{bmatrix}^\top$ and u_i denote the state and control input of vehicle i, respectively. s_i, v_i, and a_i denote position, velocity, and acceleration. The system matrices are given by

$$A_{\mathrm{b}} = \begin{bmatrix} 1 & T_s & \frac{T_s^2}{2} \\ 0 & 1 & T_s \\ 0 & 0 & 1 - \frac{T_s}{\tau_b} \end{bmatrix}, \quad B_b = \begin{bmatrix} 0 \\ 0 \\ \frac{T_s}{\tau_b} \end{bmatrix}, C_{\mathrm{b}} = I_3,$$

where T_s is the sampling time and the constant $\tau_b > 0$ is nominal engine time lag. $\Delta_i(t)$ captures the modeling differences and uncertainties caused by difference between nominal τ_b and the real value.

Define the collective state as $x = \text{col}\ (\ x_1, \cdots, x_N\ )$, the state-space equation of the platoon is

$$\begin{cases} x(t+1) = (I_N \otimes A_\text{b})x(t) + (I_N \otimes B_\text{b})u(t) + \Delta(t) \\ y_i(t+1) = C_i x(t), \end{cases} \tag{1.4}$$

where $u = \text{col}\ (\ u_1, \cdots, u_N\ )$ is the collective control input, $\Delta = \text{col}\ (\ \Delta_1, \cdots, \Delta_N\ )$ group unmodeled dynamics and external disturbances and $C_i = \begin{bmatrix} 0 \cdots \underbrace{C_b}_{i\text{th}} \cdots 0 \end{bmatrix}$ is the global output matrix.

Since each vehicle measures its own full states through onboard sensors, ensuring that the pair $(A_\text{b},\ C_\text{b})$ is observable. However, in the context of the platoon dynamic (1.4), the pair $((I_N \otimes A_\text{b}), C_i)$ remain unobservable, motivating the need for distributed estimation through inter-vehicle communication.

2 Distributed State Observer Architecture

The distributed observer provides an effective approach to estimate the full platoon state x. In this section, we introduce a new distributed observer architecture that incorporates trust-function mechanisms, which will be detailed later in Section 3. To achieve full-state estimation of all vehicles, two complementary layers are employed : a Local Observer ($\mathcal{LO}$) that uses onboard sensors, and a Distributed Observer ($\mathcal{DO}$) that fuses information exchanged through communication links. Fig. 1.1 illustrates the architecture of the proposed distributed observer for vehicle platoons.

2.1 General structure of the observer

The distributed state observer architecture considered in this study is characterized by the following set of equations (1.5) :

$$(\mathcal{LO}_i) : \hat{x}_0^{(i)}(t+1) = A_\text{b}\hat{x}_0^{(i)}(t) + B_\text{b}u_i(t) + F_i(y_i(t) - C_\text{b}\hat{x}_0^{(i)}(t)) \tag{1.5a}$$

$$(\mathcal{DO}_i^{(j)}) : \hat{x}_i^{(j)}(t{+}1) = A_\text{b}\left(\hat{x}_i^{(j)}(t) + \sum_{l \in \mathcal{N}_i} w_{il}^{(j)}(t)(\hat{x}_l^{(j)}(t) - \hat{x}_i^{(j)}(t)) + w_{i0}^{(j)}(t)\left(\hat{x}_0^{(j)}(t) - \hat{x}_i^{(j)}(t)\right)\right) {+} B_\text{b}\hat{u}_j(t) \tag{1.5b}$$

where

— $\hat{x}_0^{(i)}(t)$ is the state of the local observer $\mathcal{LO}_i$, which will try to track the states of the ith vehicle. It means that

$$\lim_{t \to \infty} \left(\hat{x}_0^{(i)}(t) - x^{(i)}(t)\right) = 0. \tag{1.6}$$

— F_i is the gain matrix of the local observer.

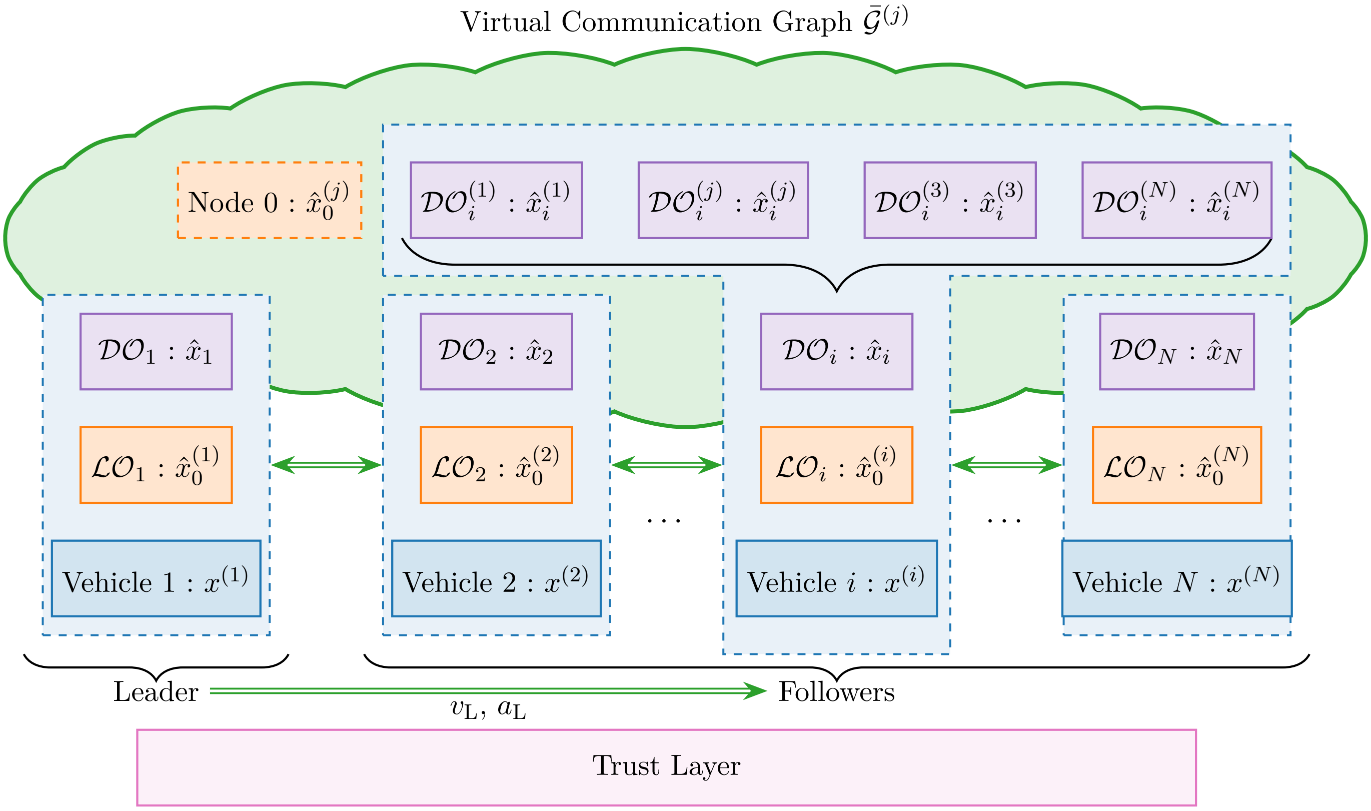
Virtual Communication Graph $\bar{\mathcal{G}}^{(j)}$
Node 0 : $\hat{x}_0^{(j)}$
$\mathcal{DO}_i^{(1)} : \hat{x}_i^{(1)}$
$\mathcal{DO}_i^{(j)} : \hat{x}_i^{(j)}$
$\mathcal{DO}_i^{(3)} : \hat{x}_i^{(3)}$
$\mathcal{DO}_i^{(N)} : \hat{x}_i^{(N)}$
$\mathcal{DO}_1 : \hat{x}_1$
$\mathcal{DO}_2 : \hat{x}_2$
$\mathcal{DO}_i : \hat{x}_i$
$\mathcal{DO}_N : \hat{x}_N$
$\mathcal{LO}_1 : \hat{x}_0^{(1)}$
$\mathcal{LO}_2 : \hat{x}_0^{(2)}$
$\mathcal{LO}_i : \hat{x}_0^{(i)}$
$\mathcal{LO}_N : \hat{x}_0^{(N)}$
Vehicle 1 : $x^{(1)}$
Vehicle 2 : $x^{(2)}$
Vehicle i : $x^{(i)}$
Vehicle N : $x^{(N)}$
Leader
Followers
v_{L}, a_{L}
Trust Layer

FIGURE 1.1 – Schematic diagram of the estimation method.

