SAE INTERNATIONAL

ARTICLE INFO
Article ID: 12-08-04-0033

doi:10.4271/12-08-04-0033

A Reinforcement Learning-Based Parameter Tuning Approach for a Secure Cooperative Adaptive Cruise Control System

Farahnaz Javidi-Niroumand[1] and Arman Sargolzaei[1]

[1]University of South Florida, Department of Mechanical Engineering, USA

Abstract

Connected and autonomous vehicles (CAVs) rely on communication channels to improve safety and efficiency. However, this connectivity leaves them vulnerable to potential cyberattacks, such as false data injection (FDI) attacks. We can mitigate the effect of FDI attacks by designing secure control techniques. However, tuning control parameters is essential for the safety and security of such techniques, and there is no systematic approach to achieving that. In this article, our primary focus is on cooperative adaptive cruise control (CACC), a key component of CAVs. We develop a secure CACC by integrating model-based and learning-based approaches to detect and mitigate FDI attacks in real-time. We analyze the stability of the proposed resilient controller through Lyapunov stability analysis, identifying sufficient conditions for its effectiveness. We use these sufficient conditions and develop a reinforcement learning (RL)-based tuning algorithm to adjust the parameter gains of the controller, observer, and FDI attack estimator, ensuring the safety and security of the developed CACC under varying conditions. We evaluated the performance of the developed controller before and after optimizing parameters, and the results show about a 50% improvement in accuracy of the FDI attack estimation and a 76% enhancement in safe following distance with the optimized controller in each scenario.

History

Received: 27 Jun 2024
Revised: 30 Oct 2024
Accepted: 24 Dec 2024
e-Available: 25 Jan 2025

Keywords

Cooperative adaptive cruise control, False data injection attack, Lyapunov stability, Parameter tuning, Reinforcement learning

Citation

Javidi-Niroumand, F. and Sargolzaei, A., "A Reinforcement Learning-Based Parameter Tuning Approach for a Secure Cooperative Adaptive Cruise Control System," *SAE Int. J. of CAV* 8(4):2025, doi:10.4271/12-08-04-0033.

ISSN: 2574-0741
e-ISSN: 2574-075X

I. Introduction

The increasing prevalence of connected and autonomous vehicles (CAVs) holds the promise of revolutionizing the transportation of people and goods. CAVs enhance safety, efficiency, and convenience by utilizing real-time data and communication with surrounding systems. This connectivity allows CAVs to detect unsafe situations and adjust to changing traffic conditions sooner than is possible with onboard sensors alone. To maintain safe distances between vehicles and smooth traffic flow, CAVs utilize features like adaptive cruise control (ACC), which can automatically adjust the vehicle's speed. Cooperative adaptive cruise control (CACC) is an extension of traditional ACC that allows vehicles to coordinate with each other, allowing them to follow each other at shorter distances while maintaining safety, thus improving efficiency. By optimizing the way vehicles interact with each other on the road, CACC has the potential to improve traffic flow, reduce congestion, and enhance fuel efficiency [1, 2]. However, as CACC systems integrate cutting-edge sensors and communication technologies, they become more vulnerable to cyberattacks such as false data injection (FDI) attacks [3]. Therefore, it is necessary to develop secure control algorithms to detect and mitigate FDI attacks in real-time. These strategies use fault detection and mitigation technologies to detect irregularities that indicate cyberattacks or system failures, allowing for faster response and reducing the impact of attacks.

The development of attack-resilient control systems for CAVs primarily involves the identification and mitigation of cyberattacks to maintain system performance. Key methodologies include using advanced vehicle control systems designed to resist cyberattacks, with specific techniques such as Lyapunov stability analysis to detect and counteract FDI attacks in real-time, as discussed in various studies [4, 5, 6]. Other approaches focus on dealing with denial of service (DoS) [7] and sensor attacks [8] using adaptive estimation and sliding mode design for quick identification and robust state estimation even when sensors are compromised.

In [9] the authors present the development of a longitudinal controller that achieves string-stable platooning by exchanging information with other cooperative vehicles via wireless connection. To validate the performance of the controller in a real-life situation, [10] implemented the CACC via two controllers to manage the approaching maneuver to the leader vehicle and to regulate car-following within the platoon. The authors of [11] have created a reliable H_∞ controller for CACC using the loop shaping design methodology to obtain the necessary tracking characteristics when robustness and performance are needed and the system is confronting competing string stability. In [12], the authors designed a control structure that addresses the heterogeneous CACC issue. They utilized online information and applied adaptive optimal control to establish the minimum headway values required to ensure vehicle string stability.

While all these papers have explored CACC system design, they have neglected to address the performance of their strategies in the presence of cyberattacks. Investigating the system's resilience to cyber attacks, specifically FDI attacks, is imperative for ensuring its robustness and reliability in real-world scenarios. Researchers have dedicated several studies to examining the vulnerabilities and countermeasures related to FDI attacks within the framework of CACC systems. In [13, 14], the authors studied the drastic impact of FDI attacks on the vehicular platoon. Other studies highlight the critical importance of ensuring security in CACC environments, where vehicles communicate with each other to maintain safe distances and synchronize speeds. In [15], a resilient state estimator was developed to provide safety and performance in a platoon of CAVs under FDI attacks. The authors in [16] propose a platoon state information fusion method using the characteristics of the communication channel and compare the status information of the platoon members with the desired status information to detect FDI attacks on the platoon. In [17], a distributed attack monitor and H_∞ CACC controllers were suggested as a way to reduce the impact of a specific type of attack known as stealthy FDI attacks. These attacks hide in system disturbances and uncertainty and are hard to spot. The authors in [18] have modeled FDI attacks as ghost vehicles inserting into the platoons of vehicles using a partial differential equation (PDE) approach. To model the FDI attack as a ghost vehicle, the attacker needs sufficient knowledge about the vehicles in the platoon. This approach focuses only on attacks that create fictitious vehicles, but FDI attacks can take many forms. Additionally, the ghost vehicle model assumes that FDI attacks result in obvious anomalies, such as a vehicle appearing out of nowhere. However, well-designed FDI attacks may evade simple anomaly detection, as they might introduce noise or small disturbances that are harder to distinguish from normal variations.

Although many studies have explored the impact of FDI attacks on CACC, most of them have not conducted a thorough stability analysis to assess the consequences of such attacks on system performance. Lyapunov-based nonlinear controllers and observers for CACC are proposed in [19, 20], capable of estimating FDI attacks in real-time. The authors demonstrated that their proposed controllers and FDI attack estimation techniques ensure semi-globally uniformly bounded tracking under FDI attacks. However, these proposed controllers, observers, and FDI attack estimators manually select parameters for limited scenarios, rendering them unsafe for other situations. For example, these papers only consider a constant safe desired distance between vehicles, whereas in real-world scenarios, the safe distance should vary with the velocities of the lead and following vehicles.

Furthermore, the FDI attack estimator works less well for time-varying FDI attacks, which their updated laws make it hard to accurately estimate.

To address the mentioned limitations, we have developed a robust CACC system featuring both stability analysis and performance assurance measures to ensure its security. However, one of the primary challenges lies in tuning the controller's parameters to guarantee the safe and efficient operation of CAVs in the face of FDI attacks. If not optimally set, the system might become either overly conservative, affecting efficiency, or insufficiently robust, making it vulnerable to disturbances and attacks. Additionally, as operational scenarios evolve, these parameters may require adjustments, underscoring the need for ongoing monitoring and re-tuning to maintain system resilience and functionality in the face of emerging cyberattacks.

There are several techniques to tune controller parameters, from manual adjustments based on experience and heuristic rules to more sophisticated automated and adaptive techniques such as particle swarm optimization (PSO) [21]. Among all these methods, machine learning (ML) can be effectively used to tune the parameters of controllers, providing a powerful approach to handling complex systems with dynamic and nonlinear behaviors that are difficult to model and tune with traditional methods [22]. One of the most promising applications of ML in control system tuning is through reinforcement learning (RL). In this approach, an agent gains decision-making skills by acting in a certain way to accomplish objectives in an environment [23]. RL-based methods are widely used in the field of control [24, 25, 26].

In this article, we use RL-based technique for tuning parameters of our developed secure CACC. RL can be used to find optimal control strategies and dynamically adjust controller parameters. By interacting with the environment through trial and error, it learns the best actions based on rewards or penalties received for its performance. RL is well-suited for environments where the system dynamics are complex and not well-understood [27, 28, 29]. The implementation of TD3 has been discussed in [30, 31, 32, 33] to tune and optimize controller dynamics.

This article contributes to the following areas: Section II will discuss the dynamic model of a CACC system while it is under FDI attack. In Section III, the design procedure of the controller, observer, and attack estimator has been provided. Section IV provides a thorough discussion of the RL tuning approach to enhance the controller and observer's performance. Section V describes the evaluation of the suggested and optimized controllers for two separate scenarios. The conclusions are presented in Section VI.

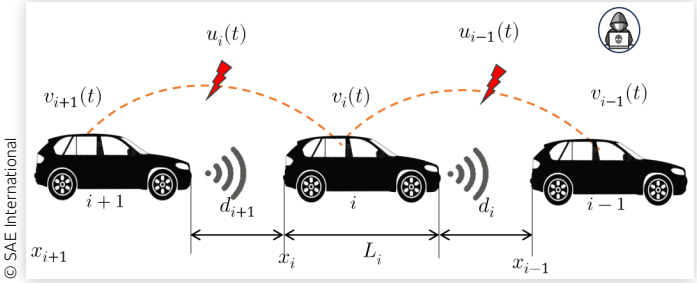
II. Dynamic Model of CACC under FDI Attack

Figure 1 illustrates a CACC system under attack. An attacker can send false information about a vehicle's position, speed, or intentions, resulting in a collision within the platoon of vehicles. In addition, attackers can disrupt traffic flow by interfering with the communication between vehicles. For instance, in FDI attacks, the attacker purposefully introduces false or modified data into CACC by targeting the communication channel between a vehicle and other vehicles or infrastructure [20, 34, 35]. This manipulation modifies the vehicle's perception of its surroundings, which may lead it to make unsafe behaviors [21].

A. Dynamic Model Representation

The dynamic behavior of individual vehicles within the CACC system encompasses longitudinal dynamics, including acceleration and braking, and can be characterized as a first-order transfer function. The CACC setup labels the leading vehicle with the index $i-1$, and the subsequent vehicles in the platoon with the index i, ranging from 2 to n, where n is the number of the vehicles. The leading vehicle adjusts its speed in accordance with a reference speed profile and transmits acceleration commands to the following vehicle via a communication

FIGURE 1 The architecture of a CACC string of vehicle under FDI attack.



channel. The follower vehicles are equipped with a radar sensor and an intervehicle communication network to monitor parameters such as the velocity and position of the leader. The dynamic model of the vehicle in the platoon is described as

$$\begin{cases} \dot{x}_i(t) = v_i(t) \\ \dot{v}_i(t) = -b_i v_i(t) + c_i u_i(t) \end{cases} \quad \text{Eq. (1)}$$

where t denotes the time, $x_i(t) \in \mathbb{R}$ is the position, $v_i(t) \in \mathbb{R}$ is the velocity, $u_i(t) \in \mathbb{R}$ is the control input signal of the follower vehicle, and $b_i \in \mathbb{R}_{>0}$ and $c_i \in \mathbb{R}_{>0}$ denote the vehicle model parameters derived based on our vehicle-in-the-loop experimental results.

