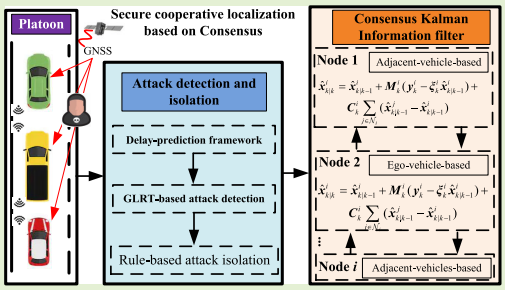
Secure Cooperative Localization for Connected Automated Vehicles Based on Consensus

Xin Xia, Runsheng Xu, and Jiaqi Ma

***Abstract*—In this article, we present secure cooperative localization for connected automated vehicles (CAVs) based on consensus estimation through leveraging shared but possibly attacked sensory information from multiple adjacent vehicles. First, the communication topology between the CAVs, node kinematic model, and node measurement model for each vehicle are introduced. Then, a consensus Kalman information filter (CKIF) is applied to fuse the shared information from connected vehicles. Since the sensory information might be attacked, an attack detection algorithm based on the generalized likelihood ratio test (GLRT) is adopted. A delay-prediction framework is proposed to maintain the accuracy and real-time performance of the detection algorithm. Next, a rule-based attack isolation method is used to defend the attack. Finally, the proposed secure cooperative localization algorithm is validated in extensive numerical simulation experiments. The results confirm that leveraging information from multiple vehicles in a cooperative manner leads to better accuracy and resilience for vehicle localization under attacks.**



***Index Terms*— Attack detection and defense, connected automated vehicles (CAVs), consensus estimation, secure cooperative localization.**

I. Introduction

CYBER-PHYSICAL vehicular and transportation systems, enabled by the Internet of Things (IoT) sensing, edge and cloud computing, 5G communication, advanced control, and drive-by-wire systems in the vehicles and infrastructure, offer opportunities to improve the performance of individual vehicles and traffic. With the development of connected vehicle (CV) and automated vehicle (AV) technologies [1], [2], [3], [4], [5], [6], cooperative driving automation (CDA) [7], as standardized by SAE J3216 [8], aims at combining both technologies in connected automated vehicles (CAVs) to enable real-time cooperation of equipped vehicles, other road users, and infrastructure. As outlined in the pioneering review of AVs and CAVs control [9], CDA technology will further improve safety, mobility, environmental sustainability, situational awareness, and operational efficiency of traffic flow. Cooperative localization is one of the critical components. It enables the downstream modules such as planning and control of CDA by leveraging shared sensory information from the CVs and infrastructures through vehicle-to-everything (V2X) communication. For example, aided by the localization information, driving safety is effectively ensured through the proposed sliding mode control algorithm [10]. The cooperation between different vehicles brings the potential to fuse diverse information to improve the localization accuracy of CAVs. However, the multimodality sensors or communication channels for the cooperation also make CAVs vulnerable to cyberattacks. This raises security issues for the localization system [11]. Aiming at leveraging the shared multisensor information from the CAVs in a secure manner, this article proposes a secure cooperative localization method for the CAVs using a consensus estimation framework with considerations of cyberattacks on the sensory information.

A. State-of-the-Art

Localization is one of the most basic modules of any automated driving platform. It has been extensively studied in the past decades [12], [13], [14], [15], [16]. Based on diverse sensors such as inertial measurement unit (IMU), magnetometer, global navigation satellite system (GNSS), camera, radar, or light detection and ranging (LiDAR) equipped on individual vehicles, the multisensor-fusion-based methods typically are represented by the GNSS/inertial navigation

Manuscript received 10 July 2023; accepted 4 September 2023. Date of publication 11 September 2023; date of current version 16 October 2023. This work was supported in part by the United States Department of Transportation (USDOT) Connected Automated Vehicles (CAV) Performance Data Project, in part by the USDOT Automated Driving System Demonstration Program, and in part by the California Resilient and Innovative Mobility Initiative (RIMI) Program. The associate editor coordinating the review of this article and approving it for publication was Dr. Geethu Joseph. *(Corresponding author: Jiaqi Ma.)*

The authors are with the UCLA Mobility Laboratory, Department of Civil and Environmental Engineering, University of California at Los Angeles, Los Angeles, CA 90095 USA (e-mail: jiaqima@ucla.edu).

Digital Object Identifier 10.1109/JSEN.2023.3312610

system (INS) integration system [13], GNSS/INS/LiDAR fusion [17], camera/LiDAR-based simultaneous localization and mapping (SLAM) [18], and map-matching-based localization [19]. These topics have been explored extensively with much significant progress. With the fast development of CAVs and intelligent transportation system (ITS), the cooperation between the system elements (vehicles and infrastructure) is enabling more possibilities (e.g., sharing and using diverse sensory information between vehicles) other than the individual-vehicle-based localization to improve the localization accuracy [20]. Also, in the event of cyberattacks, the cooperation provides more flexibility to detect and defend against the attacks that are injected into the sensors or the V2X communication to improve the system security [11].

To address the issue where GNSS fails at places such as city canyons and indoor parking lots for a short period of time, the relative distance between CAVs is integrated with the GNSS position through an extended Kalman filter (KF) in [21], and [22]. The algorithm relies more on the relative distance to constrain the localization error when the GNSS fails. When using the interdistance from radars, due to the different update rates between the GNSS, radars, and communication units, a track-matching approach using the chi-square statistic test method is used to associate the information from multiple sensors [23]. In [24], in addition to the relative distance between different vehicles from radar, the relative azimuth from the camera is also used to supplement GNSS in a Bayesian framework. To fill the gap where GNSS is unavailable, in [25], the distance between the ego vehicle and roadside unit (RSU) with a known position is integrated with the onboard sensors. Then, the algorithm will localize the ego vehicle based on a weighted linear least square algorithm. With cooperation between CAVs, the issues coming from the GNSS's sensitivity to environments have been addressed to some extent [26].

In terms of sensor fusion framework, a multisensor multivehicle framework is proposed based on global/centralized filtering and local filtering using the onboard sensors, GNSS, and relative distance from other vehicles [26]. Compared with centralized estimation in [24], and [25], distributed estimation is implemented on each vehicle. It shows greater resilience to the vulnerabilities from the failure of the sensors and communications and requires less energy-consuming communication and parallel processing [27]. In this sense, it is straightforward to take advantage of distributed estimation to design the localization algorithm for the CAVs [28]. Among distributed estimation methodologies, the consensus KF (CKF) [27], which has been utilized to localize unmanned aerial vehicles in a formation and has shown promising performance [29], is one of the appropriate methods for the localization of CAVs. Another merit of the CKF is its capability to incorporate shared information from different CAVs. In addition to the benefits stated above, it also allows the neighboring estimator to reach a consensus on the localization [27]. To the best of the authors' knowledge, the CKF [27] has not been investigated to localize the CAVs in the literature. This article aims to bridge this gap while also considering potential cyberattacks on sensors and communication data for security.

As mentioned above, the information sharing between CAVs enables them to cooperate with each other for localization. However, in the meantime, this cooperation makes sensors or communication vulnerable to attacks [30]. Before fusing sensors from different sources, i.e., vehicles or infrastructure, the information should be inspected to detect attacks or other types of faults [31]. In [28], the velocity and the position of the CAVs in a platoon are estimated in an unknown input observer (UIO). A threshold method is adopted to diagnose the faults/attacks based on the output of the UIO. Once a fault/attack is declared, the shared information from the remaining CAVs is used for estimation. In [30], a piecewise-constant attack injected in the GNSS position measurement is detected by a scheme based on a modified unbiased finite impulse response estimator. The scheme is able to generate an intermediate value only related to the attack for detecting the attack conveniently. In [31], given the multiple redundant sensors to measure the same physical variable of the CAV, the attacks are detected directly if there is a difference between the measurement from a specific sensor and the averaged measurement for all the sensors larger than a threshold. In [32], a drift attack caused by the GNSS spoofing is added to an optimizer as a variable to be solved. Then, the attack is diagnosed based on the estimated value of the attack. For the different attack detection methods in [31], stochastic variables are generated, and then, one sample or multiple samples will be used to decide whether an attack occurred [33]. The detection accuracy relies on the thresholds heavily if only one sample of the generated stochastic variable is used. This means that the performance of the attack detector depends heavily on the threshold selection. If a set of samples from the generated stochastic variable is used, the detection accuracy is usually higher. The generalized likelihood ratio test (GLRT) is a well-developed detection method that is based on a set of samples [34]. However, the relatively large set (large window size) can lead to a time delay for the attack detection such as the case with the GLRT in [34]. This time delay possibly leaves the system under attack for a short time before the detection is done. To address the time delay of this kind, this article proposes a delay-prediction framework to enhance attack detection.

Once the attack is detected, the failed sensory information can be discarded in the multisensor-fusion localization algorithms. Specifically, the corresponding node in the CKF can be adjusted to isolate the attacks in the sensory measurements.

B. Main Contributions

In this article, to achieve the multisensor-fusion localization given the shared information of CAVs, a Kalman-consensus information filter (KCIF) is applied and a delay-prediction GLRT-based attack detection method is presented for improving the security of the localization system. Specifically, two main contributions of this article are summarized as follows.

1) To leverage the shared position information from the CAVs, this article applies a KCIF to fuse the measurements from the ego vehicle and adjacent vehicle(s) with different communication topologies. Inspired by [24]

and [26] but differing from them which only consider how to fuse the shared sensory measurements when the measurements are normal, our consensus estimation framework also offers the convenience to accommodate the possible attacks properly. To the best of the authors' knowledge, no research using the KCIF has been reported for cooperatively localizing the CAVs considering the attacks in sensory measurements.

2) For detecting cyberattacks in the sensory measurements, a GLRT-based method is designed. Compared with the GLRT-based detection algorithm in [34], our proposed delay-prediction framework is not only able to detect the attack but also address the induced temporal latency of the decision made by the GLRT-based method [33]. It is worth noting that this framework can be generally integrated with a multiple-sample-based attack/fault detection algorithm regardless of the specific form of the detection algorithm to address the temporal delay issue, which is induced in the detection algorithm. Then, based on the attack indicator (AI) from the GLRT-based algorithm, a rule-based attack isolation method is integrated with the KCIF for isolating the attack data samples. The secure cooperative localization is validated via numerical simulation.

The remainder of this article is organized as follows. The problem studied in this article is formulated in Section II. The secure cooperative localization method is designed in Section III. Section IV provides test results in different communication topology and attack settings and discusses the findings and performance, and finally, Section V concludes this article.

II. Problem Formulation

In this section, the communication topology for CAV information sharing, node kinematic model, and node measurement model are presented. Based on these models, the cooperative localization algorithm is designed in the following.

