Cyber-Attack Detection in Socio-Technical Transportation Systems Exploiting Redundancies Between Physical and Social Data

Tanushree Roy, Member, IEEE, Sara Sattarzadeh, and Satadru Dey, Senior Member, IEEE

Abstract—Cyber–physical–social connectivity is a key element in intelligent transportation systems (ITSs) due to the ever-increasing interaction between human users and technological systems. Such connectivity translates the ITSs into dynamical systems of socio-technical nature. Exploiting this socio-technical feature to our advantage, we propose a cyber-attack detection scheme for ITSs that focuses on cyber-attacks on freeway traffic infrastructure. The proposed scheme combines two parallel macroscopic traffic model-based partial differential equation (PDE) filters whose output residuals are compared to make decision on attack occurrences. One of the filters utilizes physical (vehicle/infrastructure) sensor data as feedback whereas the other utilizes social data from human users’ mobile devices as feedback. The Social Data-based Filter is aided by a fake data isolator and a social signal processor that translates the social information into usable feedback signals. Mathematical convergence properties are analyzed for the filters using Lyapunov’s stability theory. Finally, we validate our proposed scheme by presenting simulation results.

Index Terms—Cyber-attack detection, socio-technical systems.

I. INTRODUCTION

A. Research Background

Security against cyber-attacks is one of the crucial criteria for today’s emerging intelligent transportation systems (ITSs). Such ITSs present both new threats as well as opportunities which were not existent in the conventional transportation systems. For example, an attacker can hack a vehicle CAN bus to introduce undesirable traffic scenarios. Similarly, compromised electronic traffic messaging boards or tampered traffic lights can cause disruptions like congestion or accidents. Additionally, more advanced attack strategies can facilitate to create VIP-lanes for attackers by rerouting traffic from specific lanes using false traffic congestion information [1]. Thus, increasing complexity of ITS magnifies attack susceptibilities which can eventually endanger the safe operation of urban transportation.

B. Significance

While large-scale cyber–physical connectivity in ITSs exposes them to unprecedented vulnerability to cyber-threats, social connectivity through human–ITS interactions generates additional information for better management of such ITSs [2], [3]. Specifically, such social connectivity generates an enormous amount of data that can be utilized meaningfully to enhance our system knowledge and transportation control algorithms. In this context, the significance of this work is in designing safe-guarding mechanisms for ITSs using a system-theoretic framework and a data-fusion strategy.

C. Motivation

Data fusion has been extensively used in the field of autonomous driving, energy, military surveillance and reconnaissance, and medical fields [4], [5], [6], [7]. Based on the classification by Durrant-Whyte [8], data fusion can be complementary, cooperative, or redundant. A data fusion is called redundant if information from multiple data sources is aggregated for a single target system. This redundant data fusion technique is essential to increase confidence in the data or to provide contingencies in case of corruption of one or more data sources. In relation to transportation, data is generated not only from vehicular and infrastructure (physical) sensors but also through (social) sensors such as mobile devices generating position data through GPS or tweets. Data from these different modalities can be fused to gather traffic information. Specifically, we intend to exploit the redundancies present in social and physical data. Such redundancy arises as physical infrastructure sensors measure vehicle positions whereas mobile device GPS or tweets also possess similar information. Having similar information from different data sources could be a key to detect anomalies in the traffic networks. In this work, we utilize such data fusion in conjunction with partial differential equation (PDE) system theoretic techniques for cyber-attack detection.

D. Main Contribution

In this article, we intend to detect cyber-attack using a second-order continuum or macroscopic traffic model. This not only enables us to characterize global and complex
behavior of traffic but also helps simulate infrastructure-level cyber-attacks without the computation burden of a vehicle-level microscopic model. Specifically, our proposed scheme contains two parallel macroscopic traffic model-based PDE filters whose output residuals are compared to make decision on attack occurrences. One of the filters utilizes physical sensor data as feedback whereas the other utilizes social data from human users’ mobile devices as feedback. The Social Data-based Filter is aided by a fake data isolator and a social signal processor (SSP) that translates the social information into usable feedback signals. We analyze the convergence properties and design the filter parameters using Lyapunov’s stability theory.

The main contribution of the work lies in 1) proposing a traffic social data assimilation strategy by coupling second-order microscopic traffic data generation dynamics to second-order macroscopic traffic dynamics; 2) constructing a long short-term memory (LSTM) recurrent neural network (RNN) structure for detection of fake social traffic data; 3) designing stable Social and Physical Data-based filters using Lyapunov stability theory; and 4) ultimately proposing cyber-attack detection scheme analyzing residuals generated by Social and Physical Data-based filters.

E. Article Organization and Notation

The remainder of this article is organized as follows. Section II discusses related work in this area, Section III discusses the traffic modeling and problem statement, Section IV discusses the design of the Social Data-based Filter while Section V introduces the Physical Data-based Filter and Section VI discusses the Comparator. Finally, Section VII presents the implementation details, Section VIII discusses the simulation results followed by a conclusion in Section IX.

Notation: The following notations have been used in this work: $w_i = \partial w/\partial t$, $w_x = \partial w/\partial x$, $\mathbf{w}(p) = (\partial w/\partial p)$, $\mathbf{\dot{w}}(p) = (\partial^2 w/\partial p^2)$; $\|w(.)\|$ denotes the spatial $L_2$ norm given as $\|w(.)\| := \sqrt{\int_0^L w^2(x)dx}$.

II. Related Work

Physical data in traffic infrastructure arises from vehicular and infrastructure sensors [3], [9]. This data, in general, requires less processing and has higher reliability in terms of availability and delay. In contrast, social data [10] is generated by the human users-in-the-loop using mobile devices, such as messages on networking platforms and GPS coordinates of mobile devices. These social data are less-structured, less reliable in terms of availability and delay, and require more processing efforts. This distinction makes it especially challenging to combine these two types of data in a system-theoretic framework.