We describe the dynamic model of the lead vehicle as

$$\begin{cases} \dot{x}_{i-1}(t) = v_{i-1}(t) \\ \dot{v}_{i-1}(t) = -b_{i-1} v_{i-1}(t) + c_{i-1} u_{i-1}(t) \end{cases} \quad \text{Eq. (2)}$$

where $x_{i-1} \in \mathbb{R}$, $v_{i-1} \in \mathbb{R}$, and $u_{i-1} \in \mathbb{R}$ denote the position, velocity, and control input of the leader vehicle, respectively. The leader vehicle dynamics are defined as $b_{i-1} \in \mathbb{R}_{>0}$ and $c_{i-1} \in \mathbb{R}_{>0}$.

B. FDI Attack Representation

FDI attacks inject faulty data into connected vehicles' communication networks, potentially disrupting system performance and leading to vehicle collisions. To introduce the FDI attack effect into the dynamic model of the CACC system, we define it as

$$f_i\left(u_{i-1}(t)\right) \triangleq u_{i-1}(t) + \beta_{i-1}(t) \quad \text{Eq. (3)}$$

where $f_i \in \mathbb{R}$ is the attack function and $\beta_{i-1}(t) \in \mathbb{R}$ is the unknown, continuous, and time-varying FDI attack added to the leader control signal, which is transmitted to the following vehicle through the communication channel.

Assumption 1. *The FDI attack is assumed to be bounded and differentiable such that* $\| \beta_{i-1}(t) \| \leq \bar{\beta} \; \forall t \geq t_0$, *where* $\bar{\beta}$ *is a known positive constant value* [4].

III. Control, Observer, and Attack Estimation Design

In this section, we examine the design of the controller, observer, and attack estimator, along with the error and auxiliary signals. The signals that are defined and computed here will be utilized in Appendix A to support the stability analysis.

A. Control Design

The initial goal of this article is to design a secure controller to maintain a safe distance between leader and follower vehicles during FDI attacks. The control signal is designed based on the Lyapunov stability analysis as

$$u_i(t) \triangleq -\frac{b_{i-1}}{c_i} v_{i-1}(t) + \frac{b_i}{c_i} v_i(t) + \frac{c_{i-1}}{c_i} \bar{u}_{i-1}(t) - \frac{c_{i-1}}{c_i} \hat{\beta}_{i-1}(t) - \frac{1}{c_i} \ddot{x}_{d_i}(t) + \frac{1}{c_i} (\alpha_i + k_i) r_i(t) + \frac{1}{c_i} \left(1 - \alpha_i^2\right) e_i(t) \quad \text{Eq. (4)}$$

where $k_i \in \mathbb{R}_{>0}$ is a gain specified for the controller that will be further optimized and $e_i : [t_0, \infty) \to \mathbb{R}$ is the tracking error between the leader and follower defined as

$$e_i(t) \triangleq x_{i-1}(t) - x_i(t) - L_i - x_{d_i}(t) \quad \text{Eq. (5)}$$

where $L_i \in \mathbb{R}$ is the length of vehicle i and $x_{d_i} : [t_0, \infty) \to \mathbb{R}$ is the desired distance between leader and follower defined as

$$x_{d_i}(t) \triangleq h_i v_i(t) + x_0 \quad \text{Eq. (6)}$$

where $h_i \in \mathbb{R}$ is the time headway and $x_0 \in \mathbb{R}$ is the standstill safe distance between the vehicles. Additionally, the compromised control signal is expressed as $\bar{u}_{i-1}(t) \triangleq u_{i-1}(t) + \beta_{i-1}$.

Furthermore, the auxiliary error signal $r_i \in \mathbb{R}$ is defined as

$$r_i(t) \triangleq \dot{e}_i(t) + \alpha_i e_i(t) \quad \text{Eq. (7)}$$

such that $\alpha_i \in \mathbb{R}_{>0}$, is a user-specified gain. We consider $\hat{\beta}_{i-1} \in \mathbb{R}$ to be the estimation of FDI attack and design it in the next subsection such that it remains bounded.

Assumption 2. *The desired distance, its first, and second derivatives are assumed to be bounded by positive known constants,* $x_{d_i}, \dot{x}_{d_i}, \ddot{x}_{d_i} \in \mathcal{L}_\infty$ [36].

B. Observer Design

Since the CACC is under FDI attack, the second objective of this article is to design an observer. Based on the Lyapunov stability analysis, we define the observer as

$$\ddot{\hat{x}}_{i-1}(t) = -b_{i-1} v_{i-1} + c_{i-1} \bar{u}_{i-1}(t) - c_{i-1} \hat{\beta}_{i-1}(t) + (l_i + \alpha_{i-1}) \tilde{r}_{i-1}(t) + \left(1 - \alpha_{i-1}^2\right) \tilde{x}_{i-1}(t) \quad \text{Eq. (8)}$$

such that l_i denotes the observer gain and $\alpha_{i-1} \in \mathbb{R}_{>0}$ is a user-defined gain that will be tuned further. Additionally,

$\hat{x}_{i-1} \in \mathbb{R}$ expresses the estimated position of the lead vehicle.

To measure the accuracy of the observer, a state estimation error $\tilde{x}_{i-1} : \left[t_0, \infty\right) \to \mathbb{R}$, is described as

$$\tilde{x}_{i-1}\left(t\right) \triangleq x_{i-1}\left(t\right) - \hat{x}_{i-1}\left(t\right) \quad \text{Eq. (9)}$$

An estimation of the auxiliary error signal $\tilde{r}_{i-1} : \left[t_0, \infty\right) \to \mathbb{R}$ can be defined as

$$\tilde{r}_{i-1}\left(t\right) \triangleq \dot{\tilde{x}}_{i-1}\left(t\right) + \alpha_{i-1}\tilde{x}_{i-1}\left(t\right) \quad \text{Eq. (10)}$$

We designed the observer to estimate the leader's control signal. To evaluate the accuracy of this estimation, an estimation error signal $\tilde{u}_{i-1} : \left[t_0, \infty\right) \to \mathbb{R}$, is defined as

$$\tilde{u}_{i-1}\left(t\right) \triangleq u_{i-1}\left(t\right) - \hat{u}_{i-1}\left(t\right) \quad \text{Eq. (11)}$$

We also define $\hat{u}_{i-1}\left(t\right) \triangleq \overline{u}_{i-1}\left(t\right) - \hat{\beta}_{i-1}\left(t\right)$ to generate

$$\tilde{u}_{i-1}\left(t\right) = u_{i-1}\left(t\right) - \overline{u}_{i-1}\left(t\right) + \hat{\beta}_{i-1}\left(t\right) \quad \text{Eq. (12)}$$

C. FDI Attack Estimation Design

Since the injected false data in the communication channel is unknown and has a nonlinear, unpredictable structure, a neural network can be used to estimate it. Following this subsection, we provide an FDI attack estimation using an adaptive neural network (ANN) approach. A simple NN with one hidden layer is sufficient for identifying and modeling FDI attacks in the CAV system. The FDI attack, denoted as β_{i-1}, takes place over a non-compact domain. Therefore, a nonlinear mapping, such as $M_{\beta_{i-1}} : [t_0, \infty) \to \zeta$ is necessary to transform time into a compact spatial domain, expressed as

$$M_{\beta_{i-1}} \triangleq \frac{c_{\beta_{i-1}}\left(t - t_0\right)}{c_{\beta_{i-1}}\left(t - t_0\right) + 1}, \quad \zeta \in \left[0,1\right],\ t \in \left[t_0, \infty\right) \quad \text{Eq. (13)}$$

where $c_{\beta_{i-1}} \in \mathbb{R}_{>0}$ is a gain that is defined by the user, as mentioned in [37]. As a result, $\beta_{i-1}(t)$, is transformed into the compact domain ζ as

$$\beta_{i-1}\left(t\right) = \beta_{i-1}\left(M^{-1}_{\beta_{i-1}}\left(\zeta\right)\right) \triangleq \beta_{M_{\beta_{i-1}}}\left(\zeta\right) \quad \text{Eq. (14)}$$

and $\beta_{M\beta_{i-1}}(\zeta)$ can be modeled using a three-layer NN as

$$\beta_{M_{\beta_{i-1}}}\left(\zeta\right) \triangleq W_i^T \sigma_i\left(V_i^T I_i\right) + \epsilon_i \quad \text{Eq. (15)}$$

where $I_i \in \mathbb{R}^{2\times 1}$ represents the input to the NN, while the $W_i \in \mathbb{R}^{(n_n+1)\times 1}$ and $V_i \in \mathbb{R}^{2\times n_n}$ denote the bounded constant ideal weights. The parameter n_n rcorresponds to the number of neurons in the hidden layer. Furthermore, $\sigma_i(\cdot) \in \mathbb{R}^{(n_n+1)\times 1}$ represents the activation function vector, and ϵ_i accounts for the functional reconstruction error.

Considering (14), the NN-based estimation of FDI attack can be described as

$$\hat{\beta}_{i-1}\left(t\right) \triangleq \hat{W}_i^T \sigma_i\left(\hat{V}_i^T I_i\right) \quad \text{Eq. (16)}$$

where $\hat{W}_i \in \mathbb{R}^{(n_n+1)\times 1}$ and $\hat{V}_i \in \mathbb{R}^{2\times n_n}$ represent the estimated ideal weights and I_i is defined as

$$I_i \triangleq \left[1 \ \ \hat{\beta}_{i-1}^T\right]^T \quad \text{Eq. (17)}$$

As a NN-based FDI attack estimator, we further propose an update law for the weight matrices. A continuously differential projection operator, proj(·), is applied as shown in [37] to avoid zero estimation for the $\hat{W}_i$ and $\hat{V}_i$ update laws as

$$\dot{\hat{W}}_i \triangleq \text{proj}\left(\Gamma_{1_i}\sigma_i\left(\hat{V}_i^T I_i\right)\psi_i^T\right) \quad \text{Eq. (18)}$$

and

$$\dot{\hat{V}}_i \triangleq \text{proj}\left(\Gamma_{2_i} I_i \psi_i^T \hat{W}_i^T \sigma_i'\left(\hat{V}_i^T I_i\right)\right) \quad \text{Eq. (19)}$$

where Γ_{1_i}, $\Gamma_{2_i} \in \mathbb{R}^{(n_n+1)\times(n_n+1)}$ are definite positive gain matrices and $\psi_i \triangleq -c_{i-1}\left(r_i + \tilde{r}_{i-1}\right)$. Furthermore, $\sigma_i'\left(\cdot\right)$ is the partial derivative of the $\sigma_i(\cdot)$ function that will be further calculated in the stability analysis proof.

