Secure Dynamic Nonlinear Heterogeneous Vehicle Platooning: Denial-of-Service Cyber-Attack Case

Mohammad Hossein Basiri and Nasser L. Azad and Sebastian Fischmeister

Abstract Connected and Automated Vehicles (CAVs), as a large class of cyber-physical systems, have recently emerged as an effective autonomous driving mechanism in intelligent transportation systems in terms of improvement in safety, fuel economy, road throughput, and driving comfort. This chapter deals with a Secure Distributed Nonlinear Model Predictive Control (Secure–DNMPC) algorithm consisting of i) detection and ii) mitigation phases to securely control a string of CAVs, namely *vehicle platoons*. The approach ensures the desired control performance of a nonlinear heterogeneous platoon equipped by different communication topologies under the premise of the existence of Denial-of-Service (DoS) attacks. The proposed method is also capable of providing safe and secure control of dynamic platoons in which arbitrary vehicles might perform cut-in and/or cut-out maneuvers. Convergence time and stability analysis of the system are also investigated in some cases. Furthermore, to handle DoS attacks modeled by an exceeding time delay in inter-vehicular data transmission, we propose the integration of an Unscented Kalman Filter (UKF) design within the controller resulting in a novel Secure–DNMPC–UKF co-design. This, in essence, estimates the delayed system states and feeds the predicted values to the Secure–DNMPC, which efficiently mitigates the attack effects. Simulation results demonstrate the fruitfulness of the proposed method.

Mohammad Hossein Basiri
Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada.
e-mail: mh.basiri@uwaterloo.ca

Nasser L. Azad
Department of Systems Design Engineering, University of Waterloo, ON, Canada.
e-mail: nlashgar@uwaterloo.ca

Sebastian Fischmeister
Department of Electrical and Computer Engineering, University of Waterloo, ON, Canada.
e-mail: sfischme@uwaterloo.ca

1 Introduction

This section introduces the general notion of Cyber-Physical Systems (CPSs) along with one of their subclasses, namely Connected and Automated Vehicles (CAVs), and explains some of the most important and prevalent security-related issues that need to be taken into account while dealing with these systems. Then, we will review the related work and explicitly state the contributions of the current chapter.

1.1 State-of-the-Art

Cyber-Physical Systems (CPSs) are among the fast emerging profound infrastructures enabling traditional physical plants to operate in a wide area and a distributed fashion. Networking, computation, communication, and control are tightly interwoven to foster a CPS [1]. These components are categorized in cyber and physical layers, each of which interacts with the other parts to receive external data, process them, and generate appropriate output signals. Never may a CPS perform without the proper and timely functioning of its constituents. Instances of CPS include, but are not limited to, automotive control, medical monitoring, robotic systems, and smart grid [2].

Apart from the physical layer, the cyber one has been broadly shown to be prone to external intelligent cyber-attacks. Data integrity, confidentiality, and availability are the major crucial concerns of cyber-security that an intelligent intruder might target [3,4]. Various attacks have been introduced to destruct one or more of the aforementioned security aspects of a CPS. False data injection, GPS spoofing, eavesdropping, Denial-of-Service (DoS), and replay attack are some of the paradigms [5–9]. Several well-known attacks on CPS include Stuxnet on a Supervisory Control and Data Acquisition (SCADA) system [10, 11], attacks on the wireless network channels in smart power grid systems [12], and compromising Anti-lock Braking System (ABS) sensors of a vehicle [13]. Hence, the security-related issues, such as attack detection and secure state estimation and control of CPS, have been converted to attracting challenges in the control community.

Constituting an important application of CPS, autonomous driving has greatly emerged during the last decade. Due to the huge growth in the number of vehicles driving in the world, traffic congestion threatens driving safety. This will potentially result in increasing the risk of accidents. Autonomous vehicles and autonomous driving are another aspects which have got a great deal of attention. Through this technological development, driving safety can be highly enhanced as most car accidents are caused by human errors and distractions while driving. Over 90% of all car accidents are caused by human errors [14]. From this point of view, self-driving cars can remove a considerable amount of human errors resulting in safer transportation. In Canada alone, there were close to 111,000 road-related injuries and over 1,800 fatalities reported in 2014 [15]. Autonomous cars have many other advantages, such as getting faster to the destination, reducing governmental costs and car ownership [16, 17].

The degree of autonomy incorporated in autonomous driving is categorized in 6 different levels (levels 0-5). Level 0 (no automation) is the most basic one in which no autonomy is incorporated. The vehicle is fully controlled by a human driver. In level 1 (driver assistance), the vehicle can assist the driver with some functions, such as steering, acceleration, or braking. In level 2 (partial automation), the vehicle lets the driver disinvolve with some of these tasks. The driver still has the main role in monitoring the environment and in taking care of most safety-critical functions. The driver is responsible for taking full control of the vehicle when needed. In level 3 (conditional automation), the vehicle performs all the environment monitoring tasks. In safe conditions, the driver can leave the safety-critical functions like braking to the vehicle; however, his attention is still required. Level 4 (high automation) of autonomy is able to take care of monitoring the environment, steering, acceleration, and braking. In addition, the vehicle is capable of changing lanes, turning and using signals. However, the vehicle can not perform decisions in more complex scenarios, such as traffic jams or merging onto the highway. Level 5 (complete automation) exploits full autonomy, which requires no human intervention, pedals, brakes, or a steering wheel.

Connected and Automated Vehicles (CAVs), as a large class of CPS, have recently emerged as an effective autonomous driving mechanism in intelligent transportation systems in terms of improvement in safety, fuel economy, road throughput, and driving comfort. Vehicles participating in a realistic platoon most likely bear variant nonlinear dynamics forming a heterogeneous platoon. Platoons could be formed based on different spacing policies and governed by different formation control techniques such as traditional linear/nonlinear controllers, optimal control methods, and more advanced consensus algorithms [18–20]. It has been widely proved in the literature that nonlinear control techniques are mandatory to achieve desired formation objectives, such as maintaining a safe gap among consecutive cars while tracking the speed profile of the leader vehicle.

CAVs can communicate with each other and exchange their date through Vehicle-to-Vehicle (V2V) and/or Vehicle-to-Infrastructure (V2I) wireless communications. Getting more developed through using more effective data communication structures, connected vehicles are equipped with different information flow topologies to facilitate and improve the efficacy of data transfers. Predecessor-follower (PF), predecessor-leader follower (PLF), two-predecessors follower (TPF), two-predecessors-leader follower (TPLF), all-predecessors follower (APF), all-predecessors-leader follower (APLF), and h–nearest neighbor are some of the instances [21, 22]. These structures can be exploited either in a unidirectional or a bidirectional data transmit (see Fig. 1).

As was mentioned before, autonomous cars can be equipped with wireless data communication devices such that they can transfer data such as inter-vehicular distance and speed. In this respect, CAVs typically take advantage of V2V and/or V2I communication environments. V2V communications can provide direct data transfer with a much lower delay compared to radars [23]. The V2V communications en-

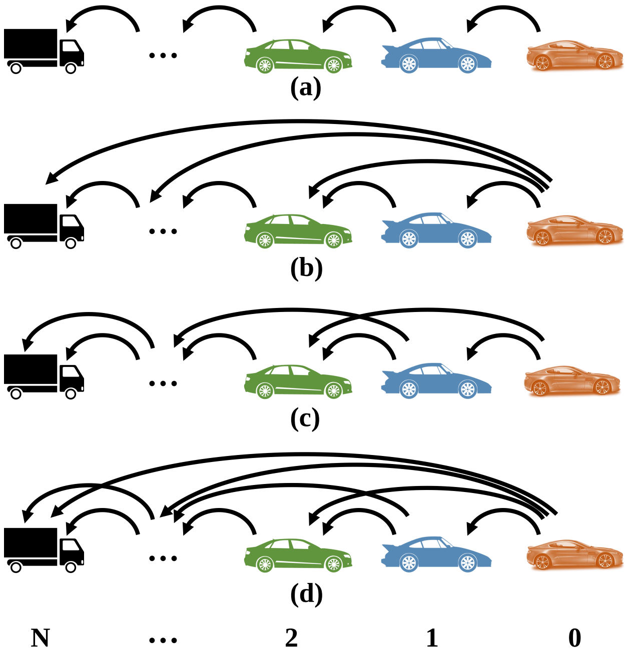


Fig. 1: Unidirectional topology: (a) PF, (b) PLF, (c) TPF, and (d) TPLF, (vehicle 0 is the Leader Vehicle (LV))

able vehicles to drive closely with short inter-vehicular distances. This will increase the amount of road throughput and reduce the need for developing more road networks. The vehicles can exchange data, such as inter-vehicular distance, speed and acceleration. In this context, Cooperative Adaptive Cruise Control (CACC) system has been widely developed, featuring the possibility of coordination between connected vehicles aiming at enhancing fuel efficiency, safety, driving comfort, and road throughput. This system, which is the advanced version of Adaptive Cruise Control (ACC), lets neighboring vehicles form a platoon, which is a string of vehicles following a common speed profile.

Despite the benefits of wireless connectivity among these vehicles, this makes the whole system susceptible to cyber-attacks. One such a prevalent attack, that has broadly drawn the attention of both cyber-security and control communities, is Denial-of-Service (DoS) attack. A DoS intelligent intruder aims at jamming communication links among cars through overwhelming the beacon node by fake requests, hence, hinders the network from processing legitimate requests. This can result in huge performance degradation and even hazardous collisions.

This chapter concerns with the control problem of a large class of CPS, namely platoon formation, which has had a leading role in autonomous driving systems. Basically, a platoon is a string of connected and automated vehicles, all driving with a pre-specified safe gap among consecutive cars and following a shared speed profile generated by the leader vehicle. This physical layer, together with the wireless connectivity among the participating vehicles, as the cyber layer, constitutes the system as a whole CPS. Vehicle platooning is well-known due to its advantages such as enhancement of road throughput, fuel economy, driving comfort, and safety; however, it suffers from the vulnerability of wireless connections among the cars to devastating malware [24–26]. In particular, an outsider attacker might compromise inter-vehicular data to fool the on-board sensors and controller of the receiver vehicle resulting in unnecessary acceleration/brake actions. Besides, he may cause a failure in the network by jamming it or injecting a huge amount of delay, which in essence makes the outdated transferred data useless. The latter is the scenario from which a DoS attacker takes advantage and will be the focus of this chapter.

1.2 Related Work

Recently, much research has been done in investigating the security of networked control systems from various perspectives [27–31]. Communication-related protection methods, such as encryption of wireless channels, are techniques to avoid receiving compromised data via the wireless infrastructures [32]. On the other hand, control-oriented concepts, such as game-theoretic methods, are also among the leading methodologies which address the security issue of general cyber-physical systems with a considerable amount of care [33, 34]. Although there has been a large amount of research addressing the security of CPS, those systems still suffer from the lack of secure performance in the presence of possible malicious intruders [31, 35, 36]. This needs to be noted that the introduced techniques for fault-tolerant control might be applied to security problems; however, the majority of those technique cannot handle the devastation imposed by an intelligent intruder [37]. The reason is that an intelligent attacker has some *a priori* knowledge of the system dynamics and/or the controller which is not the case in a random fault. Furthermore, specific components of a system may be targeted by an intelligent adversary based on his own criteria, such as optimizing the amount of consumed energy or the intended level of devastation. In this regard, different methodologies based on system/graph/game notions have been introduced to address security issues of general control systems [34, 38–40].