With increasing interest in detection of cyber-attacks on ITSs, many researchers have utilized physical data to detect denial-of-service (DoS) attack [11], [12], [13], replay attack [14], [15], false data injection [16], [17], complex congestion pattern [1], [18], attack on car platoons [19], [20], [21], [22], and identity of attacker [23]. On the other hand, most social data-based research focuses on detecting traffic incidents or location using social data (such as status update messages (SUMs) or tweets) [24], [25]. Very few works have utilized social data for real-time traffic control or estimation. For example, probe vehicle measurements have been used to estimate traffic density in [26] and [27]. However, the utilization of social data to detect cyber-attacks in transportation systems has remained underexplored. Social data can be highly useful in modeling and real-time management of traffic systems. Especially, with the present rate of engagement in social networking, the amount of social data can be enormous compared to fewer physical data that are restricted by the cost of installation. Most importantly, in case of cyber-attacks in physical part of the infrastructure, social data would provide us real-time redundancies in monitoring traffic states.

Moreover, the presence of malicious vehicles has been detected using Gaussian process regression (GPR) in [28]. However, they fail to analyze the detection sensitivity depending on fixed physical sensor location and the subtlety of speed changes for the malicious vehicle. A distributed cyberattack detection strategy has been employed in [29] through set membership criteria, this work fails to capture infrastructure-level cyberattack scenarios. In [30], spoofing attack detection strategies are proposed using probe-based traffic flow information. This work uses a discretized approximation of a first-order traffic model to obtain its attack detection scheme using a mixed-integer linear programming. While [30] uses a macroscopic traffic model, it is only first order which fails to capture realistic human driving behavior [31]. Moreover, the analysis is based on discretization in the time domain which may further add to the inaccuracy of the proposed strategy.

In our previous work [18], we designed an attack detection filter combining second-order macroscopic Aw–Rascl–Zhang (ARZ) traffic model and outlet traffic flux measurement from physical infrastructure. The major distinction between our current work and [18] lies in the following: 1) the current work utilizes both physical and social data while [18] uses only physical data; 2) the current work has two filters working in parallel while [18] uses only one filter; and 3) the current work utilizes ARZ model with traffic density and velocity states while [18] adopted ARZ model with flux and velocity states. Such difference in the ARZ model translates to different coupling in the PDE models and slight difference in backstepping transformation used for filter design.

In our recent work [10], we proposed a diagnostic framework that utilized a microscopic platoon model and system theoretic tools along with physical and social data to detect cyber-attacks affecting vehicular communication network. The first limitation of [10] is the use of a microscopic model that requires extensive computation and may be unrealistic in the context of numerous links in the traffic network or for capturing complex traffic dynamics pattern. Second, the cyber-attack considered was limited to that on vehicular communication networks and attack such as complex congestion pattern cannot be captured. Third, only connected and autonomous vehicle (CAV) dynamics is considered while in reality cyber-attack or misinformation on traffic infrastructure can disrupt traffic flow irrespective of autonomous or human-driven or heterogeneous.
vehicles. Finally, in our previous framework, no contingency for cyber-attack on social data has been considered.

This article simultaneously addresses the aforementioned limitations in our previous work [10] and bridges the research gap in using physical and social data to detect infrastructure or global cyber-attack, such as coordinated ramp metering, data spoofing on fixed sensors, adversarial, or hijacked vehicle interventions.

III. TRAFFIC MODELING AND PROBLEM STATEMENT

In this section, we describe the problem set-up in terms of the traffic model, cyber-attack threats in socio-technical traffic system, physical and social data, and cyber-attack detection scheme.

A. Macroscopic Traffic Model

In this study, we consider traffic flow on a single lane of a free-way as a one-dimensional spatial evolution along \( x \in [0, L] \), where \( L \) represents the length of the free-way. In general, traffic flow is characterized by two of the three main variables: 1) density; 2) velocity; and 3) flux. In congruence with the data measurement available to our problem, we choose density and velocity as our variables. These variables are defined as follows: 1) traffic density is the number of vehicles per unit length of the free-way and 2) traffic velocity is given by the mean distance covered per unit time.

In this framework, traffic flow is represented by the ARZ spatiotemporal macroscopic traffic model where the dynamics of the density \( \rho(x, t) \) and velocity \( v(x, t) \) of the traffic flow are given as follows [32]:

\[
\rho_t + (\rho v)_x = 0, \quad v_t + (v + V_{opt}(\rho))v_x = \frac{V_{opt}(\rho) - v}{T_r}
\]

(1)

where \( t \in [0, \infty) \) is time and \( x \in [0, L] \) is the spatial variable, \( V_{opt}(\rho) \) is the optimal velocity function and \( T_r \) is the relaxation parameter. The boundary conditions are given by

\[
\rho(0, t) = \rho_m + u, \quad \rho(L, t) = \rho^*
\]

(2)

where \( \rho_m \) is the maximum density and \( u \) is a control input at the inlet ramp metering on the free-way. Here, \( \rho^* \) is the constant nominal density maintained at the outlet of the freeway using loop detector measurement.

To facilitate our analysis, we first linearize the ARZ model (1) around nominal operating point \( (\rho^*, v^*) \). Subsequently, to decouple this linearized system, we perform the following transformation \( w := v - \rho V_{opt}(\rho^*) \), to obtain the transformed model in \((w, \dot{v})\) domain [33]

\[
w_t + k_1 w_x = -k_2 w, \quad v_t + k_3 v_x = -k_2 v
\]

(3)

where \( k_1 = v^*, k_2 = (1/T_r) \) and \( k_3 = [v^* + \rho^* V_{opt}(\rho^*)] \). Under attack, this model is modified as

\[
w_t + k_1 w_x = -k_2 w, \quad v_t + k_3 v_x = -k_2 v + \delta_1
\]

(4)

\[
w(0, t) = v(0, t) - (\rho_m + u + d_2 - \rho^*) V_{opt}(\rho^*)
\]

(5)

\[
w(L, t) = v(L, t)
\]

(6)

where \( \delta_1(x, t) \) is the in-domain distributed attack while \( \delta(t) \) represents the inlet boundary attack. Without loss of generality, we can assume here \( u = \rho^* - \rho_m \) and define \( \delta_2 := -d_2/V_{opt}(\rho^*) \) which modifies (5) as

\[
w(0, t) = v(0, t) + \delta_2.
\]

We also note here that attacks at the outlet boundary are not considered as it is trivial to analyze with our assumed outlet boundary measurement. We note here that the crucial variables used in this paper have been described in Table I.