D. Stability Analysis

To simplify future analysis, the parameter t, which represents time, is removed from the equations. Let us define z_i as

$$z_i \triangleq \left[e_i^T \ \ r_i^T \ \ \tilde{x}_{i-1}^T \ \ \tilde{r}_{i-1}^T\right]^T \quad \text{Eq. (20)}$$

and define $H_i : [t_0, \infty) \to \mathbb{R}_{\geq 0}$ as

$$H_i\left(t\right) \triangleq \frac{1}{2}\text{tr}\left(\tilde{W}_i^T \Gamma_{1_i}^{-1} \tilde{W}_i\right) + \frac{1}{2} tr\left(\tilde{V}_i^T \Gamma_{2_i}^{-1} \tilde{V}_i\right) \quad \text{Eq. (21)}$$

where $\tilde{W}_i = W_i - \hat{W}_i$ and $\tilde{V}_i = V_i - \hat{V}_i$ are the estimation errors for the ideal weight matrices and tr(·) is the trace operator. We also consider the following sufficient conditions as

$$\alpha_i > 0, \quad \alpha_{i-1} > 0, \quad k_i > \frac{c_{i-1}}{2\varepsilon_{1_i}}, \quad l_i > \frac{c_{i-1}}{2\varepsilon_{2_i}} \quad \text{Eq. (22)}$$

to ensure the validity of the theorem we further discuss in this section. The $\varepsilon_{1_i} \in \mathbb{R}_{>0}$ and $\varepsilon_{2_i} \in \mathbb{R}_{>0}$ are positive known constants. We also define $\lambda_i \in$ R as

$$\lambda_i \triangleq \frac{\varepsilon_{3_i}}{2} N_{i,max}^2 \quad \text{Eq. (23)}$$

where $\varepsilon_{3_i} \triangleq c_{i-1}(\varepsilon_{1_i} + \varepsilon_{2_i})$ is a positive known constant. Furthermore, let $\eta_{i_1}, \eta_{i_2} \in \mathbb{R}_{>0}$ be positive constants such that $\eta_{i_1} < \eta_{i_2}$.

Theorem 1. *The given open-loop error system in (5–7), controller given in (4), state estimator in (8), and FDI attack estimator in (16) ensure semi-globally uniformly ultimately bounded tracking such that*

$$\lim_{t\to\infty} \sup \| z_i(t) \| \leq \sqrt{\frac{1}{\eta_{i_1}} \left(H_{i,max} + \frac{\eta_{i_2} \lambda_i}{\chi_i} \right)} \quad \text{Eq. (24)}$$

A detailed proof of the presented theorem is provided in Appendix A.

IV. Reinforcement Learning

A. Problem Statement

In this subsection, we present an adaptive feedback control technique based on supervisory RL. The RL agent, shown in Figure 2, generates an update law for user-defined gains in the environment, which includes the controller, the observer, and the FDI attack estimator. The traditional regulatory feedback controller hierarchy operates at the supervisory level. Throughout the training phase, the RL agent learns to predict the optimal parameters, k_i, α_i, α_{i-1}, and l_i, for the controller and observer of a CACC system. Additionally, the RL-based tuning method finds the NN weight matrices Γ_{1_i} and Γ_{2_i} for the FDI attack estimator. For RL, choosing an appropriate reward function is crucial since it directly affects the behavior of the agent and how well the learning process works. We have defined the CACC reward function, $R_i \in \mathbb{R}$, as

$$\mathcal{R}_i \triangleq -e_i^2 - \tilde{\beta}_{i-1}^2 \quad \text{Eq. (25)}$$

where $\tilde{\beta}_{i-1} \in \mathbb{R}$ is the error of FDI attack estimation, defined as

$$\tilde{\beta}_{i-1} \triangleq \beta_{i-1} - \hat{\beta}_{i-1} \quad \text{Eq. (26)}$$

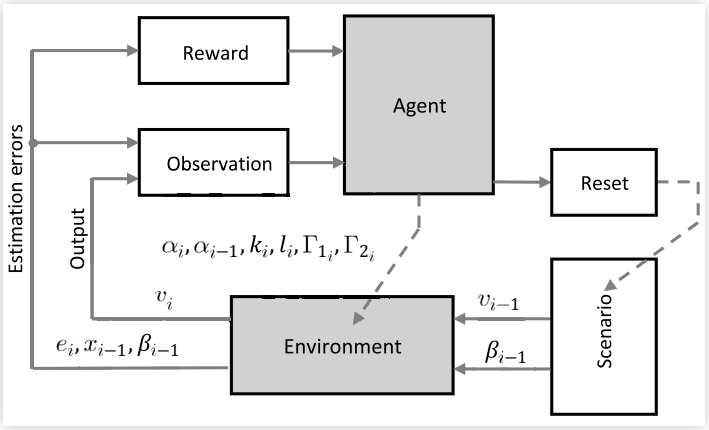
Furthermore, the RL agent receives observations from the environment in the form of a vector of states error, e_i, $\tilde{\beta}_{i-1}$, $\tilde{x}_{i-1}$, and system output v_i. In order to train and assess an RL agent, the reset function is essential. The primary purpose of reset is to reinitialize the environment to a starting state, which allows for multiple episodes of interaction. Here, the reset function is modifying the speed and FDI attack amplitude randomly to provide sufficient variation needed for training.

RL agent can incorporate a reset function that penalizes actions leading to unsafe states. The reset function heavily penalizes situations such as unsafe following distances, which could lead to collisions, as well as unreasonable estimation errors in detecting FDI attacks. By doing so, the algorithm ensures that the agent learns to avoid risky behaviors that could compromise safety or system performance. These penalties serve as safety constraints, guiding the RL agent toward safer and more reliable decision-making during training.

B. Background

Depending on system constraints, control objectives, and the nature of the environment, different RL designs can be tailored to specific types of control problems. Our target system has a continuous action space, and we have a model-free RL, since the agent does not have access

FIGURE 2 The schematic of RL agent for parameter tuning.



to the system's model. Additionally, since the proposed RL agent can learn from experiences generated by a different policy or even a random exploration strategy, it is considered as an off-policy RL.

Based on these information, deep deterministic policy gradient (DDPG), twin delayed DDPG (TD3), soft actor-critic (SAC), and proximal policy optimization (PPO) are some of the most popular model-free DRL algorithms in continuous control. TD3 is preferred over other algorithms due to its enhanced stability and efficiency in continuous control tasks. Compared to DDPG, TD3 addresses overestimation bias and unstable learning by introducing twin critics, delayed policy updates, and target smoothing. Furthermore, TD3 is simpler and more computationally efficient than SAC, especially when extensive exploration or stochastic policies are not needed. TD3's deterministic approach excels in scenarios requiring precise control, like robotics or autonomous vehicles. Furthermore, TD3's off-policy nature makes it more sample-efficient than PPO, making it suited for problems with continuous action spaces [38].

The optimization problem of RL is built around the framework of Markov decision processes (MDP) that provide a tuple $\langle \mathcal{S}_i, \mathcal{A}_i, \mathcal{P}_i, \mathcal{R}_i \rangle$ formal way to describe an environment in RL and allow for the analysis and development of optimal control strategies. The $\mathcal{S}_i$ is the set of states representing the possible configurations or conditions of the environment, $\mathcal{A}_i$ is the set of control actions available to the agent, and $\mathcal{P}_i$ is the state transition probability matrix. The agent is interacting with the environment to learn behaviors that maximize rewards. At each time step t, with a given state $s \in \mathcal{S}_i$, the agent selects actions $a \in \mathcal{A}_i$ with respect to its policy $\pi_\phi : \mathcal{S}_i \rightarrow \mathcal{A}_i$, receiving a reward $r \in \mathrm{R}_i$ and the new state of the environment s'.

C. TD3 Algorithm

Algorithm 1 offers a comprehensive overview of TD3 optimization. Two separate critic networks, or Q-networks, are used by TD3 to independently estimate the action value function: Q_{θ_1} and Q_{θ_2}. By estimating the action value as the smallest value between the two critics, the twin Q-networks reduce overestimation bias and help to limit positive bias in the policy update. The actor network proposes actions based on the current state, $\pi_\phi(s, a)$. It is often a feedforward neural network with multiple layers (e.g., input, hidden, and output). The input layer receives the state s, the hidden layers perform nonlinear transformations on the supplied data, and the output layer creates the action a in the continuous action space. The actor network increases the estimated Q-value of the critic network to maximize the expected return from the current state. A soft update mechanism updates both the target Q-networks and the target policy network, gradually tracking their parameters to the learned networks. To stabilize training, we used delayed versions of the actor and critic networks in TD3.

ALGORITHM 1 Optimization Workflow of TD3.

Initialize critic networks Q_{θ_1}, Q_{θ_2}, and actor networks π_ϕ with random parameters θ_1, θ_2, and ϕ.
Initialize target networks $\theta_1' \leftarrow \theta_1$, $\theta_2' \leftarrow \theta_2$, $\phi' \leftarrow \phi$
Initialize reply buffer $\mathcal{D}$
Initialize hyperparameters $\gamma, N, d, \delta, g, \tau$
for $t = 1$ *to* T **do**
  Select action $a_t \sim \pi(s) + g$ with exploration noise $g \sim \mathcal{N}(0, \delta)$.
  Store s_t, a_t, r_t, s_{t+1} in $\mathcal{D}$.
  Sample a mini-batch of N transitions (s_j, a_j, r_j, s_{j+1}) from $\mathcal{D}$
  $\tilde{a} \leftarrow \pi_{\phi'}(s) + g, \quad g \sim \text{clip}(\mathcal{N}(0, \tilde{\delta}, -q, q)$
  $y \leftarrow r + \gamma \min_{j=1,2} Q_{\theta_j'}(s', \tilde{a})$
  Update critics $\theta_j \leftarrow min_{\theta_j} \frac{1}{N} \sum (y - Q_{\theta_j}(s, a))^2$
  if t *mod* d **then**
    Update ϕ by the deterministic policy gradient:
    $\nabla_\phi J(\phi) = \frac{1}{N} \sum \nabla_a Q_{\theta_1}(s, a)|_{a=\pi_\phi(s)} \nabla_\phi \pi_\phi(s)$
    Update target networks:
    $\theta_j' \leftarrow \tau\theta_j + (1 - \tau)\theta_j'$
    $\phi' \leftarrow \tau\phi + (1 - \tau)\phi'$
  end
end

The actor network and critic networks are initialized with random parameters. The target networks ϕ', θ_1', and θ_2' are set to match the initial networks. A replay buffer is initialized to store experiences, and hyperparameters such as batch size, discount factor γ, delay coefficient, exploration noise, and policy noise. The agent chooses an action based on the current state with additional exploration noise at each time step, carries it out, and then watches the subsequent state and reward. The replay buffer holds the transition. During the training process, each critic network is updated using the Bellman equation. If s' is the next state following action a taken in state s, with reward r received, the target value for each Q-network is computed as

$$y_j = r + \gamma \min_{j=1,2} Q_{\theta_j'}\left(s', \pi_{\phi'}\left(s'\right) + g\right)$$
$$g \sim \text{clip}\left(\mathcal{N}\left(0, \delta\right), -q, q\right) \qquad \text{Eq. (27)}$$

where π_ϕ is the optimal policy, $Q_\theta(s, a)$ is the differentiable function approximator with parameter θ, g is Gaussian noise with 0 mean, and δ variance clipped to the range of $-q$ and q to keep the target in a small range, so the value estimate learned is with respect to a noisy policy. In deep Q-learning, the network is updated with a secondary frozen target network $Q_{\theta'}(s, a)$ to maintain a fixed objective y over multiple updates. The agent updates the actor and critic networks using mini-batches sampled from the replay buffer. Target networks are updated to provide stable training.