A. Communication Topology

The scenario shown in Fig. 1 is CAV platooning [35], and without loss of generality, this article will focus on such scenarios, i.e., longitudinal scenarios. Each vehicle in Fig. 1 is equipped with an IMU, a GNSS receiver, and a sensor such as radar, LiDAR, or camera, which can measure the relative distance between the ego vehicle and adjacent vehicles. The IMU can measure the longitudinal acceleration of the ego vehicle. The GNSS receiver provides the position of the ego vehicle, and the radar, LiDAR, or camera measures the relative distance between the ego vehicle and its neighbor. Vehicles are also equipped with V2X transceivers [such as cellular V2X or dedicated short-range communication (DSRC)] to establish vehicle to vehicle (V2V) communication and enable sharing of sensory information among adjacent vehicles through a directed graph $\mathcal{G}_d = \{V, E\}$ or undirected graph $\mathcal{G}_u = \{V, E\}$ [22]. In the graph, $V = \{1, 2, \ldots, N\}$ is the set of nodes and $E \subseteq V \times V$ is the set of edges in connection. The adjacency matrix $\mathcal{A}$ and the Laplacian matrix $\mathcal{D}$ are adopted

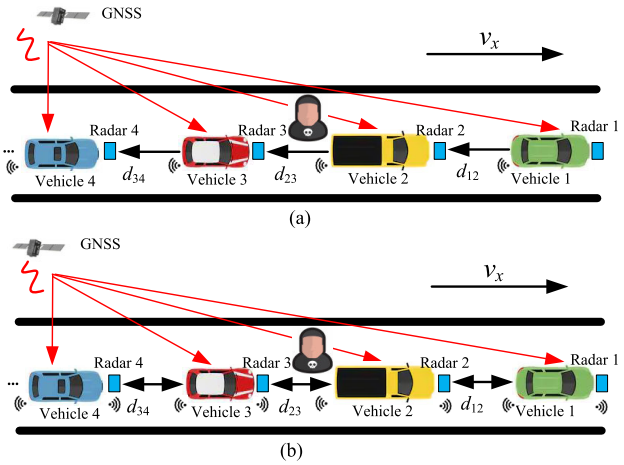


Fig. 1. Example scenarios for cooperative localization. Each vehicle is equipped with an IMU, a GNSS receiver, and a sensor such as radar, LiDAR, or camera, which can measure the relative distance/velocity between the ego vehicle and adjacent vehicles. d_{12} denotes the measured relative distance between Vehicles 1 and 2 and v_x is the velocity of the traffic flow. Vehicles are able to share sensory information through V2X communication (such as cellular V2X or DSRC). For the scenario with directed communication case shown in (a), only the following vehicle has access to its front neighbor; for the scenario with undirected communication case shown in (b), the front and the following vehicle has access to its neighbor. Other communication topologies are also possible. The GNSS will be exposed to the attacks.

to show the properties of the graph $\mathcal{G}$ [36]. The entry a_{ij} of the adjacency matrix $\mathcal{A} \in \mathbb{R}^{N\times N}$ is given as

$$\begin{cases} a_{ij} = 1, & \text{if } \{j, i\} \in E \\ a_{ij} = 0, & \text{if } \{j, i\} \notin E, \end{cases} \quad i, j = \{1, 2, \ldots, N\} \tag{1}$$

where, for the directed graph, $j, i \in E$ denotes that there is a directed edge from node j to node i, meaning that node i has access to the sensory information of node j through V2V communication for $\mathcal{G}_d = \{V, E\}$; for the undirected graph, $j, i \in E$ denotes that there is an undirected edge between nodes j and i, meaning that node i or j has access to each other's sensory information through V2V communication for $\mathcal{G}_u = \{V, E\}$. Besides, there are no self-loops, and thus, $a_{ii} = 0$, $i = 1, \ldots, N$. Node j is the neighbor of node i when $a_{ij} = 1$, and the neighbor set of node i is denoted as $\mathcal{N}_i = \{j | a_{ij} = 1\}$. Then, the entry of the degree matrix $\mathcal{D}$ for this graph is given as

$$\beta_{ij} = \begin{cases} 0, & \text{if } i \neq j \\ \sum_{k=1}^{N} a_{ik}, & \text{if } i = j, \end{cases} \quad i, j = \{1, 2, \ldots, N\}. \tag{2}$$

Accordingly, the Laplacian matrix $\mathcal{L} \in \mathbb{R}^{N\times N}$ is given as

$$\mathcal{L} = \mathcal{D} - \mathcal{A}. \tag{3}$$

In this article, the secure localization algorithm is tested based on the communication with both the directed and undirected graph typologies as two representative scenarios. There can be, however, more complicated topological scenarios. In the case of CAV platooning, there are multiple possible ways of communication, such as all predecessor and leader–predecessor [37]. The proposed methodology can be applied to analyze any type of communication topology since

it is feasible to add any nodes into the consensus estimation algorithm as long as the types of measurements, possibly from different types of sensors, for each node are homogeneous. A homogeneous node means that the node has the same capability to sense its neighbor vehicles and communicate with them such that the node is able to provide the relative position between itself and neighbor vehicles and then publish the relative position information to its neighbor vehicles. For example, the interdistance might come from camera, LiDAR, or radar sensor. The details are discussed in Section III.

B. Node Kinematic Model

In this article, since the main contribution is the cooperative localization method design for the CAVs, without the loss of generality, we simplify the kinematic model with the assumption that the vehicle is mainly maneuvered in the longitudinal direction, and therefore, a longitudinal vehicle kinematic model is used. Another reason for this assumption is that platooning and similar safety-critical applications are one of the main the application scenarios of this localization algorithm since the vehicles in a platoon follow very closely to each other and any faults and attacks can result in serious consequences [30]. Note that, although only the longitudinal vehicle kinematics is considered in this article, the lateral vehicle kinematics can also be incorporated into the overall secure localization framework by only changing the node kinematic model to address the more comprehensive driving maneuvers.

The longitudinal vehicle kinematic model is presented in (4) and (5)

$$\dot{v} = a + \omega_a \tag{4}$$

where v denotes the longitudinal velocity, a denotes the longitudinal acceleration, and ω_a is the random noise of the accelerometer

$$\dot{p} = v \tag{5}$$

where p denotes the position. By choosing the velocity and position as the states $\boldsymbol{x} = [p\ v]^\top$, we have the standard state equation

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{\Gamma}\boldsymbol{\omega} \tag{6}$$

where $\boldsymbol{A} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ is the state matrix, $\boldsymbol{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ is the input matrix of the vehicle, $\boldsymbol{u} = a$ is the input, $\boldsymbol{\omega} = \omega_a$ is the noise, and $\boldsymbol{\Gamma} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ is the input matrix of the noise. When estimating the states such as the velocity or the position by an estimator such as KF, the model described by (4) and (5) needs to be discretized as in the following equation:

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}_k \boldsymbol{x}_k + \boldsymbol{\Xi}_k \boldsymbol{u}_k + \boldsymbol{\Lambda}_k \boldsymbol{\omega}_k \tag{7}$$

where $\boldsymbol{\Phi}_k = e^{\boldsymbol{A}\Delta T} \approx \boldsymbol{I} + \boldsymbol{A}\Delta T$ is the state transition matrix of the system (6) with the discrete-time realization, $\boldsymbol{\Xi}_k = \int_0^{\Delta T} e^{\boldsymbol{A}t}\mathrm{d}t\boldsymbol{B} \approx \boldsymbol{B}\Delta T$ is the input matrix, and $\boldsymbol{\Lambda}_k = \int_0^{\Delta T} e^{\boldsymbol{A}t}\mathrm{d}t\boldsymbol{\Gamma}$.

Remark 1: In the longitudinal vehicle kinematic model, we made some simplifications such as ignoring the bias error and gravity component in the longitudinal accelerometer caused by the nonzero pitch angle of the vehicle body. It is worth noting that, although these factors are significant for the localization algorithm development, the estimation of them has been well addressed in the literature such as [38]. Leveraging the off-the-shelf algorithms is available to tackle the issues caused by these errors. Another note is that the model in (6) is based on the vehicle kinematics, which is robust against the vehicle dynamic model uncertainties, meaning that the method in this article will not be affected by the vehicle dynamic model uncertainties.

C. Node Measurement Model

Given the sensor configuration discussed in Section II-A, the pieces of information from the GNSS that provides the global position and radar (or sensors that generate the same types of measurements) that provide the interdistance between vehicles are adopted to develop the CAV measurement model. For the yellow vehicle in Fig. 1, the node measurement model is derived to implement an estimator to estimate the state of the vehicle. Through the instrumented GNSS receiver and radar, the yellow vehicle has access to the position measurement and relative distance from itself to the green vehicle in Fig. 1. Through the V2V communication, the ego yellow vehicle can also request the sensory information of its adjacent vehicles (green vehicle in $\mathcal{G}_d$, green and red vehicles in $\mathcal{G}_u$, or possibly other vehicles dependent on the communication topology) to enrich the measurements. These measurements have the potential to improve both the accuracy and robustness against the attacks on the ego yellow vehicle via the proposed consensus-based estimation. Specifically, for instance, in $\mathcal{G}_u$, the relative distance d_{12} and d_{23} and the GNSS position of the green and red vehicles are available for the yellow vehicle for enhanced localization. The measurement model is given by the relative distance as

$$p_{\mathrm{G}e} = p_e + \eta_{\mathrm{G}e} + f_{\mathrm{G}e} \tag{8}$$

where the subscript e means the ego vehicle, p_{G} is the GNSS position, p is the true position, η_{G} is the Gaussian white noise of the GNSS measurement, and f_{G} denotes the injected attack.

Along with the measurements from the ego vehicle, the GNSS positions of the adjacent vehicle $i \in \mathcal{N}_i$ can be transformed to the position measurements of the ego vehicle with the relative distance measurements. The GNSS position of vehicle i obtained through wireless communication has the same measurement model as (8) and it is given as

$$p_{\mathrm{G}i} = p_i + \eta_{\mathrm{G}i} + f_{\mathrm{G}i} \tag{9}$$

where the subscript i means the adjacent vehicle i. Then, combining (9) with the transformation given as

$$d_{\mathrm{R}ie} = d_{ie} + \eta_{\mathrm{R}ie} + f_{\mathrm{R}ie} \tag{10}$$

where $d_{\mathrm{R}ie}$ means the relative distance between the adjacent vehicle i and ego vehicle e measured by the sensors such as radar, LiDAR, or camera. d_{ie} means the true relative distance,

f_{Rie} denotes the injected attack in the measurement d_{Rie}, and η_{Rie} means the Gaussian white noise of the relative distance. Then, the measurement of the ego vehicle position can be derived as

$$p_{ei} = p_i + d_{ie} + \underbrace{\eta_{Rie} + \eta_{Gi}}_{\eta} + \underbrace{f_{Rie} + f_{Gi}}_{f} \tag{11}$$

where p_{ei} means the measurement of the ego vehicle e through its adjacent vehicle i. From (11), it can be seen that both the attacks from the GNSS f_{Gi} and the relative distance f_{Rie} will be propagated to the position measurement p_{ei} for the ego vehicle's position. In other words, though cooperative localization can potentially improve accuracy, leveraging the information from immediately adjacent vehicles or even vehicles further away (via chaining and adding up consecutive sensor data; see Remark 2) such as the leader of the platoon that uses a certain communication topology will also incur more risks due to the propagation and make the ego vehicle localization more vulnerable to any attack on the surrounding traffic. This necessitates continuous monitoring of the sensory measurements used for the cooperative localization and detection of faults and attacks. This in turn allows us to fuse those measurements from adjacent vehicles cooperatively and improve the localization accuracy in a consensus framework if those attacks are detected and isolated properly. This process of attack isolation will be discussed in Section III-C.

