2021 Fourth International Conference on Connected and Autonomous Driving (MetroCAD) | 978-1-6654-4594-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/METROCAD51599.2021.00014

Physical Invariant Based Attack Detection for Autonomous Vehicles: Survey, Vision, and Challenges

Francis Akowuah and Fanxin Kong
Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse NY
feakowua@syr.edu, fkong03@syr.edu

***Abstract*—Automobiles continue to become more autonomous and connected as increasingly integrating with information technology. Meanwhile, this advance also comes with a higher risk of various security violations on vehicles. In this paper, we study how to detect attacks on autonomous vehicles, and specially focus on physical invariant-based attack detection. A physical invariant (PI) is defined as a property that a physical system always holds, i.e., the evolution of system states (usually measured by sensors) follows immutable physical laws. We first discuss existing research efforts of PI-based attack detection and classify them according to the knowledge of physical invariants and sensor redundancy. Then, we point out several critical challenges on attack detection research efforts including data sets, benchmark and testbeds, and evaluation metrics. Finally, we highlight open problems that offer promising research opportunities.**

***Index Terms*—attack detection, autonomous driving, self-driving, autonomous vehicles, physical invariant**

I. Introduction

Automobiles have transitioned from once-closed architectures to open architectures due to the integration with information technology (IT). Increasingly, V2X technologies are getting modern vehicles more connected to the outside. The integration has enabled the development of many safety features such as collision detection, lane departure warning system, blind-spot information system, adaptive headlights, night vision, driver fatigue recognition, etc. It has also enabled convenient features such as parallel park assistance, head-up display, fingerprint entry/ignition/personalization, infotainment and telematics apps, internet connection, among others [1]. Utilizing sensor technology, the autonomy of automobiles continues to increase as seen in adaptive cruise control and self-driving efforts. The benefits of autonomous driving include increased traveling speed, decreased traffic, reduced emissions, and the extra time to perform other tasks during a commute [2].

However, the integration with IT and sensory technology has also led to increased system complexity and has exposed the autonomous vehicle to a number of cyber and physical attacks. Cyber attacks compromise the computing and networking components of the autonomous vehicle system and it includes attacks such as buffer overflow, DNS, and TCP attacks. Usually, cyber attacks are deployed through the injection of malware, software, or by unauthorized access to elements of the communication network [13]. Typical examples of cyber attacks have been demonstrated in [14]–[18]. Physical attacks, on the other hand, perturbs the physical environment of the autonomous vehicle such that it allows the injection of malicious signals into sensors and actuators. In most cases, attackers exploit the same physical channels that the target sensor uses for its operation such that it results in manipulation or disruption of sensor readings. For example, researchers in [3] demonstrate a contactless attack that uses ultrasound against ultrasonic sensors, radio against MMW radars, and laser against cameras. These sensor attacks caused Tesla Model S's ”blindness” and malfunction. A similar remote attack on camera and LiDAR is demonstrated in [7]. Shoukry et al. [19] showed how non-invasive attacks on wheel speed sensors influenced Anti-lock Braking Systems (ABS) of a vehicle to malfunction. It must be noted that ultrasonic sensors, MMW radars, cameras, and LiDAR are essential sensors that enable self-driving vehicles to function. Hence, successful attacks on sensors can have devastating effects on the system user, the system itself, and the environment at large. Table. I provides brief information about environmental sensors aiding self-driving and also gives references to attacks on these sensors.

Due to the safety-critical roles that these autonomous systems play, it is important to provide defense mechanisms for them. The research community has responded to this need and has proposed a number of solutions. To this extend, proposed defense mechanisms have involved (1) attack detection and (2) attack recovery. The former proposes solutions that raise alerts when an attack is determined to occur whereas the latter seeks to mitigate the effects of the attacks by proposing measures that enable the continuous operation of the system, even in the midst of attacks. We focus on attack detection in this work.

In this paper, we study how to detect attacks on autonomous vehicles, and specially focus on physical invariant-based attack detection. A physical invariant (PI) is defined as a property that a physical system always holds, i.e., the evolution of system states (usually measured by sensors) follows immutable physical laws. We first discuss existing research efforts of PI-based attack detection and classify them according to the knowledge of physical invariants and sensor redundancy. Then, we point out several critical challenges on attack detection

978-1-6654-4594-8/21/$31.00 ©2021 IEEE
DOI 10.1109/MetroCAD51599.2021.00014

TABLE I
A SUMMARY OF SENSORS THAT ENABLE AUTONOMOUS DRIVING SYSTEMS TO PERCEIVE THEIR ENVIRONMENT. ATTACKERS TARGET THESE ENVIRONMENTAL SENSORS AND OTHER TYPES OF SENSORS TO CAUSE MALFUNCTION AND/OR DAMAGE TO THE AUTONOMOUS VEHICLE.

Sensor	**Physical Principle / Signal**	**Range**	**Usage**	**Attack Reference**
Ultrasonic	Ultrasound	Short range/Proximity	Parking Assistance	[3]–[6]
Camera	Visible light	Short range	Traffic sign recognition Obstacle recognition Lane departure warning	[3] [7]
GPS	Microwaves	Global	Navigation Time	[8], [9]
Radar	Millimeter waves (microwave)	Short range Medium range Long range	Blind-spot warning Cross-traffic alert Collision avoidance Adaptive cruise control	[3]
LiDAR	Infrared	Long range	Collision avoidance Pedestrian detection	[7], [10]–[12]

including data sets, benchmark and testbeds, and evaluation metrics. Finally, we highlight open problems that offer promising research opportunities.

Existing surveys such as [20] considers the use of physical invariants in detecting attacks in a number of cyber-physical domains including smart grid, industrial control systems, etc. Given that each domain has distinct properties, such broad coverage does not ultimately detail the efforts and challenges that are unique to autonomous vehicles. We fill in this gap by focusing on attack detection in autonomous vehicles.

Our contributions include:

- a systematic survey of attack detection research efforts in autonomous vehicles that is presented using a new classification. The classification is based on (i) the knowledge of physical invariants required to build a model that approximates the nominal system behavior and (ii) sensor redundancy.
- identifying the limitations and challenges in undertaking attack detection research in autonomous vehicles. The identified limitation and challenges are not only applicable to attack detection but also other security research involving AVs.
- identifying and enumerating inadequately explored challenges or open problems as well as offering possible solutions.
- enumerate the vision for attack detection in AVs and present research opportunities.