V. Results

The simulation results are shown here for the CACC system that is discussed in Section III. The CACC model parameters are described in Table 1. For simplicity, the leader and follower vehicles have been considered homogeneous, with similar constants generated from a real vehicle in [19]. We evaluated the effectiveness of the implemented controller, observer, and FDI attack estimator in two distinct scenarios, each involving a different form of FDI attack.

The CACC system model, as well as the tuning approach, were designed in MATLAB/Simulink using the RL toolbox with the hyperparameters defined in Table 2. The following subsections contain a thorough description of the simulation result.

A. Scenario 1

In the initial scenario, we aim to create a drive cycle and implement two different types of FDI attacks to assess the effectiveness of the designed controller and attack estimator. The amplitude and frequency of the FDI attacks, along with the speed profile amplitude, are random values with a specific time step duration. Thus, we have developed S_{11} and S_{12} to replicate the initial drive cycle with two different FDI attack shapes. An RL-based approach tunes the developed controller, observer, and attack estimator gains, resulting in an optimized CACC system. Tables 3 and 4 display the user-defined gains selected by trial and error, as well as by applying the tuning approach.

Figure 3 shows how well the CACC algorithm worked when it was attacked in scenario S_{11}. The CACC algorithm successfully tracked the reference speed profile, identified the injected FDI attack, and maintained a safe distance between the leader and follower vehicles. The baseline controller lacks FDI attack estimation results, leading to several collisions during the simulation. The developed controller provides an attack estimation for the CACC system; however, without proper tuning, this estimation result is not precise. Using the RL tuning, we are able to follow the desired speed profile flawlessly, ensuring safe CACC algorithm operation.

In order to assess the effectiveness of the controller under a more complex FDI attack, we have created scenario S_{12}. There is a different attack signal in Figure 4, which shows how well the CACC worked with the baseline, developed, and optimized solutions. The optimized solution with RL-tuned gains is able to accurately estimate the FDI attack, follow the speed profile, and maintain a safe distance.

TABLE 1 Model parameters.

Parameter	Value	Description
b_i	0.1413	Model gain 1
c_i	6.6870	Model gain 2

TABLE 2 RL tuning parameters.

Parameter	Value	Description
N	128	Mini-batch size
γ	0.3162	Discount factor
q	0.5	Clip value
d	25	Number of iteration
τ	0.01	Soft update rate
δ	0.1	Noise variance

TABLE 3 Developed and optimized controller and observer gains.

	k_i	α_i	α_{i-1}	l_i
Developed controller	20	0.1	0.1	20
Optimized controller	25.5124	1.2711	9.3562	15.2672

B. Scenario 2

Given that there are certain situations where an FDI attack can significantly degrade system performance, the second scenario, named S_{21} and S_{22}, is designed to create the worst-case scenario by carefully planning the FDI attack and speed profile. In this scenario, when the leading vehicle applies the brakes, the FDI attack will intensify. As depicted in Figures 5 and 6, the follower vehicle is capable of tracking the leading vehicle within the initial 20 s of the maneuver. Moreover, the outcome of estimating FDI attacks can validate the effectiveness of the developed algorithm in both detection and mitigation. The visual representation showcases the NN's successful attack detection, enhanced by the RL tuning approach's refinement.

C. Discussion

The root mean square (RMS) error of the FDI attack estimation and tracking performance for the baseline, developed resilient controller, and optimized controller have been shown in Tables 5 and 6. The optimized controller significantly reduces the RMS error in each scenario for both FDI attack estimation and following distance maintenance.

TABLE 4 Developed and optimized FDI attack estimator gains.

	Γ_{1_i}	Γ_{2_i}
Developed controller	$\begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$	$\begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$
Optimized controller	$\begin{bmatrix} 3.3812 & 0 \\ 0 & 3.3812 \end{bmatrix}$	$\begin{bmatrix} 2.2145 & 0 \\ 0 & 2.2145 \end{bmatrix}$

FIGURE 3 Scenario S_{11} simulation results: (a) leader and follower vehicle speed, (b) FDI attack estimation, and (c) following distance between the vehicles.

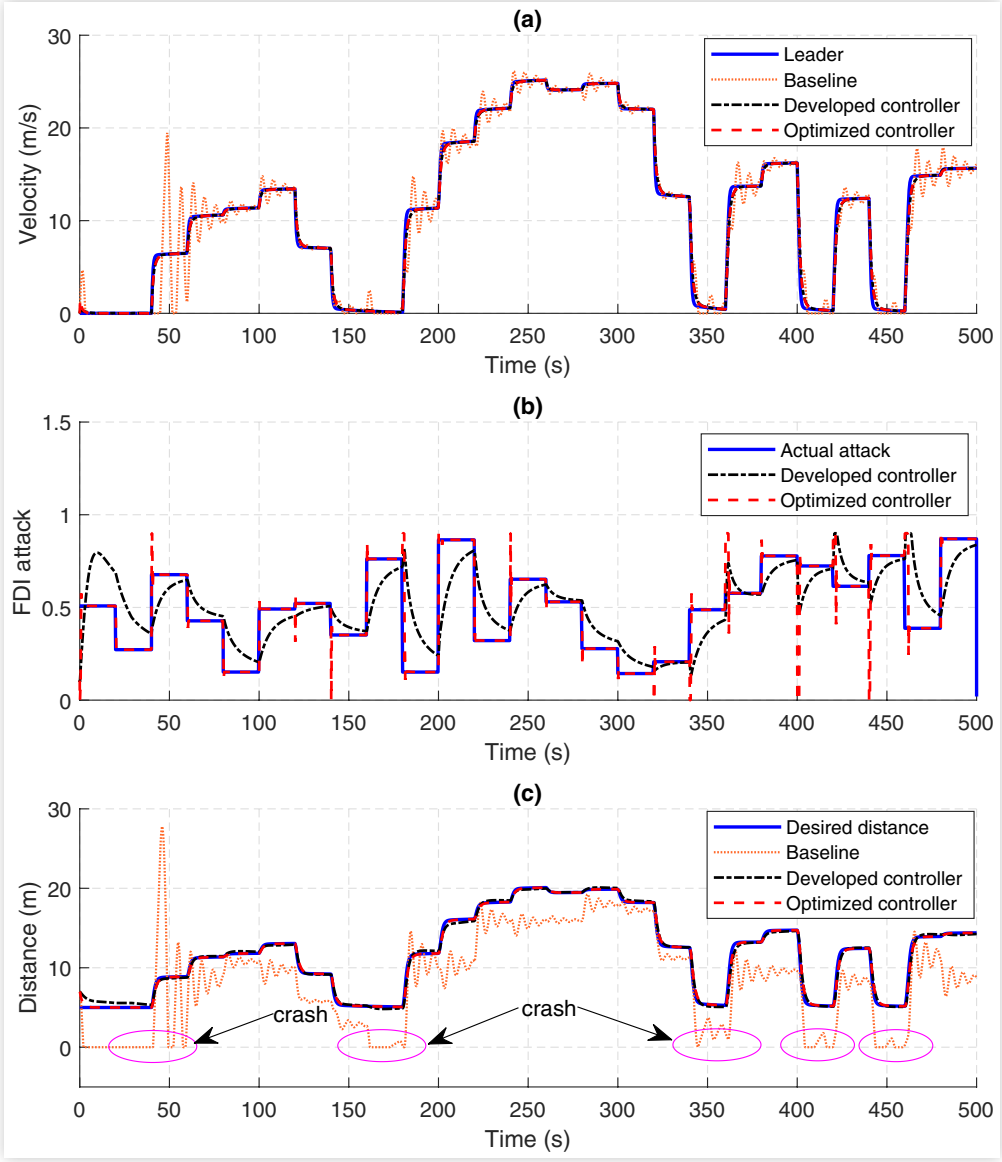


FIGURE 4 Scenario S_{12} simulation results: (a) leader and follower vehicle speed, (b) FDI attack estimation, and (c) following distance between the vehicles.

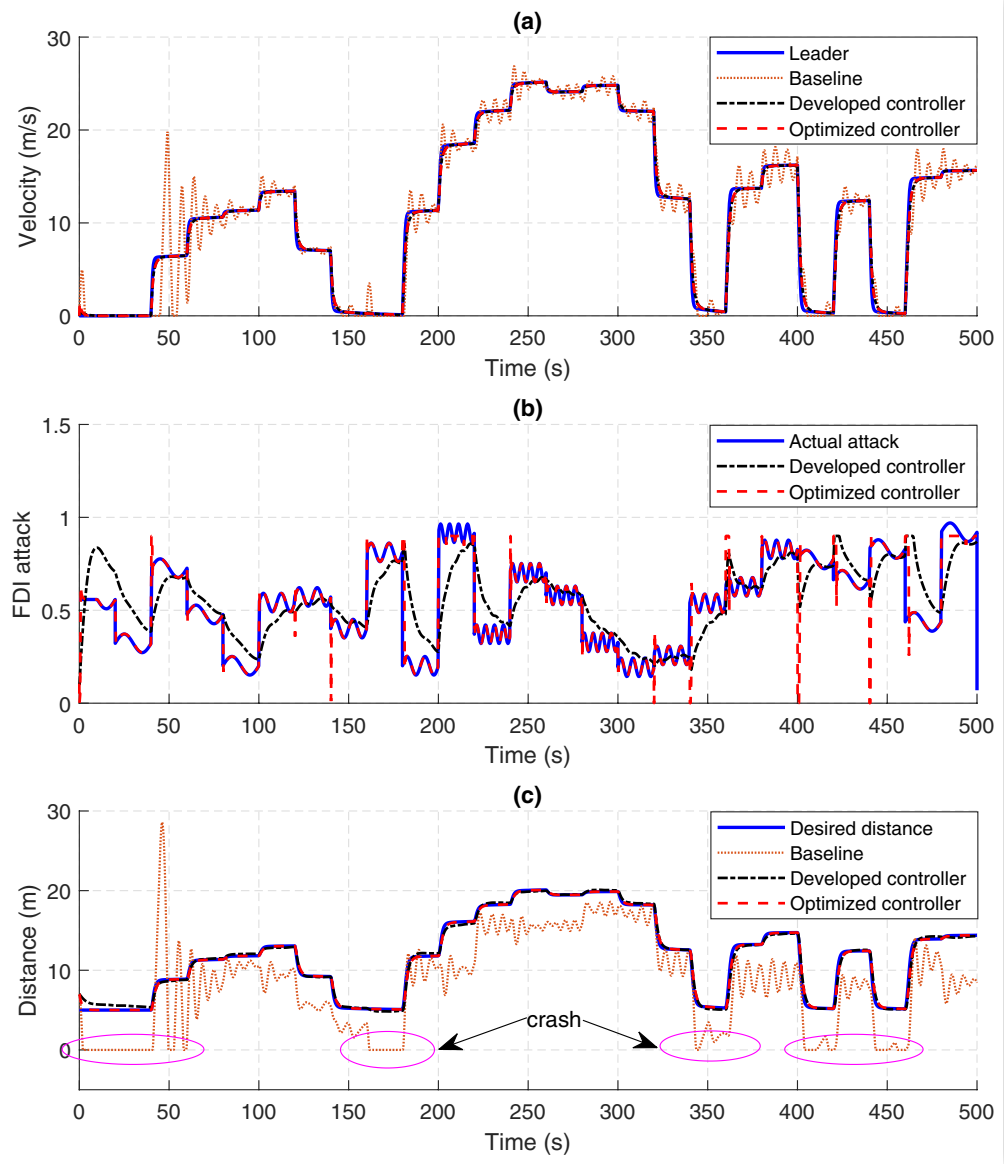


FIGURE 5 Scenario S_{21} simulation results: (a) leader and follower vehicle speed, (b) FDI attack estimation, and (c) following distance between the vehicles.