- $\mathcal{N}_i$ is the neighbors of the ith vehicle according to the communication graph $\mathcal{G}$.
- $w_{il}^{(j)}(t)$, $l \in \mathcal{N}_i^{(j)}$ is the gain scale to be designed based on the communication topology.
- $\hat{x}_i^{(j)}(t)$ is the state of the distributed observer $\mathcal{DO}_i^{(j)}$, which will try to track the states of the platoon. It means that

$$\lim_{t\to\infty} (\hat{x}_i(t) - x(t)) = 0, \tag{1.7}$$

  where $\hat{x}_i(t) = \mathrm{col}\left(\hat{x}_i^{(1)}(t), \ \cdots, \ \hat{x}_i^{(N)}(t)\right)$ is the collective states of all distributed observer.
- $w_{i0}^{(j)}$ is the gain scale to be designed, which is such that

Remark 1. *If a host forms $\hat{u}_j(t)$ (e.g., from a known controller law or an input estimator), the mismatch*

$$\Delta u_j(t) = u_j(t) - \hat{u}_j(t) \tag{1.8}$$

will be treated as part of the disturbance vector. We do not require $\hat{u}_j(t)$ for the $(\mathcal{DO})$ to run ; using it only tightens bounds.

2.2 On the structure of the observer (1.5)

The distributed observer $\mathcal{DO}_i^{(j)}$ is designed to estimate the state of each vehicle $j \in \mathcal{V}$ in the platoon from the perspective of vehicle i. The term $\sum_{l\in\mathcal{N}_i} w_{il}^{(j)} \left(\hat{x}_l^{(j)}(t) - \hat{x}_i^{(j)}(t)\right)$ takes into account communication with neighbors $l \in \mathcal{N}_i$ to adjust $\hat{x}_i^{(j)}(t)$ towards a consensus among neighboring estimates of vehicle j's state. The weights $w_{il}^{(j)}$ determine the influence of each neighbor's estimate, promoting consistency across the platoon, which will be designed later. The consensus term alone has a key limitation , it cause all vehicles estimates of a given state $\hat{x}_i^{(j)}(t)$ to converge to a common value. This is theoretically sound only for an ideal platoon with perfectly aligned states. However, practical disturbances, initial condition errors, and imperfect models break this uniformity, rendering a pure consensus estimate inaccurate for any individual vehicle's true state. To mitigate this, the local observer $\mathcal{LO}_i$ provides an accurate estimate of vehicle i's own state, $\hat{x}_0^{(i)}(t)$. The additional anchoring term $w_{i0}^{(j)}\left(\hat{x}_0^{(j)}(t) - \hat{x}_i^{(j)}(t)\right)$ injects this trusted reference, allowing each vehicle to correct model errors and maintain accurate peer estimates.

2.3 Introduction of a virtual communication graph

Each distributed observer $\mathcal{DO}_i^{(j)}$ requires reference information from the local observer $\mathcal{LO}_j$. To formally include this reference, we define a virtual node labeled as node "0" that connects to the vehicle j, where $j \in \mathcal{V}$ and its neighbors $\mathcal{N}_j$ according to the original graph $\mathcal{G}$. The resulting virtual communication graph $\bar{\mathcal{G}}^{(j)} = \{\bar{\mathcal{V}}, \bar{\mathcal{E}}^{(j)}\}$ is constructed as :

$$\begin{aligned} \bar{\mathcal{V}} &= \mathcal{V} \cup \{0\}, \\ \bar{\mathcal{E}}^{(j)} &= \mathcal{E} \cup \{(0,j) | j \in \mathcal{V}\} \cup \{(0,k) | k \in \mathcal{N}_j\}. \end{aligned} \tag{1.9}$$

The adjacency matrix $\bar{\mathcal{A}}^{(j)} \in \mathbb{R}^{(N+1)\times(N+1)}$ corresponding to $\bar{\mathcal{G}}^{(j)}$ is defined as :

$$\bar{\mathcal{A}}^{(j)} = \begin{bmatrix} 0 & 0 \\ w_0^{(j)} & \mathcal{A} \end{bmatrix}, \tag{1.10}$$

where the vector $w_0^{(j)} = \text{col}\left(w_{10}^{(j)}, \cdots, w_{N0}^{(j)}\right) \in \mathbb{R}^N$ defines the connection strengths from node 0 to the distributed observers.

This virtual communication graph provides a unified representation for integrating local estimation (through node 0) and neighbor-based consensus (through graph G), ensuring that every vehicle's local observer acts as a trusted anchor for distributed state estimation.

2.4 Main objectives

Traditional distributed observers use fixed weights w_{il} that assume all data sources are trustworthy. However, in a connected vehicle platoon, compromised nodes may send falsified information. To address this issue, the proposed design incorporates trust-based and time-varying weighting $w_{il}(t)$, allowing each vehicle to dynamically adjust its observer weights according to the reliability of received information. To address this issue, one of the key challenges is to define a quantitative measure of *trust* that captures how reliably a host vehicle can utilize information from its neighbors, despite potential uncertainties or malicious data. Based on this trust, an *adaptive update law* for the gain scaling factor $w_{il}(t)$ is designed, depending on both the computed trust value and the gain matrix F_i of the local observer ($\mathcal{LO}_i$). This trust mechanism is incorporated into the observer synthesis conditions, presented later in Section 4, ensuring an appropriate ISS convergence bound.

3 Trust Score Framework

To ensure resilient estimation, each vehicle continuously evaluates the reliability of the data it receives. This is achieved through a multi-layer trust framework that quantifies how much a host vehicle i trusts another vehicle l. The overall structure of this trust framework is illustrated in Fig. 1.2.

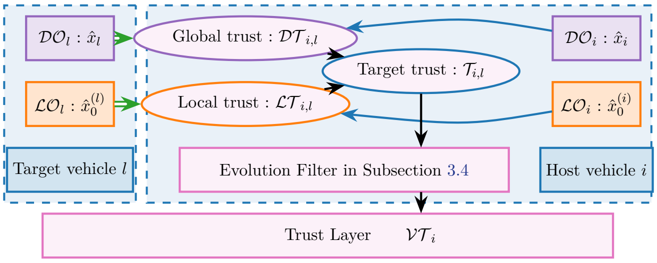
$\mathcal{DO}_l : \hat{x}_l$
$\mathcal{LO}_l : \hat{x}_0^{(l)}$
Target vehicle l
Global trust : $\mathcal{DT}_{i,l}$
Target trust : $\mathcal{T}_{i,l}$
Local trust : $\mathcal{LT}_{i,l}$
$\mathcal{DO}_i : \hat{x}_i$
$\mathcal{LO}_i : \hat{x}_0^{(i)}$
Evolution Filter in Subsection 3.4
Host vehicle i
Trust Layer $\mathcal{VT}_i$

Figure 1.2 – Trust framework.

In this context, we define the following types of trust :

- **Local Trust** $\mathcal{LT}_{i,l}$: evaluates the reliability of the local observer $\mathcal{LO}_l$ data from vehicle l based on consistency and timeliness.
- **Distributed Trust** $\mathcal{DT}_{i,l}$: measures how well the global estimates of l agree with those of i.
- **Target Trust** $\mathcal{T}_{i,l}$: Combining the Local Trust and Global Trust, the host vehicle i can evaluate the trust score for each target vehicle at time t.
- **Vehicle Trust** $\mathcal{VT}_i$: Smooth evolution of Target Trust scores over time.