Remark 2: Although, in (11), only the position of the ego vehicle e and the position of its adjacent vehicle i are associated in $\mathcal{G}_d$ or $\mathcal{G}_u$, more position measurements for the ego vehicle can be derived from the CVs through the transformation formulated in (10) as long as the communication topology is able to provide the link between the corresponding CV and the ego vehicle. In this way, not only the information from the adjacent vehicle(s) and the ego vehicle can be fused, but also more information from CVs further away is possible to be incorporated in our proposed consensus localization framework. The key to enabling this fusion is having the bridged communication between the ego vehicle and other CVs to transfer the sensory information.

Based on (8) or (11), the standard measurement model is given as

$$z = \boldsymbol{H}\boldsymbol{x} + \eta + f \tag{12}$$

where z denotes the position measurement in (8) or (11), $H = [1, 0]$, η denotes the Gaussian white noise in (8) or (11), and f denotes the attack from the position of the GNSS in (8) or the relative distance (11). For the ego vehicle, the measurements given in (8) and (11) are homogeneous, and then, the consensus estimation technique such as a CKF is adopted to fuse the pieces of information from different vehicles to improve both the position accuracy and the resilience against cyberattacks.

III. Secure Localization Method

Based on the models developed in Section II, in this section, the framework of the secure cooperative localization algorithm is first presented, the CKF for fusing the sensory information is introduced, the GLRT-based attack detection algorithm is developed, and the attack defense method is given.

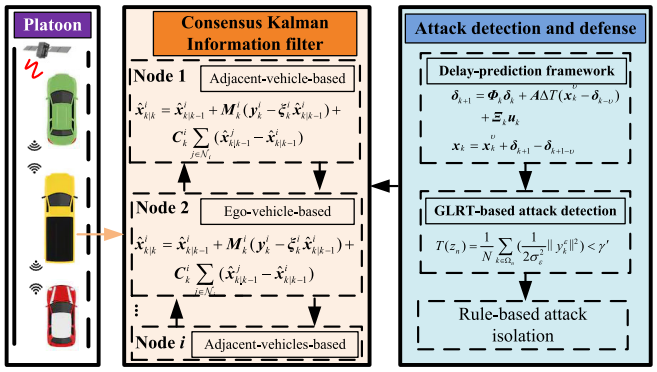


Fig. 2. Framework of the secure cooperative localization. The sensory information input to the CKIF includes the IMU information of the ego vehicle (yellow vehicle), GNSS position of the ego vehicle and adjacent vehicle(s), and relative distance from the ego vehicle to adjacent vehicle(s). Each of the positions from the ego vehicle's GNSS or derivation given by (11) from adjacent vehicle(s) drives a node in the CKIF. All the sensory measurements are monitored by the GLRT-based attack detection method. Once the attack is detected, the corresponding position measurements are isolated by a rule-based method.

A. Framework

The framework of the proposed algorithm is shown in Fig. 2. A consensus Kalman information filter (CKIF) is adopted to fuse the sensory information (i.e., the position of the GNSS and relative distance from the radar) from the ego vehicle i and the adjacent vehicles in $\mathcal{N}_i$ with the communication topology $\mathcal{G}_d$ or $\mathcal{G}_u$. For each node in the CKIF, the GLRT-based attack detection will be performed to diagnose whether the sensory measurement from the ego vehicle or its neighbor is attacked. Then, based on the attack detection results, a rule-based delay-prediction method is proposed to isolate the attack.

B. Consensus Kalman Information Filter

In this section, a CKIF is applied to achieve a consensus estimation of the states in (7) by fusing the homogeneous measurements from different vehicles in a platoon [27]. Specifically, for the communication topology with $\mathcal{G}_d$, based on the adjacency matrix $\mathcal{D}$, the ego vehicle i only has access to its front vehicle's sensory measurements such as GNSS position, and therefore, two nodes are included in the CKIF. For the communication topology $\mathcal{G}_u$, the ego vehicle has access to sensory measurements (i.e., GNSS and radar for measuring the global position and interdistance, respectively) from its front and rear vehicles, and thus, three nodes are included. In other words, the difference when applying the CKIF to estimate the states of the ego vehicle between the communication topology $\mathcal{G}_d$ and $\mathcal{G}_u$ is the number of the nodes included in the CKIF. When more vehicles are connected, i.e., the homogeneous sensory information can be shared between the CAVs, more nodes can be incorporated in the CKIF. Therefore, the CKIF framework can feasibly apply to both $\mathcal{G}_d$ and $\mathcal{G}_u$, as well as other communication topologies. The two different

nodes based on the ego vehicle's or its adjacent vehicle(s)' information are discussed correspondingly.

1) Node-Based on the Measurement From the Ego Vehicle: Based on (7), the GNSS position given in (12) can drive one node in the CKIF to estimate the position. In this node, the predicted variables (7) will provide the prior information and the GNSS position from the ego vehicle is explored to provide the measurement for the posteriori estimation.

2) Node(s) Based on the Measurement From the Adjacent Vehicle(s): Besides the measurement from the ego vehicle's GNSS position, the position measurement from the adjacent vehicles given as (11) can be leveraged to drive other node(s) in the CKIF. Based on the node kinematic model (7) for the ego vehicle and the measurement model (11) from the adjacent vehicles, for the directed communication topology $\mathcal{G}_d$, one node beside the node in Section III-B1 can be formulated, and for the undirected communication topology $\mathcal{G}_u$, two more nodes beside the node in Section III-B1 are augmented.

Remark 3: Due to the different number of the nodes in the CKIF for $\mathcal{G}_d$ and $\mathcal{G}_u$, both the localization accuracy and robustness against the attack will be difference. The intuitive speculation for the difference is that: to some extent, with more nodes in the CKIF to come to a consensus estimation, the position accuracy is higher and the robustness is also better due to the more redundancy of sensory measurements. This, however, comes at the cost of additional data communication and computational loads. In Section IV, this speculation will be exemplified and discussed.

With the nodes given in Sections III-B1 and III-B2, the CKF shown in (13) is adopted [39]

$$\begin{aligned}
\hat{\boldsymbol{x}}^i_{k|k} &= \hat{\boldsymbol{x}}^i_{k|k-1} + \boldsymbol{K}^i_k\left(\boldsymbol{Z}^i_k - \boldsymbol{H}^i_k\hat{\boldsymbol{x}}^i_{k|k-1}\right) \\
&\quad + \boldsymbol{C}^i_k\sum_{j\in\mathcal{N}_i}\left(\hat{\boldsymbol{x}}^j_{k|k-1} - \hat{\boldsymbol{x}}^i_{k|k-1}\right) \\
\boldsymbol{K}^i_k &= \boldsymbol{P}^i_k\left(\boldsymbol{H}^i_k\right)^\top\left(\boldsymbol{R}^i + \boldsymbol{H}^i_k\boldsymbol{P}^i_k\left(\boldsymbol{H}^i_k\right)^\top\right)^{-1} \\
\boldsymbol{M}^i_k &= \left(\boldsymbol{F}^i_k\boldsymbol{P}^i_k\left(\boldsymbol{F}^i_k\right)^\top + \boldsymbol{K}^i_k\boldsymbol{R}^i\left(\boldsymbol{K}^i_k\right)^\top\right) \\
\boldsymbol{F}^i_k &= \boldsymbol{I} - \boldsymbol{K}^i_k\boldsymbol{H}^i_k, \boldsymbol{C}^i_k = \gamma\boldsymbol{F}^i_k\boldsymbol{G}^i_k \\
\boldsymbol{G}^i_k &= \boldsymbol{\Phi}^i_k\boldsymbol{M}^i_k\left(\boldsymbol{\Phi}^i_k\right)^\top + \boldsymbol{Q}^i_k + \boldsymbol{P}^i_k\boldsymbol{S}^i_k\boldsymbol{P}^i_k \\
\boldsymbol{P}^i_{k+1} &= \boldsymbol{\Phi}^i_k\boldsymbol{M}^i_k\left(\boldsymbol{\Phi}^i_k\right)^\top + \boldsymbol{Q}^i, \quad \hat{\boldsymbol{x}}^i_{k+1|k} = \boldsymbol{\Phi}^i_k\hat{\boldsymbol{x}}^i_{k|k}
\end{aligned} \tag{13}$$

where the superscript i denotes node i, $\hat{\boldsymbol{x}}^i_{k|k}$ and $\hat{\boldsymbol{x}}^i_{k+1|k}$ are estimation and prediction of the state $\boldsymbol{x}^i_k$, and the matrix inversion lemma $(\mathbf{A}+\mathbf{BCD})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}(\mathbf{C}^{-1} + \mathbf{DA}^{-1}\mathbf{B})^{-1}\mathbf{DA}^{-1}$ is utilized for computation of $\boldsymbol{K}^i_k$. $\boldsymbol{H}^i$ is the measurement matrix and $\boldsymbol{I}$ is an identity matrix. $\boldsymbol{R}^i$ and $\boldsymbol{Q}^i$ denote the measurement and process noise covariance matrices, respectively. $\boldsymbol{P}^i$ denotes the state covariance matrix, and $\boldsymbol{\Phi}^i$ denotes the state transition matrix. For the convenience of implementation, by defining the weighted measurements $\boldsymbol{y}^i_k = (\boldsymbol{H}^i_k)^\top(\boldsymbol{R}^i)^{-1}\boldsymbol{z}^i_k$ for node i and the information matrix $\boldsymbol{\xi}^i_k = (\boldsymbol{H}^i_k)^\top(\boldsymbol{R}^i)^{-1}\boldsymbol{H}^i_k$, the information form of the CKF, shown in (14), i.e., CKIF, is applied

$$\begin{aligned}
\hat{\boldsymbol{x}}^i_{k|k} &= \hat{\boldsymbol{x}}^i_{k|k-1} + \boldsymbol{M}^i_k\left(\boldsymbol{y}^i_k - \boldsymbol{\xi}^i_k\hat{\boldsymbol{x}}^i_{k|k-1}\right) \\
&\quad + \boldsymbol{C}^i_k\sum_{j\in\mathcal{N}_i}\left(\hat{\boldsymbol{x}}^j_{k|k-1} - \hat{\boldsymbol{x}}^i_{k|k-1}\right) \\
\boldsymbol{M}^i_k &= \left(\left(\boldsymbol{P}^i_k\right)^{-1} + \boldsymbol{\xi}^i_k\right)^{-1} \\
\boldsymbol{C}^i_k &= \gamma\boldsymbol{F}^i_k\boldsymbol{G}^i_k \\
\boldsymbol{P}^i_{k+1} &= \boldsymbol{\Phi}^i_k\boldsymbol{M}^i_k\left(\boldsymbol{\Phi}^i_k\right)^\top + \boldsymbol{Q}^i \\
\hat{\boldsymbol{x}}^i_{k+1|k} &= \boldsymbol{\Phi}^i_k\hat{\boldsymbol{x}}^i_{k|k}.
\end{aligned} \tag{14}$$

Exchanging the prediction of the states, the node based on the measurement from the ego vehicle and the node(s) based on the measurement from the adjacent vehicle(s) attempt to reach a consensus on the (estimated) states from (7). Algorithm 1 represents the CKIF strategy, where nodes i and j denote the node(s) in Section III-B1 and the node(s) in Section III-B2.