In the recent decade, there has been a vast range of studies addressing security issues of vehicle platoons [41–48]. In essence, researchers have been concerned about possible vulnerabilities of vehicle platoons against cyber attacks as well as communication delays [24, 42–53]. Particularly, in [46], a DoS attack detection and estimation scheme based on sliding mode observer has been proposed for a linear homogeneous car following system. Also, authors of [47, 48] study the performance degradation of a linear homogeneous vehicle following controller caused by unreliable wireless communications. Various types of intrusions imposed by either insider and/or outsider adversary on connected vehicles have been investigated in literature, which include but are not limited to DoS, GPS spoofing, masquerading, insider/outsider eavesdropping [54]. Each of these attacks can potentially degrade system performance by violating one or more of the data availability, data confidentiality, and data integrity. A detailed and formal attack classification in a three-dimensional attack space is given in [55]. Network-aware control methods have also been proposed to handle possible communication failures through the platoon. Those approaches mainly consider random communication failures with an emphasis on the control/stability performance of the whole platoon without considering intelligent cyber attacks [51, 56, 57].

However, the lack of a systematic approach adhering to control performance objectives of a dynamic nonlinear heterogeneous platoon while mitigating the DoS attack effects is yet sensible. Thus, in this study, we focus on an attacked dynamic nonlinear heterogeneous platoon in which arbitrary vehicles might perform cut-in/cut-out maneuvers. Variant nonlinear dynamics of the participating cars are considered in the model to form a realistic nonlinear heterogeneous platoon.

1.3 Contributions

Contributions of this chapter are explicitly as follows. Under the premise of the existence of a DoS attacker of either a network blocker or a huge time delay injector, we propose a Secure Distributed Nonlinear Model Predictive Control (Secure–DNMPC) framework to detect and mitigate the attack effects while ensuring fulfillment of the platoon control objectives. The algorithm is flexible to adopt different communication topologies handling inter-vehicular data transfer among the vehicles. Convergence time and stability analysis of the algorithm is proved in some cases. Furthermore, in case of a DoS attacker as an exceeding time delay injector, since the transferred data are still available while the attack is underway, we propose to make use of the outdated system states and take benefit of them to implement the control strategy instead of simply ignoring the data and using the most recent one. This will effectively improve the control performance of the whole system. In essence, we propose to embed a UKF as the state observer within the design of the Secure–DNMPC to adapt the algorithm to the delayed data transmission. This results in a novel Secure–DNMPC–UKF co-design. In addition, this gives the opportunity to either consider non-ideal noisy sensors or take into account the contaminated sent data due to the noisy surrounding environment and road conditions.

1.4 Chapter Organization

The remainder of this chapter is organized as follows. Sec. 2 presents the system modeling, including the platoon model and different types of DoS attack descriptions. Sec. 3 details the design of the secure controller together with some stability analysis results. Adaptation of the algorithm to handle dynamic maneuvers together with convergence time analysis are given in Sec. 4. Sec. 5 demonstrates the simulation results on a Two-Predecessor Follower (TPF) attacked nonlinear dynamic heterogeneous platoon. Finally, Sec. 6 concludes the chapter.

2 System Modeling

In this section, we present the considered platoon model and its control objectives. In addition, we give different DoS attack descriptions on which we focus in this chapter.

2.1 Platoon Model

Let us consider a platoon of vehicles, consisting of a Leader Vehicle (LV) and N Follower Vehicles (FVs) indexed by $\mathcal{N} := \{1, \ldots, N\}$. In this chapter, we consider the longitudinal dynamics and unidirectional communication topologies. Let Δt be the discrete time interval and $p_i(t)$, $v_i(t)$, and $T_i(t)$ denote the position, velocity, and the integrated driving/breaking torque of the i-th FV at time t, respectively. For the i-th FV, we denote the vehicle mass, the coefficient of aerodynamic drag, the coefficient of rolling resistance, the inertial lag of longitudinal dynamics, the tire radius, the mechanical efficiency of the driveline, and the control input by m_i, $C_{A,i}$, $f_{r,i}$, τ_i, r_i, η_i, and $u_i(t) \in \mathbb{R}$, respectively and g is the gravity constant. The dynamics of the i-th FV can be stated via the following discrete-time nonlinear model [58]

$$\begin{cases} \boldsymbol{x}_i(t+1) = \boldsymbol{\phi}_i(\boldsymbol{x}_i(t)) + u_i(t)\,\boldsymbol{\psi}_i \\ \quad\; \boldsymbol{y}_i(t) = \boldsymbol{\gamma}\,\boldsymbol{x}_i(t), \end{cases} \tag{1}$$

where $\boldsymbol{x}_i(t) := [p_i(t), v_i(t), T_i(t)]^\top \in \mathbb{R}^3$ and $\boldsymbol{y}(t) := [p_i(t), v_i(t)]^\top \in \mathbb{R}^2$ are the states and outputs of each vehicle, respectively. Also, $\boldsymbol{\psi}_i := [0, 0, (1/\tau_i)\,\Delta t]^\top$, $\boldsymbol{\gamma} := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$, and

$$\boldsymbol{\phi}_i(\boldsymbol{x}_i(t)) := \begin{bmatrix} p_i(t) + v_i(t)\,\Delta t \\ v_i(t) + \frac{\Delta t}{m_i}\left(\frac{\eta_i}{r_i} T_i(t) - C_{A,i}\, v_i^2(t) - m_i\, g\, f_{r,i}\right) \\ T_i(t) - (1/\tau_i)\, T_i(t)\,\Delta t \end{bmatrix}. \tag{2}$$

Stacking the states, outputs, and the control input signals of all vehicles into vectors yields the platoon dynamics as follows

$$\begin{cases} X(t+1) = \boldsymbol{\Phi}(X(t)) + \boldsymbol{\Psi} U(t), \\ Y(t+1) = \boldsymbol{\Theta} \cdot X(t+1), \end{cases} \tag{3}$$

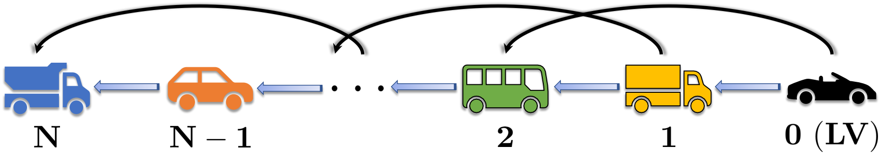


Fig. 2: TPF heterogeneous vehicle platoon with a leader and N followers

where $X(t) = [x_1(t)^\mathsf{T}, x_2(t)^\mathsf{T}, \dots, x_N(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3N\times1}, Y(t) = [y_1(t)^\mathsf{T}, y_2(t)^\mathsf{T}, \dots, y_N(t)^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{2N\times1}$, $U(t) = [u_1(t), u_2(t), \dots, u_N(t)]^\mathsf{T} \in \mathbb{R}^{N\times1}$. Besides, $\boldsymbol{\Phi} = [\phi_1^\mathsf{T}, \phi_2^\mathsf{T}, \dots, \phi_N^\mathsf{T}]^\mathsf{T} \in \mathbb{R}^{3N\times1}$, $\boldsymbol{\Psi} = \mathrm{diag}\{\psi_1, \psi_2, \dots, \psi_N\} \in \mathbb{R}^{3N\times N}$, and $\boldsymbol{\Theta} = I_N \otimes \gamma \in \mathbb{R}^{2N\times3N}$.

Let $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N\times N}$ be the adjacency matrix of the underlying platoon graph topology where $a_{ij} = 1$ ($= 0$) means that the j-th FV can (cannot) send information to the i-th FV, and $\mathcal{D} = \mathrm{diag}\{\mathrm{deg}_1, \mathrm{deg}_2, \dots, \mathrm{deg}_n\}$ be the degree matrix, where $\mathrm{deg}_i = \Sigma_{j=1}^n a_{ij}$. Also, let $p_i = 1$ ($= 0$) mean that the i-th FV is (not) pinned to the LV and gets (does not get) information from it. Suppose $\mathbb{P}_i := \{0\}$ if $p_i = 1$ and $\mathbb{P}_i := \varnothing$ if $p_i = 0$. The pinning matrix is then defined by $\mathcal{P} = \mathrm{diag}\{p_1, p_2, \dots, p_n\}$. We denote $\mathbb{N}_i := \{j | a_{ij} = 1, j \in \mathcal{N}\}$ and $\mathbb{O}_i := \{j | a_{ji} = 1, j \in \mathcal{N}\}$ as the sets of FVs which the i-th FV can get information from and send information to, respectively. The set $\mathbb{I}_i := \mathbb{N}_i \cup \mathbb{P}_i$ is the set of all vehicles sending information to the i-th FV. In this study, for convenience, we consider a dynamic heterogeneous platoon equipped by Two-Predecessor Follower (TPF) communication topology shown in Fig. 2; however, it is straightforward to adapt our algorithm to other communication topologies.

Assumption 1 The directed graph of the platoon topology contains a spanning tree rooted at the LV. This assumption is necessary for stability in both homogeneous [59] and heterogeneous [58] platooning. This ensures that all vehicles get the leader's information either directly or indirectly.

2.2 Platoon Control Objectives

The control objectives of the platoon are to track the speed profile generated by the leader while keeping the safe desired distance between the vehicles. Mathematically, we aim at $\lim_{t\to\infty} |v_i(t) - v_0(t)| = 0$ and $\lim_{t\to\infty} |p_{i-1}(t) - p_i(t) - d| = 0$ where d is the desired constant distance between every two consecutive vehicles. We also denote the distance between the i-th and j-th FVs by $d_{i,j}$.

Two types of output are considered here, which are the predicted and assumed outputs. The former is obtained by the calculated control input from optimization, which is fed to the system. The latter is obtained by shifting the optimal output of the last-step optimization problem. Let $\boldsymbol{y}_i^p(k|t)$ and $\boldsymbol{y}_i^a(k|t)$ denote the predicted output and the assumed output, respectively. We explain the details of how to obtain these two outputs in the following sections. The predicted and assumed states are denoted by $\boldsymbol{x}_i^p(k|t)$ and $\boldsymbol{x}_i^a(k|t)$, respectively.