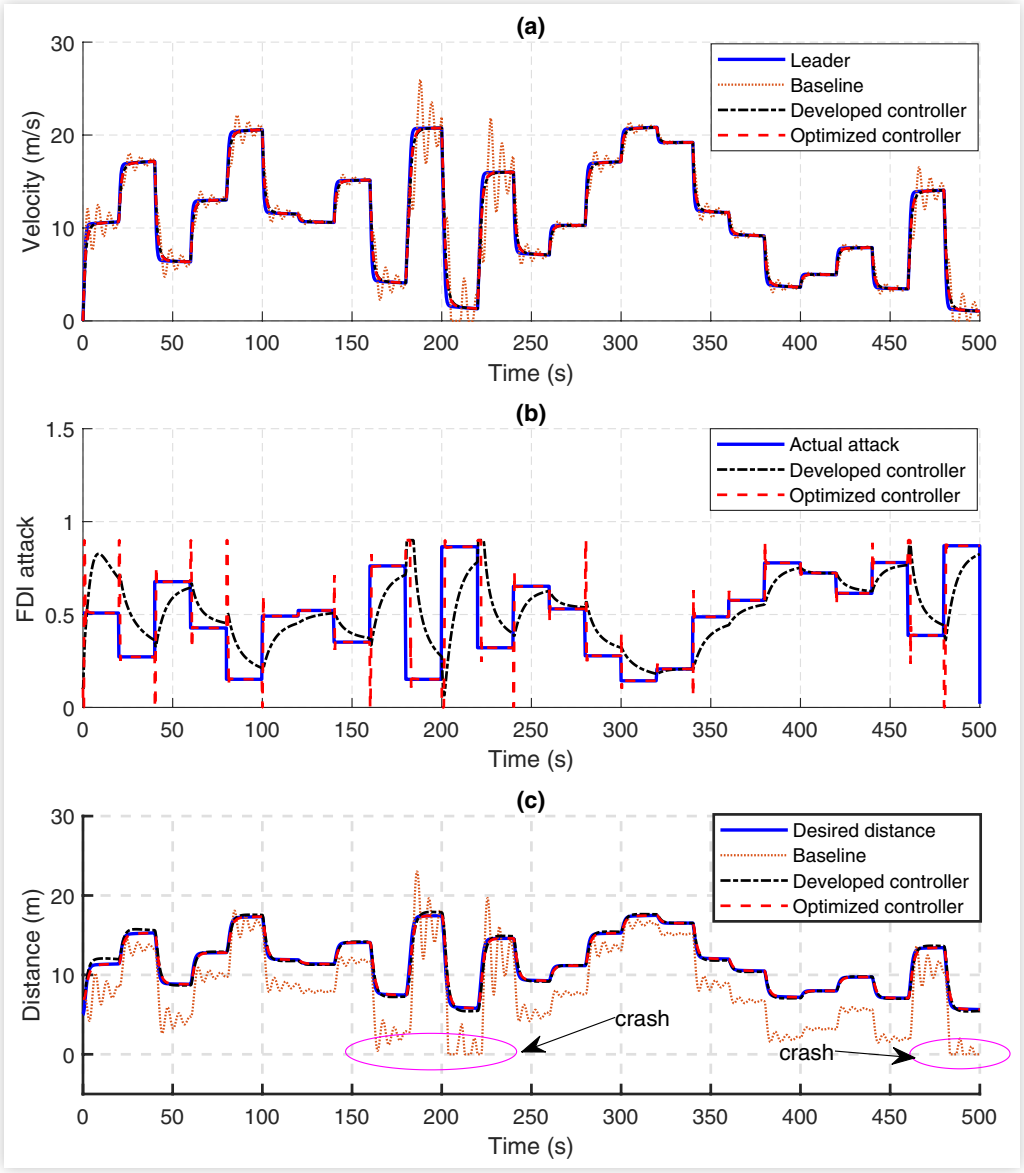


FIGURE 6 Scenario S_{22} simulation results: (a) leader and follower vehicle speed, (b) FDI attack estimation, and (c) following distance between the vehicles.

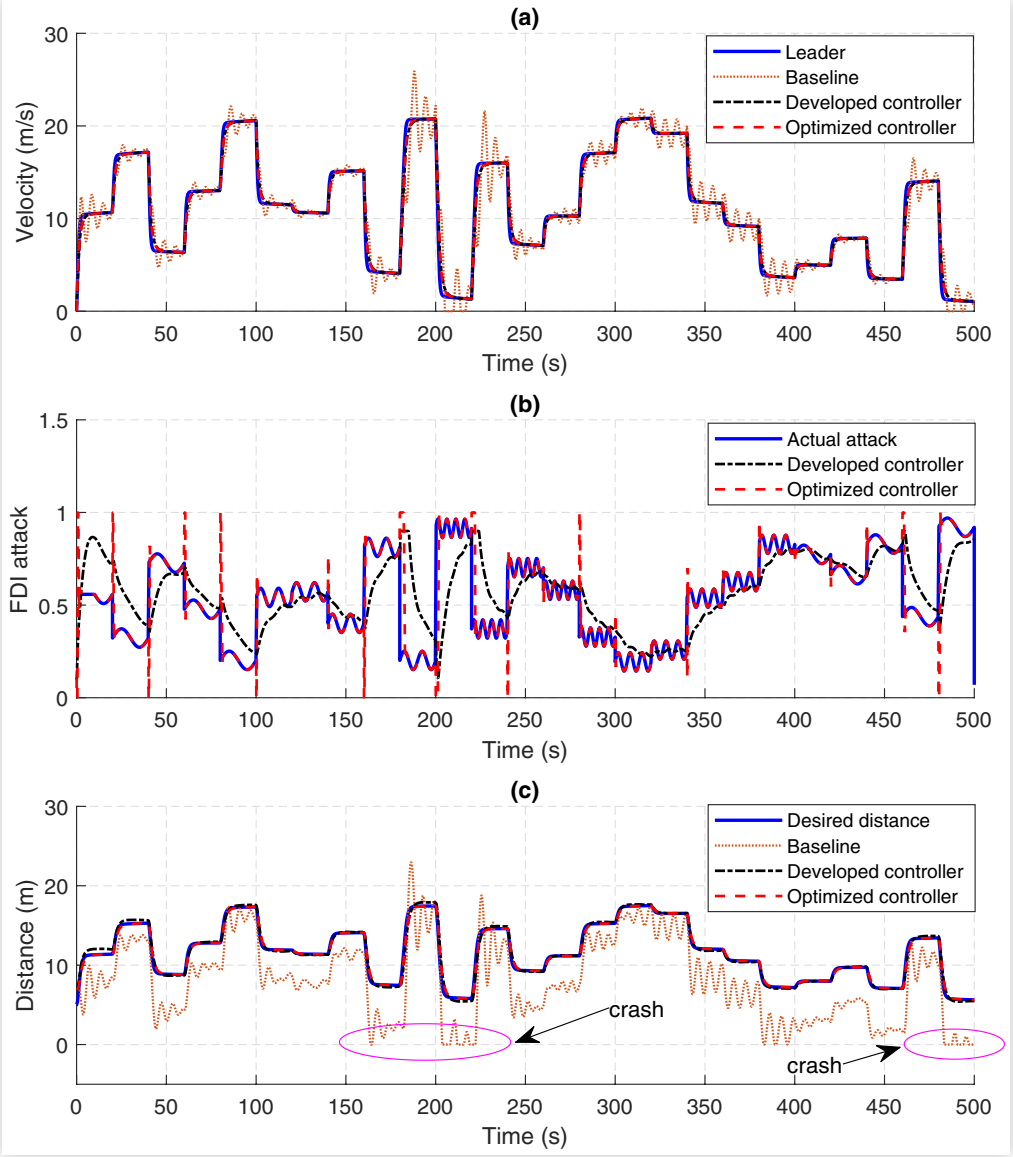


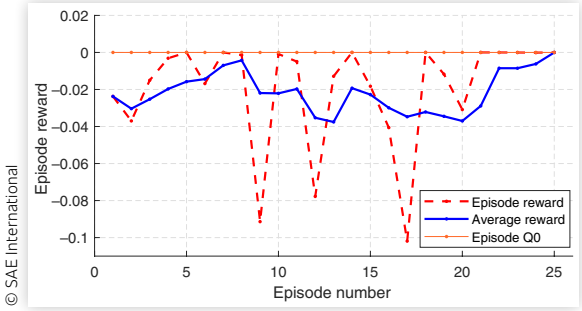
TABLE 5 RMS error of FDI attack estimation in each scenario.

	S_{11}	S_{12}	S_{21}	S_{22}
Developed controller	0.1576	0.1564	0.1883	0.1891
Optimized controller	0.0781	0.0776	0.1104	0.1151
PSO approach	0.2598	0.2545	0.3070	0.3003

TABLE 6 RMS error of following distance in each scenario.

	S_{11}	S_{12}	S_{21}	S_{22}
Baseline	4.2190	4.4637	3.7921	4.0948
Developed controller	0.2797	0.2754	0.2883	0.2837
Optimized controller	0.0647	0.0649	0.0765	0.0764
PSO approach	0.1372	0.1398	0.4007	0.4076

FIGURE 7 Training progress of TD3 algorithm.