In the following subsections, we will explain how to calculate the trust. After that, we will provide more details on how to determine vehicle trust ($\mathcal{VT}_i$) and how to utilize it effectively.

3.1 Data validity indicator

Before trust computation, every received V2V packet data is validated for authenticity and freshness :

$$\delta_{i,l}(t) = \begin{cases} 1, & \text{if message is authenticated and fresh,} \\ 0, & \text{otherwise.} \end{cases} \tag{1.11}$$

Only packets satisfying $\delta_{i,l}(t) = 1$ use in the following trust evaluation process.

3.2 Local trust

The Local Trust $\mathcal{LT}_{i,l}$ quantifies the confidence that the host vehicle i places in the local observer output of a neighboring vehicle $l \in \mathcal{N}_i$. It is based on three physical-consistency indicators computed from measurable states : velocity, distance, and acceleration. When data are missing, a hold-and-decay rule maintains continuity.

3.2.1 Velocity consistency :

The host estimates the expected velocity of the platoon leader using its last valid packet as a reference value :

$$v_{\mathrm{ref}}(t) = v_{\mathrm{L}}(t - t_{\mathrm{elap}}) + t_{\mathrm{elap}} \cdot a_{\mathrm{L}}(t - t_{\mathrm{elap}}), \tag{1.12}$$

Then, the mismatch score of the target vehicle i with the reference leader can be calculated as

$$v_{l,\mathrm{L}} = \begin{cases} \max\left(1 - \left|\dfrac{\hat{v}_0^{(l)} - v_{\mathrm{ref}}}{v_{\mathrm{ref}}}\right|,\ 0\right), & \text{if } v_{\mathrm{ref}} > 0, \\ \max\left(1 - |\hat{v}_0^{(l)}|,\ 0\right), & \text{otherwise,} \end{cases} \tag{1.13}$$

Similarly, the host vehicle $\hat{v}_0^{(i)}$ compares with its neighbors $\hat{v}_0^{(l)}$, the following mismatch score is :

$$v_{l,\mathrm{H}} = \begin{cases} \max\left(1 - \left|\dfrac{\hat{v}_0^{(l)} - \hat{v}_0^{(i)}}{\hat{v}_0^{(i)}}\right|, 0\right), & \text{if } \hat{v}_0^{(i)} > 0, \\ \max\left(1 - |\hat{v}_0^{(l)}|, 0\right), & \text{otherwise,} \end{cases} \tag{1.14}$$

The velocity mismatch is computed as a weighted average of two estimates :

$$\iota_{i,l}^{\mathrm{velocity}} = ((1 - \sigma)v_{l,\mathrm{L}} + \sigma v_{l,\mathrm{H}}) \tag{1.15}$$

where the weighted σ is selected based on the index position of the target vehicle l. If the target is following the host vehicle i, then $\sigma \in (0, 0.3)$. If the target is leading, $\sigma \in (0.7, 1)$.

3.2.2 Distance consistency :

The host vehicle i calculates the distance with respect to the target l as :

$$d_{i,l} = \hat{s}_{\mathrm{x},0}^{(l)} - \hat{s}_{\mathrm{x},0}^{(i)} - L,$$

where L is the length of the vehicle. So the mismatch between the measured distance d from its sensors and the calculated data $d_{k,i}$ is

$$\iota_{i,l}^{\mathrm{distance}} = \max\left(1 - \left|\frac{d_{i,l} - d}{d}\right|, 0\right). \tag{1.16}$$

Assumption 1. *When a vehicle is connected, it has access to basic information about its neighbors, including their type and length.*

3.2.3 Acceleration consistency :

The host vehicle can calculate the expected relative acceleration as

$$a_{\mathrm{recv}}^{\mathrm{rel}} = \hat{a}_0^{(l)} - \hat{a}_0^{(i)}, \qquad a_{\mathrm{expect}}^{\mathrm{rel}} = \frac{v_{\mathrm{rel}}(t) - v_{\mathrm{rel}}(t - n)}{n},$$

where $v_{\text{rel}} = \hat{v}_0^{(l)} - \hat{v}_0^{(i)}$ and n defines the averaging window used to smooth noise. To account for physical context, the mismatch is normalized by both distance and relative velocity :

$$\iota_{i,l}^{\text{accel}} = \max\left(\left|\frac{v_{\text{rel}}}{d_{i,l}} \cdot a_{\text{recv}}^{\text{rel}} - a_{\text{expect}}^{\text{rel}}\right|, 0\right) \tag{1.17}$$

where d_{norm} are normalization constants reflecting typical spacing in the platoon.

This formulation tightens the trust threshold when vehicles are close or moving fast, where inconsistencies are more critical, and relaxes it when they are farther apart or nearly stationary. It provides smoother and dimensionally consistent trust updates while reducing false trust drops caused by minor sensor fluctuations.

3.2.4 Fusion and update rule :

We combine all three indicators into an integrity measure $\mathcal{LT}_{i,l}$ indicates the trust of local data of the host vehicle i in the target vehicle l at time t :

$$\mathcal{LT}_{i,l}(t) = \begin{cases} f(\iota_{i,l}^{\text{velocity}}, \iota_{i,l}^{\text{distance}}, \iota_{i,l}^{\text{accel}}) & \text{if } \delta_{i,l}(t) = 1 \\ (1-\lambda_h)\mathcal{LT}_{i,l}(t-1), & \text{otherwise} \end{cases} \tag{1.18}$$

where f is a fusion function that combines the three scores. The parameter λ_h defines the slow decay during temporary losses.

Remark 2. *Function f can be a weighted average or a minimum function, depending on the desired sensitivity to individual indicators. The decay λ_h can be varied based on how many consecutive packets are missed. This prevents a single missed packet from instantly reducing trust to zero, while still penalizing prolonged disconnection.*

3.3 Distributed trust

Same concept of Local Trust but in the context of the distributed data. We denote $\mathcal{DT}_{i,l}$ represents how the host vehicle i trusts the distributed estimation received from target vehicle l, which is defined as

$$\mathcal{DT}_{i,l} = \begin{cases} \gamma_{i,l}^{\text{self}} \cdot \gamma_{i,l}^{\text{local}} \cdot \gamma_{i,l}^{\text{host}} & \text{if } \delta_{i,l}(t) = 1 \\ (1-\lambda_h)\mathcal{DT}_{i,l}(t-1), & \text{otherwise} \end{cases} \tag{1.19}$$

where γ_l^{host} and $\gamma_{i,l}^{\text{local}}$ are applied to check if the global estimates from neighbor l are consistent with the host's and the local measurements, respectively ; Both of them will be illustrated in the following.

3.3.1 Consistency with host's global estimate :

To assess whether the global estimate from the target l is trustworthy, the host can compare it directly to its own global estimate. We use the Mahalanobis distance, which normalizes discrepancies based on covariance :

$$r_{i,l}^{(j)} = \left(\hat{x}_i^{(j)} - \hat{x}_l^{(j)}\right)^\top \Sigma_{\text{host}}^{-1} \left(\hat{x}_i^{(j)} - \hat{x}_l^{(j)}\right), \tag{1.20}$$

where Σ_{local} is the diagonal covariance matrix which captures variance and correlations between each state channel defined as

$$\Sigma_{\text{local}} = \text{diag}\left(\sigma_1^2, \sigma_2^2, \cdots, \sigma_n^2\right), \tag{1.21}$$

where n is the number of states in the vehicle, and σ_k^2 is the variance of discrepancies about the different states.

And then, add up these differences to get the total discrepancy score across the platoon as the following

$$R_{i,l} = \sum_{j=1}^{N} r_{i,l}^{(j)}. \tag{1.22}$$

A small $R_{i,l}$ means the global estimate from vehicle l is close to the host's, suggesting higher trustworthiness.

Based on the total discrepancy score across the platoon, the trust score can be converted to be

$$\gamma_{i,l}^{\text{host}} = \exp\left(-R_{i,l}\right). \tag{1.23}$$

where $\gamma_{i,l}^{\text{host}} \in (0, 1]$. A value near 1 means high trust (small discrepancy), while a value near 0 means low trust (large discrepancy).