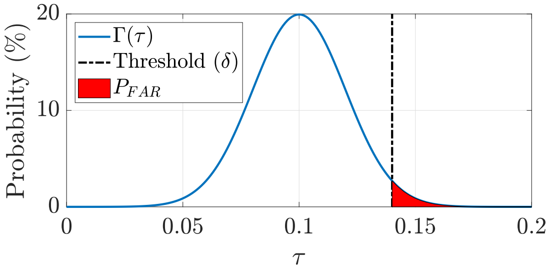


Fig. 3: Probability distribution function of the false alarm rate and the threshold

2.3 Attack Description

We mainly focus on a widespread cyber-attack, called the DoS attack. Basically, endangering the security of the system, a DoS attacker jams the network by flooding it with fake requests such that the shared network gets overwhelmed by these demands; hence, becomes too busy to process the legitimate requests sent by the authorized users [27, 60]. This inherently causes packet loss or at least suffering delays in data transfers. In our application, we study two different DoS attack modeling introduced in the literature, i.e.,

- The DoS attacker is able to block the communication link among two nonconsecutive neighboring vehicles, which results in missing inter-vehicular data received by the follower vehicle. In essence, if the communication link among vehicle i and $i-2$ is attacked during $t \in [t_0, t_1]$, the vehicle i is only able to receive the valid data up to $t = t_0$ and has the exact same data until the attack is over, i.e., vehicle i will restart to receive updated data from vehicle $i-2$ at $t > t_1$. In the rest of the chapter, we denote $\tau_a = t_1 - t_0$ as the attack period for notational convenience.
- Another prevalent DoS attack type is to view the intruder as who injects a relatively large delay in the data transmission network. Hence, in this case, during the attack period τ_a, the follower vehicles receive the data with the time delay τ_r. This time delay is much larger compared to a threshold for a practical DSRC network.[1] The threshold can be calculated based on the acceptable probability of false alarm rate P_{FAR}
$$P_{\text{FAR}} = \int_{\delta}^{\infty} \Gamma(\tau) d\tau \leq P_{\text{FAR, acceptable}}, \tag{4}$$
where $\Gamma(\tau)$ is the probability density function of the time delay, and δ is the chosen threshold (shown in Fig. 3) [46]. The threshold δ can also be determined using Monte-Carlo simulations, False Positives and True Negatives [31].

We will propose countermeasures in subsequent sections to face both of the aforementioned attack modelings.

[1] It should be noted that the acceptable time delay heavily depends on the application. Here, we focus on the automotive control application.

3 Secure Controller Design for Dynamic Heterogeneous Platooning

Details of the proposed secure controller are given in this section. In addition, some preliminary concepts needed to develop the method is explained. Furthermore, closed-loop system stability along with the convergence time analysis are presented in this section.

3.1 Overview

To countermeasure the DoS attacker explained in the previous section, we take advantage of a modified version of the Distributed Nonlinear Model Predictive Control (DNMPC) approach proposed in [58], called Secure–DNMPC, which aims at mitigating the effects of the attack while achieving the desired control objectives. The algorithm basically consists of two main phases, namely i) detection and ii) mitigation phase. In the first phase, we seek to detect if a DoS attack is underway. If an attack is detected in which the attacked communication link corresponds to the ego-vehicle with its immediate preceding or following vehicle, then the algorithm ignores the data received through the V2V link (until the attack is over) and switches to the on-board sensors followed by the implementation of the DNMPC. Otherwise, if the blocked link corresponds to the farther neighbors of the ego-vehicle, the victim vehicle makes use of the most recent updated data prior to the attack commence, and the mitigation phase starts by performing Secure–DNMPC. Inherently, in the second phase, each vehicle solves a local optimal control problem detailed as follows to generate its own optimal control input signal, which is used to compute the assumed states. The assumed states are then exchanged with the neighbors. Moreover, if the intruder targets the communication link by injecting a huge amount of time delay in data transmission, denoted by τ_r, the algorithm switches to the Secure–DNMPC–UKF mode to make use of the delayed states as much as possible. Specifically, in this case, the controller employs the observer to estimate the delayed states and provides the controller with the predicted data. The mechanism of the controller in both of the above-mentioned cases are detailed in the following sections.

Assumption 2 As a standard assumption and from a practical point of view, we assume that the attacker has a limited resource of energy preventing him from jamming the network ceaselessly [34, 61, 62].

Remark

It is notable that the DoS attacker never attacks a link between two consecutive vehicles. The reason is that in the algorithm, the positions and velocities are transmitted, which can be reliably measured by on-board sensors mounted on an ego-vehicle such as GPS and radar. Thus, once a follower detects that those quantities are no longer updated, it can switch to its redundant sensors to obtain real-time data.

3.2 Design of the Secure Controller

Consider a predictive horizon N_p for the model predictive control employed to control the platoon. Suppose the predicted control inputs over the horizon are $\mathcal{U}_i^p(t-\tau) := \{u_i^p(0|t-\tau), \ldots, u_i^p(N_p-1|t-\tau)\}$ which need to be calculated by the following optimization problem, which is the local NMPC problem that each vehicle needs to solve at each time instant t

$$\underset{\mathcal{U}_i^p(t-\tau)}{\text{minimize}} \quad J_i(\boldsymbol{y}_i^p, u_i^p, \boldsymbol{y}_i^a, \boldsymbol{y}_{-i}^a) \tag{5a}$$

$$\begin{aligned}
\text{subject to} \quad & \boldsymbol{x}_i^p(k+1|t-\tau) = \boldsymbol{\phi}_i(\boldsymbol{x}_i^p(k|t-\tau)) + u_i^p(k|t-\tau)\,\boldsymbol{\psi}_i, && \text{(5b)}\\
& \boldsymbol{y}_i^p(k|t-\tau) = \boldsymbol{\gamma}\,\boldsymbol{x}_i^p(k|t-\tau), && \text{(5c)}\\
& \boldsymbol{x}_i^p(0|t-\tau) = \boldsymbol{x}_i(t-\tau), && \text{(5d)}\\
& u_i^p(k|t-\tau) \in \mathfrak{U}_i, && \text{(5e)}\\
& \boldsymbol{y}_i^p(N_p|t-\tau) = \frac{1}{|\mathbb{I}_i|}\sum_{j\in\mathbb{I}_i}\big(\boldsymbol{y}_j^a(N_p|t-\tau) + \widetilde{\boldsymbol{d}}_{i,j}\big), && \text{(5f)}\\
& T_i^p(N_p|t-\tau) = h_i(v_i^p(N_p|t-\tau)), && \text{(5g)}
\end{aligned}$$

where $\boldsymbol{y}_{-i}(t) := [\boldsymbol{y}_{i_1}^\top, \ldots, \boldsymbol{y}_{i_m}^\top]^\top$ (if $\{i_1, \ldots, i_m\} := \mathbb{N}_i$), $\mathfrak{U}_i = \{u_i \mid u_i \in [\underline{u}_i, \bar{u}_i]\}$ defines the feasible bounds on the control input, $|\mathbb{I}_i|$ is the cardinality of $\mathbb{I}_i$, $\widetilde{\boldsymbol{d}}_{i,j} := [d_{i.j}, 0]^\top$, and τ is either τ_a or τ_r depending on the attack model. The last two terminal constraints are to make the DNMPC algorithm stable. For a detailed description of the above constraints, the interested reader is referred to [58].

The objective function (5a) is defined as the summation of all local cost functions

$$\begin{aligned}
J_i(\boldsymbol{y}_i^p, u_i^p, \boldsymbol{y}_i^a, \boldsymbol{y}_{-i}^a) := \sum_{k=0}^{N_p-1} \Big(& \|\boldsymbol{y}_i^p(k|t-\tau) - \boldsymbol{y}_{\mathrm{des},i}(k|t-\tau)\|_{\boldsymbol{Q}_i} \\
& + \|u_i^p(k|t-\tau) - h_i(v_i^p)\|_{R_i} + \|\boldsymbol{y}_i^p(k|t-\tau) - \boldsymbol{y}_i^a(k|t-\tau)\|_{\boldsymbol{F}_i} \\
& + \sum_{j\in\mathbb{N}_i} \|\boldsymbol{y}_i^p(k|t-\tau) - \boldsymbol{y}_j^a(k|t-\tau) - \widetilde{\boldsymbol{d}}_{i,j}\|_{\boldsymbol{G}_i} \Big),
\end{aligned} \tag{6}$$

in which, for a weight matrix $\boldsymbol{A} \geq 0$, $\|\boldsymbol{x}\|_{\boldsymbol{A}} := \boldsymbol{x}^\top \boldsymbol{A}\,\boldsymbol{x}$. In (6), $0 \leq \boldsymbol{Q}_i, \boldsymbol{F}_i, \boldsymbol{G}_i \in \mathbb{R}^2$ and $0 \leq R_i \in \mathbb{R}$ are the weight matrices which are the NMPC regularization factors. In fact, the matrices $\boldsymbol{Q}_i$, R_i, $\boldsymbol{F}_i$, $\boldsymbol{G}_i$ penalize for deviation of the predicted output from the desired output $\boldsymbol{y}_{\mathrm{des},i}(k|t-\tau)$, deviation of the predicted control input from the equilibrium, deviation of the predicted output from the assumed output, and deviation of the predicted output from the neighbors' assumed trajectories, respectively. For the i-th FV, the desired state and control signal are $\boldsymbol{x}_{\mathrm{des},i}(t) := [p_{\mathrm{des},i}(t), v_{\mathrm{des},i}(t), T_{\mathrm{des},i}(t)]^\top$ and $u_{\mathrm{des},i}(t) := T_{\mathrm{des},i}(t)$, respectively, where $p_{\mathrm{des},i}(t) := p_0(t) - i\,d$, $v_{\mathrm{des},i}(t) := v_0$, $T_{\mathrm{des},i}(t) := h_i(v_0)$ where $h_i(v_0) := (r_i/\eta_i)(C_{A,i}\,v_0^2 + m_i\,g\,f_{r,i})$ is the external drag. The desired output is $\boldsymbol{y}_{\mathrm{des},i}(t) := \boldsymbol{\gamma}\,\boldsymbol{x}_{\mathrm{des},i}(t) \in \mathbb{R}^2$.

Having injected the DoS attack on the communication network of two nonconsecutive vehicles, the follower car fails to receive updated data from its neighbor. It should be noted that what distinguishes the intelligent intruder from an intrinsic network time delay is that the data received after a huge time delay is no longer useful to generate the correct control input. To combat this attacker, we propose to integrate an Unscented Kalman Filter (UKF) within our Secure–DNMPC such that the receiver can estimate the missing data and feed the predicted values to the NMPC controller. Consequently, the NMPC controller ignores the delayed states and makes use of the predicted values as long as the attack is running. We refer to this mode of the controller as the Secure–DNMPC–UKF mode. The controller is then switched back to Secure–DNMPC once either the attack is over or the injected time delay falls below the specified threshold.

Embedding the UKF within our design takes us one more step closer to a more realistic vehicle platoon system. In particular, through our proposed co-design, we can take the process and sensor noise into account as well, which is of high importance, especially for measurement sensors. From one side, assuming ideal non-noisy sensors, as done in most of the existing works in the literature, is a contrived assumption. On the other hand, the signals sent through the environment from one vehicle to another will be most likely compromised by some noise due to surrounding weather and road conditions.

The reason for choosing UKF over Extended Kalman Filter (EKF) is to avoid the propagation of the state distribution approximation error through the system dynamics caused by the first-order linearization performed in EKF. This is vital in terms of ensuring the safety of the platoon as the propagated error in the true posterior mean and covariance of the transformed Gaussian random variable may be large and cause unsafe driving behavior. Remarkably, adopting UKF does not impose anymore computational burden compared to EKF. The interested reader is referred to [63] for more details on the superiority of UKF over EKF for nonlinear state estimation. Fig. 4 shows a flowchart illustrating the procedure of the Secure–DNMPC–UKF co-design.