The remainder of this paper is organized as follows. Section II provides a brief background information. We describe the two taxonomies in sections III and IV. The challenges that researchers face is discussed in section V. We point out open problems and research opportunities in VI and conclude the paper in section VII.

II. PRELIMINARIES

A. *Scope of study*

While the autonomous vehicle faces both cyber attacks and physical attacks we only focus on the latter. Defenses against cyber attacks are relatively advanced due to the many traditional cybersecurity techniques already available. Comparatively, proposed solutions for defending against physical attacks are few and more challenging. We survey research efforts that seek to detect physical attacks in autonomous vehicles. Since the physical properties are often measured with sensors, we focus on research works that address the detection of false sensor attacks.

We systematically survey publications in the past ten years (2010-2020) that appeared in the computer security and system conferences such as CCS, AsiaCCS, Usenix Security, ACSAC, S&P, NDSSand ICCPS. In order to increase coverage, we included relevant papers that were cited by these papers as well as those which cited these papers. The selection criteria for including a paper in the survey were papers (to the best of our knowledge) where the proposed attack detection solution monitors the evolution of the system states (usually measured by sensors), actuator and/or control signals, and then raise an alarm whenever the observed signals digress from a model of the physical system.

B. *Physical invariant*

A physical invariant is the property of a physical system that always holds under some transformation. They remain unchanged due to immutable physical laws. Compared with information technology systems, one unique property of physical systems such as autonomous vehicles is that the physical evolution of system states has to follow the laws of nature. For example, when a vehicle is uphill and no brake is applied, the force of gravity pulls the vehicle downward. The velocity, acceleration, position, and orientation of the vehicle can be confirmed or measured by sensors under normal conditions.

Successful attacks on sensors often cause the sensors to report values that are in violation of the physical invariant of the physical system. Following the example given above, a successful spoofing attack on the GPS can cause the GPS sensor to report values that indicate the vehicle is stationary although the vehicle is actually moving downwards. Hence developing mechanisms to detect the violation has been the basis of many attack detection research efforts. We discuss details of such mechanisms and provide a classification of them in section III.

C. Overview of autonomous vehicles

An autonomous vehicle (AV) can be defined as a machine that operates and performs missions or tasks under its own power, without human input or supervision. It is worth mentioning that autonomous vehicles do not always operate fully autonomously but can also operate semi-autonomously. That is, a human operator may maintain control of the vehicle, however, some control functions of the vehicle are autonomous. For example, even though most current automobiles are not fully autonomous, manufacturers incorporate semi-autonomous features such as adaptive cruise control and self-parking assistance.

Autonomous vehicles discover and navigate their environment by collecting and combining information from various sensors such as ultrasonic sensors, cameras, GPS, radio detection and ranging (radar), light detection and ranging (LiDAR), and on-board computers. Ultrasonic sensors, designed for low-speed scenarios such as parking assistance, are proximity sensors that detect objects within several meters from the vehicle. They detect objects by transmitting and receiving mechanical waves. While front-looking cameras take images that allow traffic sign recognition and lane departure warning, rear-facing cameras assist the driver when reversing or parking. Cameras rely on visible light to take images. GPS provides geographical location and timing information. Radars are used in various scenarios depending on their range [3]. The short-range radars are used for blind-spot and cross-traffic alerts. The medium-range radars together with LiDARs are used for collision avoidance and pedestrian detection. The long-range radars are useful for high-speed adaptive cruise control. Radars rely on millimeter waves whereas LiDARs rely on infrared. Many other sensors are used in autonomous vehicles to measure various physical phenomena. Table. I shows a summary of these sensors that enable autonomous driving systems to perceive and navigate their environment. Also, a subset of sensors that were extracted from an automotive dataset [21] are shown in Table II.

Autonomous vehicles may also manage communication from other autonomous vehicles. [22]. This is enabled by wireless networking around the vicinity of the vehicle known as vehicle to vehicle (V2V). One of the reasons for connecting vehicles is to rapidly share a vehicle's data such as speed, location, activity, camera images, etc so that collisions can be prevented. For instance, the cameras in nearby cars can take different angles of the environment, which when put together, can assist a vehicle's self-driving system to make better decisions than images from only its cameras. Making a good decision based on rich environmental information can prevent many collisions. Also, to prevent accidents, one vehicle can send a warning to nearby vehicles that it is experiencing a brake failure. Upon receiving such a warning signal, the nearby vehicles may stop or take other precautionary measures to prevent a collision.

TABLE II
SOME SENSORS THAT USED IN VEHICLES TO MEASURE VARIOUS PHYSICAL PHENOMENA.

CAN bus Sensors	GPS Sensors	IMU Sensors
ASR	Acceleration	Accelerometer_X
AccPedal	Current_sec	Accelerometer_Y
AirIntakeTemperature	Direction	Accelerometer_Z
AmbientTemperature	Distance	Body_acceleration_X
BoostPressure	Velocity	Body_acceleration_Y
BrkVoltage		Body_acceleration_Z
EngineSpeed_CAN		G_force
EngineTemperature		Magnetometer_X
Kickdown		Magnetometer_Y
MFS_Tip_Down		Magnetometer_Z
MFS_Tip_Up		Velocity_X
SteerAngle		Velocity_Y
Trq_FrictionLoss		Velocity_Z
Trq_Indicated		
VehicleSpeed		
WheelSpeed_FL		
WheelSpeed_FR		
WheelSpeed_RL		
WheelSpeed_RR		
Yawrate		

ASR = Acceleration Slip Regulation, ACC = Acceleration,
BRK = Break, MFS = Misfiring System, TRQ = Torque,
FL = Front Left, FR = Front Right, RL = Rear Left,
RR = Rear Right, G = Gravity

The advancement in technologies such as multi-core, sensor technologies, artificial intelligence, and robotics have allowed autonomous vehicles to perform computations and analysis that enables self-driving or autonomous driving. The autonomous driving system consists of various components such as real-time operating systems, machine and deep learning models, and sensors.

D. Control systems

Generally, feedback control systems have four components namely (1) plant, (2) sensors, (3) controller and (4) actuator. Fig. 1 shows the feedback control system. The plant is the physical system. Sensors monitor or measure physical phenomenon such as speed, temperature, orientation among others. Sensor measurements (y_k) are transmitted to the controller. The controller, based on y_k, issues control commands (u_k) to the actuator. In an auto-cruise system, an example control command is "increment speed/throttle by 2 mph". An actuator is a device that physically carries out the command. A motor that turns the wheels in an autonomous vehicle is an example actuator.