3.3.2 Consistency with host's local measurement :

2 Thing : Use Self and Target Value Distributed state Host vehicle i have relative measurement to nearby vehicles (e.g., distance and relative velocity to a predecessor and a follower) and use that to checks if the distributed estimates from target l are consistent with these measurements.

Let $y_{i,j}(t) \in R^{m_{ij}}$ is the relative measurement of vehicle i to vehicle j (e.g., $y_{i,j}(t) = s_j(t) - s_i(t)$ for relative position).

The distributed estimate of target l should satisfy expectively :

$$H(\hat{x}_l^{(j)}(t) - \hat{x}_l^{(i)}(t)) \approx y_{i,j}(t).$$

where $H \in R^{m_{ij} \times n}$ is the measurement matrix that extracts the relevant states (e.g., position) from the state vector.

Define the error between the global estimate and the local measurement as

$$\varepsilon_{i,l}^{(j)}(t) = \left(\hat{x}_l^{(j)}(t) - \hat{x}_l^{(i)}(t)\right) - y_{i,j}(t), \tag{1.24}$$

Similar with the (1.20), by applying the covariance matrix Σ_{local}, we can get

$$E_{i,l}^{(j)}(t) = (\varepsilon_{i,l}^{(j)}(t))^{-1}\Sigma_{\text{local}}^{-1}\varepsilon_{i,l}^{(j)}(t). \tag{1.25}$$

Then, the total consistency error can be get by summing up the errors over all measurable vehicles :

$$E_{i,l}(t) = \sum_{j\in\mathcal{N}_i} E_{i,l}^{(j)}(t). \tag{1.26}$$

A higher $E_{i,l}(t)$ suggests that the distributed estimate of target l contradicts with local sensor relative measurements.

Convert the error into a trust factor as

$$\gamma_{i,l}^{\text{local}}(t) = \exp\left(-E_{i,l}(t)\right), \tag{1.27}$$

where $\gamma_{i,l}^{\text{local}}(t) \in (0, 1]$, with lower values indicating poor consistency.

***Remark** 3. We can estimated covariance Σ_{host} and Σ_{local} by collecting the differences in vehicle i from a window of recent received data from l in normal condition. This avoids the need for each node to transmit its own covariance.*

3.4 Target trust and temporal evolution

We can combine the global trust and the local trust to get one single trust of the target l by following :

$$\mathcal{T}_{i,l}(t) = \mathcal{LT}_{i,l} \cdot \mathcal{DT}_{i,l}. \tag{1.28}$$

$\mathcal{T}_{i,l}(t)$ indicates the quality of target l data at the time t , that will facilitate the decision-making process for the host vehicle.

Because this value can vary abruptly, a temporal-evolution mechanism is introduced to obtain a smoother, interpretable estimate.

3.4.1 Discretization into trust quality :

We define a five-level trust quality among $q = 5$ categories : *Unreliable*, *Poor*, *Acceptable*, *Good*, and *Excellent*. When the host vehicle i computes a trust value $\mathcal{T}_{i,l}(t) \in [0, 1]$ for a neighboring vehicle l, the result is discretized into a one-hot vector $r_{i,l}(t) \in \mathbb{R}^q$. For example, if $\mathcal{T}_{i,l}(t) = 0.72$, it falls into the "Good" category (4th level), so $r_{i,l}(t) = [0, 0, 0, 1, 0]^\top$.

3.4.2 Accumulated trust history :

Over time, the accumulated trust vector $R_{i,l}(t)$ for target vehicle l is updated using a weighted aggregation of past interactions :

$$R_{i,l}(t + 1) = (1 - \mathcal{T}_{i,l}(t) \cdot \beta) \cdot R_{i,l}(t) + r_{i,l}(t), \tag{1.29}$$

where $\beta \in [0, 1]$ as a tunable weight controls how quickly past experiences decay.

Remark 4. *The choice of β is crucial for effectively capturing the dynamics of trust over time. A higher β places more emphasis on recent observations, allowing the trust score to adapt quickly to changes in behavior. Conversely, a lower β gives more weight to historical data, resulting in a smoother trust evolution that is less sensitive to transient fluctuations.*

3.4.3 Normalized trust distribution :

Finally, at the current step t (in the following, we omit t for simplifying), a normalized trust distribution vector $S_{i,l}(k)$ is computed by incorporating an a priori distribution α and a confidence parameter C :

$$S_{i,l}(k) = \frac{R_{i,l}(k) + C \cdot \alpha(k)}{C + \sum_{j=1}^{q} R_{i,l}(j)}, \tag{1.30}$$

which provides a probabilistic interpretation of the trustworthiness of the target vehicle l, blending historical observations with prior beliefs to produce a robust and adaptive trust estimate.

3.4.4 Expected trust level :

The target vehicle trust score $\mathcal{VT}_{i,l}$ is then calculated as :

$$\mathcal{VT}_{i,l} = \sum_{j=1}^{q} \frac{j-1}{q-1} \cdot S_{i,l}(j). \tag{1.31}$$

While $\mathcal{VT}_{i,l}$ provides a stable, pairwise evaluation between directly connected vehicles, a more global view is required for the observer to weigh all available information. This motivates the definition of a generalized trust vector, introduced next.

3.5 Generalized trust vector

The target trust value $VT_{i,l}$ obtained in the previous subsection quantifies how much the host vehicle i trusts a specific neighbor l. However, for the distributed observer to operate reliably, each vehicle must maintain an opinion about all other vehicles in the platoon, including those beyond its direct communication range. To achieve this, we define a generalized trust vector O_i that aggregates and propagates local trust information in a robust one-hop manner. For vehicle i, the vector is defined as

$$O_i = [O_i(1), O_i(2), \ldots, O_i(N)], \tag{1.32}$$

where O_i represents the current opinion of vehicle i regarding all platoon members, while $O_i(j) \in [0, 1]$ specifies vehicle i's current trust opinion toward vehicle j.

1) Self-Trust. Each vehicle assumes full confidence in its own state :

$$O_i(i) = 1, \tag{1.33}$$

2) Direct-Neighbor Trust. For every neighbor $j \in \mathcal{N}_i$ with which vehicle i exchanges data directly, the trust opinion is set as :

$$O_i(j) = VT_{i,j}, \tag{1.34}$$

where $VT_{i,j}$ is the smoothed target-trust value derived in the previous subsection.

3) One-Hop Trust Propagation. For any non-neighbor vehicle $j \notin \mathcal{N}_i \cup \{i\}$, the host vehicle i estimates $O_i(j)$ using only information from its direct neighbors $\mathcal{N}_i$.

$$S_{i,j} = \Big\{ O_k(j) \mid k \in \mathcal{N}_i,\ VT_{i,k} > \theta_{\min},\ \delta_{i,k}(t) = 1 \Big\} \tag{1.35}$$

Here, $\theta_{\min}$ is the minimum acceptable trust threshold,

Each valid neighbor k is associated with a *credibility weight*

$$w_{i,k} = \frac{VT_{i,k}}{\sum_{p \in S_{i,j}} VT_{i,p}} \tag{1.36}$$

If the credible set $S_{i,j}$ is non-empty, the propagated opinion is computed using a *weighted median* operator :

$$O_i(j) = \mathrm{wMed}\big(\{O_k(j)\}_{k \in S_{i,j}}, \{w_{i,k}\}_{k \in S_{i,j}}\big), \tag{1.37}$$

which limits the influence of outliers or manipulated values.

4) Index Distance-Based Fallback. If the credible set $S_{i,j}$ is empty, vehicle i resorts to a *host-centric distance weighting* of all its neighbors :

$$O_i(j) = \frac{\sum_{k \in \mathcal{N}_i} g_{i,k}\, O_k(j)}{\sum_{k \in \mathcal{N}_i} g_{i,k}}, \qquad g_{i,k} = \frac{1}{1 + |i - k|}. \tag{1.38}$$

where $|i - k|$ is the index distance between vehicles i and k. The closer a neighbor k is to host i, the larger its influence $g_{i,k}$. This fallback guarantees that the trust vector remains fully populated even when no verified propagated opinion exists.