Before delving into the details of the Secure–DNMPC–UKF algorithm, a quick overview of the basics of Unscented Kalman Filtering is given in the following.

? Principles of Unscented Kalman Filtering

Unscented Kalman Filter, as a nonlinear state observer, basically relies on the unscented transformation to capture the statistical properties of state estimates via nonlinear functions. The observer initially captures the mean and covariance of the state estimates through a set of so-called sigma points. The algorithm makes use of those sigma points as the inputs of the process and measurement functions to generate

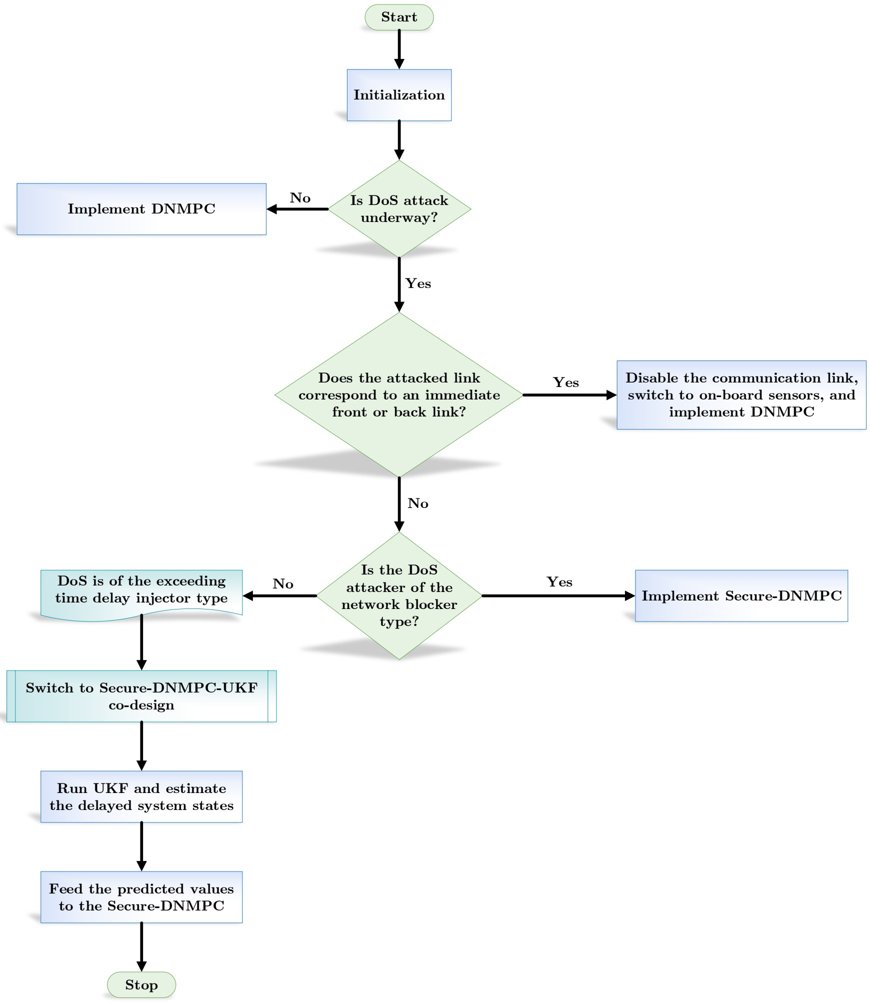

Start
Initialization
Is DoS attack underway?
No
Implement DNMPC
Yes
Does the attacked link correspond to an immediate front or back link?
Yes
Disable the communication link, switch to on-board sensors, and implement DNMPC
No
Is the DoS attacker of the network blocker type?
Yes
Implement Secure-DNMPC
No
DoS is of the exceeding time delay injector type
Switch to Secure-DNMPC-UKF co-design
Run UKF and estimate the delayed system states
Feed the predicted values to the Secure-DNMPC
Stop

Fig. 4: Procedure of the proposed Secure–DNMPC–UKF co-design

a new set of states. Subsequently, a set of state estimates and state estimation error covariance are obtained using the mean and covariance of the previously transformed points.

Let us consider an n-state nonlinear system described by the following nonlinear state transition and measurement functions comprised by additive zero-mean process noise $\boldsymbol{w}[k] \sim (0, \boldsymbol{Q}[k])$ and measurement noise $\boldsymbol{v}[k] \sim (0, \boldsymbol{R}[k])$

$$\begin{cases} \boldsymbol{x}[k+1] = f(\boldsymbol{x}[k], u_s[k]) + \boldsymbol{w}[k] \\ \quad\;\; \boldsymbol{y}[k] = h(\boldsymbol{x}[k], u_m[k]) + \boldsymbol{v}[k] \end{cases} \tag{7}$$

The filter takes the following steps to obtain the state estimates and the state estimation error covariance

1) The filter is initialized with an initial value for state $x[0]$ and state estimation error covariance matrix P

$$\hat{\boldsymbol{x}}[0|-1] = \mathbb{E}(\boldsymbol{x}[0]) \tag{8}$$

$$P[0|-1] = \mathbb{E}[(\boldsymbol{x}[0] - \hat{\boldsymbol{x}}[0] - 1)(\boldsymbol{x}[0] - \hat{\boldsymbol{x}}[0] - 1)^\top] \tag{9}$$

   where $\hat{\underline{x}}[k]$ is the state estimate at time k and $\hat{\boldsymbol{x}}[k_1|k_0]$ denotes the state estimate at time k_1 using the measurement data up to time k_0.

2) Having used the measurement data $\boldsymbol{y}[k]$ at each time instant k, the filter updates the state estimate and the state estimation error covariance:

   a) Choose the sigma points $\hat{\boldsymbol{x}}^{(i)}[k|k-1]$ at time k

$$\hat{\boldsymbol{x}}^{(0)} = \hat{\boldsymbol{x}}[k|k-1] \tag{10}$$

$$\hat{\boldsymbol{x}}^{(i)}[k|k-1] = \hat{\boldsymbol{x}}[k|k-1] + \Delta\boldsymbol{x}^{(i)}, \quad i = 1, 2, \ldots, 2n \tag{11}$$

$$\Delta\boldsymbol{x}^{(i)} = (\sqrt{cP[k|k-1]})_i, \quad i = 1, 2, \ldots, n \tag{12}$$

$$\Delta\boldsymbol{x}^{(n+i)} = -(\sqrt{cP[k|k-1]})_i, \quad i = 1, 2, \ldots, n \tag{13}$$

   where $c = \alpha^2(n+\kappa)$ is a scaling factor and $(\sqrt{cP})_i$ is the i-th column of the $\sqrt{cP}$ matrix [64].

   b) For each of the sigma points, use the nonlinear measurement function to compute the predicted measurements

$$\hat{\boldsymbol{y}}^{(i)}[k|k-1] = h(\hat{\boldsymbol{x}}^{(i)}[k|k-1], u_m[k]), \quad i = 1, 2, \ldots, 2n \tag{14}$$

   c) In order to obtain the predicted measurement at time k, integrate the predicted measurements

$$\hat{\boldsymbol{y}}[k] = \Sigma_{i=0}^{2n} W_n^{(i)} \hat{\boldsymbol{y}}^{(i)}[k|k-1] \tag{15}$$

$$W_n^{(0)} = 1 - \frac{n}{\alpha^2(n+\kappa)} \tag{16}$$

$$W_n^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \ldots, 2n \tag{17}$$

   d) By adding the measurement noise $\boldsymbol{R}[k]$, estimate the covariance matrix of the predicted measurement

$$P_{\boldsymbol{y}} = \Sigma_{i=0}^{2n} W_c^{(i)} (\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k])(\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k])^\top + \boldsymbol{R}[k] \tag{18}$$

$$W_c^{(0)} = (2 - \alpha^2 + \beta) - \frac{n}{\alpha^2(n+\kappa)} \tag{19}$$

$$W_c^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \ldots, 2n \tag{20}$$

For the details on effects of parameters α, β, and κ the reader is referred to [64].

e) Estimate the cross-covariance between $\hat{\boldsymbol{x}}[k|k-1]$ and $\hat{\boldsymbol{y}}[k]$

$$P_{\boldsymbol{xy}} = \frac{1}{2\alpha^2(n+\kappa)} \Sigma_{i=1}^{2n} (\hat{\boldsymbol{x}}^{(i)}[k|k-1] - \hat{\boldsymbol{x}}[k|k-1])(\hat{\boldsymbol{y}}^{(i)}[k|k-1] - \hat{\boldsymbol{y}}[k|k-1])^\top \tag{21}$$

Note that $\hat{\boldsymbol{x}}^{(0)}[k|k-1] - \hat{\boldsymbol{x}}[k|k-1] = 0$.

f) Compute the estimated state and state estimation error covariance at time step k

$$K = P_{\boldsymbol{xy}} P_{\boldsymbol{y}}^{-1} \tag{22}$$

$$\hat{\boldsymbol{x}}[k|k] = \hat{\boldsymbol{x}}[k|k-1] + K(\boldsymbol{y}[k] - \hat{\boldsymbol{y}}[k]) \tag{23}$$

$$P[k|k] = P[k|k-1] - KP_{\boldsymbol{y}} K_k^\top \tag{24}$$

where K is the Kalman gain.