Algorithm 1 CKIF

Input : $\boldsymbol{\Phi}^i_k$; z_{Gi} (position measurement from ego vehicle i) and z_{ij} (position measurement from ego vehicle's adjacent vehicles j); $\boldsymbol{H}$; $\boldsymbol{Q}^i$; $\boldsymbol{R}^i$ and $\boldsymbol{R}^j$; $\hat{\boldsymbol{x}}^j_{k|k-1}$ (prediction states of node j); initial state $\boldsymbol{x}^i(0)$;initial state covariance $\boldsymbol{P}^i(0)$

Output: $\bar{\boldsymbol{x}}^i_{k|k}$

while *GNSS updated* **do**

Compute the information vector $\boldsymbol{y}^i_k = \boldsymbol{H}^\top(\boldsymbol{R}^i)^{-1}\boldsymbol{z}_{Gi} + \sum_{j\in\mathcal{N}_i}(\boldsymbol{H}^\top(\boldsymbol{R}^j)^{-1}\boldsymbol{z}_{ij})$;

Compute the information matrix $\boldsymbol{\xi}^i_k = \boldsymbol{H}^\top(\boldsymbol{R}^i)^{-1}\boldsymbol{H} + \sum_{j\in\mathcal{N}_i}(\boldsymbol{H}^\top(\boldsymbol{R}^j)^{-1}\boldsymbol{H})$;

Compute the consensus Kalman state estimation $\hat{\boldsymbol{x}}^i_{k|k} = \hat{\boldsymbol{x}}^i_{k|k-1} + \boldsymbol{M}^i_k(\boldsymbol{y}^i_k - \boldsymbol{\xi}^i_k\hat{\boldsymbol{x}}^i_{k|k-1}) + \boldsymbol{C}^i_k\sum_{j\in\mathcal{N}_i}(\hat{\boldsymbol{x}}^j_{k|k-1} - \hat{\boldsymbol{x}}^i_{k|k-1})$ $\boldsymbol{M}^i_k = ((\boldsymbol{P}^i_k)^{-1} + \boldsymbol{\xi}^i_k)^{-1}$; $\boldsymbol{C}^i_k = \gamma\boldsymbol{F}^i_k\boldsymbol{G}^i_k$;

Update the state and its error covariance $\boldsymbol{P}^i_{k+1} = \boldsymbol{\Phi}_k\boldsymbol{M}^i_k\boldsymbol{\Phi}^\top_k + \boldsymbol{Q}^i$; $\bar{\boldsymbol{x}}^i_{k+1|k} = \boldsymbol{\Phi}_k\hat{\boldsymbol{x}}^i_{k|k}$;

end

C. Attack Detection and Defense

In Section III-B, the measurements from the ego vehicle and its adjacent vehicles are fused through the KCIF. In the real implementation, the GNSS or the radar has the potential risk to be exposed to two kinds of attacks, including false data injection or denial of service [40]. For the two sensors, the most common attack studied in the literature is the false data injection [30]. The considered attacker, which has the access to the sensors and compromises them to falsify the data, will inject false data by adding a drift error or large noisy error such as outliers into sensory measurements. The latter can usually be detected by a chi-square method [41] and compensated by

an adaptive KF [42]. Also, it is more complex to address the drift attack [33]

Thus, in this section, the drift attack model similar to [30] is introduced, and then, its detection and isolation method shown in Fig. 2 is presented.

1) Attack Model: For our study, the false data injection attack, which is a drift error, is considered. The drift error does not frequently change and is injected in the original GNSS position or the interdistance [30], i.e., given $\forall k$, there exist $\exists k_1, k_2 \in \mathbb{N}^+$ such that the attack signal ε_k

$$\varepsilon_k = c_k, \quad \forall k \in [k_1, k_2], \; k_2 - k_1 \geq n - 1, \; n \in \mathbb{N}^+ \tag{15}$$

where c_k is a constant. This kind of error for the GNSS position or the interdistance is commonly encountered [30] and is investigated in this article. Note that another common kind of false data injection attack, which is a large noise error, will not be discussed because it can be addressed by a method such as chi-square method and adaptive KF in [42].

2) Delay-Prediction Framework: For the attack detection in a KF framework, usually, the innovation or residual is used to determine whether an attack has occurred [33]. In our case, the residual $\boldsymbol{\varepsilon}$ of the CKIF given in (16) is generated for the attack detector

$$\boldsymbol{\varepsilon}_k = z_k - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k} \tag{16}$$

where z_k is from (12) and $\hat{\boldsymbol{x}}_{k|k}$ is the estimated state from the CKIF. As stated in Section I-A, attack detection can be done based on only one sample [31] or multiple samples of the residual or innovation [33]. In this work, the GLRT-based method in [43], which is a multiple-sample-based method, is adopted to detect the attack due to its high detection accuracy. However, the tradeoff is that several samples of the residual or innovation are required.

Fig. 3 shows the delay-prediction-based attack detection framework. The input u_k and measurement z_k for the KF are delayed by $\tau = \upsilon\Delta T$ first. The delayed $u_{k-\upsilon}$ and $z_{k-\upsilon}$ are used to perform the time update and measurement update of the KF to obtain the estimated states $x_{k-\upsilon}$. Furthermore, $\hat{x}_k$ is predicted. $x_{k-\upsilon}$ and $\hat{x}_k$ are then used to generate the residual $\varepsilon_{k-\upsilon}$ and pseudo innovation ζ_k of the KF. These two variables will be used as input to the GLRT-based attack detection algorithm. The detailed process of attack detection is discussed in the following.

In the real-time implementation, a buffer, such as Buffer 2 in Fig. 3, is used to save the current and historical information, but this operation will induce a time delay for the decision due to the moving average effect. This means that given Buffer 2 of the residual or innovation at time t, the decision made belongs to time t_2 instead of time t. There will be a lag $\varrho = t - t_2$, which is related to the size of the samples of the residual or innovation. This lag will prevent us from instantly isolating the attack, making the estimated states influenced by the attacks in this delay. Although smaller sets will reduce the time delay, they will degrade the attack detection accuracy. To explicitly account for the time delay of the GLRT-based attack detection algorithm, a delay-prediction framework is proposed.

Given the lag ϱ for the GLRT-based method no matter how we choose the window size of the samples, we delay the

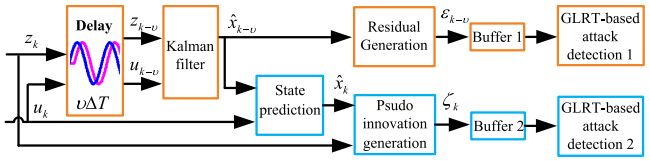


Fig. 3. Delay-prediction-based attack detection framework.

IMU information, the GNSS position of the ego vehicle, and the position derived from the adjacent vehicles for a time τ to estimate the states $\hat{\boldsymbol{x}}(t-\tau)$ in (7) of the CAVs by the CKIF at time $t-\tau$. This active delay will allow us to save the fresh sensory information from $t-\tau$ to t to diagnose the attack in the measurements as long as we can predict the current estimated states $\hat{\boldsymbol{x}}(t)$ based on $\hat{\boldsymbol{x}}(t-\tau)$ and the fresh IMU information free from the attacks. Then, using the predicted states and the fresh measurements from the ego vehicle's GNSS and adjacent vehicles, we can generate a set of pseudo innovations and save it in buffer 2 of Fig. 3 for GLRT-based attack detection. The decision made based on buffer 2 by the GLRT-based method at t_2 is prior to the states $\hat{\boldsymbol{x}}(t-\tau)$ at the time $t-\tau$, and using this decision, the attack could be isolated instantly by tuning the measurement covariance matrix in the corresponding nodes in the CKIF. Then, the time delay issue of the multiple-sample-based attack detection algorithm could be addressed. However, this mechanism works when the measurement transits from normal status to an attacked status but has a deficiency when the measurement transits from an attacked status back to a normal status. To tackle this, based on $\hat{\boldsymbol{x}}(t-\tau)$, Buffer 1 in Fig. 3 with the residual generated by (16) is also reserved for another GLRT-based attack decision method. The decision from this GLRT-based detector tagged to t_1 is able to reflect the status of the measurement at the time $t-\tau$ to some extent. Note that although there is also a latency between $t-\tau$ and t_1, this latency will not cause a huge impact because it just delays a short term to use the normal measurements instead of attacked ones. Thereby, based on the decisions from the GLRT-based method using Buffers 1 and 2 in Fig. 3, the attack model studied in this article can be detected properly.

In the following, the state prediction and pseudo innovation generation are presented. Based on the estimated states $\hat{\boldsymbol{x}}(t-\tau)$ and the IMU information, the states $\hat{\boldsymbol{x}}(t)$ is predicted by the following equation [44]:

$$\begin{aligned} \dot{\boldsymbol{\delta}}(t) &= \boldsymbol{A}(t)\left(\hat{\boldsymbol{x}}(t-\tau) + \boldsymbol{\delta}(t) - \boldsymbol{\delta}(t-\tau)\right) + \boldsymbol{B}(t)\boldsymbol{u}(t) \\ \hat{\boldsymbol{x}}(t) &= \hat{\boldsymbol{x}}(t-\tau) + \boldsymbol{\delta}(t) - \boldsymbol{\delta}(t-\tau) \end{aligned} \tag{17}$$

where $\tau = \upsilon\Delta T$ is the actual delay time, υ is the window size of the required innovation for the GLRT-based attack detection algorithm, $\hat{\boldsymbol{x}}(t-\tau)$ is the delayed estimated states, $\boldsymbol{u}$ is the input in (7), and $\boldsymbol{\delta}$ is the intermediate states. From (17), it can be seen that, given the delay estimated states $\hat{\boldsymbol{x}}(t-\tau)$ and $\boldsymbol{u}(t)$, the states at current timestamp t can be predicted. For the real implementation, (17) is discretized and we have

$$\begin{aligned} \boldsymbol{\delta}_{k+1} &= \boldsymbol{\Phi}_k\boldsymbol{\delta}_k + \boldsymbol{A}\Delta T\left(\hat{\boldsymbol{x}}_{k-\upsilon} - \boldsymbol{\delta}_{k-\upsilon}\right) + \boldsymbol{\Xi}_k\boldsymbol{u}_k \\ \hat{\boldsymbol{x}}_k &= \hat{\boldsymbol{x}}_{k-\upsilon} + \boldsymbol{\delta}_{k+1} - \boldsymbol{\delta}_{k+1-\upsilon} \end{aligned} \tag{18}$$

where $k \geq \upsilon$ and $\hat{\boldsymbol{x}}_k$ is the predicted states for the current timestamp k. Once we have the predicted states, a buffer will

be used to save the predicted states from $k+1-\upsilon$ to $k+1$. Then, the pseudo innovation $\boldsymbol{\zeta}_k$ in (19) will be computed and tested by the GLRT-based method to determine whether an attack has occurred shown in the blue blocks in the lower branch in Fig. 3

$$\boldsymbol{\zeta}_k = \boldsymbol{z}_k - \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k-\upsilon}. \tag{19}$$

The term pseudo is used here because, rigorously, the predicted state used for computing the innovation is the one-step prediction from the KF, which is the condition that the innovation satisfies the Gaussian noise distribution [45]. The prediction step size υ used in (18) depends on the buffer size, which will be used in the GLRT-based algorithm and is larger than one apparently. On the one hand, the prediction errors have been proved stable and bounded in [44]. On the other hand, from our experience, only ten samples of the innovation (around 1 s in the time domain) are enough for the GLRT-based algorithm to detect the attack. In this regard, this short-time prediction based on the IMU information will not generate a large cumulative error in the position [19]. Compared with the attack, this cumulative error within a short time is negligible.