B. Cyber-Attack: Threat Model Formulation

The traffic model, described in (4), (6), and (7), captures two broad categories of cyber-attacks based on their acting location: in-domain and boundary attacks. The in-domain cyber-attack \( \delta_1(x, t) \) is a distributed attack on the traffic states and models attacks on communication layers of CAVs, corruption of vehicle navigation systems, or hijacked rogue vehicles. These attacks can be generated by appropriating the controller of CAVs to create arbitrary velocity patterns in the traffic [34, 35].

The in-domain attack can also model cyber-attacks on the central management system since the latter can affect large-scale traffic disruptions by compromising a wide range of traffic management devices such as traffic electronic message boards or traffic reporting services on the freeway [36]. The nature of these attacks can be DoSs, false-data injection, sensor deception, replay attack, data spoofing, or mixed threat modalities [10]. On the other hand, the inlet cyber-attack \( \delta_2(t) \) models attacks on traffic control infrastructure, such as ramp-metering or loop detectors. These attacks can be orchestrated by hacking into the software of these devices/sensors/control boxes or the central management system [1], [35].

Additionally, in a socio-technical traffic system, cyber-attacks may arise from corrupted social data. Such attacks can be initiated through the generation of fake tweets or messages and spams. Utilization of social signals for traffic management system is thus endangered by such cyber-attacks. Furthermore, corrupted social data may include malware or...
phishing links that can jeopardize the operation of central management systems. These can also lead to false panic scenarios and severely impact traffic management systems. The different cyber-attack scenarios are shown in Fig. 1.

There are few instances of cyber-attacks on a transportation system that cannot be identified by the proposed approach. Passive cyber-attacks such as eavesdropping over networks to track vehicles or data theft from traffic camera fall into this category. Additionally, cyber-attacks that do not directly impact traffic states remain undetected by this method. For example, DoS attack or ransomware attacks on e-ticketing system.

C. Physical and Social Data

The measurement of the outlet traffic flow $v(L, t)$ is used as a physical measurement data from the infrastructure sensors. Concurrently, we also consider $N$ social agents or human-in-the-loop, who are traveling in cars on this freeway and are transmitting social data, such as geo-tags, GPS coordinates from portable devices, messages, SUMs, or tweets. When any car transmits positional information through social data, it not only provides us with the position knowledge of this vehicle but also provides us unique access to information of traffic at these in-domain points.

Note that the social data contains vehicle-level (i.e., microscopic) information whereas the model variables are global traffic quantities, such as density, flow velocity, and flux. In order to translate the microscopic data to macroscopic variables, we leverage the connection between the macroscopic ARZ model and microscopic car-following model [32]. This motivates us to identify social-data generating segments of the macroscopic traffic as microscopic. Subsequently, we define the dynamics of each social-data transmitting vehicle using a modified car-following model [32]:

$$\ddot{z}_i(t) = \frac{C_y B(t)}{B^2(t) + 1} + \frac{1}{T_v} V_{opt} \left( \frac{\Delta X}{B(t)} - \dot{z}_i(t) \right)$$

where $B(t) = \dot{b}_i(t) - \ddot{z}_i(t)$ and $\dot{z}_i(t) \forall i \in \{1, \ldots, N\}$ represent the position of the $i$th car transmitting social data, $\Delta X$ is the length of the car, and $b_i$ is the position of the micro–macro interface. The interface $b_i$ here acts analogous to the preceding car in the car-following model. The constant parameter $C_y = v^*(\Delta X)^y$ where $y$ is a non-negative parameter.

Since we obtain positional information from the social data of the $i$th social agents, we assume that the position data of their vehicle $z_i$ is known. The optimal velocity function $V_{opt}$ is monotonic and bounded. This implies that it is invertible.

Thus, the position of the interface $b_i$ can be computed from (8) by solving a nonlinear differential equation of the form $\dot{b}_i = h(b_i, z_i, \ddot{z}_i)$. This enables us to define local density as

$$\rho_i = \Delta X/(b_i - z_i) \approx \rho(z_i, t).$$

These $\rho(z_i, t)$ serve then serve as measurements for the Social Data-Based Filter.

D. Adversarial Attack Detection Approach

Based on the above setup, the cyber-attack detection scheme is shown in Fig. 2. The scheme contains Social Data-based Filter, Physical Data-based Filter, and Comparator. The Social Data-based Filter utilizes $\rho(z_i, t)$ as feedback whereas the Physical Data-based Filter utilizes $v(L, t)$ for employing output injection. Both of these filters output residual signals. These residual signals are processed by the Comparator to produce a decision of attack-detection. In the following sections, we will discuss the Filters and the Comparator.

IV. SOCIAL DATA-BASED FILTER

Online social networking (OSN) has grown extensively over the recent years. Among various providers of OSN, Twitter alone provides networking platform to roughly one-fourth of the U.S. adult population of which 42% are on the platform daily.1 Furthermore, about 40% of these SUMs or tweets contain information, such as geo-tags and geographical location information [37]. This suggests that the data from OSN can be leveraged successfully as a social data signal in the context of traffic systems.

To achieve the aforementioned objective, we propose a Social Data-based Filter (see Fig. 3). This filter consists of three separate stages: Stage I is the fake social data isolator (FSDI), Stage II is SSP, and finally Stage III consists of the social signal residual generator (SSRG). These three stages work in tandem to ensure high-quality prediction from the Social Data-based Filter.