3) Now the state and state estimation error covariance can be predicted at time instant $k+1$

a) Choose the sigma points $\hat{\boldsymbol{x}}^{(i)}[k|k]$ at time instant k.

$$\hat{\boldsymbol{x}}^{(0)}[k|k] = \hat{\boldsymbol{x}}[k|k] \tag{25}$$

$$\hat{\boldsymbol{x}}^{(i)}[k|k] = \hat{\boldsymbol{x}}[k|k] + \Delta\boldsymbol{x}^{(i)}, \quad i = 1, 2, \ldots, 2n \tag{26}$$

$$\Delta\boldsymbol{x}^{(i)} = (\sqrt{cP[k|k]})_i, \quad i = 1, 2, \ldots, n \tag{27}$$

$$\Delta\boldsymbol{x}^{(n+i)} = -(\sqrt{cP[k|k]})_i, \quad i = 1, 2, \ldots, n \tag{28}$$

$$\tag{29}$$

b) In order to get the predicted states at time $k+1$, combine the predicted states

$$\hat{\boldsymbol{x}}[k+1|k] = \Sigma_{i=0}^{2n} W_n^{(i)} \hat{\boldsymbol{x}}^{(i)}[k+1|k] \tag{30}$$

$$W_n^{(0)} = 1 - \frac{n}{\alpha^2(n+\kappa)} \tag{31}$$

$$W_n^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \ldots, 2n \tag{32}$$

4) To account for the process noise, add $\boldsymbol{Q}[k]$ and compute the covariance of the predicted state

$$P[k+1|k] = \Sigma_{i=0}^{2n} W_c^{(i)} (\hat{\boldsymbol{x}}^{(i)}[k+1|k] - \hat{\boldsymbol{x}}[k+1|k])(\hat{\boldsymbol{x}}^{(i)}[k+1|k] - \hat{\boldsymbol{x}}[k+1|k])^\top + \boldsymbol{Q}[k] \tag{33}$$

$$W_c^{(0)} = (2 - \alpha^2 + \beta) - \frac{n}{\alpha^2(n+\kappa)} \tag{34}$$

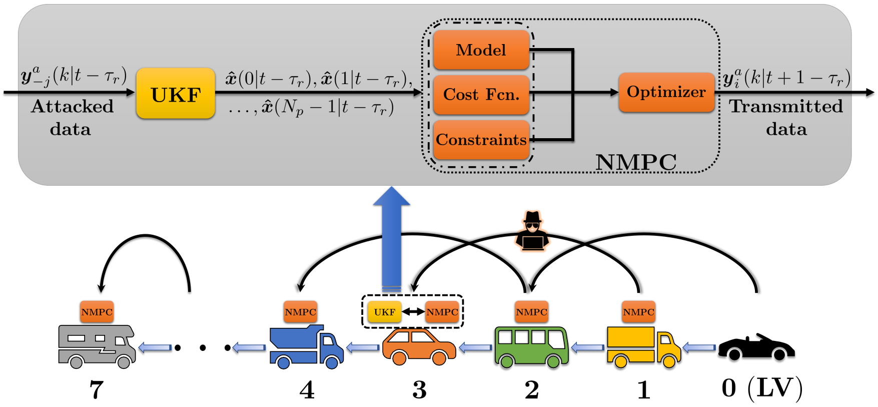


Fig. 5: Schematic of the proposed Secure–DNMPC–UKF co-design

$$W_c^{(i)} = \frac{1}{2\alpha^2(n+\kappa)}, \quad i = 1, 2, \dots, 2n \tag{35}$$

For more details on the observer for the case of non-additive process/measurement noise, please see [64].

The proposed algorithm is summarized in Algorithm 1, which is the extended version of the authors' previous work [65] for static platoons. We further note that $\boldsymbol{y}_i^a(t)$ represents the data sent by the vehicle i to the set $\mathbb{O}_i$ while $\boldsymbol{y}_{-j}^a$ denotes the data received by the vehicle i from its neighbors $j \in \mathbb{N}_i$. Superscript a, p, and $*$ are to distinguish between assumed, predicted, and optimal quantities, respectively. The assumed quantities are the ones transmitted by the vehicles in the platoon. Fig. 5 illustrates the Secure–DNMPC–UKF co-design in which $\hat{\boldsymbol{x}}(k|t)$ denotes the estimated state at time instant k using the measured data up to time t.

3.3 Stability Analysis of Secure–DNMPC

In this section we study the stability of the Secure–DNMPC algorithm incorporating the time delay τ imposed by the DoS attacker. Prior to stability analysis, let us first introduce the following Lemma.

Lemma 1 ([58]) *For any platoon wherein all the vehicles can receive data (directly/indirectly) from the leader vehicle, the eigenvalues of $(\mathcal{D}+\mathcal{P})^{-1}\mathcal{A}$ lie within the unit circle disk, i.e.*

$$\left|\lambda_i\left\{(\mathcal{D}+\mathcal{P})^{-1}\mathcal{A}\right\}\right| < 1. \tag{36}$$

Now, we can prove the stability of the Secure–DNMPC algorithm.

Algorithm 1 Secure–DNMPC–UKF for Dynamic Nonlinear Heterogeneous Vehicle Platooning under DoS Attack

```
 1: Initialization:
    Assumed values for vehicle $i$ are set at time $t = 0$,
        $u_i^a(k|0) = h_i(v_i(0)), \boldsymbol{y}_i^a(k|0) = \boldsymbol{y}_i^p(k|0), \quad k = 0, 1, \dots, N_p - 1$
 2: while $t \leq t_{\text{final}}$ do
 3:     Cut-in/cut-out CHECK                        ▹ Check to see if cut-in/cut-out occurred
 4:     Adjust data send-to/receive-from vehicles based on the occurred cut-in/cut-out
 5:     if $p_{-j}^a(t) = p_{-j}^a(t-1), j \in \mathbb{N}_i$ then          ▹ Check to see if a DoS is underway
 6:         if $j = i - 1$ or $j = i + 1$ then               ▹ Check to see if the attacked link
 7:                                          corresponds to a predecessor or a follower
 8:             Disable communication link, switch to on-board sensors, $\tau \leftarrow 0$, and Go to: 13
 9:         else
10:             if Attacker blocks the communication link then ▹ Check to see if the attacker is of
11:                                                                     the blockage type
12:                 $\tau \leftarrow \tau_a$
13:                 for Each vehicle $i$ do                          ▹ Implement Secure–DNMPC
14:                     Solve Problem 5 at time $t > 0$ and yield $u_i^*(k|t-\tau), \; k = 0, 1, \dots, N_p - 1$
15:                     Compute: $\begin{cases} \boldsymbol{x}_i^*(k+1|t-\tau) = \boldsymbol{\phi}_i(\boldsymbol{x}_i^*(k|t-\tau)) + \boldsymbol{\psi}_i u_i^*(k|t-\tau), \\ \boldsymbol{x}_i^*(0|t-\tau) = \boldsymbol{x}_i(t-\tau), \quad k = 0, 1, \dots, N_p - 1 \end{cases}$
16:                     Compute: $u_i^a(k|t-\tau+1) = \begin{cases} u_i^*(k+1|t-\tau), & k = 0, 1, \dots, N_p - 2 \\ h_i\left(v_i^*(N_p|t-\tau)\right), & k = N_p - 1 \end{cases}$
17:                     Compute: $\begin{cases} \boldsymbol{x}_i^a(k+1|t-\tau+1) = \boldsymbol{\phi}_i\left(\boldsymbol{x}_i^a(k|t-\tau+1)\right) + \boldsymbol{\psi}_i u_i^a(k|t-\tau+1) \\ \boldsymbol{x}_i^a(0|t-\tau+1) = \boldsymbol{x}_i^*(1|t-\tau), \quad k = 0, 1, \dots, N_p - 1 \end{cases}$
18:                     Compute: $\boldsymbol{y}_i^a(k|t-\tau+1) = \boldsymbol{\gamma} \boldsymbol{x}_i^a(k|t-\tau+1), \quad k = 0, 1, \dots, N_p - 1$
19:                     Send $\boldsymbol{y}_i^a(k|t-\tau+1)$ to the vehicles lie in the set $\mathbb{O}_i$, and receive $\boldsymbol{y}_{-j}^a(k|t-\tau+1)$ from neighboring vehicles $j \in \mathbb{N}_i$ and compute $\boldsymbol{y}_{\text{des},i}(k|t-\tau+1)$
20:                     Exert the first element of the optimal control signal $u_i(t-\tau) = u_i^*(0|t-\tau)$
21:                 end for
22:             else if Attacker injects exceeding delay $\tau_r > \delta$ then ▹ Check to see if the attacker
23:                                                  is of the exceeding time delay injector type
24:                 $\tau \leftarrow \tau_r$
25:                 Switch to Secure–DNMPC–UKF mode
26:                 Estimate the delayed states via UKF
27:                 Implement the Secure–DNMPC using the predicted states coming from the UKF
28:             end if
29:         end if
30:     end if
31: end while
```

Theorem 1 ([65]) *If a platoon which is under a DoS attack satisfies the condition in Lemma 1, then the terminal output of the system controlled by the Secure–DNMPC proposed in Algorithm 1 asymptotically converges to the desired state, i.e.*

$$\lim_{t \to \infty} \left| \boldsymbol{y}_i^p(N_p|t-\tau) - \boldsymbol{y}_{des,i}(N_p|t-\tau) \right| = 0. \tag{37}$$

4 Dynamic Platoon Control: Handling Cut-in/Cut-out Maneuvers

In this section, we consider a dynamic heterogeneous platoon wherein arbitrary vehicle(s) might perform cut-in/cut-out maneuvers. Here, we demonstrate the ability of the proposed algorithm to handle dynamic maneuvers while the platoon is subject to the cyber-attack.

First, we consider a secure dynamic heterogeneous platoon and prove some results based on which we extend the results to an insecure platoon. Assume there exist N_{ci} cut-in and N_{co} cut-out maneuvers in total while the number of initial FVs in the platoon is N. Let $\mathcal{N}_{ci} := \{1, \dots, N_{ci}\}$ and $\mathcal{N}_{co} := \{1, \dots, N_{co}\}$. We denote the time of the i-th cut-in and the j-th cut-out maneuvers by $t_{ci,i}$ and $t_{co,j}$, respectively. The following theorem determines the time of convergence of a dynamic platoon including possible cut-in and cut-out maneuvers.

Lemma 2 ([58, Theorem 2]) *If Assumption 1 is satisfied, then Problem (5) guarantees convergence of the output to the desired output in at most N time steps, i.e., $\boldsymbol{y}_i^p(N_p|t) = \boldsymbol{y}_{des,i}(N_p|t), \forall t \geq N$, for a static platoon (without any dynamic maneuvers).*

Theorem 2 *When having cut-in and/or cut-out maneuvers in a secured dynamic platoon, if Assumption 1 is satisfied, the Problem (5) guarantees convergence of the output to the desired output in at most*

$$\begin{aligned} t_{conv,\ secure} := &\max_{i,j} \left[t_{ci,i}, t_{co,j} \mid \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj} \right] \\ &+ N + N_{ci} - N_{co}, \end{aligned} \tag{38}$$

time steps, i.e., $\boldsymbol{y}_i^p(N_p|t) = \boldsymbol{y}_{des,i}(N_p|t), \forall t \geq t_{conv,\ secure}$.

Proof Let $\mathcal{L} := \mathcal{D} - \mathcal{A}$ be the Laplacian matrix of the underlying platoon graph topology. When a new cut-in or cut-out occurs, some new chaos is introduced to the system so we can consider the latest cut-in/cut-out maneuver. Considering the latest cut-in, one vehicle is added to the number of existing vehicles, let it be N. If the platoon graph is unidirectional and satisfies Assumption 1, the new $\mathcal{A} \in \mathbb{R}^{(N+1)\times(N+1)}$ is a lower-triangular matrix. Moreover, according to [58, Lemma 4], we have $\mathcal{D} + \mathcal{P} > 0$, yielding the eigenvalues of $(\mathcal{D} + \mathcal{P})^{-1}\mathcal{A}$ to be zero and this matrix to be nilpotent with degree at most $N + 1$. Based on [58, Lemma 1] and [58, Theorem 1], $\boldsymbol{y}_i^p(N_p|t)$ converges to the desired output in at most $N + 1$ steps. Extending this to N_{ci} cut-in maneuvers requires $N + N_{ci}$ time steps after the latest cut-in. Similar analysis can be performed for the cut-out maneuvers, resulting in $N - N_{co}$ time steps after the latest cut-out because the number of vehicles has been reduced. In general, having N_{ci} cut-in and N_{co} cut-out maneuvers will need $N + N_{ci} - N_{co}$ time steps after the latest maneuver which can be formulated as $\max_{i,j}[t_{ci,i}, t_{co,j} \mid \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj}]$. □

Corollary 1 *Lemma 2, for the static platoon, is a special case of Theorem 2 which is for a dynamic platoon.*

Proof When neither cut-in nor cut-out happen, the time of convergence is $t_{\text{conv, secure}} = 0 + N + 0 + 0 = N$ according to Theorem 2. □

Special Cases

Four special cases of the dynamic platoon are as follows:

E.g. 1) One cut-in happens at $t = 0$ and one cut-out happens at $t = N$: According to Theorem 2, the platoon converges in $t = N + N + 1 - 1 = 2N$. It is correct because before the cut-out, the platoon contains $N + 1$ vehicles until time N. When cut-out happens, the platoon is changed to a platoon with N vehicles which converges in N time steps according to Lemma 2.