In Section III-C3, based on the residual and pseudo innovation, the GLRT-based attack detection algorithm is introduced.

Remark 4: In this proposed delay-prediction framework, the benefit is to maintain the detection accuracy and real-time performance for the multiple-sample-based method. Although we combine this framework with the GLRT-based method in [34], and [43], note that this framework can be generalized to any multiple-sample-based attack/fault detection method when a temporal delay is induced, which is meaningful to the community.

3) GLRT-Based Attack Detection: Given the set of the residual and innovation from Buffers 1 and 2 in Fig. 3 and motivated by [34], the attack detection is formulated as a binary hypothesis testing problem, where the detector can choose between the two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$ defined as

$$\begin{aligned} &\mathcal{H}_0 : \text{The attack has occurred} \\ &\mathcal{H}_1 : \text{There is no attack.} \end{aligned} \tag{20}$$

Since the residual in (16) or the pseudo innovation (19) will be used to detect the attack, in order to derive the probability density function (pdf) for each hypothesis, the attack and noise model in the residual or pseudo innovation are specified. The model y_k, which represents the innovation $\boldsymbol{\zeta}_k$ or the residual $\boldsymbol{\varepsilon}_k$ at the time instance k, is defined as [34]

$$\begin{aligned} y_k &= s_k(\theta) + \varpi_k \\ s_k(\theta) &= s_k^{\lambda}(\theta), \quad \varpi_k = \varpi_k^{\lambda}. \end{aligned} \tag{21}$$

Here, $s_k^{\lambda}(\theta)$ represents the attack signal λ, θ denotes the set of unknown parameters of the signal, and ϖ_k^{λ} is the noise in λ. Without the attack signal, the residual becomes noise assumed to satisfy the Gaussian distributed zero-mean condition [45]. The pseudo innovation is also assumed approximated to this condition [45]. Then, for the two hypotheses, the following conditions hold:

$$\begin{aligned} &\mathcal{H}_0 : \exists k \in \Omega_n \quad \text{s.t. } s_k^{\lambda}(\theta) \neq 0 \\ &\mathcal{H}_1 : \forall k \in \Omega_n \quad \text{s.t. } s_k^{\lambda}(\theta) = 0 \end{aligned} \tag{22}$$

where $\Omega_n = \{l \in \mathbb{N} : n \leq l \leq n+N-1\}$ and N is the window size of the residual or the pseudo innovation, which is related to the lag issue mentioned in Section III-C2. From (22), it can be seen that if there is no attack in the position from the ego vehicle's GNSS or adjacent vehicle(s), the signal component except the noise should be zero. Otherwise, it should be none zero and can be detected. With (21), the residual or the pseudo innovation originates from a family of pdfs as in (with $i \in \{0, 1\}$)

$$p(z_n; \theta, \mathcal{H}_i) = \prod_{k \in \Omega_n} p\left(y_k^{\lambda}; \theta, \mathcal{H}_i\right) \tag{23}$$

where $z_n \triangleq \{y_k\}_{k=n}^{n+N-1}$ denotes the residual or the pseudo innovation sequence from time instant n to $n+N-1$, $p(*;\theta)$ denotes a pdf depending on the parameter θ, i.e., $p(z_n; \theta, \mathcal{H}_i)$ means the pdf for the two hypotheses based on the parameter θ given the residual or the pseudo innovation sequence z_n, and the details of the pdfs are defined as

$$p\left(y_k^{\lambda}; \theta, \mathcal{H}_i\right) = \frac{1}{\left(2\pi\sigma_{\lambda}^2\right)^{3/2}} \exp\left(-\frac{1}{2\sigma_{\lambda}^2} \| y_k^{\lambda} - s_k^{\lambda}(\theta) \|^2\right) \tag{24}$$

where σ_{λ} denotes the noise variance of the residual or the pseudo innovation [the noise variance of the position from the ego vehicle's GNSS or adjacent vehicle(s)]. Then, the GLRT [34] is derived to determine whether $\mathcal{H}_1$ [there is no attack in the position from the ego vehicle's GNSS or adjacent vehicle(s)] happens when

$$L_G(z_n) = \frac{p\left(z_n; \hat{\theta^1}, \mathcal{H}_1\right)}{p\left(z_n; \hat{\theta}^0, \mathcal{H}_0\right)} > \gamma \tag{25}$$

where $\hat{\theta^1}$ and $\hat{\theta}^0$ are the maximum likehood estimates of the unknown parameters when $\mathcal{H}_1$ is true and the unknown parameters when $\mathcal{H}_0$ is true, respectively, and γ is a threshold. In the real implementation, γ is a tuning parameter given the tolerant false alarm probability, i.e., the probability of deciding on the hypothesis $\mathcal{H}_1$ when hypothesis $\mathcal{H}_0$ is true. In the real-world application, we have set the probability of the false alarm rate as 0.1 and we will have the probability of the detection accuracy as 0.95, which is sufficient for the real-world application [34]. In this case, under $\mathcal{H}_0$ (there is an attack), the signal is completely unknown since the CAVs have no prior information for the attack signal, and then, $\hat{\theta}^0 = \{y_k\}_{k=n}^{n+N-1}$ and

$$p\left(z_n; \hat{\theta}^0, \mathcal{H}_0\right) = \frac{1}{\left(2\pi\sigma_{\lambda}^2\right)^{3N/2}}. \tag{26}$$

Under the $\mathcal{H}_1$ hypothesis, we have

$$p\left(z_n; \hat{\theta}^1, \mathcal{H}_1\right) = \frac{1}{\left(2\pi\sigma_{\lambda}^2\right)^{3N/2}} \exp\left(-\frac{1}{2\sigma_{\lambda}^2} \| y_k^{\lambda} \|^2\right). \tag{27}$$

With (25)–(27), an attack in the position from the ego vehicle's GNSS or adjacent vehicle(s) $\mathcal{H}_1$ can be detected if

$$T(z_n) = \frac{1}{N}\sum_{k\in\Omega_n}\left(\frac{1}{2\sigma_\lambda^2}\|y_k^\lambda\|^2\right) < \gamma' \tag{28}$$

where $\gamma' = -(2/N)\ln(\gamma)$. This means that if the energy of the residual or the pseudo innovation is less than a certain threshold γ', there is no attack considered in the position measurement. Otherwise, an attack has occurred and an AI is set. As can be seen in Fig. 3, corresponding to Buffers 1 and 2, there will be two indicators AI_1 and AI_2, respectively. Then, taking AI_1 and AI_2, we have AI, which could handle the cases when the position measurement transitions from normal status to an attacked mode or from an attack mode to a normal status. The holistic attack detection process is shown in Fig. 3. After having the AI, a rule-based strategy to defend the attack in the CKIF is designed and will be discussed in Section III-C4.

Remark 5: In this work, GLRT-based attack detection is selected for our application due to its high detection accuracy and conciseness [34]. Other similar multiple-sample-based methods, such as sequential probability ratio tests, are also applicable to be integrated with the proposed delay-prediction framework [33]. Another notice is that the inputs to the detection algorithm are the residual and innovation. These pieces of information are the difference between the prior information provided by the IMU and the actual sensory measurements. These kinds of inputs are chosen because the prior IMU information is free from attacks. In other words, due to the possible attacks in the measurements, it is challenging to directly design the attack detection method based on the redundant measurements.

4) Attack Defense Method: In this section, based on the AI, a rule-based attack isolation method is designed to prevent the localization results to be affected by the attack.

Once an AI is declared for a certain sensory measurement, the measurement update will be isolated in the CKIF. Specifically, the isolation will be executed by increasing the corresponding element in the measurement matrix $\boldsymbol{R}$ to an infinite value. This operation will prevent the corresponding node to be affected by the attack. However, in the meantime, the measurements in the nodes without the attack will be continuously leveraged for the measurement update. It can be seen that as long as not all the measurements are attacked, there always exists the measurement update in the nodes of the CKIF. The worst case is that all the measurements are attacked, and then, the CKIF will run in a time update mode, meaning that the states will be estimated consecutively by integrating the acceleration from the IMU.

When the attacks disappear, the temporary changes to the measurement matrix $\boldsymbol{R}$ will be canceled and the CKIF will run normally. The details of the algorithm of the secure cooperative localization method are given in Algorithm 2.

Remark 6: It can be seen that the CKIF well fits the secure cooperative localization problem from the sensor fusion and attack defense perspectives: not only it can handle the measurements from different vehicles conveniently, i.e., when vehicles are connected or disconnected to the ego vehicles, the added or deleted measurement can be accommodated by adding or removing the corresponding node in the CKIF, but also it can deal with the attacked sensors by adapting the measurement covariance matrix. In addition, the proposed attack detection and defense method improves the resilience to the attack for the CKIF.

Algorithm 2 Secure Cooperative Localization Method

Input : $\boldsymbol{\Phi}_k^i$; z_{Gi} (position measurement from ego vehicle i) and z_{ij} (position measurement from ego vehicle's adjacent vehicles j); $\boldsymbol{H}$; $\boldsymbol{Q}^i$; $\boldsymbol{R}^i$ and $\boldsymbol{R}^j$; $\hat{\boldsymbol{x}}_{k|k-1}^j$ (prediction states of node j); initial state $\boldsymbol{x}^i(0)$;initial state covariance $\boldsymbol{P}^i(0)$

Output: $\bar{\boldsymbol{x}}_{k|k}^i$

```
while GNSS updated do
    GLRT-based attack detection for the ego vehicle measurement based on (28);
    if z_Gi is attacked then
        Isolate it from the CKIF by enlarging the element in the covariance matrix;
    else
        for j ∈ N_i do
            Run GLRT-based algorithm to test the adjacent vehicle(s) measurement;
            if z_Gi is attacked then
                Isolate it from the CKIF by enlarging the element in the covariance matrix;
            end
        end
    end
    Run Algorithm. 1;
end
```

IV. Results and Discussion

In this section, the proposed secure cooperative localization method is validated by numerical simulations. The localization results with the directed or undirected communication and attack settings are exemplified and discussed.

A. System Requirements and Case Study Settings

In the case study, the CAV platooning application with four CAVs is considered to show the detailed localization results. Also, for the statistical analysis, ten vehicles in the platooning are included to implement the proposed algorithm. On each vehicle, sensors, including an IMU to obtain the longitudinal accelerations, a normal GNSS receiver without differential corrections to obtain the positions, and a LiDAR sensor to calculate the relative distance between the ego vehicle and its neighbor(s), are required. For exchanging the sensor information, an onboard unit (Celluler-V2X or DSRC) is necessary to transmit and receive the shared information from the CAVs wirelessly with a directed or undirected communication topology. In addition to that, each vehicle should be installed with a computer that can process the sensory

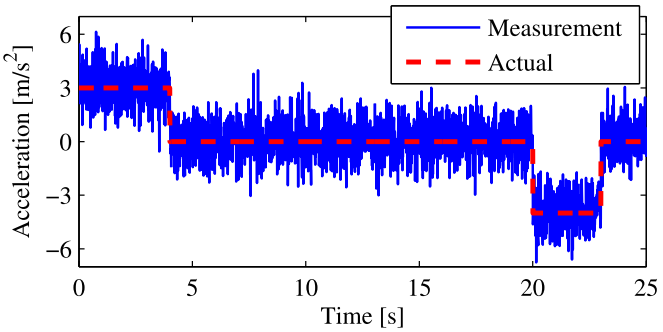


Fig. 4. Acceleration of the leading vehicle. The red line shows the actual acceleration and the blue line represents the acceleration measurement from the IMU.