The purpose of the three stages is gate-keeping, preprocessing, and social signal residual generation. Here, the Stage I FSDI detects cyber-attack on the social signals by classifying fake or illegitimate social data. This stage then passes legitimate signals onto Stage II. At this stage, the legitimate social signals undergo preprocessing in terms of acquiring meta-data

---

1https://www.omnicoreagency.com/twitter-statistics/
network having the following structure: 1) tokenized input is received by a sequential model of neural index of the word in a stored list (vocabulary). Henceforth, this to integers. This integer representation denotes the positional first processed by a tokenizer algorithm that converts words from geo-tags or obtaining location information from known landmarks in the target area. Once this position information is received, it is sent to Stage III. In the case of GPS data, Stage II directly passes this to Stage III. Finally, at Stage III, the SSRG utilizes these position measurements to generate the social signal residual. This residual is then sent to the Comparator block (along with the physical signal residual) in order to obtain an attack decision.

A. Fake Social Data Isolator

With vast social connectivity, Twitter and other OSN platforms have become a thriving ground for spamming accounts or tweets, bots, fake news, ad spam or phishing links, linkbait-and-switch, malware or “too-good-to-be-true” spams. For instance, Twitter systems flagged 3.2 million spamming accounts on their platform in 2018. This signifies the importance of isolating fake tweets or spam from credible social signals in order to retrieve meaningful traffic information from these social data. In other words, we must ensure that the social data feedback is credible.

Thus, social signals that arrive at the Social Data-based filter in the form of SUMs or strings are analyzed to isolate credible social data from fake or spam tweets. To do so, strings are first processed by a tokenizer algorithm that converts words to integers. This integer representation denotes the positional index of the word in a stored list (vocabulary). Henceforth, this tokenized input is received by a sequential model of neural network having the following structure: 1) an embedding layer; 2) two layers of LSTM-RNN [38]; and 3) two fully connected layers with 8 and 1 neurons. We add a sigmoid activation function to the single neuron of the fully connected layer to evaluate the binary classifier output. The FSDI network structure based on LSTM-RNN is shown in Fig. 4.

The embedding layer of the sequential model maps words of similar meaning closer to one another thereby extracting key features of the data set. The number of outputs of the embedding layer denotes the number of features for our network model. In this problem, we have set the feature size equal to the size of the vocabulary. These features are then sent to the LSTM layers of the network.

Like any standard recurrent network, the LSTM-RNN has the ability to derive long-term dependency information from the input data. This enables the neural network model to capture contextual information from the embedded social signal and effectively distinguish between fake and genuine ones. A standard LSTM recurrent neuron has the following gates: 1) input gate; 2) candidate gate; 3) forget gate; and 4) output gate. The first LSTM layer in the sequence contains 48 neurons while the second 24 neurons. The input to the LSTM network, if needed, is padded with zeros or truncated to standardize an input length and the size of the input layer for LSTM is equal to the size of the vocabulary obtained from the tokenizer. We set our learning rate to be 0.01 and binary cross-entropy as the loss function. Additionally, a drop-out of 0.4 is applied after each LSTM layer and Adam optimizer with $l_1$ kernel regularization is chosen to reduce fitting error and avoid overfitting of the models. Finally, we add a sigmoid activation function to the single neuron of the fully connected layer to evaluate the binary classifier output.

The isolated credible social data can then be utilized by the SSP in Stage II.

Remark 1: It should be noted here that the classification of credible vs illegitimate tweets has been attempted in the literature using content-based and account-based strategies [39]. Here, we have restricted ourselves to content-based classification problems as the account-based study is beyond the scope of this work.

B. Social Signal Processor

We utilize the SSP introduced in our previous work [10] to generate positions or trajectories information of vehicles in the traffic. The processor first collects social data in the form of meta-data such as geo-tags or location data from tweets as well as GPS signals from the mobile devices (when turned ON). Thereafter, position information from GPS signals or geo-tags is passed on to SSRG as is. Simultaneously, this processor also ensures that the collected tweets are relevant. The relevant texts are defined as those which provide information about the present location of a user (connected to the vehicle) while irrelevant texts do not. For example, among the following tweets—"I’m near Pho 11 Vietnamese Restaurant," "#Beef #Pho @Pho 11 Vietnamese Restaurant is terrific!" and “ Last night party at pho 11 was a blast”—only the first is relevant as it can be used to extract present position data of the vehicle. This filtering is achieved using standard natural language processing (NLP) techniques [10]. In [40], it is shown how temporal information in tweets can be used in extract fine-grained locations.

Thus, using the knowledge of the position of landmarks along the traffic of interest and the GPS or geo-tags signals of the social data from the vehicles, the positions or trajectories of the vehicles can be successfully obtained from the SSP. Once position is obtained, it is used to estimate the density of the traffic in those positions or along those trajectories. These
estimates are eventually passed as a feedback to the SSRG in Stage III.

**C. Social Signal Residual Generator**

Position measurement from the SSP is subsequently used to generate the local density using (8) and (9). This density measurements is used by the SSRG and are given by $y_i(t) = \rho(z_i, t) \forall i = 1, 2, \ldots, N$, where $N$ represents the number of social-agents or human-in-the-loop transmitting social data. In transformed variables, this can be expressed as $y_i(t) = (|v(z_i, t) - w(z_i, t)|/\hat{V}_{opt}(\rho^*)) \forall i = 1, 2, \ldots, N$. These $N$ output can also be represented as a vector [41]

$$y_i = \int_0^L c(x) v(x, t) - w(x, t) \frac{dv(x, t)}{\hat{V}_{opt}(\rho^*)} dx$$

where $c(x) = [c_1(x), c_2(x), \ldots, c_N(x)]^T$ and $c_i(x)$ is defined in terms of dirac delta functions $c_i(x) = \delta(x - z_i) \forall i = 1, 2, \ldots, N$. Next, the dynamics of the model-based SSRG is given by

$$\tilde{w}_i = -k_1 \tilde{w}_x - k_2 \tilde{w} + \alpha \hat{V}_{opt}(\rho^*) c(x) \tilde{V}_x \hat{v} (y_i - \tilde{y}_i)$$
$$\tilde{v}_i = -k_2 \tilde{v}_x - k_2 \tilde{w} + \beta \hat{V}_{opt}(\rho^*) c(x) \hat{v} (y_i - \tilde{y}_i)$$

where $\tilde{w}, \tilde{v}$, and $\tilde{y}_i$ represent the estimates of $w, v$ and $y_i$, respectively; $\alpha = \text{diag}(\alpha_1, \alpha_2, \ldots, \alpha_N)$, $\beta = \text{diag}(\beta_1, \beta_2, \ldots, \beta_N)$ and $\alpha_i, \beta_i \forall i$ are the filter gains. Here, we assume that the inlet velocity $v(0, t)$ is measured and known and is thus used as the boundary condition for the filter as well.