E. Threat model

The papers that we reviewed have a general threat model like the one shown in Fig 2. The attacker compromises the integrity and availability of physical components (sensor and actuator).
(1)Integrity: the attacker is able to create interference in the autonomous vehicle's physical environment such that it alters the sensor readings. Hence, the transmitted readings do not reflect the actual state of the system, that is, y_k is no longer equal to z_k as shown in the figure. Typical examples of such attacks are spoofing, data injection attacks (transduction

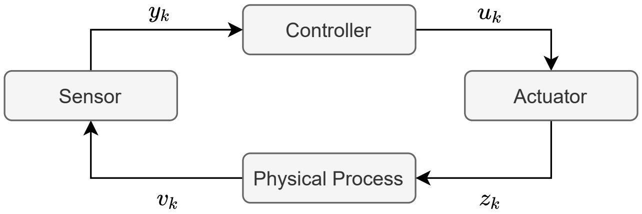

Fig. 1. The general feedback control system. y_k represent sensor reading, u_k represent controller input. z_k and v_k are the actuator output and actual physical state respectively.

attacks) [23]. The attacker may also launch replay attacks, thereby transmitting sensor values that were valid at an earlier time but no longer reflect the current state. Similarly, in a successful actuator attack, the command input issued by the controller u_k is manipulated by the attacker such that the action v_k performed by the actuator is not the same as u_k.
(2)Availability: these are the attacks that may thwart the state measurements from reaching the controller. Denial of Service (DoS)attacks are typical examples. It is worth noting that transduction attacks began as DoS attacks [24].

We note that although signal injection is often assumed in the solutions that we surveyed, it may also be done through software attacks (malware) [25]. Either way, in the end, all the attacks considered replace the sensor signal y_k and actuator signal u_k with an attacked signal y_a and u_a respectively. Since y_k and u_k represent the physics of the system, security monitoring architectures incorporate them in their attack detection solutions.

F. General Security Monitoring

Fig.3 shows the general security monitoring architecture that is used by attack detection papers that leverage the physics of the system for detection. The input to the detector is either the sensor readings y_k or the control commands u_k. Some solutions also receive both y_k and u_k as input. The detector uses various algorithms (which we discuss below) to identify anomalous sensor measurements or control commands.

Authors in [20] note that the idea of monitoring y_k and u_k has been applied in the dynamical systems fault-tolerance domain for decades. We must, however, distinguish that fault-tolerance theory does not focus on attack detection, rather, it is aimed at detecting and responding to equipment failures and random faults.

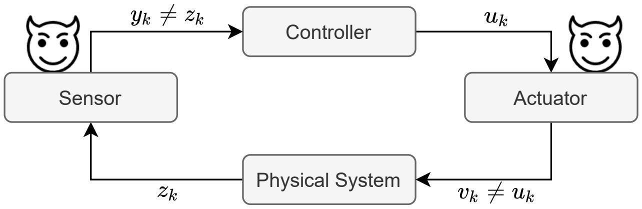

Fig. 2. Attacks that compromise the sensor reading and actuator output are considered.

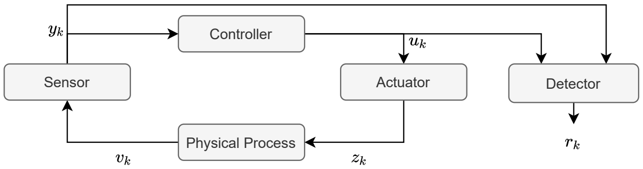

Fig. 3. General architecture for security monitoring. The input to the detector may be y_k, u_k or both

III. Classification based on physical invariant knowledge

Physical systems have properties that are guarded by immutable physical laws. When attacks are successfully launched, they violate these laws. In order to determine such a violation, it is essential to have a model that accurately approximates the nominal system behavior. The observed behavior can then be compared with the expected behavior (based on the model) to determine a violation of the physical invariant. This has been the general idea behind many attack detection publications in recent years.

Building an accurate model to approximate the nominal system behavior requires knowledge about the system and its dynamics. Modeling the complete system dynamics requires in-depth knowledge and expertise which may not always be available. Hence, recent publications have used two approaches to learn system dynamics. We group publications into two groups namely (1) black box and (2) grey box based on how they capture the system dynamics or physical invariants in their model. Further, we discuss how the model is used for attack detection.

A. Black-box approach

Publications in this category treat the system as a black box and build a model from the system data, such as sensor readings, control input and output, and system logs. The insight of this approach is that, when the system operating in a normal state, the data or readings captured by the sensor are directly proportional to the system obeying physical laws. Therefore, the data model that is built from the system data reflects the physical invariant of the system. The popular tools that have been employed in publications to learn system behavior from system data are machine and deep learning techniques. The techniques mine for relevant information and/or relationships among nominal system data.

The black box attack detection approach often has two phases: online and offline phases. The offline and online phases are summarized in Fig. 4 and Fig. 6 respectively. The offline phase or the *model training* phase starts with collecting data about the system usually consisting of sensor or actuator data. The data collected is pre-processed in order to improve the quality of the data as well as transform it into a form that is required by the chosen machine or deep learning model.

TABLE III
TAXONOMY BASED ON PHYSICAL INVARIANT.

Black Box	Grey Box	Correlation
He et. al [26] Li et. al [27] Van et. al [28] Javed et. al [29] Shin et. al [30]	Quinonez et. al [25] Choi et. al [31]	He et. al [26] Ganesan et. al [32] Li et. al [27] Parker et. al [33] Guo et. al [34]

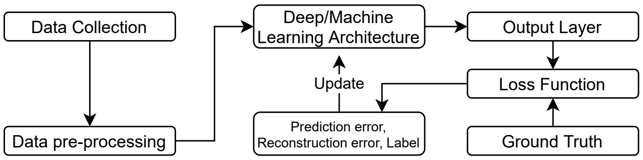
Data Collection
Deep/Machine Learning Architecture
Output Layer
Update
Loss Function
Data pre-processing
Prediction error, Reconstruction error, Label
Ground Truth

Fig. 4. The offline phase

The data pre-processing step may include one or more of the following: handling null values, handling categorical values, standardization, and one-hot encoding. The pre-processed data is fed into the machine or deep learning model such as a convolutional neural network (CNN), recurrent neural network (RNN), autoencoder, regression model, etc. The researchers in the papers we selected make different contributions at this stage. Some combine two or more DL/ML models so their trained models can learn certain patterns of interest. Others also reuse existing DL/ML architectures or make simple changes to existing ones. During model training, the output data of the DL/ML model is compared with ground truth data and a loss function calculates a score such as the reconstruction error, prediction error or assigns a label. The training process continues by optimizing and updating the model using the score obtained in the previous step. The output of the offline phase is a trained model that is capable of predicting or classifying observed system behavior.