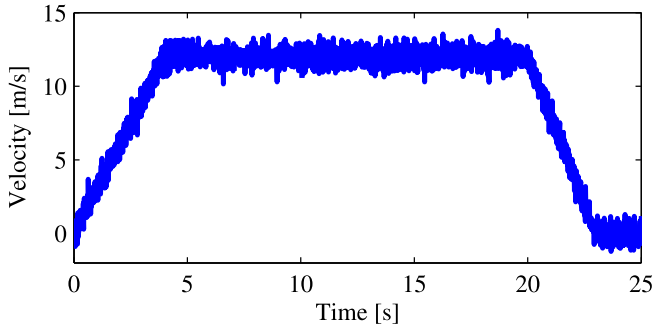


Fig. 5. Velocity of the leading vehicle.

data and run the proposed algorithms. The requirement for the computer is similar to our previous work in [46].

To simulate the real sensory measurement, noise is added to these sensors. For the acceleration measurement, it is assumed the noise satisfies $\omega_a \sim \mathcal{N}(0, (1\ \text{m}^2)^2)$ [47]. $\mathcal{N}(\mu, \sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. In order to simulate the acceleration zero-bias instability of the IMU, a constant bias error between -0.1 and 0.1 m/s^2 is added to the acceleration (0.05 m/s^2 in our case). The noise of the GNSS position satisfies $\eta_G \sim \mathcal{N}(0, 3\ \text{m}^2)$ [48]. During the platooning operation, the GNSS of the vehicles is compromised by a drift attack, which is larger than 3σ of the noise, at a certain time. The relative distance obtained from the LiDAR senor has the noise $\eta_R \sim \mathcal{N}(0, 1\ \text{m}^2)$ [49]. The following distance of the CAVs in the platoon is 30 m and it is assumed the controllers in the CAVs can make the vehicles keep the same longitudinal acceleration. Note that although, in the real platoon application, there will be transient response between vehicles, as long as the sensory information between CAVs can be measured and shared, the control behavior will not affect our cooperative localization algorithm. The acceleration of the leading vehicle (green vehicle) in Fig. 1 is shown in Fig. 4. The red line represents the actual acceleration of the leading vehicle and the blue line shows the noisy acceleration measured by the IMU. First, in $t = 0$–4 s, the formation accelerates at 3 m/s^2 and the velocity will reach 12 m/s shown in Fig. 5. After that, the formation will keep this velocity until t=20 s. Then, between $t = 20$ and 23 s, the formation starts to decelerate at -4 m/s^2 and the CAVs will stop at $t = 23$ s.

TABLE I
ATTACKS INJECTED IN THE POSITION MEASUREMENTS

Vehicle	Value of attack	Time interval
Vehicle 1	-10m	t =8-14s and t =15-19s
Vehicle 2	10m	t =10-13s and t =20-23s
Vehicle 3	-15m	t =2-5s and t =13-16s
Vehicle 4	\	\

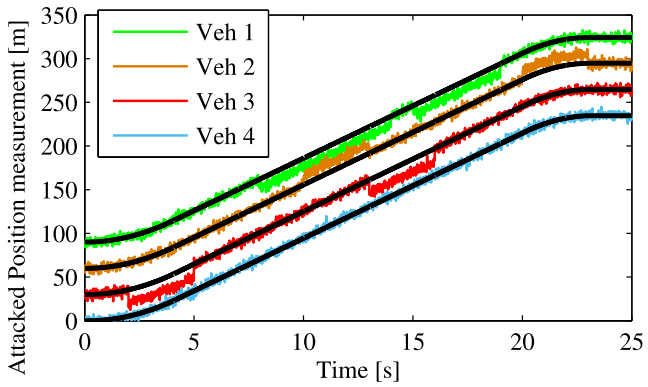


Fig. 6. Position of CAVs. Vehicles 1–4 mean the CAVs from the green vehicle to the blue vehicle in Fig. 1 driven with the acceleration in Fig. 4.

B. Results

1) Localization With Directed Communication Topology: The secure cooperative localization results of the CAVs with the directed communication are discussed in this section in terms of the attack detection and the position estimation results.

First, in order to simulate the drift attacks in the measured position, an offset position error is added to the GNSS position measurement at certain intervals during the platooning. Note that, since in (11), the attacks from the GNSS position of other adjacent vehicles or the relative distance result in the attacks f, the contributions from the attacks with different sources to the position measurement are the same. For attack detection, as long as there is an attack in f from the GNSS position or relative distance, it can be detected without knowing that it comes from the GNSS position or relative distance. Thus, to make the validation concise, only the attacks in the GNSS position are used in the case study. The attacked GNSS position measurements are shown in Fig. 6. The curves of Vehicles 1–4 represent the position measurements from the GNSS of all vehicles in Fig. 1. The GNSS in Vehicles 1–3 is attacked and the injected attacks are shown in Table I. The attacks from communication and sensors are considered. For simulating the attacks such as data tampering in the communication, the GNSS position of Vehicles 1–3 in the platoon is attacked by adding a drift error. Regarding the attacks for the sensors, the GNSS in a certain region is spoofed such as in $t = 10$–13 s, both Vehicles 2 and 3 are attacked.

The actual position of Vehicles 1–4 is represented by the center black lines in each colored curve in Fig. 6. The attack detection results are shown in Fig. 7, the black dashed line gives the actual status if an attack occurs, and the red line means the detected AI α for Vehicles 1–3. To be specific,

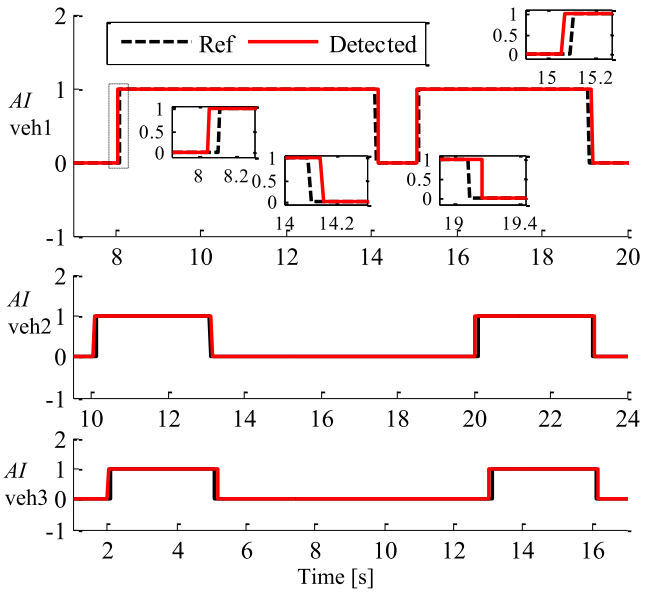


Fig. 7. Attack detection results. AI denotes the attack indicator and veh means vehicle.

for instance, the partial enlargement views in Fig. 7 provide the exact moments when the attacks happen or disappear for Vehicle 1. Since the time delay τ or the prediction horizon in our delay-prediction framework is set as 0.1 s, for our implementation, we can see that α veh 1 for both the reference and detected is behind the actual moment when the first attack occurs at $t = 8$ s given in Table I. However, α for Vehicle 1 (red line) can be set prior to the reference (black dashed line), meaning that the delayed estimator (KCIF) can be notified in advance when the attack is going to occur. This is because based on Buffer 2 in Fig. 3, the attack can be detected. Then, the delayed estimator KCIF is able to isolate the attack ahead. Also, from Fig. 7, it can be seen that the attack can also be detected accurately without any false positive. When the GNSS measurement transits from an attacked status to normal status ($t = 14$ s and $t = 19$ s in Fig. 7), it can be seen that the detection is behind the reference for a short term (less than 0.1 s) due to using Buffer 1 in Fig. 3 for the recovering process. The cost of this delay is that the KCIF continues to keep in the attack isolation mode for a short term without leveraging the valid GNSS measurement during this delay. However, the cost is negligible since, in the isolation mode, the corresponding node in the KCIF will run in the time update mode based on the IMU information and the cumulative error is small. Similar attack detection results can be seen for Vehicle 2 (α Vehicle 2) and Vehicle 3 (α Vehicle 3) with drift errors with different magnitudes. In addition, during t = 10–13 s, when both the GNSS measurements in Vehicles 1 and 2 are attacked, the attack detection algorithm can still provide the accurate detection results. Then, based on these estimated states in the KCIF, through (18), the states at the current moment can be predicted by using the information from the IMU. Thereby, the attack detection results demonstrate that the latency issue of the GLRT attack detection has been addressed and our

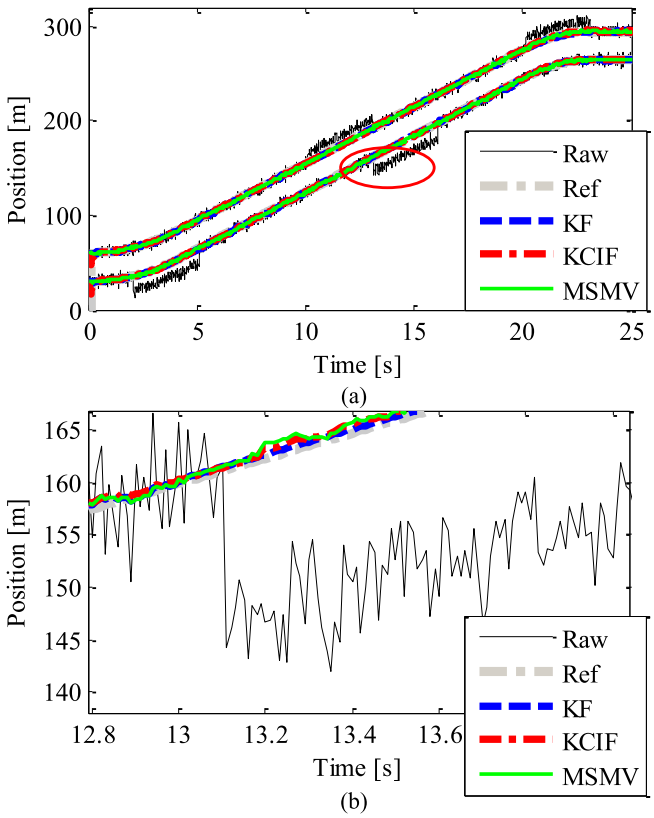


Fig. 8. Position error of Vehicles 2 and 3 with directed communication topology. MSMV, KF, and KCIF mean the results of the state-of-the-art cooperative localization method MSMV in [26], the normal KF, and the KCIF, respectively. Veh means vehicle.

proposed delay-prediction framework can detect the attack accurately.