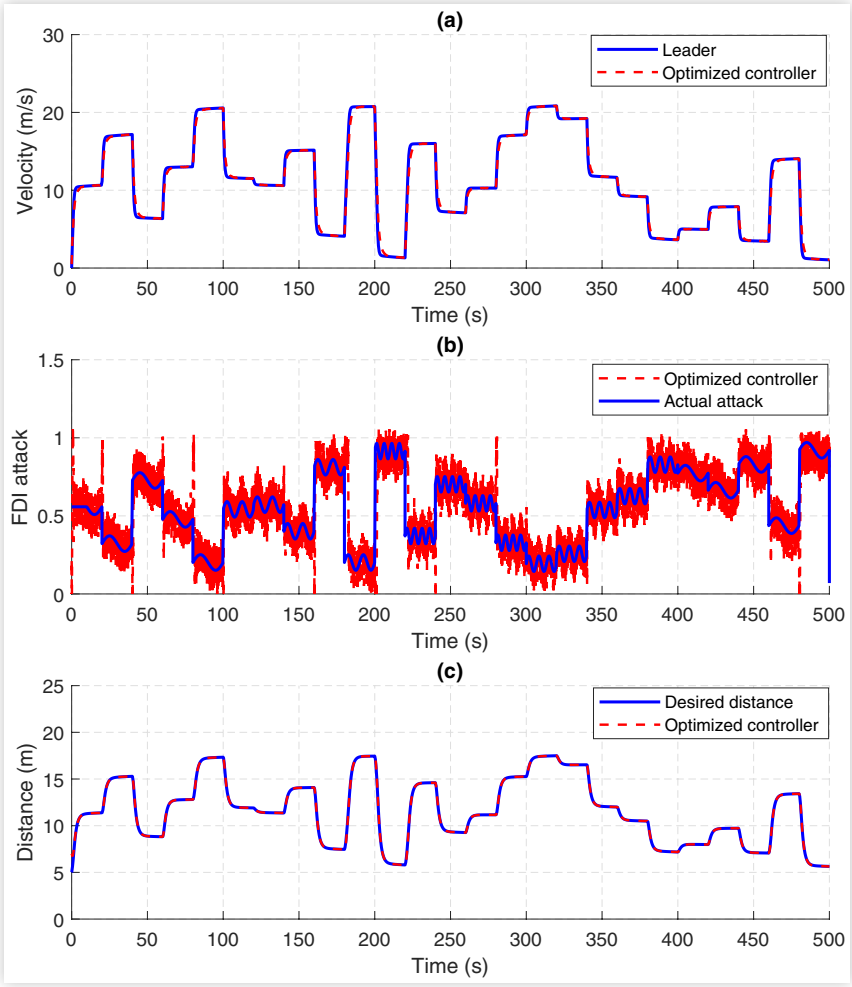


To discuss the effectiveness of the proposed parameter optimization approach we have compared it with the one introduced in [21]. We have applied the controller and observer parameters generated by the PSO approach into our case. Since the FDI attack estimator in [21] has not been considered in the optimization, we have used the parameters generated by RL agent for it. The results in Tables 5 and 6 demonstrate that the RL-based optimization significantly outperforms the PSO approach in tracking the FDI attack and maintaining a safe following distance, achieving an average RMS error improvement of 66.84%.

The RL training progress is shown in Figure 7. As illustrated, training was completed within 25 episodes, which is notably fewer than the 74 epochs required for convergence in the PSO approach presented in [21]. The episode reward, the average reward, and the episode Q0 converge to each other at the end of training and this convergence is a sign of successful training. Episode Q0 is determined by conducting inferences on the critic at the start of each episode. This value serves as an indicator of the critic's training effectiveness. Ideally, if the critic were perfect and could precisely predict the expected long-term reward based on the current observation at the episode's onset, Q0 would align closely with the actual total reward obtained during that episode.

The performance of the optimized solution was evaluated in the presence of both disturbance and measurement noise, as shown in Figure 8. To simulate disturbance, a sinusoidal signal with an amplitude of 1 and a frequency matching

FIGURE 8 Scenario S_{22e} simulation results under extreme condition: (a) leader and follower vehicle speed, (b) FDI attack estimation, and (c) following distance between the vehicles.



the velocity update rate was added to the follower vehicle's velocity. Additionally, measurement noise was introduced to the estimated FDI attack as Gaussian white noise with a variance of 0.01 and band limit of 0.2. The CACC was tested under extreme conditions in Scenario S_{22}, which represents the worst-case scenario. We designated this scenario as S_{22_e}. As shown in this figure, the CACC under FDI attack maintained safety with zero collision risk, even under extreme conditions involving disturbances and measurement noise. The FDI attack estimator also performed effectively, estimating the noise combined with the actual attack.

VI. Conclusion

In this article, we present a secure controller and observer, as well as an FDI attack estimator for a CACC system. We have used MATLAB/Simulink to build the system and control models that allow us to make detailed evaluations of the CACC system's performance under several realistic scenarios. The purpose of these scenarios is to accurately replicate an actual driving cycle and a worst-case FDI attack situation in order to thoroughly evaluate the effectiveness of the CACC system and the resilience of its controller. In addition, a stability analysis was performed using Lyapunov stability theory to guarantee the system's stability. The system's user-defined parameters in each scenario of CACC performance are a time-consuming trial-and-error process. To solve this issue, we developed a TD3 RL tuning approach to optimize the performance of a CACC system in the presence of FDI attacks.

Our findings demonstrate that using a RL-based tuner significantly reduces the RMS error in the distance between the leader and follower vehicles during CACC maneuvers. This demonstrates that the optimized tuning method enhances safety and security significantly. Additionally, it improves attack estimation accuracy, highlighting the method's robustness in mitigating FDI attacks on the CACC system.

Acknowledgements

Partial support for this research was provided by the National Science Foundation under Grant No. ECCS-EPCN-2241718. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the sponsoring agency.

Contact Information

Farahnaz Javidi-Niroumand, corresponding author
farahnaz.javidi@gmail.com

Dr. Arman Sargolzaei
a.sargolzaei@gmail.com

References

1. Faghihian, H. and Sargolzaei, A., "Energy Efficiency of Connected Autonomous Vehicles: A Review," *Electronics* 12, no. 19 (2023): 4086.
2. Othman, B., De Nunzio, G., Sciarretta, A., Di Domenico, D. et al., "Connectivity and Automation as Enablers for Energy-Efficient Driving and Road Traffic Management," in Lackner, M., Sajjadi, B., and Chen, W.Y. (eds), *Handbook of Climate Change Mitigation and Adaptation* (Cham, Switzerland: Springer, 2022), 2337-2376.
3. Niroumand, F.J., Bonab, P.A., and Sargolzaei, A., "Security of Connected and Autonomous Vehicles: A Review of Attacks and Mitigation Strategies," *SoutheastCon* 2024 (2024): 1197-1204.
4. Sargolzaei, A., Allen, B.C., Crane, C.D., and Dixon, W.E., "Lyapunov-Based Control of a Nonlinear Multiagent System with a Time-Varying Input Delay under False-Data-Injection Attacks," *IEEE Transactions on Industrial Informatics* 18, no. 4 (2021): 2693-2703.
5. Hu, L., Wang, Z., Han, Q.-L., and Liu, X., "State Estimation under False Data Injection Attacks: Security Analysis and System Protection," *Automatica* 87 (2018): 176-183.
6. Victorio, M., Sargolzaei, A., and Khalghani, M.R., "A Secure Control Design for Networked Control Systems with Linear Dynamics under a Time-Delay Switch Attack," *Electronics* 10, no. 3 (2021): 322.
7. Biron, Z.A., Dey, S., and Pisu, P., "Real-Time Detection and Estimation of Denial of Service Attack in Connected Vehicle Systems," *IEEE Transactions on Intelligent Transportation Systems* 19, no. 12 (2018): 3893-3902.
8. Han, J., Ju, Z., Chen, X., Yang, M. et al., "Secure Operations of Connected and Autonomous Vehicles," *IEEE Transactions on Intelligent Vehicles* 8 (2023): 4484-4497.
9. Lidström, K., Sjöberg, K., Holmberg, U., Andersson, J. et al., "A Modular CACC System Integration and Design," *IEEE Transactions on Intelligent Transportation Systems* 13, no. 3 (2012): 1050-1061.
10. Milanés, V., Shladover, S.E., Spring, J., Nowakowski, C. et al., "Cooperative Adaptive Cruise Control in Real Traffic Situations," *IEEE Transactions on Intelligent Transportation Systems* 15, no. 1 (2013): 296-305.
11. Trudgen, M. and Mohammadpour, J., "Robust Cooperative Adaptive Cruise Control Design for Connected Vehicles," in *Dynamic Systems and Control Conference*, Columbus, OH, 2015, Vol. 57243, V001T17A004, American Society of Mechanical Engineers.
12. Zhu, Y., Zhao, D., and Zhong, Z., "Adaptive Optimal Control of Heterogeneous CACC System with Uncertain Dynamics," *IEEE Transactions on Control Systems Technology* 27, no. 4 (2018): 1772-1779.

13. Sun, M., Al-Hashimi, A., Li, M., and Gerdes, R., "Impacts of Constrained Sensing and Communication Based Attacks on Vehicular Platoons," *IEEE Transactions on Vehicular Technology* 69, no. 5 (2020): 4773-4787.
14. Taylor, S.J., Ahmad, F., Nguyen, H.N., Shaikh, S.A. et al., "Safety, Stability and Environmental Impact of FDI Attacks on Vehicular Platoons," in *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, Budapest, Hungary, 2022, 1-6, IEEE.
15. Dutta, R.G., Hu, Y., Yu, F., Zhang, T. et al., "Design and Analysis of Secure Distributed Estimator for Vehicular Platooning in Adversarial Environment," *IEEE Transactions on Intelligent Transportation Systems* 23, no. 4 (2020): 3418-3429.
16. Gao, K., Cheng, X., Huang, H., Li, X. et al., "False Data Injection Attack Detection in a Platoon of CACC in RSU," in *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, Wuhan, China, 2022, 1324-1329, IEEE.
17. Yang, T., Murguia, C., Nesic, D., and Yuen, C., "Attack-Resilient Design for Connected and Automated Vehicles," arXiv preprint arXiv:2306.10925, 2023.
18. Biroon, R.A., Biron, Z.A., and Pisu, P., "False Data Injection Attack in a Platoon of CACC: Real-Time Detection and Isolation with a PDE Approach," *IEEE Transactions on Intelligent Transportation Systems* 23, no. 7 (2021): 8692-8703.
19. Ansari-Bonab, P., Holland, J.C., Cunningham-Rush, J., Noei, S. et al., "Secure Control Design for Cooperative Adaptive Cruise Control under False Data Injection Attack," *IEEE Transactions on Intelligent Transportation Systems* 25 (2024): 9723-9732.
20. Cunningham-Rush, J., Holland, J., Noei, S., and Sargolzaei, A., "Designing and Testing a Secure Cooperative Adaptive Cruise Control under False Data Injection Attack," in *2023 IEEE Conference on Dependable and Secure Computing (DSC)*, Tampa, FL, 2023, 1-8, IEEE.
21. Holland, J.C., Javidi-Niroumand, F., Ala'J, A., and Sargolzaei, A., "A Testing and Verification Approach to Tune Control Parameters of Cooperative Driving Automation under False Data Injection Attacks," *IEEE Access* 12 (2024): 19848-19859.
22. Su, J., Wu, J., Cheng, P., and Chen, J., "Autonomous Vehicle Control through the Dynamics and Controller Learning," *IEEE Transactions on Vehicular Technology* 67, no. 7 (2018): 5650-5657.
23. Naeem, M., Rizvi, S.T.H., and Coronato, A., "A Gentle Introduction to Reinforcement Learning and Its Application in Different Fields," *IEEE Access* 8 (2020): 209320-209344.
24. Lewis, F.L., Vrabie, D., and Vamvoudakis, K.G., "Reinforcement Learning and Feedback Control: Using Natural Decision Methods to Design Optimal Adaptive Controllers," *IEEE Control Systems Magazine* 32, no. 6 (2012): 76-105.
25. Pradhan, S.K. and Subudhi, B., "Real-Time Adaptive Control of a Flexible Manipulator Using Reinforcement Learning," *IEEE Transactions on Automation Science and Engineering* 9, no. 2 (2012): 237-249.
26. Wang, Z. and Hong, T., "Reinforcement Learning for Building Controls: The Opportunities and Challenges," *Applied Energy* 269 (2020): 115036.
27. Wei, C., Xiong, Y., Chen, Q., and Xu, D., "On Adaptive Attitude Tracking Control of Spacecraft: A Reinforcement Learning Based Gain Tuning Way with Guaranteed Performance," *Advances in Space Research* 71, no. 11 (2023): 4534-4548.
28. Shuprajhaa, T., Sujit, S.K., and Srinivasan, K., "Reinforcement Learning Based Adaptive PID Controller Design for Control of Linear/Nonlinear Unstable Processes," *Applied Soft Computing* 128 (2022): 109450.
29. Buşoniu, L., De Bruin, T., Tolić, D., Kober, J. et al., "Reinforcement Learning for Control: Performance, Stability, and Deep Approximators," *Annual Reviews in Control* 46 (2018): 8-28.
30. Tatyana, K. and Prakapovich, R., "Automatic Tuning of the Motion Control System of a Mobile Robot Along a Trajectory Based on the Reinforcement Learning Method," in Tuzikov, A.V., Belotserkovsky, A.M., and Lukashevich, M.M. (eds), *International Conference on Pattern Recognition and Information Processing* (Cham, Switzerland: Springer, 2021), 234-244.
31. Tufenkci, S., Alagoz, B.B., Kavuran, G., Yeroglu, C. et al., "A Theoretical Demonstration for Reinforcement Learning of PI Control Dynamics for Optimal Speed Control of DC Motors by Using Twin Delay Deep Deterministic Policy Gradient Algorithm," *Expert Systems with Applications* 213 (2023): 119192.
32. Chowdhury, M.A., Al-Wahaibi, S.S., and Lu, Q., "Entropy-Maximizing TD3-Based Reinforcement Learning for Adaptive PID Control of Dynamical Systems," *Computers & Chemical Engineering* 178 (2023): 108393.
33. Chen, X., Wang, R., Cui, Y., Jin, X. et al., "TD3 Tuned PID Controller for Autonomous Vehicle Platooning," SAE Technical Paper 2023-01-7108 (2023), doi:https://doi.org/10.4271/2023-01-7108.
34. Zhao, C., Gill, J.S., Pisu, P., and Comert, G., "Detection of False Data Injection Attack in Connected and Automated Vehicles via Cloud-Based Sandboxing," *IEEE Transactions on Intelligent Transportation Systems* 23, no. 7 (2021): 9078-9088.
35. Bonab, P.A., Holland, J., and Sargolzaei, A., "An Observer-Based Control for a Networked Control of Permanent Magnet Linear Motors under a False-Data-Injection Attack," in *2023 IEEE Conference on Dependable and Secure Computing (DSC)*, Tampa, FL, 2023, 1-8, IEEE.
36. Patre, P.M., MacKunis, W., Kaiser, K., and Dixon, W.E., "Asymptotic Tracking for Uncertain Dynamic Systems via a Multilayer Neural Network Feedforward and Rise Feedback Control Structure," *IEEE Transactions on Automatic Control* 53, no. 9 (2008): 2180-2185.