The online phase deploys the trained model so that it can make predictions or classifications when the system is running. The anomaly detection algorithm, in most of the papers surveyed, compares the output of the trained model with the observed signals and then calculates an anomaly score using time-window approaches or statistical methods such as cumulative sum (CUSUM), chi-square, etc. The detector raises an alert whenever the anomaly score exceeds a certain pre-determined threshold.

B. Grey-box approach

Attack detection solutions in this category have some knowledge about the system and even know the physical invariant. Instead of learning the structure of the model, such papers make their contributions by learning the parameters of the invariants utilizing techniques such as system identification (SI). Such solution is provided in [25] and [31]. Generally, these solutions also have two phases: offline and online phases as shown in Fig. 5. The offline phase extracts the physical invariants that are used to build a model that captures the underlying or expected relationships between the sensors and actuators. In other words, the model captures the expected inputs and outputs of the system. The techniques used at this phase may also capture the expected relationship among sensors.

The solutions in this category have explored both linear [31] and non-linear approaches [25] to describe the physical invariants of AVs. The linear approach assumes a Linear Dynamical State-space (LDS) system which is widely used in system dynamics and control. LDS is given as:

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ y_k &= Cx_k \end{aligned} \tag{1}$$

where $x_k \in \mathbb{R}^n$ denotes the autonomous vehicle's physical-state vectors; $u_k \in \mathbb{R}^m$ is the control input vectors; $y_k \in \mathbb{R}^p$ denotes the AV's output vectors from measurements of sensors. A, B, and C are the system matrices that are unique for each physical process. Hence, each AV has unique values for A, B and C. The proposed solutions in this category use various techniques to learn these system matrices' parameters, popular among them is system identification.

System identification (SI) is a control system engineering methodology that is used to learn the parameters for the system matrices. The two inputs to the SI method are (1) a control invariant template i.e. equation of a certain degree/form with unknown coefficients/parameters and (2) a vehicle profiling measurement data set including the system inputs, outputs, and states. The vehicle profiling measurement data set is obtained by letting the subject autonomous vehicle perform a set of missions or rides. The runtime inputs (target states) and system states are measured and recorded during the execution of the missions. When the needed inputs are provided, the SI method then performs computations that instantiate the unknown system matrices (A, B, and C). The resultant equation, therefore, becomes the model for the system which is used in the online phase to predict the behaviors of the autonomous vehicle based on inputs and states. Essentially, the resultant equation serves as the control invariants of the vehicle [31].

Although the linear invariant approach works for a wide number of dynamical systems, autonomous vehicles tend to follow a non-linear invariant as noted in [25], [35]–[37]. This

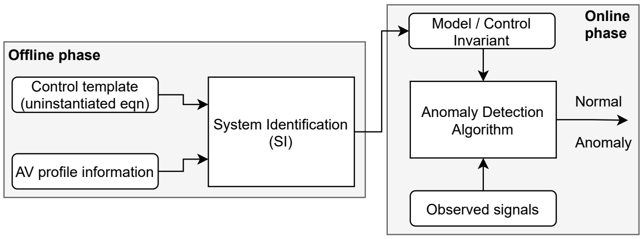
Offline phase
Control template (uninstantiated eqn)
AV profile information
System Identification (SI)
Online phase
Model / Control Invariant
Anomaly Detection Algorithm
Normal
Anomaly
Observed signals

Fig. 5. The general workflow for the grey-box approach. It consists of (1) offline phase where parameters of the control template are learned and (2) online phase where the anomaly detection algorithm uses model predictions and observed signals to determine presence or absence of anomaly

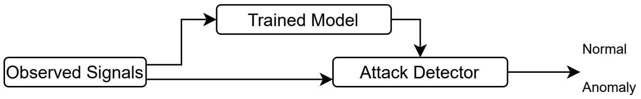

Fig. 6. The online phase

approach requires more complex equations than LDS. Authors in [25] indicate that the physical invariants of the quad-copter used in their experiment can be described with 12 non-linear differential equations “that exploit Newton and Euler equations for the 3D motion of a rigid body”. The equations oversee the position, speed, angles, and angular speed of the quad-copter. Note here that each type of autonomous vehicle will have its own set of non-linear differential equations that describe its physical invariants. The parameters of these equations are learned using the SI method discussed above for linear systems. Besides the non-linear equations' input, the parameters are learned in the same way. Particularly, the learning of non-linear parameters is formulated as an optimization problem which is given as [25]:

$$\min_{\mathcal{P}} \sum_{t=1}^{T} (H_t(\mathcal{P}, U_t) - Y_t)^2 \quad (2)$$

where U and Y are the input and output data respectively; $\mathcal{P}$ refer to the set of unknown parameters $\{p_1, p_2 ...\}$. $H_t(\mathcal{P}, U_t)$ denote the estimated output at each sampling instant t for the given parameters $\mathcal{P}$ and the input U_t. Note that $H_t(\mathcal{P}, U_t)$ is the solution the differential equations $F(\cdot)$. The goal of Eqn. 2 is to find the parameters $\mathcal{P}$ that better fit the data. In other words, Eqn. 2 seeks to find the set of parameters $\mathcal{P}$ that minimize the least square error between the estimated output $H_t(\mathcal{P}, U_t)$ and the measured output Y. Once the unknown parameters are computed, the resultant equations, therefore, become the model for the system which is used in the online phase to predict the behaviors of the autonomous vehicle based on inputs and states.

The online phase consists of an anomaly detection mechanism or algorithm that simply compares the predictions of the model that was built during the offline phase with observed signals or states. The difference between the predictions and the observed states, also called the *residual error*, are accumulated in two ways. The first approach accumulates the residual error as long as no attack has been detected as was done in [25]. The accumulation is reset whenever an attack is detected. The second approach accumulates the residual error for a set period of time (window) and then resets whenever the time window expires [31]. Either way, an alarm is raised whenever the accumulated residual error exceeds a predetermined threshold.

While the solutions in this category are robust in their attack detection role, they remain weak against stealthy attacks mainly due to perturbations and uncertainties in the model. Stealthy attacks create small deviations over time by spoofing or creating malicious data that allow the system to behave seemingly normally. Stealthy attacks are hard to defend against and remain an open problem in autonomous vehicle attack detection. Researchers in [25] are the first to propose a solution to stealthy attacks in autonomous vehicles. We discuss this further in section VI.