After having the attack detection results, the KCIF is able to defend the attacks by the rule-based attack isolation approach as in Section III-C4 and estimate the position of the CAVs. The localization results and the partial enlargement for Vehicles 2 and 3 are given in Fig. 8 to show the performance of the consensus estimation framework. For the comparison purpose, the position results from a normal KF, which is based on the sensors in an individual vehicle, and a state-of-the-art approach multi-sensor multi vehicle (MSMV) in [26] are also presented. To implement the KF and MSMV, both of them are integrated with the proposed attack detection algorithm. Note that the normal KF can only fuse the information from an individual vehicle, but the CKIF and MSMV can leverage the information from both the ego vehicle and other CAVs. This is the major difference between the normal KF and the other two methods. From Fig. 8(a), it can be seen that both the positions from the KF, KCIF, and MSMV follow well with the reference to some extent because both the methods have been integrated with our proposed attack detection method, preventing them from being attacked. In another aspect, in the partial enlargement figure shown in Fig. 8(b), we can see that the attack in the raw position does not affect the estimated position of KF, KCIF, and MSMV. From the position error in Table II and Fig. 9, differences can be identified between the KF and the cooperative localization approaches, including

TABLE II
LOCALIZATION ACCURACY COMPARISON

Method	Vehicle 2		Vehicle 3	
	AME	RMSE	AME	RMSE
KF	0.504	0.611	0.411	0.509
MSMV $\mathcal{G}_d$	0.463	0.596	0.384	0.474
KCIF $\mathcal{G}_d$	0.397	0.486	0.309	0.386
MSMV $\mathcal{G}_u$	0.431	0.538	0.351	0.468
KCIF $\mathcal{G}_u$	0.334	0.398	0.308	0.378
MSMV $\mathcal{G}_t$(4 vehicles)	0.357	0.442	0.357	0.442
KCIF $\mathcal{G}_t$(4 vehicles)	0.329	0.391	0.309	0.380
MSMV $\mathcal{G}_t$(10 vehicles)	0.222*	0.279	0.223	0.281
KCIF $\mathcal{G}_t$(10 vehicles)	0.228	0.267*	0.222*	0.268*

KF, MSMV $\mathcal{G}_d$, KCIF $\mathcal{G}_d$, MSMV $\mathcal{G}_u$, KCIF $\mathcal{G}_u$, MSMV in [26] $\mathcal{G}_t$, and KCIF $\mathcal{G}_t$ denote the normal Kalman filter, MSMV with the directed communication topology, KCIF with the directed communication topology, MSMV with the undirected communication topology, KCIF with the undirected communication topology, MSMV with totally connected topology, and KCIF with totally connected topology; the best performance is also marked by *. AME and RMSE mean the absolute mean error and root mean square error in meter, respectively.

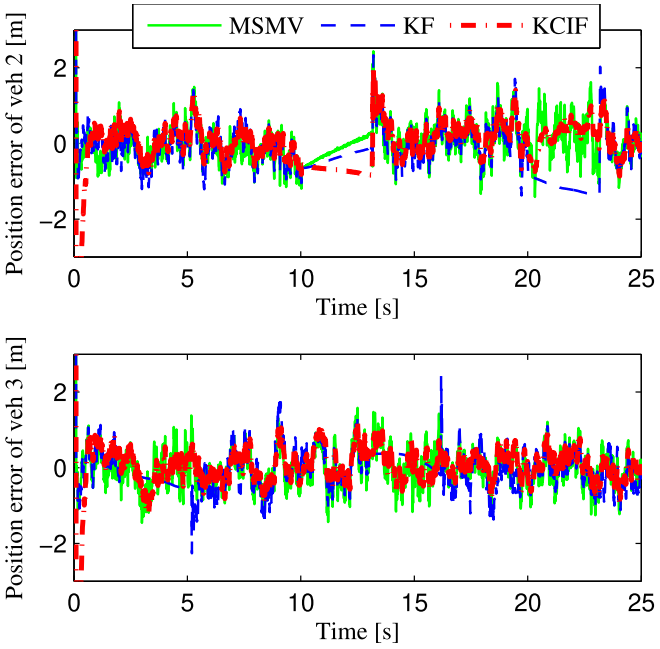


Fig. 9. Position error of Vehicles 2 and 3 with directed communication topology. Veh means vehicle.

KCIF and MSMV. For Vehicle 2, it can access the sensory measurements of its front vehicle (Vehicle 1), which include the position from the GNSS in Vehicle 2 and (11) based on the relative distance obtained from the sensor and the GNSS sensors in Vehicle 1. They can be leveraged in the KCIF and MSMV to estimate the position when there is a certain sensor attacked. Thus, the absolute mean error (AME) and RMSE of the position from both KCIF and MSMV are smaller than that from the KF. In $t = 10$–13 s, both the GNSSs in Vehicles 1 and 2 are attacked, and therefore, the position errors for the KF, KCIF, and MSMV are similar and drift a bit because there are no measurements to correct the errors coming from the IMU. However, in $t = 20$–23 s, due to the attack in Vehicle 2, the KF runs in a time update mode without

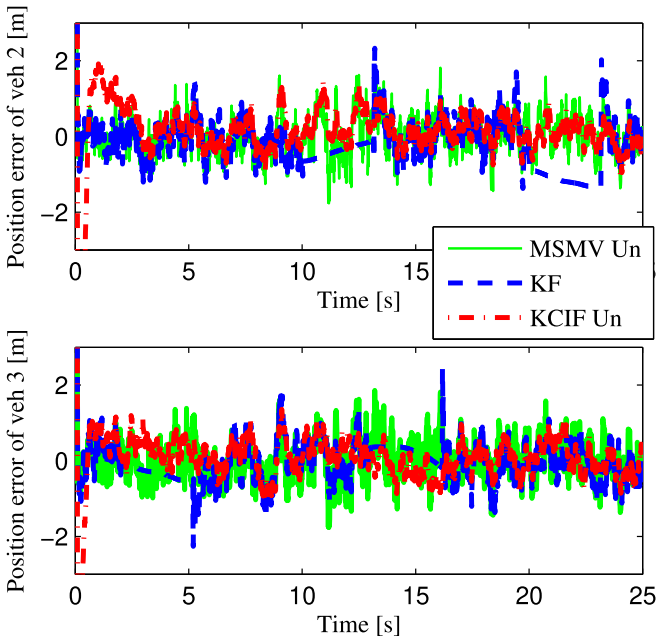


Fig. 10. Position error of Vehicles 2 and 3 with undirected and fully connection communication topology. KF, MSMV Un, and KCIF Un mean the results of the normal KF, the state-of-the-art cooperative localization method MSMV in [26] with the undirected communication topology, and the KCIF with the undirected communication topology, respectively.

measurement updates and the position error starts to drift due to the acceleration bias error in the IMU. For the KCIF and MSMV, since the GNSS in Vehicle 1 is not attacked in $t = 20$–23 s, this information could still be leveraged in the KCIF to correct the errors from the prediction process. For Vehicle 3, the position error drifts for the KF when there is an attack in $t = 2$–5 s and $t = 13$–16 s but for the KCIF and MSMV since there is no overlap when the attacks happen between Vehicles 2 and 3, which can be inferred from Table I. The measurement updates continue all the time to correct the errors from the prediction process. Based on the comparison between the KF and the cooperative localization methods (KCIF and MSMV), it can be seen that exhausting the information cooperatively from the ego vehicle and its adjacent vehicle in a directed communication topology in the KCIF improves the redundancy for the localization algorithm and makes it more secure regarding the attacks. Comparing the KCIF and MSMV, we can see that the KCIF method shows superior performance regarding AME and RMSE. This is because KCIF is a suboptimal version of the Kalman-consensus filter in [27], which has shown better performance than the existing distributed KFs such as the MSMV in [26].

2) Localization With Undirected Communication Topology: To further investigate the performance of our method in cases where more vehicles cooperate with each other, localization results for the CAVs with undirected communication are presented. Under the undirected communication topology, the CAV can not only share information with its front CAV cooperatively but also with its rear CAV, meaning that more information can be fed into the KCIF and MSMV. From the position errors shown in Fig. 10, it can be seen that in $t = 10$–13 s, although the GNSS in Vehicles 1 and 2 is

attacked, the derived position from Vehicle 3 is leveraged in the KCIF and MSMV and the drift error in Fig. 9 has been compensated. With more vehicles connected, the security of our localization algorithm is improved because there is more redundant information that can be fused to compensate for the influence of the attacks. From another aspect, for the localization accuracy, it can be seen from Table II that, from the KF to the KCIF with undirected communication $\mathcal{G}_u$, the accuracy increases in terms of AME and RMSE. With more information from the vehicles that are used, from Table II, the AME and RMSE of both KCIF and MSMV with $\mathcal{G}_u$ decrease to some extent compared with those of the KF. In addition, the proposed secure cooperative localization in cases with more CAVs totally connected in $\mathcal{G}_t$ (4–10 CAVs) is tested. From Table II, the accuracy in terms of the AME and RMSE in $\mathcal{G}_t$ where the sensory information from all ten vehicles is used in our estimation algorithm improves by 35.4% and 36.6%, respectively, compared with the one in $\mathcal{G}_d$ where sensory information from only two vehicles is used. It is worth noting that, with the number of vehicles increasing from four to ten, it can be found that the difference between the KCIF and MSMV decreases. It is possible because the random noise in the position can not only be compensated by the filter algorithm but also by the average effect. Therefore, it shows that when more vehicles share information cooperatively, both localization accuracy and security can be advanced further.

Please note that in this article, it is assumed that the shared information can be provided by radar, camera, or LiDAR through V2V/V2X communication. However, in a real application, due to the limitation of sensor range or occlusion issues of these sensors in some scenarios such as roads with high traffic density, the object cannot be detected, and thus, the interdistance between the vehicles is not accessible if a traditional object detection algorithm based on sensors in an individual vehicle is used [49]. Accordingly, in this case, the CAV may not be able to share the interdistance between itself and its neighbor CAVs and the communication edge in the cooperative localization algorithm needs to be disconnected. The consequence of this issue is similar to a measurement that has been attacked. However, in our recent paper [49], if a V2V-based object detection algorithm is used, the limitation of sensor range and occlusion issues can be resolved, and thus, the robustness of our cooperative localization can be enhanced further.

V. Conclusion

In this article, a secure cooperative localization method for the CAVs is proposed and validated by numerical simulations. It can be concluded from the following results.

1) The sensory information from the ego vehicle and the cooperation with its adjacent vehicle(s) in a directed or undirected communication topology can be well leveraged by the consensus estimation. To some extent, more nodes in the CKIF enable the algorithm to have higher localization accuracy and better security.
2) The injected attacks in the sensory measurement can be detected accurately by the GLRT-based method and the temporal lag for the decision from the GLRT-based method has been resolved. With the detection results, the proposed secure cooperative localization has shown resilient performance to the attacks. Both the security and the localization accuracy have been improved compared with a normal centralized KF.

Acknowledgment

This project belongs to OpenCDA ecosystem.

References

[1] Y. Li, Z. He, Y. Li, Z. Gao, R. Chen, and N. El-Sheimy, "Enhanced wireless localization based on orientation-compensation model and differential received signal strength," *IEEE Sensors J.*, vol. 19, no. 11, pp. 4201–4210, Jun. 2019.