37. Chakraborty, I., Mehta, S.S., Doucette, E., and Dixon, W.E., "Control of an Input Delayed Uncertain Nonlinear System with Adaptive Delay Estimation," in *2017 American Control Conference (ACC)*, Seattle, WA, 2017, 1779-1784, IEEE.
38. Babic, M., "Analysis and Evaluation of Reinforcement Learning Algorithms for a Continuous Control Problem," PhD thesis, Hochschule für Angewandte Wissenschaften Hamburg, 2024.

Appendix A: Proof of Stability

Calculations

To proceed with stability analysis, we need to determine $\dot{e}_i, \dot{r}_2, \dot{\tilde{x}}_{i-1}$, and $\dot{\tilde{r}}_{i-1}$. These calculations are explained in detail as we go further in this section. To obtain $\dot{r}_i$ we take the time derivative of (7). Substituting (5) results in the closed-loop tracking error of the system, expressed as

$$\dot{r}_i(t) = \ddot{x}_i(t) - \ddot{x}_{i-1}(t) + \ddot{x}_{d_i} + \alpha_i \dot{e}_i(t) \quad \text{Eq. (A.1)}$$

replacing $\ddot{x}_i, \ddot{x}_{i-1}$ from the model into (A.1) and substituting $\dot{e}_i(t)$ from (7) yields

$$\begin{aligned}\dot{r}_i(t) = &-b_{i-1}v_{i-1}(t) + c_{i-1}u_{i-1}(t) + a_i v_i(t) \\ &-c_i u_i(t) - \ddot{x}_{d_i}(t) + \alpha_i r_i(t) - \alpha^2 e_i(t)\end{aligned} \quad \text{Eq. (A.2)}$$

to introduce the effect of FDI attack in the error dynamic, $u_{i-1}(t)$ term is substituted from (12)

$$\begin{aligned}\dot{r}_i(t) = &-b_{i-1}v_{i-1}(t) + b_{i-1}\bar{u}_{i-1} - c_{i-1}\beta_{i-1}(t) \\ &+b_i\dot{v}_i - c_i u_i(t) - \ddot{x}_{d_i}(t) + \alpha_i r_i(t) - \alpha_i^2 e_i(t)\end{aligned} \quad \text{Eq. (A.3)}$$

since the actual FDI attack signal $\beta_{i-1}(t)$ is unknown, we replace it from (26)

$$\begin{aligned}\dot{r}_i(t) = &-b_{i-1}v_{i-1}(t) + b_{i-1}\bar{u}_{i-1} - c_{i-1}\tilde{\beta}_{i-1}(t) - c_{i-1}\hat{\beta}_{i-1}(t) \\ &+b_i\dot{v}_i - c_i u_i(t) - \ddot{x}_{d_i}(t) + \alpha_i r_i(t) - \alpha_i^2 e_i(t)\end{aligned} \quad \text{Eq. (A.4)}$$

by applying the designed control law for $u_i(t)$ from (4), the tracking error can be generated as

$$\dot{r}_i(t) = -k_i r_i(t) - e_i(t) - c_{i-1}\tilde{\beta}_{i-1}(t) \quad \text{Eq. (A.5)}$$

Furthermore, to find the $\dot{\tilde{r}}_{i-1}$ we are taking the derivative of (10) with respect to time as

$$\dot{\tilde{r}}_{i-1}(t) = \dddot{\tilde{x}}_{i-1}(t) + \alpha_{i-1}\ddot{\tilde{x}}_{i-1}(t) \quad \text{Eq. (A.6)}$$

substituting $\tilde{x}_{i-1}(t)$ from (9), results

$$\begin{aligned}\dot{\tilde{r}}_{i-1}(t) = &-b_{i-1}v_{i-1}(t) + c_{i-1}u_{i-1}(t) + \alpha_{i-1}\tilde{r}_{i-1}(t) \\ &-\alpha_{i-1}^2\tilde{x}_{i-1}(t) - \ddot{\tilde{x}}_{i-1}(t)\end{aligned} \quad \text{Eq. (A.7)}$$

considering $u_{i-1}(t)$ as in (12), FDI attack estimation error in (26), and the observation rule as in (8) and substitute them in (A.7), we derive tracking error estimation as

$$\dot{\tilde{r}}_{i-1}(t) = -c_{i-1}\tilde{\beta}_{i-1}(t) - l_i\tilde{r}_{i-1}(t) - \tilde{x}_{i-1}(t) \quad \text{Eq. (A.8)}$$

To enhance the stability proof, we also need to calculate the estimation mismatch for the FDI attack, denoted as $\tilde{\beta}_{i-1}$ as follows. Substituting (14), (15), and (16) into (26) we obtain

$$\tilde{\beta}_{i-1}(t) = W_i^T\sigma_i\left(V_i^T I_i\right) - \hat{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \epsilon_i \quad \text{Eq. (A.9)}$$

by adding and subtracting $\hat{W}_i^T\sigma_i\left(V_i^T I_i\right)$ to (A.9), we have

$$\tilde{\beta}_{i-1} = \tilde{W}_i^T\sigma_i\left(V_i^T I_i\right) + \hat{W}_i^T\left[\sigma_i\left(V_i^T I_i\right) - \sigma_i\left(\hat{V}_i^T I_i\right)\right] + \epsilon_i \quad \text{Eq. (A.10)}$$

by adding and subtracting $\hat{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right)$, we have

$$\begin{aligned}\tilde{\beta}_{i-1} = &\tilde{W}_i^T\left[\sigma_i\left(V_i^T I_i\right) - \sigma_i\left(\hat{V}_i^T I_i\right)\right] \\ &+\hat{W}_i^T\left[\sigma_i\left(V_i^T I_i\right) - \sigma_i\left(\hat{V}_i^T I_i\right)\right] \\ &+\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \epsilon_i\end{aligned} \quad \text{Eq. (A.11)}$$

applying a Taylor's series approximation around $\sigma_i\left(V_i^T I_i\right)$, the FDI attack approximation error can be represented as

$$\begin{aligned}\tilde{\beta}_{i-1} = &\tilde{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i + \tilde{W}_i^T\mathcal{O}_i\left(\tilde{V}_i^T I_i\right) \\ &+\hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i + \hat{W}_i^T\mathcal{O}_i\left(\tilde{V}_i^T I_i\right) \\ &+\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \epsilon_i\end{aligned} \quad \text{Eq. (A.12)}$$

where $\mathcal{O}_i$ represents the higher-order terms of the Taylor series approximation. By making some simplifications, we have

$$\tilde{\beta}_{i-1} = \tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i + N_i \quad \text{Eq. (A.13)}$$

where

$$N_i \triangleq \tilde{W}_i^T \sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i + W_i^T \mathcal{O}_i\left(\tilde{V}_i^T I_i\right) + \epsilon_i \quad \text{Eq. (A.14)}$$

and the partial derivative of σ_i defined as

$$\sigma_i'\left(\hat{V}_i^T I_i\right) \triangleq \left.\frac{\partial \sigma_i\left(V_i^T I_i\right)}{\partial V_i^T I_i}\right|_{\hat{V}_i^T I_i} \quad \text{Eq. (A.15)}$$

Remark 1. *Based on Assumption* 1*, the constant values of W_i and V_i and the properties of the proj(·) smooth operator, $\hat{W}_i$ and $\hat{V}_i$ are bounded. So, by using the mean value theorem as it has been discussed in* [37]*, N_i is bounded such that $\|N_i\| \leq N_{i,max}$, where $N_{i,\,max} \in \mathbb{R}_{>0}$ is a positive known constant.*

Remark 2. *Based on Assumption* 1 *and Remark* 1*, $\tilde{W}_i$ and $\tilde{V}_i$ are bounded. Therefore, H_i as defined in* (21) *is bounded by $\|H_i\| \leq H_{i,max}$, where $H_{i,\,max} \in \mathbb{R}_{>0}$.*