E.g. 2) One cut-out happens at $t = 0$ and one cut-in happens at $t = N$: According to Theorem 2, the platoon converges in $t = N + N + 1 - 1 = 2N$, which is correct because in $t \in [0, N]$, the platoon includes $N - 1$ vehicles until time N. When cut-in happens, the platoon is modified to a platoon with N vehicles which converges in N time steps according to Lemma 2.

E.g. 3) Both cut-in and cut-out happen at $t = 0$: According to Theorem 2, the platoon converges in $t = 0 + N + 1 - 1 = N$, which is correct because the platoon includes N vehicles which converges in N time steps according to Lemma 2.

E.g. 4) Both cut-in and cut-out happen at $t = N$: According to Theorem 2, the platoon converges in $t = N + N + 1 - 1 = 2N$, which is correct because in $t \in [0, N]$, the platoon includes N vehicles. After the cut-in/cut-out actions the platoon still includes N vehicles which converges in N time steps according to Lemma 2.

Corollary 2 *When having cut-in and/or cut-out maneuvers in an insecure dynamic platoon, if Assumption 1 is satisfied, the convergence time of the output to the desired output is upper bounded by* $t_{\textit{conv, secure}} + \max\{\tau_r, \tau_a\}$*, i.e.,*[2]

$$\begin{aligned} t_{\textit{conv, insecure}} \leq & \max_{i,j}\left[t_{ci,i}, t_{co,j} \mid \forall i \in \mathcal{N}_{ci}, \forall j \in \mathcal{N}_{cj}\right] \\ & + N + N_{ci} - N_{co} + \max\{\tau_r, \tau_a\}, \end{aligned} \tag{39}$$

time steps, i.e., $\boldsymbol{y}_i^p(N_p|t-\tau) = \boldsymbol{y}_{des,i}(N_p|t-\tau), \forall t \geq t_{\textit{conv, insecure}}$.

5 Simulation Results

A heterogeneous platoon consisted of seven different vehicles is considered where they can exchange inter-vehicular data among each other through the TPF commu-

[2] Although, this upper bound might be conservative in some cases (such as in the scenario studied in subsection 5.2), it provides a safe margin for the convergence time of the controller.

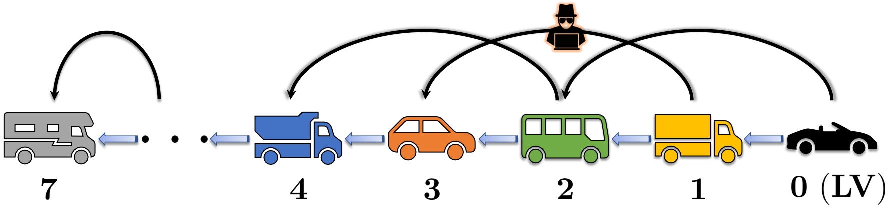


Fig. 6: TPF heterogeneous attacked vehicle platoon with a leader and 7 followers

Table 1: Parameters of the participating vehicles in the platoon

Vehicle index	m_i (kg)	τ_i (sec)	$C_{A,i}$ (N sec^2 m^{-2})	r_i (m)
1	1035.7	0.51	0.99	0.30
cut-in	1305.9	0.63	1.00	0.40
2	1849.1	0.75	1.15	0.38
3	1934.0	0.78	1.17	0.39
4	1678.7	0.70	1.12	0.37
5	1757.7	0.73	1.13	0.38
6	1743.1	0.72	1.13	0.37
7	1392.2	0.62	1.06	0.34

nication topology. It is assumed that the communication link connecting the vehicle 1 and 3 is subject to a DoS attack. Therefore, vehicle 3 cannot receive real-time data, including the position and velocity of vehicle 1 while the attack is performing (see Fig. 6). Remarkably, to emulate a practical scenario, based on Assumption 2 the external intruder is only able to cause communication degradation among the vehicles for a finite time period. In the following simulations, the DoS attacker starts jamming the communication link from vehicle 1 to 3. Seven different vehicles with realistic parameters form the platoon wherein the leader vehicle starts driving at $v_0(0) = 20m/s$ for one second, then it accelerates to reach $v_0(2) = 22m/s$ and continues with this velocity until the end of the simulation. The prediction horizon and desired spacing among consecutive vehicles have been chosen as $N_p = 20$, and $d = 10$ meters, respectively. The parameters of the participating vehicles in the platoon are listed in Table 1, which is in accordance with [66]. We have extended the code in [67] for our security analysis.

! Remark

To select an appropriate value for the prediction horizon, one has to notice as τ increases, N_p needs to be decreased in order to let the vehicles have enough time to exchange and update their data prior to the attack occurrence. On the other hand, too small values for N_p results in frequent rapid oscillations in the control input which makes the controller unimplementable in practice.

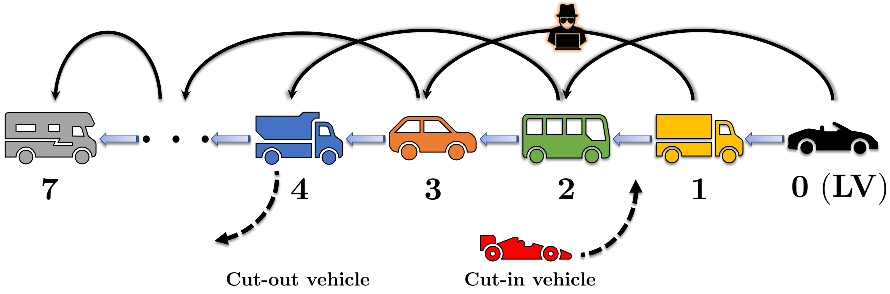


Fig. 7: TPF dynamic heterogeneous attacked vehicle platoon with cut-in and cut-out vehicles

5.1 DoS Attack Modeled as a Network Blocker

In this part, we take one more step to effectively control the dynamic heterogeneous platoon endangered by an intelligent DoS intruder. As was previously described, the attacker could jam the communication network among any two nonconsecutive vehicle to prohibits a follower vehicle from receiving updated data. Having made an expressive scenario incorporating both cut-in and cut-out actions while taking into account a DoS attack, we consider a same setting for the attacked platoon presented in the previous section except assuming a vehicle merges with the platoon at $t = 2$sec to be placed in front of the second FV. Furthermore, we let the fourth FV to perform a cut-out action at $t = 4$sec (see Fig. 7). We note that the desired distance among the vehicles ($d = 10$ meters) provides enough space for a regular vehicle to cut-in. The attack happens on the communication environment among the first and third FVs in the time interval $t \in [3, 6]$. Although these tight actions might not seem to happen in practice, they are chosen to challenge the algorithm largely. Fig. 8 demonstrate the driving quantities of the respective platoon.

From Fig. 8a, one can see that despite the blockage of the data transfer link from vehicle 1 to 3, there is no collision in the platoon, and the safety has been ensured. Besides, Fig. 8b reveals that the Secure–DNMPC algorithm effectively mitigates the DoS attack and the followers begin to keep tracking the leader's speed profile shortly after the attack is over. Convergence of torque and acceleration are also demonstrated in Fig. 8c, and 8d. It is worth mentioning that by reducing its speed, the second FV has increased its gap with the first FV to make the desired distance of 10m for the cut-in vehicle. Consequently, the following vehicles have lessened their velocity to keep the desired distance. Fig. 8b verifies this fact. A similar analysis exists for the cut-out maneuver where the following vehicles have increased their velocity to reach the desired distance from the vehicles in front.

As one can see, the spacing and speed tracking objectives have been safely fulfilled. To have a clearer look at the spacing objective, Fig. 9 shows the magnified absolute positions and the spacing error of consecutive vehicles. Since all the spacing

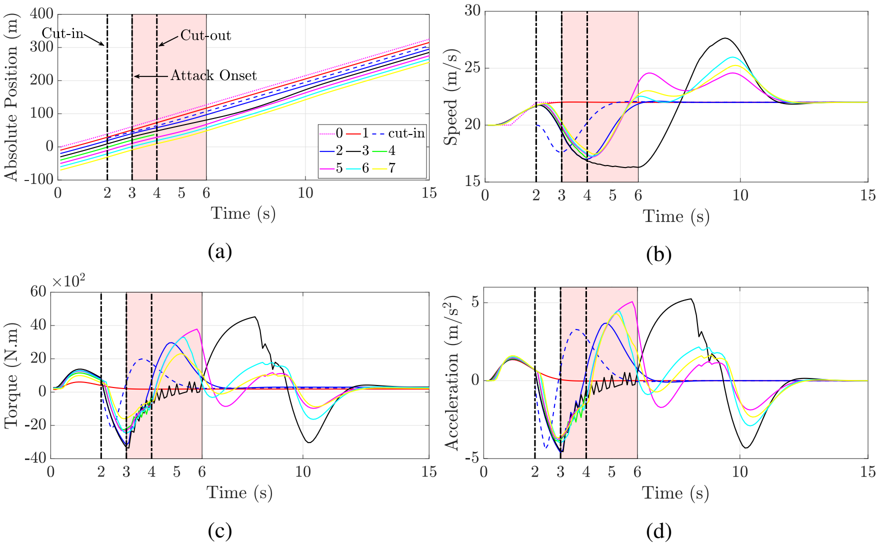


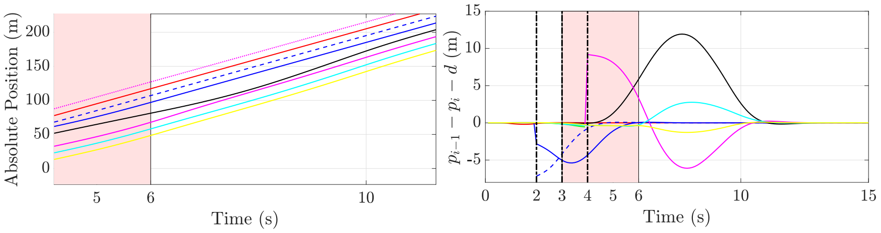
Fig. 8: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC

errors in Fig. 9b are greater than −10 meters, no collision has occurred. Moreover, the relative spacing error shows jumps in the distance error (blue and purple curves) because of the cut-in/cut-out maneuvers.[3] As expected, the spacing error for the cut-out maneuver (purple curve in Fig. 9b) has an opposite sign with respect to the cut-in error (blue curve in Fig. 9b). Furthermore, we see that convergence has been reached in less than 14s which coincides with Corollary 2 because $t_{\text{conv, insecure}} \leq \max(2, 4) + 7 + 1 - 1 + 3 = 14$s.