[2] J. Betz et al., "Autonomous vehicles on the edge: A survey on autonomous vehicle racing," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 458–488, 2022.

[3] G. Chen et al., "Planning and tracking control of full drive-by-wire electric vehicles in unstructured scenario," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, 2023.

[4] G. Elghazaly, R. Frank, S. Harvey, and S. Safko, "High-definition maps: Comprehensive survey, challenges, and future perspectives," *IEEE Open J. Intell. Transp. Syst.*, vol. 4, pp. 527–550, 2023.

[5] A. Gholamhosseinian and J. Seitz, "Vehicle classification in intelligent transport systems: An overview, methods and software perspective," *IEEE Open J. Intell. Transp. Syst.*, vol. 2, pp. 173–194, 2021.

[6] E. Thonhofer et al., "Infrastructure-based digital twins for cooperative, connected, automated driving and smart road services," *IEEE Open J. Intell. Transp. Syst.*, vol. 4, pp. 311–324, 2023.

[7] R. Xu et al., "The OpenCDA open-source ecosystem for cooperative driving automation research," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 4, pp. 2698–2711, Apr. 2023.

[8] S. Nallamothu et al., "Detailed concept of operations: Transportation systems management and operations/cooperative driving automation use cases and scenarios," United States. Federal Highway Admin., Tech. Rep. FHWA-HRT-20-064, 2020.

[9] W. Liu et al., "A systematic survey of control techniques and applications in connected and automated vehicles," *IEEE Internet Things J.*, early access, Aug. 21, 2023, doi: 10.1109/JIOT.2023.3307002.

[10] M. Hua, G. Chen, B. Zhang, and Y. Huang, "A hierarchical energy efficiency optimization control strategy for distributed drive electric vehicles," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, vol. 233, no. 3, pp. 605–621, Feb. 2019.

[11] M. T. Arafin and K. Kornegay, "Attack detection and countermeasures for autonomous navigation," in *Proc. 55th Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2021, pp. 1–6.

[12] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough, and A. Mouzakitis, "A survey of the state-of-the-ar t localization techniques and their potentials for autonomous vehicle applications," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 829–846, Apr. 2018.

[13] Y. Li et al., "Toward location-enabled IoT (LE-IoT): IoT positioning techniques, error sources, and error mitigation," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4035–4062, Mar. 2021.

[14] W. Liu, X. Xia, L. Xiong, Y. Lu, L. Gao, and Z. Yu, "Automated vehicle sideslip angle estimation considering signal measurement characteristic," *IEEE Sensors J.*, vol. 21, no. 19, pp. 21675–21687, Oct. 2021.

[15] W. Liu, L. Xiong, X. Xia, Y. Lu, L. Gao, and S. Song, "Vision-aided intelligent vehicle sideslip angle estimation based on a dynamic model," *IET Intell. Transp. Syst.*, vol. 14, no. 10, pp. 1183–1189, Oct. 2020.

[16] L. Gao, L. Xiong, X. Xia, Y. Lu, Z. Yu, and A. Khajepour, "Improved vehicle localization using on-board sensors and vehicle lateral velocity," *IEEE Sensors J.*, vol. 22, no. 7, pp. 6818–6831, Apr. 2022.

[17] K.-W. Chiang, G.-J. Tsai, Y.-H. Li, Y. Li, and N. El-Sheimy, "Navigation engine design for automated driving using INS/GNSS/3D LiDAR-SLAM and integrity assessment," *Remote Sens.*, vol. 12, no. 10, p. 1564, May 2020.

[18] X. Xia, N. P. Bhatt, A. Khajepour, and E. Hashemi, "Integrated inertial-LiDAR-based map matching localization for varying environments," *IEEE Trans. Intell. Vehicles*, early access, Jul. 26, 2023, doi: 10.1109/TIV.2023.3298892.

[19] Y. Li, Z. He, Z. Gao, Y. Zhuang, C. Shi, and N. El-Sheimy, "Toward robust crowdsourcing-based localization: A fingerprinting accuracy indicator enhanced wireless/magnetic/inertial integration approach," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 3585–3600, Apr. 2019.

[20] F. B. Günay, E. Öztürk, T. Çavdar, Y. S. Hanay, and A. U. R. Khan, "Vehicular ad hoc network (VANET) localization techniques: A survey," *Arch. Comput. Methods Eng.*, vol. 28, no. 4, pp. 3001–3033, Jun. 2021.

[21] F. Lobo, D. Grael, H. Oliveira, L. Villas, A. Almehmadi, and K. El-Khatib, "Cooperative localization improvement using distance information in vehicular ad hoc networks," *Sensors*, vol. 19, no. 23, p. 5231, Nov. 2019.

[22] M. Elazab, A. Noureldin, and H. S. Hassanein, "Integrated cooperative localization for vehicular networks with partial GPS access in urban canyons," *Veh. Commun.*, vol. 9, pp. 242–253, Jul. 2017.

[23] M. A. Hossain, I. Elshafiey, and A. Al-Sanie, "Cooperative vehicle positioning with multi-sensor data fusion and vehicular communications," *Wireless Netw.*, vol. 25, no. 3, pp. 1403–1413, Apr. 2019.

[24] G. Xiao, X. Song, H. Cao, S. Zhao, H. Dai, and M. Li, "Augmented extended Kalman filter with cooperative Bayesian filtering and multi-models fusion for precise vehicle localisations," *IET Radar, Sonar Navigat.*, vol. 14, no. 11, pp. 1815–1826, Nov. 2020.

[25] S. Ma, F. Wen, X. Zhao, Z.-M. Wang, and D. Yang, "An efficient V2X based vehicle localization using single RSU and single receiver," *IEEE Access*, vol. 7, pp. 46114–46121, 2019.

[26] P. Yang, D. Duan, C. Chen, X. Cheng, and L. Yang, "Multi-sensor multi-vehicle (MSMV) localization and mobility tracking for autonomous driving," *IEEE Trans. Veh. Technol.*, vol. 69, no. 12, pp. 14355–14364, Dec. 2020.

[27] R. Olfati-Saber, "Kalman-consensus filter: Optimality, stability, and performance," in *Proc. 48th IEEE Conf. Decis. Control (CDC), 28th Chin. Control Conf.*, Dec. 2009, pp. 7036–7042.

[28] M. Pirani et al., "Cooperative vehicle speed fault diagnosis and correction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 2, pp. 783–789, Feb. 2019.

[29] B. Gong, S. Wang, M. Hao, X. Guan, and S. Li, "Range-based collaborative relative navigation for multiple unmanned aerial vehicles using consensus extended Kalman filter," *Aerosp. Sci. Technol.*, vol. 112, May 2021, Art. no. 106647.

[30] Z. Ju, H. Zhang, and Y. Tan, "Deception attack detection and estimation for a local vehicle in vehicle platooning based on a modified UFIR estimator," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3693–3705, May 2020.

[31] T. Yang and C. Lv, "A secure sensor fusion framework for connected and automated vehicles under sensor attacks," 2021, *arXiv:2103.00883*.

[32] J. Shen, J. Y. Won, Z. Chen, and Q. A. Chen, "Drift with devil: Security of multi-sensor fusion based localization in high-level autonomous driving under GPS spoofing," in *Proc. 29th USENIX Secur. Symp. (USENIX Security)*, 2020, pp. 931–948.

[33] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 3, pp. 636–653, May 2010.

[34] I. Skog, P. Handel, J. O. Nilsson, and J. Rantakokko, "Zero-velocity detection—An algorithm evaluation," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 11, pp. 2657–2666, Nov. 2010.

[35] Y. Guo and J. Ma, "SCoPTO: Signalized corridor management with vehicle platooning and trajectory control under connected and automated traffic environment," *Transportmetrica B, Transp. Dyn.*, vol. 9, no. 1, pp. 673–692, Jan. 2021.

[36] Y. Zheng, S. E. Li, K. Li, and L.-Y. Wang, "Stability margin improvement of vehicular platoon considering undirected topology and asymmetric control," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1253–1265, Jul. 2016.

[37] Z. Wang, Y. Bian, S. E. Shladover, G. Wu, S. E. Li, and M. J. Barth, "A survey on cooperative longitudinal motion control of multiple connected and automated vehicles," *IEEE Intell. Transp. Syst. Mag.*, vol. 12, no. 1, pp. 4–24, Spring. 2020.

[38] Y. Li, X. Niu, Y. Cheng, C. Shi, and N. El-Sheimy, "The impact of vehicle maneuvers on the attitude estimation of GNSS/INS for mobile mapping," *J. Appl. Geodesy*, vol. 9, no. 3, pp. 183–197, Jan. 2015.

[39] X. Xia, E. Hashemi, L. Xiong, and A. Khajepour, "Autonomous vehicle kinematics and dynamics synthesis for sideslip angle estimation based on consensus Kalman filter," *IEEE Trans. Control Syst. Technol.*, vol. 31, no. 1, pp. 179–192, Jan. 2023.

[40] C. Kwon, W. Liu, and I. Hwang, "Security analysis for cyber-physical systems against stealthy deception attacks," in *Proc. Amer. Control Conf.*, Jun. 2013, pp. 3344–3349.

[41] R. Wang, Z. Xiong, J. Liu, J. Xu, and L. Shi, "Chi-square and SPRT combined fault detection for multisensor navigation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 3, pp. 1352–1365, Jun. 2016.

[42] A. Almagbile, J. Wang, and W. Ding, "Evaluating the performances of adaptive Kalman filter methods in GPS/INS integration," *J. Global Positioning Syst.*, vol. 9, no. 1, pp. 33–40, Jun. 2010.

[43] X. Xia, E. Hashemi, L. Xiong, A. Khajepour, and N. Xu, "Autonomous vehicles sideslip angle estimation: Single antenna GNSS/IMU fusion with observability analysis," *IEEE Internet Things J.*, vol. 8, no. 19, pp. 14845–14859, Oct. 2021.

[44] L. Khosravian, J. Trumpf, and R. E. Mahony, "State estimation for nonlinear systems with delayed output measurements," in *Proc. CDC*, Dec. 2015, pp. 6330–6335.

[45] W. Ding, J. Wang, C. Rizos, and D. Kinlyside, "Improving adaptive Kalman estimation in GPS/INS integration," *J. Navigat.*, vol. 60, no. 3, pp. 517–529, Sep. 2007.

[46] Z. Meng, X. Xia, R. Xu, W. Liu, and J. Ma, "HYDRO-3D: Hybrid object detection and tracking for cooperative perception using 3D LiDAR," *IEEE Trans. Intell. Vehicles*, early access, Jun. 12, 2023, doi: 10.1109/TIV.2023.3282567.

[47] STMicroelectronics. *ASM330LHB Specification*. Accessed: Apr. 26, 2023. [Online]. Available: https://www.st.com/resource/en/datasheet/asm330lhb.pdf

[48] Ublox. *Ublox ZED-F9T-10B Specification*. [Online]. Available: https://content.u-blox.com/sites/default/files/ZED-F9T-10B_DataSheet_UBX-20033635.pdf

[49] X. Xia et al., "An automated driving systems data acquisition and analytics platform," *Transp. Res. C, Emerg. Technol.*, vol. 151, Jun. 2023, Art. no. 104120.