6) Final Expression. Combining all cases, the complete definition of the generalized trust vector is

$$O_i(j) = \begin{cases} 1, & j = i, \\ VT_{i,j}, & j \in \mathcal{N}_i, \\ \mathrm{wMed}\big(\{O_k(j)\}, \{w_{i,k}\}\big), & S_{i,j} \neq \varnothing, \\ \dfrac{\sum_{k \in \mathcal{N}_i} g_{i,k}\, O_k(j)}{\sum_{k \in \mathcal{N}_i} g_{i,k}}, & S_{i,j} = \varnothing. \end{cases} \tag{1.39}$$

This generalized representation allows each vehicle to maintain a resilient, full-platoon trust profile that supports the adaptive observer weight design described in the following section.

4 Weight Design for observer based on the trust

The trust framework provides each vehicle with a set of quantified opinions $\mathcal{O}_{i,j}$ representing the reliability of information from other vehicles. This section shows how these values are used to define adaptive observer weights $w_{il}^{j}(t)$ and establishes conditions for stability of the estimation error dynamics.

4.1 Weight based on the trust

Based on the virtual communication graph $\bar{\mathcal{G}}^{(j)}$ as that includes the local observer node 0, each vehicle i identifies its legitimate neighbors at time t :

$$\begin{aligned}\mathcal{LN}_i^{(j)}(t) = &\Big\{\, l \in N_i \mid O_i(l,t) \geq \theta_{\min}\Big\} \\ &\cup \ \Big(\{0\} \text{ if } \mathcal{LT}_{i,j}(t) \geq \theta_{\min}\Big)\end{aligned} \tag{1.40}$$

To limit the influence of any single neighbor, a normalization factor is introduced.

$$n_{il}^{(j)}(t) = \max\Big\{\kappa, \Big|\mathcal{LN}_i^{(j)}(t)\Big| + 1\Big\} \geq 1, \forall l \in \mathcal{LN}_i^{(j)} \tag{1.41}$$

where $\kappa > 0$ is the parameter to limit the maximum influence from the neighbors of ith vehicle. Then, the observer's weight gain can be chosen as

$$w_{il}^{(j)}(t) = \begin{cases} \dfrac{1}{n_{il}^{(j)}(t)}, & l \in \mathcal{LN}_i^{(j)}(t) \\ 0, & \text{otherwise.} \end{cases} \tag{1.42}$$

According to the weight gain, the communication graph will be switched depending on the time. The following assumption is required.

Assumption 2 (*Jointly connected graph). There exists a scalar constant $T > 0$ such that for all $t \geq 0$, the union graph $\cup_{[t,t+T]}\bar{\mathcal{G}}^{(j)}(t)$ contains a spanning tree.*

Assumption 2 means that for all $t \geq 0$, every node $i \in \mathcal{V}$ is reachable from node "0 in the union graph $\cup_{[t,t+T]}\bar{\mathcal{G}}^{(j)}(t)$. This ensures the connectivity of the proposed switching communication topology, which is required for our observer design method. For more details on this assumption, we refer the reader to [**hao2024eventtriggered**].

4.2 Stability of the error

In this section, we explore the conditions that guarantee the stability of the estimation error dynamics of the proposed observer given in (1.5). The error dynamics are given as follows :

$$e_0^{(j)}(t+1) = (A_\mathrm{b} - F_j C_\mathrm{b})\, e_0^{(j)}(t) + d_0^j, \tag{1.43}$$

$$\begin{aligned}e_{\mathcal{DO}}^{(j)}(t+1) = &\Big(w_0^{(j)} \otimes A_\mathrm{b}\Big)\, e_0^{(j)}(t) \\ &+ \Big(\Big(I_N - \bar{\mathcal{L}}^{(j)}\Big) \otimes A_\mathrm{b}\Big)\, e_{\mathcal{DO}}^{(j)}(t) + d_{\mathcal{DO}}^j,\end{aligned} \tag{1.44}$$

with

$$\bar{\mathcal{L}}^{(j)} = \begin{bmatrix} \sum_{l\in\{\mathcal{N}_1\cup\{0\}\}} w_{1l}^{(j)} & -w_{12}^{(j)} & \cdots & -w_{1N}^{(j)} \\ -w_{21}^{(j)} & \sum_{l\in\mathcal{N}_2} w_{2l}^{(j)} & \cdots & -w_{1N}^{(j)} \\ \vdots & \vdots & \ddots & \vdots \\ -w_{N1}^{(j)} & -w_{N2}^{(j)} & \cdots & \sum_{l\in\mathcal{N}_N} w_{Nl}^{(j)} \end{bmatrix}$$

$$\begin{aligned} d_0^j &= -\Delta_j, \\ d_{DO}^j &= (I_N \otimes B_b)(\hat{u}_j - u_j) - (1_N \otimes \Delta_j) \\ e_{\mathcal{DO}}^{(j)}(t) &= \operatorname{col}\left(e_1^{(j)}(t), \cdots, e_N^{(j)}(t) \right) \end{aligned}$$

where $w_0^{(j)} = \operatorname{col}\left(w_{10}^{(j)}, \cdots, w_{N0}^{(j)} \right)$ is a vector describing the connection between the local observer ($\mathcal{LO}_j$) and other distributed observers ($\mathcal{DO}_i$), $i \in \mathcal{V}$; $\bar{\mathcal{L}}^{(j)}$ is the virtual weighted Laplacian matrix of the virtual graph $\bar{\mathcal{G}}^{(j)}$.

Before stating the main theorem of this section, we need to make the following assumption which is necessary to develop an ISS bound on the estimation errors.

Assumption 3. *The disturbance vectors $d_0^j(t)$ and $d_{\mathrm{DO}}^j(t)$ are bounded. That is, there exist $\bar{d}_0 \geq 0$ and $\bar{d}_{\mathrm{DO}} \geq 0$ such that $|d_0^j(t)| \leq \bar{d}_0$ and $|d_{\mathrm{DO}}^j(t)| \leq \bar{d}_{\mathrm{DO}}$, for all $t \geq 0$.*

Now, we are ready to state the following theorem, which provides the main sufficient conditions ensuring the stabilization of the error system (1.43)–(1.44).

Theorem 1. *Consider a vehicle platoon (1.4) over a directed weighted communication network $\bar{\mathcal{G}}^{(j)}$. Let (1.5) be the corresponding distributed observer. Assume that the following conditions hold true :*

1. *For all $j \in \mathcal{V}$, there exists F_j such that the matrix $(A_b - F_j C_b)$ is Schur stable.*
2. *For all $i \in \mathcal{V}$, $w_{i0}^{(j)} \geq 0$ and $w_{il}^{(j)} \geq 0$ satisfy*

$$\sum_{l\in\bar{\mathcal{G}}^{(j)}} w_{il}^{(j)} \leq 1. \tag{1.45}$$

3. *There exists $\alpha \in (0,1)$ and a matrix norm $\|\cdot\|_*$ such that, uniformly over all admissible switches,*

$$\|\left(I_N - \bar{\mathcal{L}}^{(j)} \right) \otimes A_b\|_* \leq \alpha < 1. \tag{1.46}$$

Then, the estimation error (1.43)–(1.44) is ISS. In particular, there exist constants $c_0, c_1 > 0$ and $\lambda \in (0,1)$ such that

$$\|e_0^{(j)}(t)\| \leq \lambda^t \|e_0^{(j)}(0)\| + c_0 \bar{d}_0, \tag{1.47}$$

$$\begin{aligned} \|e_{DO}^{(j)}(t)\| &\leq \lambda^t \|e_{DO}^{(j)}(0)\| \\ &\quad + c_1 \Big(\|e_0^{(j)}(0)\| + \bar{d}_0 + \bar{d}_{DO} \Big). \end{aligned} \tag{1.48}$$

A sketch of the proof. The proof is straightforward and relies on Assumptions 2 and 3. Its simplicity stems from the fact that the coupled system (1.43)–(1.44) has a cascade structure. Condition (1) guarantees that (1.43) is ISS. Condition (2), ensured by Assumption 2, makes it possible to verify Condition (3) by applying the Gershgorin theorem. Finally, Condition (3), together with (2), ensures that (1.44) is ISS. □

Remark 5. *The conditions of Theorem 1 can be verified using a Lyapunov stability approach. This analysis yields explicit expressions for the matrix F_j and the constants c_0, c_1, α, and λ, depending on the selected Lyapunov functions. Since the overall system exhibits a cascade structure, system (1.43) can be analyzed independently from system (1.44), which effectively leads to a separation principle. Due to the page limit constraint, these detailed results are not presented as a separate theorem.*

5 Simulation Results

This section evaluates the performance of the proposed trust-aware distributed observer for a connected vehicle platoon under various cyberattack scenarios. The main objectives of the simulation are to :

- Assess the observer resilience to falsified and delayed information.
- Analyze the evolution of trust scores for detecting and isolating malicious nodes.
- Quantify estimation performance degradation under different types of attacks.