Thus, the errors for the SSRG can be defined by $\tilde{W} := w - \tilde{w}$ and $\tilde{V} := v - \tilde{v}$ and their dynamics is given by the difference of (11)–(13) and (4), (6), (7)

$$\tilde{W}_i = -k_1 \tilde{W}_x - k_2 \tilde{W} + \alpha \hat{V}_{opt}(\rho^*) c(x) \tilde{V}_x \hat{v} (y_i - \tilde{y}_i)$$
$$\tilde{V}_i = -k_2 \tilde{V}_x - k_2 \tilde{W} + \beta \hat{V}_{opt}(\rho^*) c(x) \hat{v} (y_i - \tilde{y}_i) + \delta_1$$

The residual for the Social Data-based Filter is given by

$$r_s(t) = (y_i - \tilde{y}_i)^2 \hat{V}_{opt}^2 = \Sigma_i (\tilde{V}(z_i, t) - \tilde{W}(z_i, t))^2$$

where the dynamics of $z_i$ is given by (8). This residual signal $r_s(t)$ converges to zero under no attack and remains stable under attack scenarios. The proof of this can be found in the Appendix A of this article.

**V. PHYSICAL DATA-BASED FILTER**

Unlike the Social Data-based Filter, the Physical Data-based Filter obtains its measurement from the outlet of the traffic (at $x = L$) in the form of flow measurement and is given by $y_p(t) = v(L, t)$.

As before, using the system model (4), (6), (7) and output injection, we write the Physical Data-based Filter model

$$\hat{v}_i = -k_3 \hat{v}_x - k_2 \hat{v} + y_1(x) \hat{v}_p - \hat{y}_p$$

where $\hat{v}, \hat{v}$ and $\hat{y}_p$ represent the estimates of $w, v$ and $y_p$, respectively, obtained from the Physical Data-based Filter, and $\gamma_1(x)$ and $\gamma_2(x)$ are filter gains.

The errors for the Physical Data-based Filter is then defined as $\tilde{W} := w - \tilde{w}$ and $\tilde{V} := v - \tilde{v}$ and their dynamics is given by the difference of (19)–(21) and (4), (6), (7)

$$\tilde{W}_i = -k_1 \tilde{W}_x - k_2 \tilde{W} + \gamma_1(x)(y_p - \tilde{y}_p)$$
$$\tilde{V}_i = -k_3 \tilde{V}_x - k_2 \tilde{W} + \gamma_2(x)(y_p - \tilde{y}_p) + \delta_1$$

The residual for the Physical Data-based Filter is given by

$$r_p(t) = \tilde{V}^2(L, t) dx$$

This residual $r_p(t)$ converges to zero under no attack and remains stable under attack scenarios. The proof of this can be found in Appendix B of this article.

**Remark 2:** Due to the unpredictability of human/social factors, the behavior of the structure of the social data may vary over time. In order to adapt to such temporal fluctuations, the training of LSTM-RNN must be adaptive, structurally, functionally, as well as parametrically, over time.

**Remark 3:** In this framework, we have also assumed that the passengerly Internet-enabled mobile devices. Utilizing these devices, the passenger can log into a mobile application when starting their ride in an autonomous vehicle. Similar to ride-sharing applications like Uber, this application will pair/match passengers with vehicles. This enables us to focus on a subset of globally generated tweets. In addition to leveraging the text-based social messaging service, we also include the social/mobile device generated GPS of the passengers. This would provide us with additional redundancy to support our discriminant system.

**VI. COMPARATOR**

As shown in Fig. 3, the residuals generated by the Social Data-based Filter $r_s(t)$ and Physical Data-based Filter $r_p(t)$ are propagated to the Comparator block. The purpose of the Comparator is to process generated residuals in a meaningful way in order to provide an attack-detection decision.

From the previous theoretical analysis, it is evident that under no attack scenarios both the residuals converge to zero. However, in practice, even under no attack these residuals will never be identically zero because of system uncertainties from model and measurement inaccuracies. Accordingly, we define a threshold such that residual signals greater than that predetermined value would indicate the presence of an attack. Such computation of threshold is standard in the community of fault detection and generally obtained under no fault scenarios [42]. In this work, we chose the threshold by finding the maximum value of the residual under no attack conditions [10].

Once these thresholds are obtained for both the Physical and Social Data-based Filters, both the residuals are compared to their corresponding thresholds, and a logical output is generated. In case of the Physical Data-based Filter, if the threshold is given by $r_{th,p}$, then $r_p(t) \geq r_{th,p}$ produces a high
and a low otherwise. Similarly, for the Social Data-based Filter with a threshold $r_{th,s}$, if $r_s(t) \geq r_{th,s}$ then it produces a high, otherwise a low.

Next, these logical signals are compared to determine the final attack detection in the following way: if at least one of the residuals produces a high, then a positive attack decision is confirmed. In other words, only if both the filters produce a low signal, a negative attack decision is made. This implies that the proposed cyber-attack detection scheme utilizes the redundancies of both physical data and social data to increase effective attack detection. The complete attack-detection logic for the Comparator block is shown in Table II.