5.2 DoS Attack Modeled as an Exceeding Time Delay Injection in the Data Transmission

Inherent communication delay of standard 802.11p-based DSRC network ranges from tens to hundreds of milliseconds [68–70]. Here, to ensure modeling a highly devastating attacker, we assume the time delay imposed by the intruder is $\tau_r = 2.5$sec. In addition, non-ideal sensors are assumed in the simulations, i.e., an additive zero-mean white Gaussian noise with variance $\sigma^2 = 0.01$ is considered on both the

[3] Note that the jump in the relative spacing error of the third FV (black curve) is due to the DoS attack.



(a)

(b)

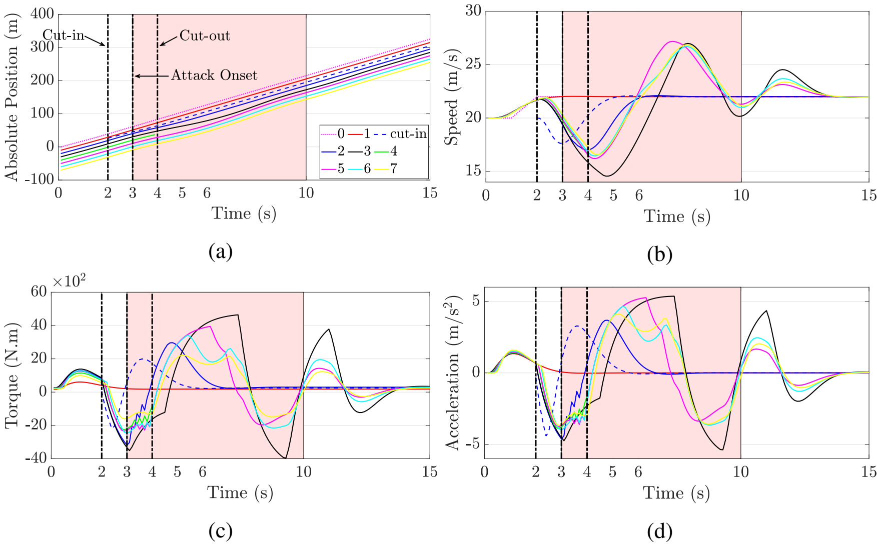
Fig. 9: (a) Magnified absolute position and (b) spacing error of the TPF dynamic heterogeneous DoS attacked platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC

position and velocity sensors.[4] To challenge more the algorithm we introduce a severer attack which happens for a longer period of time, i.e., in the time range $t \in [3, 10]$. It is worth noting that cut-in and cut-out maneuvers are still in effect at $t = 2$sec and $t = 4$sec, respectively. Fig. 10 shows the performance of the proposed co-design controller on the attacked platoon with cut-in and cut-out actions. As is demonstrated by the driving quantities, safe distance and velocity tracking requirements have been fulfilled. Furthermore, the convergence has been reached in less than 18s which again verifies Corollary 2 because $t_{\text{conv, insecure}} \le \max(2, 4) + 7 + 1 - 1 + \max(2.5, 7) = 18$s. It would also be insightful to compare the results to the case where UKF is not embedded in the design. Fig. 11 demonstrates the resulting driving behavior when only relying on the controller leaving out the estimation phase. Occurring collision and violating the control objectives clearly prove the critical role of the observer design.

It is noticeable that by comparing the previous scenarios (Figs. 8, 10, and 11), it reveals that embedding the UKF within our controller design, also has the advantage of reducing the oscillations in the control input caused by the cyber attack. This generation of a smoother control input enhances the driving comfort in practice.

We highlight that the proposed algorithm has also been successfully tested on different platoon formations such as Two-Predecessor Leader Follower (TPLF), with different spacing policies such as Constant Time Headway (CTH) policy, and also on Federal Test Procedure (FTP) drive cycle to emulate urban driving.

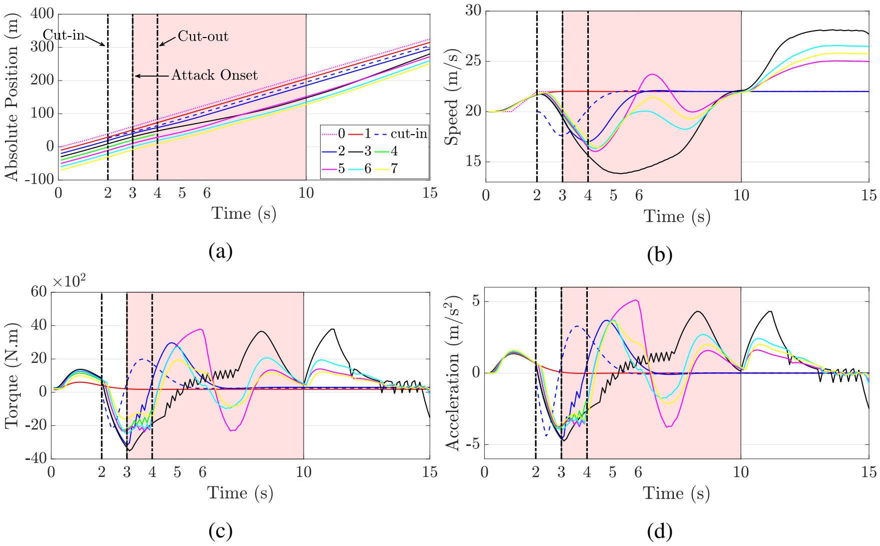
[4] This could also be considered as the environment effects on the transmitted signals. Modeling the environment effect with white Gaussian noise in V2V communications is widely used in literature [71, 72].



(a) (b)

(c) (d)

Fig. 10: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked platoon with cut-in/cut-out maneuvers equipped by Secure–DNMPC–UKF co-design



(a) (b)

(c) (d)

Fig. 11: (a) Absolute position, (b) speed, (c) torque, and (d) acceleration of the TPF dynamic heterogeneous DoS attacked platoon with cut-in/cut-out maneuvers without UKF design

6 Conclusion and Future Directions

This chapter dealt with a broadly concerned control problem, namely the dynamic heterogeneous platoon control. A platoon mainly consists of networking and data transmission among the vehicles, forming the cyber layer, and the physical environment composed of the participant cars, forming the physical layer. This cyber-physical system is highly prone to cyber-attacks endangering the wireless connectivity among the vehicles. This vulnerability to external attackers needs to be fully addressed as an insecure communication layer in a platoon can cause manipulated and/or missing data received by the followers resulting in dangerous hazards. In this chapter, we focused on the widespread so-called DoS attack in which the intelligent intruder targets the wireless links by overwhelming the node by invalid requests, hence, either blocks the network or prevents it from timely data transfer. We proposed a Secure–Distributed Nonlinear Model Predictive Control (Secure–DNMPC) framework to ensure a safe and secure dynamic platooning which fulfills both the safe distancing between the cars and speed tracking requirements. The method is capable of handling cut-in/cut-out maneuvers under the premise of the existence of a cyber DoS attack. The algorithm is basically comprised of detection and mitigation phases.

Furthermore, we introduced a novel Secure–DNMPC–UKF co-design for the case when the DoS attacker injects a huge amount of time delay in the network compared to the intrinsic practical DSRC time delay. This makes use of the available but outdated data to estimate and predict future states. The proposed approach also provides the opportunity to consider non-ideal sensors which contaminate the measured data. In addition, compromised signals sent through a realistic noisy environment can be considered as well. Simulation results demonstrated the efficacy of the introduced technique. As a future direction, one can think of generalizing the given algorithm to a multi-platooning scenario in which two or more attacked platoons drive in parallel, and arbitrary vehicles wish to exit their own platoon and merge with an adjacent one. Also, other types of attacks and the corresponding countermeasures could be considered.

Acknowledgment

The authors would like to thank Dr. Yang Zheng (SEAS and CGBC at Harvard University, Cambridge, MA, USA) for the fruitful discussions during this work. Also, the authors would like to thank the support from Natural Sciences and Engineering Research Council (NSERC) of Canada.

References

1. Wang, X.: Cyber-Physical Systems: A Reference. Springer-Verlag Berlin Heidelberg (2021)
2. Song, H., Fink, G., Jeschke, S.: Security and Privacy in Cyber-Physical Systems. Wiley Online Library (2017)
3. Guo, S., Zeng, D.: Cyber-Physical Systems: Architecture, Security and Application. Springer (2019)
4. Koç, Ç.K.: Cyber-Physical Systems Security. Springer (2018)
5. Liu, H., Niu, B., Li, Y.: False-data-injection attacks on remote distributed consensus estimation. IEEE Transactions on Cybernetics (2020)
6. Humayed, A., Lin, J., Li, F., Luo, B.: Cyber-physical systems security – A survey. IEEE Internet of Things Journal **4**(6) (2017) 1802–1831
7. Ashibani, Y., Mahmoud, Q.H.: Cyber physical systems security: Analysis, challenges and solutions. Computers & Security **68** (2017) 81–97
8. Ali, S., Al Balushi, T., Nadir, Z., Hussain, O.K.: Cyber Security for Cyber Physical Systems. Volume 768. Springer (2018)
9. Basiri, M.H., Azad, N.L., Fischmeister, S.: Distributed time-varying kalman filter design and estimation over wireless sensor networks using owa sensor fusion technique. In: 2020 28th Mediterranean Conference on Control and Automation (MED), IEEE (2020) 325–330
10. Langner, R.: Stuxnet: Dissecting a cyberwarfare weapon. IEEE Security & Privacy **9**(3) (2011) 49–51
11. Farwell, J.P., Rohozinski, R.: Stuxnet and the future of cyber war. Survival **53**(1) (2011) 23–40
12. Liu, Y., Ning, P., Reiter, M.K.: False data injection attacks against state estimation in electric power grids. ACM Transactions on Information and System Security (TISSEC) **14**(1) (2011) 13
13. Shoukry, Y., Martin, P., Tabuada, P., Srivastava, M.: Non-invasive spoofing attacks for anti-lock braking systems. In: International Workshop on Cryptographic Hardware and Embedded Systems, Springer (2013) 55–72
14. Singh, S.: Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report (2015)
15. Transport Canada: Canadian motor vehicle traffic collision statistics. (2014)
16. Road Safety in Canada. (2011) [Online]. Available: http://www.tc.gc.ca/eng/motorvehiclesafety/tp-tp15145-1201.htm.
17. Godsmark, P., Kirk, B., Gill, V., Flemming, B.: Automated vehicles: The coming of the next disruptive technology. (2015)
18. Gao, F., Hu, X., Li, S.E., Li, K., Sun, Q.: Distributed adaptive sliding mode control of vehicular platoon with uncertain interaction topology. IEEE Transactions on Industrial Electronics **65**(8) (2018) 6352–6361
19. Li, Y., Tang, C., Peeta, S., Wang, Y.: Nonlinear consensus-based connected vehicle platoon control incorporating car-following interactions and heterogeneous time delays. IEEE Transactions on Intelligent Transportation Systems (2018)
20. Liu, P., Kurt, A., Ozguner, U.: Distributed model predictive control for cooperative and flexible vehicle platooning. IEEE Transactions on Control Systems Technology (99) (2018) 1–14
21. Zheng, Y., Li, S.E., Wang, J., Cao, D., Li, K.: Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies. IEEE Transactions on Intelligent Transportation Systems **17**(1) (2016) 14–26
22. Pirani, M., Hashemi, E., Simpson-Porco, J.W., Fidan, B., Khajepour, A.: Graph theoretic approach to the robustness of k-nearest neighbor vehicle platoons. IEEE Transactions on Intelligent Transportation Systems **18**(11) (2017) 3218–3224
23. Van Arem, B., Van Driel, C.J., Visser, R.: The impact of cooperative adaptive cruise control on traffic-flow characteristics. IEEE Transactions on Intelligent Transportation Systems **7**(4) (2006) 429–436
24. Rawat, D.B., Bajracharya, C.: Vehicular cyber physical systems. Technical report, Springer (2017)