5.1 Simulation setup

The simulation considers a platoon of four fully connected vehicles, where each vehicle exchanges information with all others through vehicle-to-vehicle (V2V) communication. All vehicles share similar nominal parameters, with small modeling variations treated as uncertainties.The proposed trust-aware distributed observer runs onboard each vehicle to estimate states and evaluate the reliability of shared data. Table 1.1 summarizes the main parameters used for the vehicle model and the trust framework.

Table 1.1 – Main simulation parameters.

Parameter	**Value**	**Description**
T_s	0.1 s	Sampling time
τ	1.5 s	Engine lag constant
w_v, w_d, w_a	1.2, 2.0, 0.9	Local trust weights
$\beta_{\text{trust}}, C, k$	0.5, 0.2, 5	Trust decay and regularization
d_{norm}	15 m	Normalization constants
$\theta_{\min}$	0.5	Minimum trust threshold
N	4	Number of vehicles

5.2 Attack scenarios

To assess the robustness of the vehicle platooning system, six attack cases are defined, as summarized in Table 1.2. All attacks are launched by the leader vehicle and broadcast to every follower in the platoon. Each attack is active during the interval $[10, 15]$ s. For reproducibility , the observer is evaluated under the assumption that the follower controllers are known, thereby isolating and emphasizing the observer capability to detect and mitigate malicious data. The parameter p denotes the probability of the attack or fault occurrence.

Table 1.2 – Six attack cases in the scenario.

Case	**Type**	**Target**	**Parameters**
1	Bias	Position	bias $= -5$ m
2	Faulty	Position	int. $= 10$, $p = 0.3$
3	Bias	Velocity	bias $= -2$ m/s
4	Faulty	Velocity	int. $= 2.5$, $p = 0.3$
5	Faulty	Acceleration	int. $= 1.0$, $p = 0.3$
6	Drop	All	$p_{\text{drop}} = 0.5$

5.3 Results

Figure 1.3 presents a heatmap of the normalized impact scores for each vehicle across all six attack cases. The normalized impact score is obtained by dividing the raw combined error by the maximum observed combined error across all vehicles and attack scenarios. This normalization enables a clear comparison of the relative severity of each attack on individual vehicles.

To quantitatively assess the influence of different cyberattack scenarios, both raw combined errors and normalized impact scores were analyzed. The raw combined error provides a direct measure of deviation in distance, velocity, and acceleration estimation, whereas the normalized impact score offers a more balanced indicator of overall system degradation relative to nominal operating performance.

Across all cases, the DoS attack (Case 6) exhibited the highest normalized impact score (0.872), confirming it as the most severe attack scenario. This high value reflects a significant degradation in estimation accuracy and control performance during the 5-second attack window. In contrast, the velocity bias attack (Case 3) with a bias of -2 m/s produced the lowest normalized impact score (0.240), indicating minimal influence on system stability and tracking capability.

From the vehicle perspective, V4 was identified as the most affected vehicle, reaching the maximum normalized impact score (0.872), whereas vehicle V2 demonstrated the highest resilience, with the lowest average impact score (0.362). This suggests that V4 local dynamics or communication dependencies make it more sensitive to external disruptions, while V2 maintains more stable performance under varying attack conditions.

Overall, the relatively low mean normalized impact scores across all cases indicate that the proposed trust-aware distributed observer effectively mitigates diverse types of cyberattacks and maintains accurate state estimation for each vehicle in the platoon.

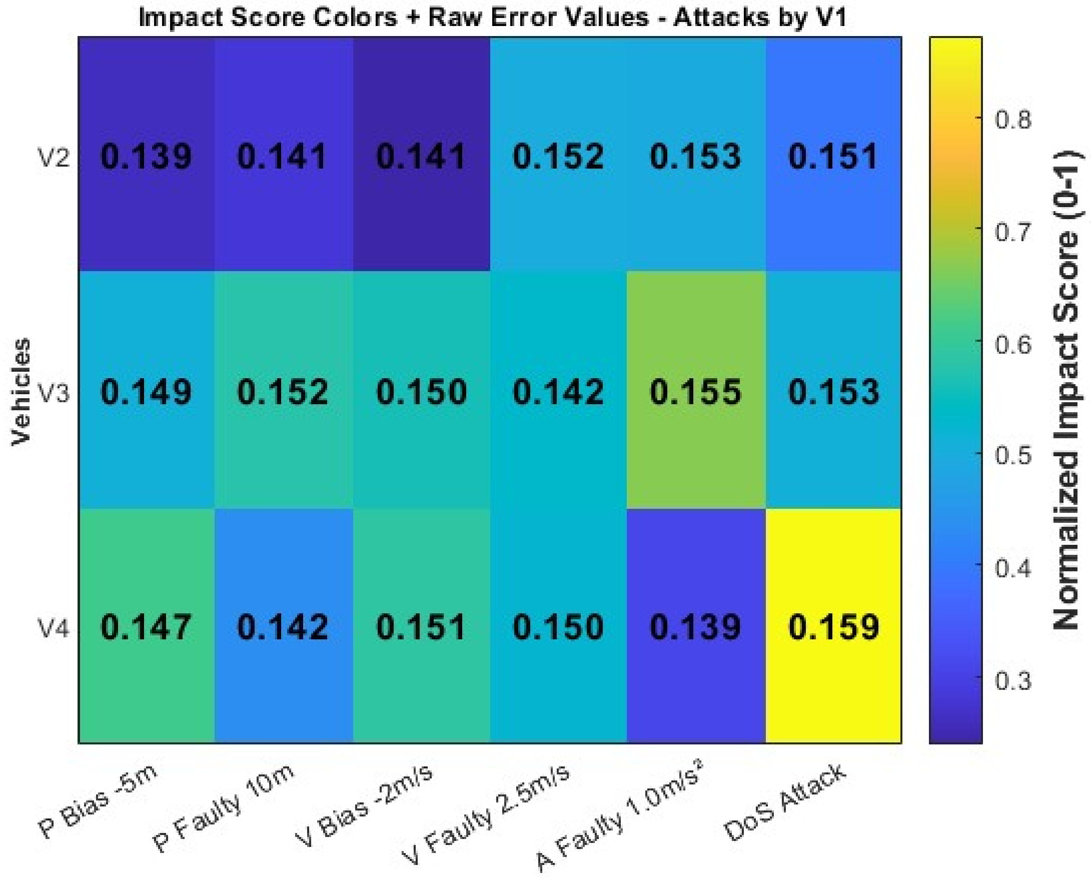

FIGURE 1.3 – Heatmap of normalized impact scores for all vehicles and attack cases.

In Figure 1.4, the trust score evolution for the DoS attack (Case 6) is illustrated. During the attack period $[10, 15]$ s, these 3 vehicles (V2, V3, V4) exhibit a noticeable drop in trust values below the threshold of 0.5, indicating successful detection of the communication disruption. Once the attack ceases, the trust scores gradually recover, demonstrating the system ability to restore confidence and resume normal operation. This behavior confirms the effectiveness of the proposed trust mechanism in identifying and isolating malicious data in real time.