**TABLE II**

| Physical Residual | Social Residual | Decision |
|-------------------|-----------------|----------|
| Low               | Low             | No       |
| High              | Low             | Yes      |
| Low               | High            | Yes      |
| High              | High            | Yes      |

**VII. IMPLEMENTATION**

In this section, we discuss the implementation details of the proposed cyber-attack detection strategy, namely, the Social and Physical Data-based Filters, and the Comparator.

A. Implementation of Social Data-Based Filter

The primary goal of the Social Data-based Filter is to generate the social residual $r_s(t)$ for the Comparator. It achieves that in three sequential stages: first, it identifies credible social data from passengers (Stage-FSDI), then converts them to equivalent position signals for their vehicles (Stage-SSP), and finally generates the residual (Stage-SSRG). The inputs to the filter are time instant $t$, social signal $S$, and model parameter set $P$ which contains $\rho^s, \rho_m, \nu^s, \Delta X,$ and $T_r$. Algorithm 1 details these stages where *Social Filter* designates the main function along with ancillary functions for FSDI, SSP, SSRG, and local density calculation.

Once a social signal reaches the Social Data-based filter (line 22), it is categorized as string data from SUMs or float data GPS signal (line 23). The float data from GPS provides position information directly to the SSRG (lines 10, 11, and 24) while the string data is passed through FSDI and SSP in order to extract equivalent position signals (lines 5–9 and 24).

In order to implement the FSDI (lines 1–3), we first implement the LSTM-based fake data isolator based on the structure discussed in Section IV-A. Thereafter, we randomly select a portion (65%) of the tweets and designate them as our training dataset. The rest of the tweets are used for testing the performance of our FSDI system (discussed in Section VIII-A). The training tweets/strings are henceforth tokenized/parsed into individual words (line 2). These tokens constitute as inputs to the LSTM. The performance of the LSTM is discussed in Section VIII-A.

Once the network is trained, it can provide a flag denoting whether a received text-based social signal is credible or not (line 3). Once the credible texts are identified, they are forwarded to the SSP to extract equivalent position signals.

This block is implemented in the functional block $SSP$ in Algorithm 1. $SSP$ either uses GPS signal from the mobile devices directly or extract equivalent position signals from credible social text data by using NLP. A detailed discussion of the implementation has been described in our previous work [10]. In Algorithm 1, this is denoted by the function NLP (line 7).

Thereafter, the generated equivalent position signals $z(t)$ (from line 7 or 11) are sent to obtain the local density (line 24). The first task of $Local\_Density$ function is to solve (8) to obtain moving micro–macro boundary variable $b(t)$ (line 14). This is then utilized to obtain the local density $\rho(z, t)$ which can approximate the global density at the position of the data-generating vehicle $\rho(z, t)$ (line 15). Subsequently, this local density is used by SSRG as a measurement (line 25).

The implementation has been described in our previous work [10]. In Algorithm 1, this is denoted by the function NLP (line 7).

In Algorithm 1, this is denoted by the function NLP (line 7).

Finally, the Social Data-based filter generates the social residual $r_s(t)$ (line 26) following Algorithm 1.

This is then utilized to obtain the local density (line 24). In order to solve the PDE filter model (11)–(13) to obtain $\bar{v}, \bar{w}$, the computational task of the implementation has been described in our previous work [10]. In Algorithm 1, this is denoted by the function NLP (line 7).

Field $P$ which contains $\rho^s, \rho_m, \nu^s, \Delta X,$ and $T_r$. Finally, the Social Data-based filter generated the social residual $r_s(t)$ (line 26) following Algorithm 1.

Field $P$ which contains $\rho^s, \rho_m, \nu^s, \Delta X,$ and $T_r$. Finally, the Social Data-based filter generated the social residual $r_s(t)$ (line 26) following Algorithm 1.

Field $P$ which contains $\rho^s, \rho_m, \nu^s, \Delta X,$ and $T_r$. Finally, the Social Data-based filter generated the social residual $r_s(t)$ (line 26) following Algorithm 1.
Algorithm 2: Generate \(r_p(t)\)

**Input:** Time instant \(t\), physical signal \(y_p(t)\), parameter set \(\mathbb{P}\).

**Output:** Physical Residual \(r_p(t)\).

1. **Function** Physical Filter \((t, y_p(t), \mathbb{P})\):
   2. \(\hat{y}_p(t) \leftarrow \hat{v}(L, t)\);
   3. Solve PDE Filter Model (19)-(21) to obtain \(\hat{v}, \hat{w}\);
   4. \(r_p(t) \leftarrow (y_p(t) - \hat{y}_p(t))^2\);
   5. return \(r_p(t)\);

Algorithm 3: Attack Flag Generation

**Input:** Residuals \(r_s(t)\) and \(r_p(t)\), Residual thresholds \(r_{th,s}\) and \(r_{th,p}\).

**Output:** Attack Flag (Boolean).

1. **Function** Comparator \((r_s(t), r_p(t), r_{th,s}, r_{th,p})\):
   2. Define \(\Delta r_s := r_s(t) - r_{th,s}\);
   3. Define \(\Delta r_p := r_p(t) - r_{th,p}\);
   4. if \(\Delta r_s \geq 0 \ OR \ \Delta r_p \geq 0\) then
      5. Flag = True;
   6. else
      7. Flag = False;
   8. return Flag;

B. Implementation of Physical Data-Based Filter

Similar to the Social Data-based Filter, the primary objective of the Physical Data-based Filter is to generate the physical residual signal \(r_p(t)\) for the Comparator. The inputs to this filter algorithm are given by time instant \(t\), physical signal \(y_p(t)\) and parameter set \(\mathbb{P}\). We note here that the physical signal is the measurement of velocity from the outlet of traffic freeway under consideration and it can be readily available from devices such as loop detectors. Thus, the physical signal does not require any additional preprocessing (line 2) in order to be utilized by the model-based filter (19)-(21) (line 3). Thus, Algorithm 2 shows the function Physical Filter which can be used to generate the residual \(r_p(t)\) (26) (lines 4 and 5).