IV. Classification based on correlation

Immutable physical laws cause multiple sensors to exhibit correlations that can be exploited for attack detection. The multiple sensors could be measuring the same system state or not. Multiple sensors measuring the same physical state are called *homogeneous sensors* whereas those measuring different physical system states are referred to as *heterogeneous sensors*. We classify publications that exploit correlation that naturally exists among sensors for attack detection purposes into two groups: (1) homogeneous sensors and (2) heterogeneous sensors attack detection.

A. Homogeneous sensors attack detection

Multiple sensors measuring the same physical phenomenon are expected to have their measurements correlating. When this natural redundancy is not observed, it could be an indication of a possible attack, and this has been the basis for publications in this category. For instance, when four wheel speed sensors are used to monitor the speed of a vehicle's wheel, they should all report similar readings under normal operation.

Researchers in [38] propose a switching algorithm that searches for a combination of sensors that have not been compromised and generates estimates that are insensitive to sparse malicious attacks. The algorithm assumes that some of the redundant sensors have been compromised.

Although this is a good approach to attack detection, it has some limitations. First, it increases the cost of production as multiple sensors of the same type have to be deployed. This leads to increased power consumption. Also, more space will be required to accommodate the multiple sensors leading to increased weight. In applications where a lighter weight is desired, this approach may be impractical. On the other hand, fooling the attack detection may be easier since the same attack strategy and equipment can be used to attack the multiple sensors simultaneously. For example, the attacker may successfully cause all speed sensors to report 5mph thereby preserving the correlation.

B. Heterogeneous attack detection

The attack detection solutions in this category hinge on the observation that some set of sensors within an autonomous vehicle are correlated in terms of their readings [26], [27], [32], [34]. Remember that this observation is guided by physical laws. For instance, as a car moves faster, naturally, the wheels spin faster, the engine speed increases, and the pressure applied to the pedals also increases. Therefore, this natural phenomenon causes effects on sensors that monitor the wheel speed, engine speed, and pedal. Given that this natural redundancy holds all the time due to physical laws, a violation of the observation to some degree could be an indication of an attack. Hence, the proposed solutions in this category capture this physical invariant by using various methods that exploit the correlation or the natural redundancy that exists among the different sensors. Generally, the detector raises an alert whenever the natural redundancy no longer holds due to attacks.

The methods used to exploit the correlation are varied including cluster analysis [32], Pearson correlation analysis [39], autoencoders [26], regression [27]. In cluster analysis, researchers first build tools to determine the context and the cluster that the identified context belongs to. This is done for each time window. Then, a pairwise cross-correlation is performed and the results are compared with the expected correlation values for that cluster. The calculated deviation from the cluster's mean correlation value is reported as standard deviation from the mean.

In the regression method, the authors formulate the problem as a machine learning regression problem. The regression model uses statistical processes to estimate the relationships among correlated sensors. The model predicts sensor values which are then compared with the observed sensor values. A deviation is calculated and if it exceeds a threshold, an alert is raised.

EVAD [34] utilizes the frequency domain to detect attacked sensors after Fourier transform. They also organize the correlations of sensors into a ring architecture in order to reduce the computation overhead. EVAD exploits both the time domain and the frequency domain property of sensor data as the criterion to detect anomalies.

Researchers in [33] also considered a system where multiple sensors measure the same physical variable. The solution assumes that some of the redundant sensors are attacked. The work develops a resilient sensor fusion algorithm for attack detection.

Unlike the homogeneous approach discussed above, this approach does not increase the cost of production since no extra sensors are needed. Therefore, the power consumption, space, and weight remain the same for these solutions. Also, this approach tends to be more robust to attack since, to fool the detection and maintain the correlation, the attacker has to launch attacks against multiple types of sensors. Based on the fact that each type of sensor relies on different physical principles to operate, the attacker needs multiple strategies, equipment, and varying proximity to the sensor in order to launch a successful attack simultaneously.

V. Challenges

In this section, we discuss some of the challenges that researchers proposing attack detection methods face. We do not discuss the challenges in any particular order of importance.

A. Training data for data models

From our discussion above, we see that machine and deep learning techniques are valuable for building attack detection solutions. These tools, however, require enormous training data. The first challenge is that the publicly-available datasets are sparse and they contain no or very few attack datapoints. One of the reasons for this is that, especially for real-life datasets, attacks rarely occurred in the past because vehicles were then closed system [26]. Even with modern-day vehicles that are becoming open systems, successful attacks do not happen often. Hence, with such limited attack scenarios in the dataset, the machine and deep learning models are constrained in learning the attack patterns as expected to build robust models that are able to recognize attacks. In other words, CPS attack monitoring models that are trained with insufficient data tend to respond unfavorably to events or scenarios that they have not been seen before [40]. This data sparsity problem was one of the causes of the 2016 Tesla crash [41].

It is worth mentioning that some proposed methods [26] have responded to this data sparsity challenge by leveraging unsupervised machine/deep learning techniques. The models are trained to learn the nominal behavior of the plant under study from only normal data. Then using the principle of inclusion-exclusion, an alarm is raised whenever the sensor under scrutiny does not produce data that are indications of normal activity. However, the false positive and false negative rates are not promising for practical applications.

Further, the normal data available are not sufficient since they usually do not contain all the normal behavior scenarios. For instance, during the data collection stage, if the autonomous vehicle does not perform certain activities, maneuvers, or tasks, the data associated with these normal behaviors will not be captured in the dataset. Hence, unsupervised learning techniques/models which only learn from normal data are misled to classify even normal autonomous vehicle activities as abnormal.

Lastly, the sensor data obtained from autonomous vehicles can be corrupted, noisy, faulty, missing, and may contain redundant data [40], [42]. Sensors tend to be sensitive to interference in their environment which can lead to data corruption. In most situations or applications, such interference is inevitable and in others, some measures can be taken to reduce the noise. Data may also be corrupted due to the interactions occurring among system components. Lossy communication channels especially those between the sensor and data collection point contribute to data corruption. Identifying that a dataset is corrupted may require some system expertise and can be challenging. The consequences of building an attack monitor on corrupt data are quite obvious.