5.4 Distributed State Estimation for Platoon Control

In this section, we present a platoon control framework that leverages distributed state estimation to enhance longitudinal control performance. The approach adopts a constant time headway spacing (CTHS) policy to maintain small inter-vehicle distances, improving traffic flow and safety. Two controllers are employed : an inner controller, based on the Intelligent Driver Model (IDM), for local longitudinal control using estimates from the local observer, and an outer controller, the Cooperative Adaptive Cruise Control (CACC), for distributed coordination using estimates from the distributed observer. These controllers are combined to form a robust final control strategy, balancing local responsiveness and platoon-wide consensus.

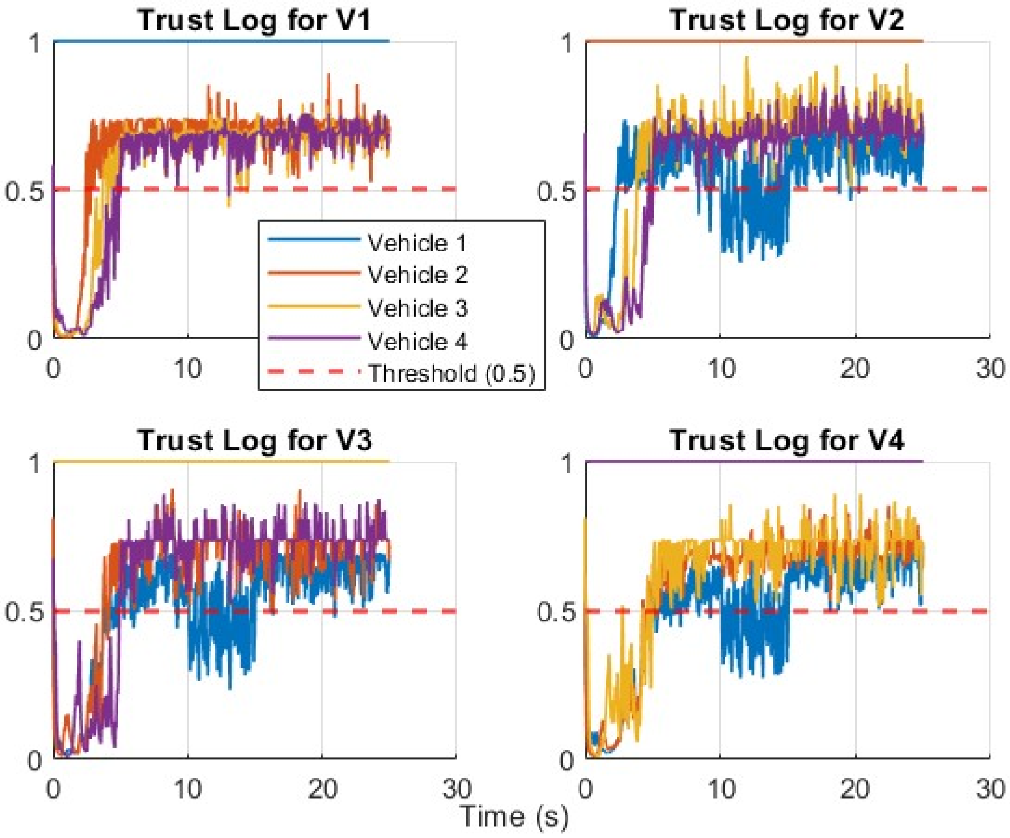

FIGURE 1.4 – Trust score evolution during DoS attack (***Case 6***).

5.4.1 Notation

5.4.2 Inner/Local Controller : Intelligent Driver Model (IDM)

The IDM is a car-following model that computes a vehicle's desired acceleration based on its own state and the relative state to its predecessor. Here, it serves as the inner controller for local longitudinal control, utilizing the local observer's estimate of the vehicle's own state, $\hat{x}_0^{(i)}$. The IDM acceleration for vehicle i is given by :

$$a_{\mathrm{IDM},i} = \alpha \left[1 - \left(\frac{\hat{v}_0^{(i)}}{v_0} \right)^{\delta} - \left(\frac{s^*(\hat{v}_0^{(i)}, \Delta v_{i-1,i})}{s_{i-1,i}} \right)^2 \right], \tag{1.49}$$

where :

- $\hat{v}_0^{(i)}$ is the estimated velocity of vehicle i from $\mathcal{LO}_i$.
- $s_{i-1,i} = s_{i-1} - s_i - L$ is the actual relative distance to the predecessor, with L as the vehicle length (assuming direct sensor measurement for simplicity, as the query specifies IDM uses only local observer estimates, but relative distance typically requires predecessor data).
- $\Delta v_{i-1,i} = v_{i-1} - \hat{v}_0^{(i)}$ is the actual relative velocity, using the predecessor's true velocity v_{i-1} from sensors.

— $s^*(\hat{v}_0^{(i)}, \Delta v_{i-1,i}) = s_0 + T\hat{v}_0^{(i)} + \frac{\hat{v}_0^{(i)}\Delta v_{i-1,i}}{2\sqrt{\alpha\beta}}$ is the desired minimum gap.

— α, β, δ, v_0, s_0, and T are model parameters : maximum acceleration, comfortable deceleration, free-flow exponent, desired velocity, minimum gap, and time headway, respectively.

Since $\mathcal{LO}_i$ provides only $\hat{x}_0^{(i)}$ and not the predecessor's state, we assume vehicle i uses sensor data (e.g., radar) for $s_{i-1,i}$ and $\Delta v_{i-1,i}$, consistent with standard IDM implementations, while adhering to the query's directive to use local observer estimates for the vehicle's own state.

5.4.3 Cooperative Controller : Cooperative Adaptive Cruise Control (CACC)

The CACC enhances platoon coordination by leveraging information from multiple preceding vehicles via the distributed observer $\mathcal{DO}_i$. For vehicle i, the CACC control input is :

$$\begin{aligned} u_{\text{CACC},i}(k) = \sum_{j=1}^{i-1} [\kappa_s \left(\hat{s}_i^{(j)}(k) - \hat{s}_0^{(i)}(k) - d_{i,j}(k)\right) \\ +\kappa_v \left(\hat{v}_i^{(j)}(k) - \hat{v}_0^{(i)}(k)\right) + \kappa_a \left(\hat{a}_i^{(j)}(k) - \hat{a}_0^{(i)}(k)\right)] \end{aligned} \tag{1.50}$$

where :

— $\hat{s}_i^{(j)}(k)$, $\hat{v}_i^{(j)}(k)$, $\hat{a}_i^{(j)}(k)$ are the estimated position, velocity, and acceleration of vehicle j from $\mathcal{DO}_i^{(j)}$.

— $\hat{s}_0^{(i)}(k)$, $\hat{v}_0^{(i)}(k)$, $\hat{a}_0^{(i)}(k)$ are the estimated position, velocity, and acceleration of vehicle i from $\mathcal{LO}_i$ (included for consistency, though the query specifies distributed observer estimates for CACC).

— $d_{i,j}(k) = d + h\hat{v}_0^{(i)}(k)$ is the desired spacing between vehicles i and j, with d as a constant gap and h as the time headway (for $j = i-1$, this aligns with the CTHS policy).

— κ_s, κ_v, κ_a are control gains for position, velocity, and acceleration errors, respectively.

This formulation ensures that vehicle i adjusts its behavior based on the estimated states of all preceding vehicles, promoting consensus and stability across the platoon.

5.4.4 Final Controller

The final control input integrates the IDM and CACC controllers to balance local and cooperative objectives. The target acceleration is :

$$a_{\text{target},i} = (1-\gamma(t))a_{\text{IDM},i} + \gamma(t)u_{\text{CACC},i}, \tag{1.51}$$

where $\gamma(t) \in [0,1]$ is a tuning parameter that is influenced by the opinion score mentioned in Section 3. For this context, we choose $\gamma(t) = \min(\mathcal{O}_i)$. To prevent abrupt changes of the mixing 2 type controller, a first-order filter is applied :

$$u_i(t) = u_i(t-1) + \tau_f(a_{\text{target},i} - u_i(t-1)), \tag{1.52}$$

where τ_f is the filter time constant.