C. Implementation of Comparator

The implementation strategy for Comparator is shown in Algorithm 3. The inputs to this algorithm are the residuals and corresponding thresholds from the Social Data-based Filter and the Physical Data-based Filter, i.e., \(r_s(t), r_{th,s}\) and \(r_p(t), r_{th,p}\), respectively. The algorithm implements the attack detection logic given in Table II (lines 4–7) and generates an attack detection flag as its output (line 8).

**Remark 4:** In transportation, usual control-theoretic cyber-attack detection schemes are realized by analyzing residual from physical data only. This is accomplished by using strategies similar to Algorithm 2 alone. In contrast, in our proposed scheme we use, namely Algorithms 1–3 to detect cyber-attacks. Here, Algorithm 1 meaningfully links the microscopic vehicle dynamics model (social data generators) to the macroscopic traffic dynamics model. Next, utilizing this link, Algorithm 1 generates a redundant social residual. Finally, Algorithm 3 employs the residuals from Algorithms 1 and 2 to detect cyber-attacks. Thus, the novelty of our proposed algorithm lies in the utilization of social data in conjunction with physical data for effective cyber-attack detection.

VIII. SIMULATION RESULTS

In this section, we present simulation results to show the effectiveness of our cyber-attack detection scheme using the redundancies in Social and Physical Data-based Filters. We first test the performance of the LSTM for fake data isolation. Then we simulate both the microscopic model (8) and the equivalent linearized macroscopic model (1) under equivalent operating conditions. The microscopic model is used to generate vehicle-level data for Social Data-based Filter whereas the macroscopic model is used to generate physical measurement data for the Physical Data-based Filter. In order to emulate realistic scenarios, we have injected noise in both physical and social measurements. The source of noise in physical data is the inaccuracies in loop detector measurements whereas the noise in social data arises from intermittent and delayed nature of GPS signals and tweets. The prescribed thresholds for the Social and Physical Data-based Filters are chosen as \(1 \times 10^{-5}\) and \(2 \times 10^{-5}\), respectively.

A. Performance of LSTM for Fake Data Isolation

To evaluate the performance of the LSTM-based FSDI module, we have generated a set of artificial tweets. For our problem, real-life fake data could not be gathered due to the lack of available dataset. In order to show the proof-of-concept for the scheme, we generated synthetic data containing legitimate tweets resembling real tweets mentioning landmark locations. Here, we have used 1333 samples of tweets and our goal was to create social data that are related to landmarks present along the North and South Atherton Street of the State College area. A portion of these tweets was generated by mining original tweets (from Twitter), SUMs (from swarmapp.com) or Twitter flagged tweet pool and/or by altering the name of the landmarks in original tweets or SUMs. Common strategies to create fake/corrupted data were by adding or manipulating original tweets in the following way: 1) addition of spam links; 2) inclusion of “too-good-to-be-true” claims; 3) mention of fake or out-of-scope landmarks; and 4) fake tweets (such as fake restaurant promotions, false donation promises in exchange of retweets, etc). Another portion of the tweets was collected from the spam database of [43] and similarly altered to reflect the name of the landmarks in interest. In this dataset, we have 695 legitimate tweets and 638 illegitimate tweets. An example of a legitimate tweet is “Long queue at Sheetz” while that of an illegitimate tweet is “Bicycle shop has free servicing, I am here! bestcheapserice.com.”

To train our LSTM neural network for fake data isolation/classification, we split the dataset described in Section IV-A into 35% testing and 65% training datasets. Similar to the training data, the testing data is parsed into words and the corresponding tokens are used as input to the LSTM for testing.
This classification model is tested using multiple metrics. The confusion matrix for the test dataset is shown in Fig. 5. First, we evaluate the performance of the network using the definition of accuracy $\frac{\text{Correctly predicted cases with fault and no fault}}{\text{Total number of cases}}$. The accuracy of our network on the test dataset is 92% with 8% misclassification.

As the error costs of positive and negative classification are different, we look at the Sensitivity metric of the trained network which is defined as follows: Sensitivity $\frac{\text{Correctly identified cases with fault}}{\text{All faulty cases}}$. We intended to build a highly sensitive network such that a maximum number of false social data are flagged. The Sensitivity of our network was obtained to be 95%. Additionally, the F-score for our classifier is evaluated to be 0.925 [44].

B. Performance of Detection Scheme Under In-Domain Cyber-Attacks

In this study, we illustrate the advantage of using both social and physical data as opposed to just physical data. We show three cases showing the advantage of our proposed method. 

**Case I:** We inject a stealthy in-domain attack at 100 s to the velocity profile. This attack is “stealthy” in the sense that it does not show up at the physical data sensor of the system (i.e., in $\rho(L)$) [18]. Essentially, this attack acts as a high amplitude perturbation occurring somewhere in-domain. The velocity and density response under this attack is shown in Fig. 6 where it is evident that the effect of in-domain attack does not show up significantly at the outlet measurement. Fig. 6 also shows the residual response under attack where the residual for both the filters remains below the threshold until the injection of attack at 100 s. However, since the outlet measurement is used by the Physical Data-based Filter, the residual of the Physical Data-based Filter remains below its threshold even after the attack injection. On the other hand, as Social Data-based Filter uses in-domain data from the vehicles, its residual crosses its threshold within 2.5 s of the attack. This provides a high to the comparator which can make a positive attack detection decision using Table II. This implies that such attacks would remain undetected if Physical Data-based Filter is used exclusively. However, they will be detected by our scheme as we exploit the redundancies between social and physical data.

It should also be noted here that Social Data-based Filter can only detect an attack if the social sensors are in the spatiotemporal impact zone of the attack. Strictly speaking, this implies that the social sensor must lie in the zone of influence of the attack propagation. This indicates that the location and timing of the social sensor in relation to the injected attack is a key point in this setting.