B. Testbeds

The availability or access to rich/practical autonomous vehicle testbed is another challenge that researchers face. In most of the papers reviewed, evaluations are not performed on systems that mimic the resources that are available on real autonomous vehicles thereby reducing the practicality of the proposed solutions. Rather, experiments are carried out using simulated data that were run on computing resources that differ a lot from resources available on autonomous vehicles. For instance, the operating systems that the experiments are simulated are not a real-time OS. Also, the CPU/GPU capabilities and memory capacity available on experimental systems are higher compared with what is available on autonomous vehicles.

In part, high-end autonomous vehicle testbeds are expensive to acquire, limiting research groups, especially those in developing countries, from testing out their novel ideas and designs. Although cheaper testbeds are available, usually, they do not possess all the sensors that may be required for the particular research. It is also possible to custom-build autonomous vehicle testbeds, however, assembling all the components requires expertise that may not be available in the research group or the university at large. Even in instances where the expertise is available, the process of building the testbed can be time-consuming. From our own experience, it has taken more than a year to build an autonomous vehicle testbed. Further, the sharing of testbeds amongst research groups especially those whose physical geography is farther apart may be hampered by travel restrictions by governments, a pandemic, or other factors.

C. *Benchmark for comparing related work*

It is difficult to fairly and accurately compare the effectiveness and efficiency of the various proposed attack detection methods due to the absence of "standardized" benchmark data. Given that each research effort evaluates their work on the data that the researchers generate or simulate, it is difficult to tell if the proposed solutions are applicable to only their data or work with other new data. A common benchmark can facilitate result comparison as well inspire research proposals that perform better than existing solutions.

Also, many researchers who have access to good testbeds or even simulate good autonomous vehicle data often do not make their data and source code publicly available. Such availability to the public not only aids the repeatability of the research method but also allows others to use the data and compare the results.

D. *Standard evaluation metrics*

Another challenge regarding research result comparison is the lack of standard evaluation metrics. Usually, different metrics are used for evaluating the proposed attack detection method. This makes it difficult to know which proposal is better and even how an existing solution should be improved based on a metric. A standard evaluation metric can guide the current as well as the future development of evaluative metrics for attack detection methods in autonomous vehicles. A common metric can also help the peer review process so that reviewers can make a better judgment of papers under review and/or make suggestions that improve research efforts.

VI. Open Problems and Research Opportunities

Although the research community has responded to the need for viable attack detection methods to protect autonomous vehicles due to the safety-critical roles that they play, a number of problems still remain, thereby offering research opportunities. In this section, we discuss some of these open problems.

A. *Real-time detection and usability*

Many of the research efforts have focused on the accuracy of detecting the attacks, however, they have not adequately addressed the real-time constraints of the attack detection and the usability of the attack detector [43].

The timing constraint we are referring to is what we call *detection deadline*. It is the time before which the attack must be detected. Ultimately, an untimely attack detection is equally damaging as no detection at all. It is important that attack detectors raise an alarm before any damaging effects occur. For instance, an alarm should be raised before the autonomous vehicle hits an object and not after. The usability of a detector refers to the false alarm rate, and a lower rate is desirable since that translates to better usability.

Meeting both the timing and usability constraints is non-trivial. This calls for techniques or solutions that will be able to calculate the accurate detection deadline before which attacks must be detected and at the same time achieve lower false alarms. In other words, adaptive attack detection solutions that balance timing and usability constraints are needed to protect autonomous vehicles.

B. *Recovery after attack detection*

As important as it is to detect attacks when they occur, it is also essential to provide mitigation measures that respond to the attack after it has been detected. We refer to these measures as *attack recovery*. While many research efforts have focused on attack detection, comparatively, very few have addressed attack recovery such as [44] [45] [46] [47] and [48]. Authors in [13] who reviewed 32 security survey papers indicated that only 8 addressed some form of response to detected attacks.

As an effective way of improving attack-resilience, attack recovery solutions should be able to develop mechanisms that can estimate system states that are accurate enough to control the autonomous vehicle irrespective of the compromised components. The recovery measures should also meet timing constraints and must be usable. Similar to our discussion above, any response to an attack should complete before the damage is caused. This means recovery solutions should not only be able to estimate system states but also be able to calculate *recovery deadline* before which the system enters an unsafe state. Zhang et al. [46] have started this research direction but we believe more adaptive real-time solutions can be pursued in order adequately improve attack-resilience.

C. *Distinguishing between faults and attacks*

Abnormal behavior in autonomous vehicles may not always be a result of an attack. For instance, an autonomous sensor may produce anomalous readings for a number of such as poor weather and other environmental conditions, magnetic field interferences or even sensor aging [28]. Obviously, the response to an attack should be different from the response to a fault. Failure to rightly classify an attack or vice-versa may result in serious operational failure [49] [50]. Sometimes replacing a faulty component is all that is required to resume normal behavior.

Hence, it is important for the proposed solution to be able to distinguish between faults and attacks so that the right response is applied. Fault detection and attack detection to a greater extent, remain two separate domains. This gap needs to be bridged in order to produce robust and usable attack detection solutions. An attack detection solution that incorporates fault detection can improve its usability. Some attempts in this direction have been made in other cyber-physical domains [50] but remain inadequately addressed in autonomous vehicles which are more dynamic than the static system addressed in [50].

D. Context-aware attack detectors

Attack detectors can include contextual information in concluding that an actual attack is occurring which leads to building usable and robust detectors. Ref [51] defines context as the "additional information that is not directly used as measurement data but is related to the measurements in an unknown but structured way". The environmental conditions in which the autonomous vehicle is placed have varied effects on the state of the system. For example, potholes can cause speed sensor readings to briefly break sensor correlation as well as exceed detector residual thresholds thereby raising false attack alerts. If the detector can harness the contextual information of the pothole presence, it can make a more accurate decision of not raising an alert in this scenario. None of the papers that met our survey criteria considers the context in their solutions. Given the complexity of cyber-physical systems such as autonomous vehicles, ignoring its context information completely in attack detection solutions is an indication that the solution could fail in real-world scenarios. More research efforts that incorporate contextual information in sensor attack detectors are therefore needed. Context-aware detectors have been discussed in other CPS domains [51]–[56]. [57] investigates the extent to which context information may be used to improve the security and survivability of CPS in general. Comparatively, specific contextual-aware physical invariant-based attack detectors for autonomous CPS are limited. Wasicek et al. [58] propose a context-aware intrusion detection in automotive control systems, however, their solution targets controller protection rather than sensor attack detection. RAID [59] is one work that attempts to incorporate road context in the proposed intrusion detection system. RAID extracts road contexts from sensory information using a lightweight machine learning model. The extracted road context and the corresponding in-vehicle network frames are validated to ensure there is no significant deviation.