25. Kargl, F., Papadimitratos, P., Buttyan, L., Müter, M., Schoch, E., Wiedersheim, B., Thong, T.V., Calandriello, G., Held, A., Kung, A., et al.: Secure vehicular communication systems: implementation, performance, and research challenges. IEEE Communications magazine **46**(11) (2008) 110–118
26. Koopman, P., Wagner, M.: Autonomous vehicle safety: An interdisciplinary challenge. IEEE Intelligent Transportation Systems Magazine **9**(1) (2017) 90–96
27. Amin, S., Cárdenas, A.A., Sastry, S.S.: Safe and secure networked control systems under denial-of-service attacks. In: International Workshop on Hybrid Systems: Computation and Control, Springer (2009) 31–45
28. Mo, Y., Sinopoli, B.: Secure control against replay attacks. In: Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on, IEEE (2009) 911–918
29. Mo, Y., Garone, E., Casavola, A., Sinopoli, B.: False data injection attacks against state estimation in wireless sensor networks. In: Decision and Control (CDC), 2010 49th IEEE Conference on, IEEE (2010) 5967–5972
30. Chabukswar, R., Sinopoli, B., Karsai, G., Giani, A., Neema, H., Davis, A.: Simulation of network attacks on SCADA systems. In: First Workshop on Secure Control Systems. (2010) 587–592
31. Basiri, M.H., Thistle, J.G., Simpson-Porco, J.W., Fischmeister, S.: Kalman filter based secure state estimation and individual attacked sensor detection in cyber-physical systems. In: 2019 American Control Conference (ACC), IEEE (2019) 3841–3848
32. Siegel, J.E., Erb, D.C., Sarma, S.E.: A survey of the connected vehicle landscape-architectures, enabling technologies, applications, and development areas. IEEE Transactions on Intelligent Transportation Systems **19**(8) (2018) 2391–2406
33. Zhu, Q., Basar, T.: Game-theoretic methods for robustness, security, and resilience of cyber-physical control systems: Games-in-games principle for optimal cross-layer resilient control systems. In: IEEE control systems. Volume 35. (2015) 45–65
34. Li, Y., Shi, L., Cheng, P., Chen, J., Quevedo, D.E.: Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach. IEEE Transactions on Automatic Control **60**(10) (2015) 2831–2836
35. Cardenas, A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., Sastry, S., et al.: Challenges for securing cyber physical systems. In: Workshop on future directions in cyber-physical systems security. Volume 5. (2009)
36. Fawzi, H., Tabuada, P., Diggavi, S.: Secure estimation and control for cyber-physical systems under adversarial attacks. IEEE Transactions on Automatic control **59**(6) (2014) 1454–1467
37. Cardenas, A.A., Amin, S., Sastry, S.: Secure control: Towards survivable cyber-physical systems. In: 2008 The 28th International Conference on Distributed Computing Systems Workshops, IEEE (2008) 495–500
38. Pasqualetti, F., Dörfler, F., Bullo, F.: Attack detection and identification in cyber-physical systems. IEEE transactions on automatic control **58**(11) (2013) 2715–2729
39. Liu, Y.C., Bianchin, G., Pasqualetti, F.: Secure trajectory planning against undetectable spoofing attacks. Automatica **112** (2020) 108655
40. Weerakkody, S., Liu, X., Son, S.H., Sinopoli, B.: A graph-theoretic characterization of perfect attackability for secure design of distributed control systems. IEEE Transactions on Control of Network Systems **4**(1) (2016) 60–70
41. Zhang, T., Zou, Y., Zhang, X., Guo, N., Wang, W.: Data-driven based cruise control of connected and automated vehicles under cyber-physical system framework. IEEE Transactions on Intelligent Transportation Systems (2020)
42. Basiri, M.H., Pirani, M., Azad, N.L., Fischmeister, S.: Security of vehicle platooning: A game-theoretic approach. IEEE Access **7**(1) (2019) 185565–185579
43. Parkinson, S., Ward, P., Wilson, K., Miller, J.: Cyber threats facing autonomous and connected vehicles: Future challenges. IEEE Transactions on Intelligent Transportation Systems **18**(11) (2017) 2898–2915
44. Petit, J., Shladover, S.E.: Potential cyberattacks on automated vehicles. IEEE Transactions on Intelligent transportation systems **16**(2) (2014) 546–556

45. Zhang, T., Antunes, H., Aggarwal, S.: Defending connected vehicles against malware: Challenges and a solution framework. IEEE Internet of Things journal **1**(1) (2014) 10–21
46. Biron, Z.A., Dey, S., Pisu, P.: Real-time detection and estimation of denial of service attack in connected vehicle systems. IEEE Transactions on Intelligent Transportation Systems **19**(12) (2018) 3893–3902
47. Lei, C., Van Eenennaam, E., Wolterink, W.K., Karagiannis, G., Heijenk, G., Ploeg, J.: Impact of packet loss on cacc string stability performance. In: 2011 11th International Conference on ITS Telecommunications, IEEE (2011) 381–386
48. Ploeg, J., Semsar-Kazerooni, E., Lijster, G., van de Wouw, N., Nijmeijer, H.: Graceful degradation of cacc performance subject to unreliable wireless communication. In: 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), IEEE (2013) 1210–1216
49. Azees, M., Vijayakumar, P., Deborah, L.J.: Comprehensive survey on security services in vehicular ad-hoc networks. IET Intelligent Transport Systems **10**(6) (2016) 379–388
50. Dadras, S., Gerdes, R.M., Sharma, R.: Vehicular platooning in an adversarial environment. In: Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ACM (2015) 167–178
51. Öncü, S., Ploeg, J., Van de Wouw, N., Nijmeijer, H.: Cooperative adaptive cruise control: Network-aware analysis of string stability. IEEE Transactions on Intelligent Transportation Systems **15**(4) (2014) 1527–1537
52. Qin, W.B., Orosz, G.: Experimental validation of string stability for connected vehicles subject to information delay. IEEE Transactions on Control Systems Technology (2019)
53. Qin, W.B., Gomez, M.M., Orosz, G.: Stability analysis of connected cruise control with stochastic delays. In: 2014 American Control Conference, IEEE (2014) 4624–4629
54. Laurendeau, C., Barbeau, M.: Threats to security in DSRC/WAVE. In: International Conference on Ad-Hoc Networks and Wireless, Springer (2006) 266–279
55. Teixeira, A., Shames, I., Sandberg, H., Johansson, K.H.: A secure control framework for resource-limited adversaries. Automatica **51** (2015) 135–148
56. Harfouch, Y.A., Yuan, S., Baldi, S.: An adaptive switched control approach to heterogeneous platooning with intervehicle communication losses. IEEE Transactions on Control of Network Systems **5**(3) (2017) 1434–1444
57. Dolk, V.S., Ploeg, J., Heemels, W.M.H.: Event-triggered control for string-stable vehicle platooning. IEEE Transactions on Intelligent Transportation Systems **18**(12) (2017) 3486–3500
58. Zheng, Y., Li, S.E., Li, K., Borrelli, F., Hedrick, J.K.: Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies. IEEE Transactions on Control Systems Technology **25**(3) (2017) 899–910
59. Zheng, Y., Li, S.E., Wang, J., Cao, D., Li, K.: Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies. IEEE Transactions on Intelligent Transportation Systems **17**(1) (2016) 14–26
60. Yuan, Y., Zhu, Q., Sun, F., Wang, Q., Başar, T.: Resilient control of cyber-physical systems against denial-of-service attacks. In: 2013 6th International Symposium on Resilient Control Systems (ISRCS), IEEE (2013) 54–59
61. Zhang, H., Cheng, P., Shi, L., Chen, J.: Optimal denial-of-service attack scheduling with energy constraint. IEEE Transactions on Automatic Control **60**(11) (2015) 3023–3028
62. Sun, Q., Zhang, K., Shi, Y.: Resilient model predictive control of cyber-physical systems under dos attacks. IEEE Transactions on Industrial Informatics (2019)
63. Wan, E.A., Van Der Merwe, R.: The unscented kalman filter for nonlinear estimation. In: Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373), IEEE (2000) 153–158
64. Simon, D.: Optimal state estimation: Kalman, H infinity, and nonlinear approaches. John Wiley & Sons (2006)
65. Basiri, M.H., Azad, N.L., Fischmeister, S.: Attack resilient heterogeneous vehicle platooning using secure distributed nonlinear model predictive control. In: 2020 28th Mediterranean Conference on Control and Automation (MED), IEEE (2020) 307–312

66. Wang, J.Q., Li, S.E., Zheng, Y., Lu, X.Y.: Longitudinal collision mitigation via coordinated braking of multiple vehicles using model predictive control. Integrated Computer-Aided Engineering **22**(2) (2015) 171–185
67. Zheng, Y.: DMPC for platoons. (2019) [Online]. Available: https://github.com/zhengy09/DMPC_for_platoons.
68. Yao, Y., Rao, L., Liu, X., Zhou, X.: Delay analysis and study of IEEE 802.11-p based DSRC safety communication in a highway environment. In: 2013 Proceedings IEEE INFOCOM, IEEE (2013) 1591–1599
69. Wang, Y., Duan, X., Tian, D., Lu, G., Yu, H.: Throughput and delay limits of 802.11-p and its influence on highway capacity. Procedia-Social and Behavioral Sciences **96** (2013) 2096–2104
70. Ma, X., Chen, X., Refai, H.H.: Performance and reliability of DSRC vehicular safety communication: a formal analysis. EURASIP Journal on Wireless Communications and Networking **2009** (2009) 1–13
71. Kukshya, V., Krishnan, H.: Experimental measurements and modeling for vehicle-to-vehicle dedicated short range communication (DSRC) wireless channels. In: IEEE Vehicular Technology Conference, IEEE (2006) 1–5
72. Sabouni, R., Hafez, R.M.: Performance of DSRC for V2V communications in urban and highway environments. In: 2012 25th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), IEEE (2012) 1–5