Stability Proof

Proof. We define Lyapunov candidate function V_{L_i} as

$$V_{L_i} \triangleq \frac{1}{2}e_i^2 + \frac{1}{2}r_i^2 + \frac{1}{2}\tilde{x}_{i-1}^2 + \frac{1}{2}\tilde{r}_{i-1}^2 + H_i \quad \text{Eq. (A.16)}$$

where $V_{L_i} : \mathbb{R}^n \to \mathbb{R}$ is a continuous positive definite and continuously differentiable function, such that

$$\eta_{1_i}\left\|z_i\right\|^2 \leq V_{L_i} \leq \eta_{2_i} z_i^2 + H_{i,max} \quad \text{Eq. (A.17)}$$

Taking the time derivative of (A.16) results

$$\begin{aligned}\dot{V}_{L_i} &= e_i\dot{e}_i + r_i\dot{r}_i + \tilde{x}_{i-1}\dot{\tilde{x}}_{i-1} + \tilde{r}_{i-1}\dot{\tilde{r}}_{i-1} \\ &\quad - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\dot{\hat{W}}_i\right) - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\dot{\hat{V}}_i\right)\end{aligned} \quad \text{Eq. (A.18)}$$

Substituting (7), (10), (A.5), and (A.8) into (A.18), yields

$$\begin{aligned}\dot{V}_{L_i} &= e_i\left(r_i - \alpha_i e_i\right) + r_i\left(c_i\tilde{\beta}_{i-1} - k_i r_i - e_i\right) \\ &\quad + \tilde{x}_{i-1}\left(\tilde{r}_{i-1} - \alpha_{i-1}\tilde{x}_{i-1}\right) \\ &\quad + \tilde{r}_{i-1}\left(-c_{i-1}\tilde{\beta}_{i-1} - l_i\tilde{r}_{i-1} - \tilde{x}_{i-1}\right) \\ &\quad - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\dot{\hat{W}}_i\right) - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\dot{\hat{V}}_i\right)\end{aligned} \quad \text{Eq. (A.19)}$$

After some simplification, we have

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 \\ &\quad - l_i\tilde{r}_{i-1}^2 - c_{i-1}\tilde{r}_{i-1}\tilde{\beta}_{i-1} + c_{i-1}r_i\tilde{\beta}_{i-1} \\ &\quad - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\dot{\hat{W}}_i\right) - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\dot{\hat{V}}_i\right)\end{aligned} \quad \text{Eq. (A.20)}$$

introducing ψ_i as in (18) and (19), (A.20) can be defined as

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 \\ &\quad - l_i\tilde{r}_{i-1}^2 + \psi_i\tilde{\beta}_{i-1} \\ &\quad - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\dot{\hat{W}}_i\right) - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\dot{\hat{V}}_i\right)\end{aligned} \quad \text{Eq. (A.21)}$$

substituting $\tilde{\beta}_{i-1}$ from (A.13) yields

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 - l_i\tilde{r}_{i-1}^2 \\ &\quad + \psi_i\left(\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i\right) \\ &\quad + \psi_i N_i - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\dot{\hat{W}}_i\right) - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\dot{\hat{V}}_i\right)\end{aligned} \quad \text{Eq. (A.22)}$$

substituting $\dot{\hat{W}}_i$ and $\dot{\hat{V}}_i$ from (18) and (19) yields

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 - l_i\tilde{r}_{i-1}^2 \\ &\quad + \psi_i^T\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \psi_i^T\hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i \\ &\quad - tr\left(\tilde{W}_i^T\Gamma_{1i}^{-1}\Gamma_{1i}\sigma_i\left(\hat{V}_i^T I_i\right)\psi_i^T\right) \\ &\quad - tr\left(\tilde{V}_i^T\Gamma_{2i}^{-1}\Gamma_{2_i}I_i\psi_i^T\hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\right) \\ &\quad + c_{i-1}\left(-r_i - \tilde{r}_{i-1}\right)N_i\end{aligned} \quad \text{Eq. (A.23)}$$

with some simplification we have

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 - l_i\tilde{r}_{i-1}^2 \\ &\quad + \psi_i^T\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right) + \psi_i^T\hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i \\ &\quad - tr\left(\tilde{W}_i^T\sigma_i\left(\hat{V}_i^T I_i\right)\psi_i^T\right) \\ &\quad - tr\left(\tilde{V}_i^T I_i\psi_i^T\hat{W}_i^T\sigma_i'\left(\hat{V}_i^T I_i\right)\right) \\ &\quad - c_{i-1}r_i N_i - c_{i-1}\tilde{r}_{i-1}N_i\end{aligned} \quad \text{Eq. (A.24)}$$

since $tr(ba^T) = a^T b$, we are able to simplify the extra terms as

$$\begin{aligned}\dot{V}_{L_i} &= -\alpha_i e_i^2 - k_i r_i^2 - \alpha_{i-1}\tilde{x}_{i-1}^2 - l_i \tilde{r}_{i-1}^2 \\ &+\psi_i^T \tilde{W}_i^T \sigma_i\left(\hat{V}_i^T I_i\right) + \psi_i^T \hat{W}_i^T \sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i \\ &-\psi_i^T \tilde{W}_i^T \sigma_i\left(\hat{V}_i^T I_i\right) - \psi_i^T \hat{W}_i^T \sigma_i'\left(\hat{V}_i^T I_i\right)\tilde{V}_i^T I_i \\ &-c_{i-1} r_i N_i - c_{i-1}\tilde{r}_{i-1} N_i\end{aligned} \quad \text{Eq. (A.25)}$$

the derivative of Lyapunov candidate function can be written as

$$\begin{aligned}\dot{V}_{L_i} &\le -\alpha_i \| e_i \|^2 - k_i \| r_i \|^2 \\ &-\alpha_{i-1} \| \tilde{x}_{i-1} \|^2 - l_i \| \tilde{r}_{i-1} \|^2 \\ &-c_{i-1} \| r_i \| \| N_i \| - c_{i-1} \| \tilde{r}_{i-1} \| \| N_i \|\end{aligned} \quad \text{Eq. (A.26)}$$

Using Young inequality, the last two terms of (A.26) can be upper bounded as

$$c_{i-1} \| r_i \| \| N_i \| < \frac{c_{i-1}}{2\varepsilon_{1_i}} \| r_i \|^2 + \frac{c_{i-1}\varepsilon_{1_i}}{2} \| N_i \|^2 \quad \text{Eq. (A.27)}$$

$$c_{i-1} \| \tilde{r}_{i-1} \| \| N_i \| < \frac{c_{i-1}}{2\varepsilon_{2_i}} \| \tilde{r}_{i-1} \|^2 + \frac{c_{i-1}\varepsilon_{2_i}}{2} \| N_i \|^2 \quad \text{Eq. (A.28)}$$

Applying (A.27) and (A.28) into (A.26) we are able to make simplification as

$$\begin{aligned}\dot{V}_{L_i} &\le -\alpha_i \| e_i \|^2 - k_i \| r_i \|^2 \\ &-\alpha_{i-1} \| \tilde{x}_{i-1} \|^2 - l_i \| \tilde{r}_{i-1} \|^2 \\ &+\frac{c_{i-1}}{2\varepsilon_{1_i}} \| r_i \|^2 + \frac{c_{i-1}\varepsilon_{1_i}}{2} \|N_i\|^2 \\ &+\frac{c_{i-1}}{2\varepsilon_{2_i}} \|\tilde{r}_{i-1}\|^2 + \frac{c_{i-1}\varepsilon_{2_i}}{2} \|N_i\|^2\end{aligned} \quad \text{Eq. (A.29)}$$

Rearranging the inequality results in

$$\begin{aligned}\dot{V}_{L_i} &\le -\alpha_i \| e_i \|^2 - \left(k_i - \frac{c_{i-1}}{2\varepsilon_{1_i}}\right) \| r_i \|^2 - \alpha_{i-1} \| \tilde{x}_{i-1} \|^2 \\ &-\left(l_i - \frac{c_{i-1}}{2\varepsilon_{2_i}}\right) \| \tilde{r}_{i-1} \|^2 + \frac{c_{i-1}}{2}(\varepsilon_{1_i} + \varepsilon_{2_i}) \| N_i \|^2\end{aligned} \quad \text{Eq. (A.30)}$$

based on the definition in (A.14) and Remark 1, N_i is upper bounded. Considering (23) yields

$$\begin{aligned}\dot{V}_{L_i} &\le -\alpha_i \| e_i \|^2 - \left(k_i - \frac{c_{i-1}}{2\varepsilon_{1_i}}\right) \| r_i \|^2 - \alpha_{i-1} \| \tilde{x}_{i-1} \|^2 \\ &-\left(l_i - \frac{c_{i-1}}{2\varepsilon_{2_i}}\right) \| \tilde{r}_{i-1} \|^2 + \lambda_i\end{aligned} \quad \text{Eq. (A.31)}$$

and using the definition in (20) we have

$$\dot{V}_{L_i} \le -\chi_i \|z_i\|^2 + \lambda_i \quad \text{Eq. (A.32)}$$

where $\chi_i \triangleq \min\left\{\alpha_i, \alpha_{i-1}, k_i - \frac{c_{i-1}}{2e_{1i}}, l_i - \frac{c_{i-1}}{2e_{2i}}\right\}$. Based on (A.16) we have

$$\dot{V}_{L_i} \le -\frac{\chi_i}{\eta_{i_2}} V_{L_i} + \frac{\chi_i}{\eta_{i_2}} H_{i,max} + \lambda_i \quad \text{Eq. (A.33)}$$

by solving the resultant differential equation in (A.33) we have

$$\begin{aligned}V_{L_i}(t) &\le V_{L_i}(0)\exp\left(-\frac{\chi_i}{\eta_{i_2}} t\right) \\ &+\left(H_{i,max} + \frac{\eta_{i_2}\lambda_i}{\chi_i}\right)\left(1 - \exp\left(-\frac{\chi_i}{\eta_{i_2}} t\right)\right)\end{aligned} \quad \text{Eq. (A.34)}$$

Considering the property described in (A.16), we are able to conclude the upper bound in (24) for system errors. Based on this conclusion and the Lyapunov stability theorem, $z_i \in \mathcal{L}_\infty$ and there exists γ_i such that the system errors $e_i, r_i, \tilde{r}_{i-1}, \tilde{x}_{i-1}$ stay within their allowed ranges. This implies that globally and uniformly bounded tracking could be ensured. □

Tracking Bound Calculation

The solution to the Lyapunov equation in (A.33), which is a linear differential equation, has been expressed in (A.34). By substituting the lower bound of V_{L_i} from (A.16), we have

$$\eta_{1_i}\left\|z_i(t)\right\|^2 \le V_{L_i}(0)\exp\left(-\frac{\chi_i}{\eta_{i_2}} t\right) + \left(H_{i,\max} + \frac{\eta_{i_2}\lambda_i}{\chi_i}\right)\left(1 - \exp\left(-\frac{\chi_i}{\eta_{i_2}} t\right)\right) \quad \text{Eq. (A.35)}$$

therefore, as $t \to \infty$ the (A.35) can be expressed as

$$\lim_{t\to\infty} \sup \left\|z_i(t)\right\|^2 \le \frac{1}{\eta_{i_1}}\left(H_{i,max} + \frac{\eta_{i_2}\lambda_i}{\chi_i}\right) \quad \text{Eq. (A.36)}$$

and the condition in (24) will be concluded.