VII. Conclusion

Autonomous vehicles play important and safety-critical roles in modern society. Defending these systems is not only desirable but indispensable. In this paper, we have systematically surveyed publications that specifically monitor the physics of the system for attack detection purposes. We present two taxonomies that are based on (1) how physical invariants are captured to build a model and (2) the correlation among sensors. We discussed the general techniques that are used in each category. Further, we highlighted the challenges that researchers face when undertaking attack detection for autonomous vehicles. Such challenges include data set, benchmark, testbed, and evaluation metrics. Lastly, we discussed some open problems that offer research opportunities including real-time adaptive attack detection, real-time recovery systems, contextual-aware attack detectors, and detectors that accurately distinguish between faults and attacks.

Acknowledgment

We would like to thank the anonymous reviewers for their constructive comments. This work was supported in part by NSF CCF-1720579 and NSF CCF-2028740.

References

[1] P. Muyan-Ozcelik and V. Glavtchev, "Gpu computing in tomorrow's automobiles." [Online]. Available: https://www.nvidia.com/content/nvision2008/tech_presentations/Automotive_Track/NVISION08-GPU_Computing_in_Tomorrows_Automobiles.pdf

[2] Z. Dong, W. Shi, G. Tong, and K. Yang, "Collaborative autonomous driving: Vision and challenges," in *2020 International Conference on Connected and Autonomous Driving (MetroCAD)*, 2020, pp. 17–26.

[3] C. Yan, W. Xu, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," *DEF CON*, vol. 24, no. 8, p. 109, 2016.

[4] B. S. Lim, S. L. Keoh, and V. L. Thing, "Autonomous vehicle ultrasonic sensor vulnerability and impact assessment," in *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*. IEEE, 2018, pp. 231–236.

[5] S. Lee, W. Choi, and D. H. Lee, "Securing ultrasonic sensors against signal injection attacks based on a mathematical model," *IEEE Access*, vol. 7, pp. 107 716–107 729, 2019.

[6] W. Xu, C. Yan, W. Jia, X. Ji, and J. Liu, "Analyzing and enhancing the security of ultrasonic sensors for autonomous vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5015–5029, 2018.

[7] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote attacks on automated vehicles sensors: Experiments on camera and lidar," *Black Hat Europe*, vol. 11, p. 2015, 2015.

[8] J. Petit and S. E. Shladover, "Potential cyberattacks on automated vehicles," *IEEE Transactions on Intelligent transportation systems*, vol. 16, no. 2, pp. 546–556, 2014.

[9] S. Narain, A. Ranganathan, and G. Noubir, "Security of gps/ins based on-road location tracking systems," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 587–601.

[10] Y. Cao, C. Xiao, B. Cyr, Y. Zhou, W. Park, S. Rampazzi, Q. A. Chen, K. Fu, and Z. M. Mao, "Adversarial sensor attack on lidar-based perception in autonomous driving," in *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, 2019, pp. 2267–2281.

[11] B. G. Stottelaar, "Practical cyber-attacks on autonomous vehicles," Master's thesis, University of Twente, 2015.

[12] H. Shin, D. Kim, Y. Kwon, and Y. Kim, "Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications," in *International Conference on Cryptographic Hardware and Embedded Systems*. Springer, 2017, pp. 445–467.

[13] J. Giraldo, E. Sarkar, A. A. Cardenas, M. Maniatakos, and M. Kantarcioglu, "Security and privacy in cyber-physical systems: A survey of surveys," *IEEE Design & Test*, vol. 34, no. 4, pp. 7–17, 2017.

[14] C. Miller and C. Valasek, "Adventures in automotive networks and control units," *Def Con*, vol. 21, no. 260-264, pp. 15–31, 2013.

[15] ——, "Remote exploitation of an unaltered passenger vehicle," *Black Hat USA*, vol. 2015, no. S 91, 2015.

[16] K. Koscher, S. Savage, F. Roesner, S. Patel, T. Kohno, A. Czeskis, D. McCoy, B. Kantor, D. Anderson, H. Shacham *et al.*, "Experimental security analysis of a modern automobile," in *2010 IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2010, pp. 447–462.

[17] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno *et al.*, "Comprehensive experimental analyses of automotive attack surfaces." in *USENIX Security Symposium*, vol. 4. San Francisco, 2011, pp. 447–462.

[18] I. Foster, A. Prudhomme, K. Koscher, and S. Savage, "Fast and vulnerable: A story of telematic failures," in *9th {USENIX} Workshop on Offensive Technologies ({WOOT} 15)*, 2015.

[19] Y. Shoukry, P. Martin, P. Tabuada, and M. Srivastava, "Non-invasive spoofing attacks for anti-lock braking systems," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2013, pp. 55–72.

[20] J. Giraldo, D. Urbina, A. Cardenas, J. Valente, M. Faisal, J. Ruths, N. O. Tippenhauer, H. Sandberg, and R. Candell, "A survey of physics-based attack detection in cyber-physical systems," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–36, 2018.

[21] C. Kaiser, A. Stocker, and A. Festl, *Automotive CAN bus data: An Example Dataset from the AEGIS Big Data Project*, Jul. 2019. [Online]. Available: https://doi.org/10.5281/zenodo.3267184

[22] J. Raiyn, "Data and cyber security in autonomous vehicle networks," *Transport and Telecommunication Journal*, vol. 19, no. 4, pp. 325–334, 2018.

[23] K. Fu and W. Xu, "Risks of trusting the physics of sensors," *Communications of the ACM*, vol. 61, no. 2, pp. 20–23, 2018.

[24] Y. Son, H. Shin, D. Kim, Y. Park, J. Noh, K. Choi, J. Choi, and Y. Kim, "Rocking drones with intentional sound noise on gyroscopic sensors," in *24th {USENIX} Security Symposium ({USENIX} Security 15)*, 2015, pp. 881–896.

[25] R. Quinonez, J. Giraldo, L. Salazar, and E. Bauman, "Savior: Securing autonomous vehicles with robust physical invariants," *Usenix*, 2020.

[26] T. He, L. Zhang, F. Kong, and A. Salekin, "Exploring inherent sensor redundancy for automotive anomaly detection," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*. IEEE, 2020, pp. 1–6.