5.4.5 Expected Distance (Spacing)

To validate the control strategy, we derive the expected steady-state spacing :

— **IDM Steady-State Spacing** : When $a_{\mathrm{IDM},i} = 0$ and $\Delta v_{i-1,i} = 0$:

$$1 - \left(\frac{\hat{v}_0^{(i)}}{v_0}\right)^{\delta} = \left(\frac{s_0 + T\hat{v}_0^{(i)}}{s_{i-1,i}}\right)^{2},$$

yielding :

$$s_{i-1,i} = \frac{s_0 + T\hat{v}_0^{(i)}}{\sqrt{1 - \left(\frac{\hat{v}_0^{(i)}}{v_0}\right)^{\delta}}}.$$

— **CACC Steady-State Spacing** : For the CTHS policy, when the platoon reaches consensus (all velocities equal), the spacing between vehicle i and $i-1$ is :

$$\boxed{s_i = d + \frac{h\,\hat{v}_0^{(i)}}{i-1}\,.}$$

These expressions allow comparison with simulation results, assessing the controllers' ability to maintain desired spacing under various conditions, including cyber-attacks.

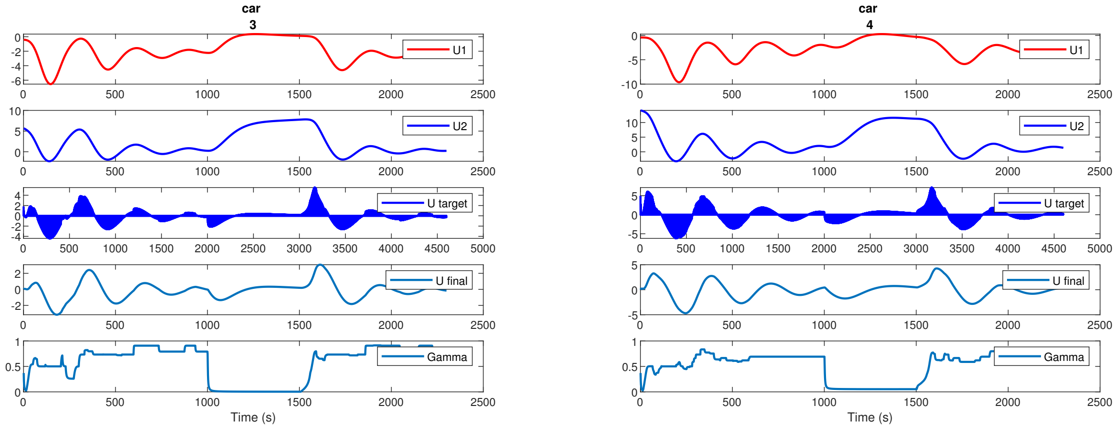

FIGURE 1.5 – Controller of vehicle 2 and 3.

In figure 1.5, we can see the control input of vehicle 2 and 3, which is the acceleration of the vehicle. Smoothly switched between 2 controllers, and no abrupt change in the control input.

In figure 1.6, we can see the relative state of the vehicle in platoon, which is the difference between the position of the vehicle and the position of its predecessor. That prouve even in attack scenario, the vehicle can keep good distance No accident or crash happen in the platoon, which is a good sign of the robustness of the platoon control framework. Also that show the expected distance between the vehicle and its predecessor is maintained, which is the desired spacing in the platoon.

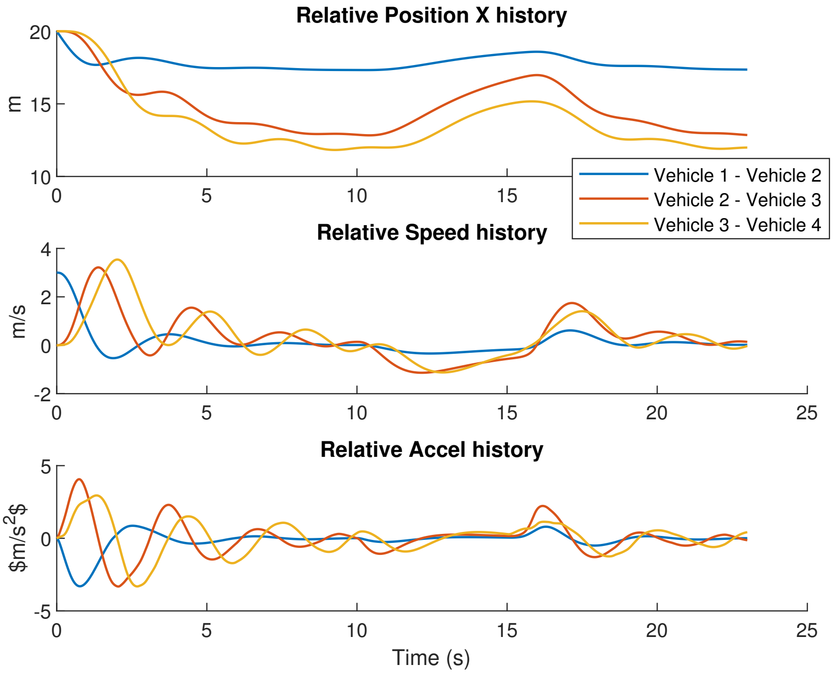

Figure 1.6 – Relative state of the vehicle in platoon

6 Conclusion and Future Work

This paper presented a distributed observer-based platoon control framework capable of maintaining stability and safety under cyberattacks. The approach combines local and distributed observers with a trust evaluation mechanism to assess the reliability of inter-vehicle data, ensuring robust operation even with compromised nodes. Future work will focus on enhancing the trust mechanism by coupling it more closely with the controller, enabling trust evaluation for non-neighboring vehicles, and improving adaptability under dynamic communication topologies. We will also investigate machine-learning based trust estimation and extend the observer design by using matrix gains to improve estimation accuracy and performance.

RÉSUMÉ

MOTS CLÉS

Estimation, mot clé 2, mot clé 3, mot clé 4

ABSTRACT

The reliability and safety of Connected and Autonomous Vehicles (CAVs) rely heavily on the real-time availability of precise dynamic variables. However, equipping vehicles with high-end sensors to measure every necessary variable—such as sideslip angles or vehicle trajectories—is often economically unfeasible or physically impossible. Furthermore, CAVs are increasingly vulnerable to sensor faults and cyber-attacks targeting inter-vehicle communications (V2V/V2I), which can lead to critical failures. To address these challenges without incurring the high cost of physical hardware redundancy, this thesis proposes the development of an embedded electronic board functioning as a resilient "soft sensor" based on analytical redundancy.

The research adopts a hybrid modeling methodology that integrates physical differential equations with online learning-based neuro-adaptive neural networks. This approach allows for the accurate representation of complex, time-varying parameters and unknown inputs within the vehicle's dynamics. Based on these models, advanced nonlinear estimation algorithms—specifically Unknown Input Observers (UIO) and neuro-adaptive observers—are developed to reconstruct missing state variables and ensure system resilience against data loss and malicious signal injection.

The proposed estimation strategies are applied to two primary control challenges:

- Resilience to Cyber-Attacks in Adaptive Cruise Control (ACC): The thesis investigates specific architectures for detecting and mitigating cyber-attacks, such as Denial of Service (DoS) and False Data Injection (FDI), within autonomous and cooperative ACC systems.
- Vehicle Tracking and Platooning: Novel estimation algorithms are proposed to handle the nonlinear dynamics and unknown forces inherent in tracking surrounding vehicles, which is essential for decision-making in maneuvers like lane changes and platooning.

The final contribution of this work is the design and fabrication of a prototype embedded electronic board. This hardware integrates the developed algorithms, validating the theoretical contributions and demonstrating a cost-effective, scalable solution for enhancing the autonomy, security, and fault tolerance of next-generation vehicles.

KEYWORDS

State Estimation, keyword 2, keyword 3, keyword 4