[27] H. Li, L. Zhao, M. Juliato, S. Ahmed, M. R. Sastry, and L. L. Yang, "Poster: Intrusion detection system for in-vehicle networks using sensor correlation and integration," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 2531–2533.

[28] F. van Wyk, Y. Wang, A. Khojandi, and N. Masoud, "Real-time sensor anomaly detection and identification in automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1264–1276, 2019.

[29] A. R. Javed, M. Usman, S. U. Rehman, M. U. Khan, and M. S. Haghighi, "Anomaly detection in automated vehicles using multistage attention-based convolutional neural network," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[30] J. Shin, Y. Baek, J. Lee, and S. Lee, "Cyber-physical attack detection and recovery based on rnn in automotive brake systems," *Applied Sciences*, vol. 9, no. 1, p. 82, 2019.

[31] H. Choi, W.-C. Lee, Y. Aafer, F. Fei, Z. Tu, X. Zhang, D. Xu, and X. Deng, "Detecting attacks against robotic vehicles: A control invariant approach," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 801–816.

[32] A. Ganesan, J. Rao, and K. Shin, "Exploiting consistency among heterogeneous sensors for vehicle anomaly detection," SAE Technical Paper, Tech. Rep., 2017.

[33] J. Park, R. Ivanov, J. Weimer, M. Pajic, and I. Lee, "Sensor attack detection in the presence of transient faults," in *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*, 2015, pp. 1–10.

[34] F. Guo, Z. Wang, S. Du, H. Li, H. Zhu, Q. Pei, Z. Cao, and J. Zhao, "Detecting vehicle anomaly in the edge via sensor consistency and frequency characteristic," *IEEE Transactions on Vehicular Technology*, 2019.

[35] T. D. Gillespie, *Fundamentals of vehicle dynamics*. Society of automotive engineers Warrendale, PA, 1992, vol. 400.

[36] A. Chovancová, T. Fico, L. Chovanec, and P. Hubinsk, "Mathematical modelling and parameter identification of quadrotor (a survey)," *Procedia Engineering*, vol. 96, pp. 172–181, 2014.

[37] T. Luukkonen, "Modelling and control of quadcopter," *Independent research project in applied mathematics, Espoo*, vol. 22, 2011.

[38] J. Kim, C. Lee, H. Shim, Y. Eun, and J. H. Seo, "Detection of sensor attack and resilient state estimation for uniformly observable nonlinear systems having redundant sensors," *IEEE Transactions on Automatic Control*, vol. 64, no. 3, pp. 1162–1169, 2018.

[39] P. Sharma, J. Petit, and H. Liu, "Pearson correlation analysis to detect misbehavior in vanet," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*. IEEE, 2018, pp. 1–5.

[40] J. Weimer, R. Ivanov, S. Chen, A. Roederer, O. Sokolsky, and I. Lee, "Parameter-invariant monitor design for cyber–physical systems," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 71–92, 2017.

[41] E. Ackerman, "Fatal tesla self-driving car crash reminds us that robots aren't perfect," *IEEE-Spectrum*, vol. 1, 2016.

[42] K. Xie, X. Ning, X. Wang, D. Xie, J. Cao, G. Xie, and J. Wen, "Recover corrupted data in sensor networks: A matrix completion solution," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1434–1448, 2016.

[43] F. Akowuah and F. Kong, "Real-time adaptive sensor attack detection in autonomous cyber-physical systems," in *27th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*. IEEE, 2021.

[44] F. Kong, M. Xu, J. Weimer, O. Sokolsky, and I. Lee, "Cyber-physical system checkpointing and recovery," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2018, pp. 22–31.

[45] F. Kong, O. Sokolsky, J. Weimer, and I. Lee, "State consistencies for cyber-physical system recovery," in *Workshop on Cyber-Physical Systems Security and Resilience (CPS-SR)*, 2019.

[46] L. Zhang, X. Chen, F. Kong, and A. A. Cardenas, "Real-time recovery for cyber-physical systems using linear approximations," in *41st IEEE Real-Time Systems Symposium (RTSS)*. IEEE, 2020.

[47] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic control*, vol. 59, no. 6, pp. 1454–1467, 2014.

[48] M. Pajic, J. Weimer, N. Bezzo, P. Tabuada, O. Sokolsky, I. Lee, and G. J. Pappas, "Robustness of attack-resilient state estimators," in *2014 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2014, pp. 163–174.

[49] Y. Li, H. Fang, and J. Chen, "Anomaly detection and identification for multiagent systems subjected to physical faults and cyber attacks," *IEEE Transactions on Industrial Electronics*, 2019.

[50] A. Anwar, A. N. Mahmood, and Z. Shah, "A data-driven approach to distinguish cyber-attacks from physical faults in a smart grid," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 2015, pp. 1811–1814.

[51] R. Ivanov, J. Weimer, and I. Lee, "Context-aware detection in medical cyber-physical systems," in *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 2018, pp. 232–241.

[52] ——, "Towards context-aware cyber-physical systems," in *2018 IEEE Workshop on Monitoring and Testing of Cyber-Physical Systems (MT-CPS)*. IEEE, 2018, pp. 10–11.

[53] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, "Soi-kf: Distributed kalman filtering with low-cost communications using the sign of innovations," *IEEE Transactions on signal processing*, vol. 54, no. 12, pp. 4782–4795, 2006.

[54] R. P. Mahler, *Advances in statistical multisource-multitarget information fusion*. Artech House, 2014.

[55] B. Sinopoli, L. Schenato, M. Franceschetti, K. Poolla, M. I. Jordan, and S. S. Sastry, "Kalman filtering with intermittent observations," *IEEE transactions on Automatic Control*, vol. 49, no. 9, pp. 1453–1464, 2004.

[56] A. K. Sikder, H. Aksu, and A. S. Uluagac, "6thsense: A context-aware sensor-based attack detector for smart devices," in *26th {USENIX} Security Symposium ({USENIX} Security 17)*, 2017, pp. 397–414.

[57] K. Wan and V. Alagar, "Context-aware security solutions for cyber-physical systems," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 212–226, 2014.

[58] A. Wasicek, M. D. Pesé, A. Weimerskirch, Y. Burakova, and K. Singh, "Context-aware intrusion detection in automotive control systems," in *Proc. 5th ESCAR USA Conf*, 2017, pp. 21–22.

[59] J. Jiang, C. Wang, S. Chattopadhyay, and W. Zhang, "Road context-aware intrusion detection system for autonomous cars," in *International Conference on Information and Communications Security*. Springer, 2019, pp. 124–142.