**Case II:** Social data is nonstationary as well as intermittent. This implies that under certain scenarios, social data sensors might not be available near the spatiotemporal impact zone of an attack. In this case, if a physical sensor is present in the zone, Physical Data-based Filter can detect these attacks while the Social Data-based filter cannot. For example, in our setting, if the in-domain attack is injected closer to the physical sensor at the outlet at 100 s. This is evident from the traffic velocity and density profile in Fig. 7. We can also observe from Fig. 7 that the residuals of both the filters remain below the threshold after the attack injection. On the other hand, since no vehicles are present in the immediate impact zone to transmit social data, the residual for the Social Data-based Filter remains below the threshold, signaling a low. Using these two signals, the comparator can decide an attack has occurred in the system using Table II.

It is to be noted that physical sensors are fixed and do not have the maneuverability of social data sensors. Moreover, they are expensive to install and cannot be deployed in large numbers. However, we see from this case study that...
the availability of both physical and social data from vehicle/infrastructure sensors and human users’ mobile devices, respectively. We exploit the redundancy between these social and physical data and design our cyber-attack detection scheme based on a macroscopic traffic model. Essentially, the scheme consists of a Social Data-based Filter and a Physical Data-based Filter running in parallel and producing residuals. The attack decision is made by comparing these two residuals. We have analyzed the mathematical properties of such filters using Lyapunov’s stability theory. Furthermore, we performed simulation studies that illustrate the efficacy of the proposed scheme. As for future studies, we plan to explore the effectiveness of the proposed scheme under various types of uncertainties related to social data generation and processing.

APPENDIX A

CONVERGENCE OF RESIDUAL DYNAMICS FOR SOCIAL DATA-BASED FILTER

Theorem 1: Consider the error dynamics (15)–(17), residual definition (18), and the dynamics of social data sensors (8). The residual signal is asymptotically stable in the sense of

\[ r_s(t) \leq c_1 r_s(0) \exp(-\lambda_s t) \]

for some \( c_1 > 0 \) and \( \lambda_s > 0 \) without any attack. Furthermore, the residual signal is input-to-state stable in the sense of

\[ r_s(t) \leq c_2 r_s(0) \exp(-\lambda_s t) + c_2 \left( \| \delta_1(., t) \|^2 + \delta_2^2(t) \right) \]

for some \( c_2 > 0 \) under attack. These conditions can be guaranteed if there exists a positive scalar \( \lambda_s \) such that LMI condition \((P + \lambda_s I) < 0\) is satisfied and matrix \( P = \sigma(P_i) \), \( \forall i \) where \( \sigma \) represents the maximum singular value and matrix \( P_i \) is given as follows:

\[ P_i = \frac{1}{\Delta x_i} \begin{bmatrix} \alpha_i & -\frac{1}{2}(\alpha_i - \beta_i) \\ -\frac{1}{2}(\alpha_i - \beta_i) & -\beta_i \end{bmatrix} \]

where \( \Delta x_i = \tilde{x}_i - \tilde{x}_{i+1} \) and for all \( i = 1, \ldots, N \), the interval \([\tilde{x}_{i+1}, \tilde{x}_i]\) contains the position \( z_i \) of the \( i \)th vehicle.

Proof: Consider the following Lyapunov candidate functional: \( \mathcal{E}(t) = \mathcal{E}_1(t) + \mathcal{E}_2(t) \), where \( \mathcal{E}_1 = \int_0^t (e^{-\lambda_s} \tilde{W}^2(., .)) \) and \( \mathcal{E}_2 = \int_0^t (e^{-\lambda_s} \tilde{V}^2(., .)) \) dx and take derivatives with respect to time. Now using the mean value theorem, we can assert that for each position \( z_i \) of the \( i \)th vehicle we can find an interval \([\tilde{x}_{i+1}, \tilde{x}_i]\) \( \forall i \in \{1, \ldots, N\} \) such that [45]

\[ e^{-\lambda_s} h(z_i, t) = \sum_{i} \frac{1}{\Delta x_i} \int_{\tilde{x}_{i+1}}^{\tilde{x}_i} e^{-\lambda_s} h(x, t) dx \]

where \( h(.) = \alpha_i \tilde{W}^2(., .) - \beta_i \tilde{V}^2(., .) + (\beta_i - \alpha_i) \tilde{V}(.,.) \tilde{W}(.,) \). We note here that \( 0 < \tilde{x}_{i+1} < \cdots < \tilde{x}_i < L \). Let us define two more points \( \tilde{x}_{N+1} := 0 \) and \( \tilde{x}_0 = L \). This implies any integral over the domain \([0, L]\) can be written as: \( \int_0^L f(x) dx = \sum_{i=0}^{N+1} \int_{\tilde{x}_{i+1}}^{\tilde{x}_i} f(x) dx \). Using this relation, (30) and Young’s Inequality in \( \mathcal{E}_1(t) \) and \( \mathcal{E}_2(t) \) yields

C. Pros and Cons of the Scheme

The major advantage of our scheme is the utilization of the additional number of measurements from social data without added infrastructure for physical measurement. Moreover, we take advantage of the dynamic nature of these sensors compared to fixed physical sensors. However, the inaccuracy and arbitrariness of geographical location information present in social data impacts their efficacy. Therefore, the data preprocessing time and the intermittent, unreliable nature of social data can also impact the accuracy of our proposed detection scheme. Thus, this scheme can be deployed on a smaller scale for critical areas in the beginning, until these challenges are addressed on a larger scale.

IX. CONCLUSION

In this article, we explore a cyber-attack detection scheme for socio-technical transportation systems. We consider despite such disadvantages of physical sensors, they can provide valuable attack information in case social sensors are unavailable. Thus, a faster response and mitigation can be undertaken using the output of the Physical Data-based Filter as well.

Case III: Finally, we present the case where the residuals of both the filters are affected by a cyber-attack. In this scenario, a wide-spread in-domain attack is injected the effect of which can be seen from the velocity and density profiles of the traffic in Fig. 8. Such a case is an ideal scenario where there are social data sensors as well as physical data sensors in the impact zone of the attack and both the residuals cross their thresholds after the attack injection. Consequently, the comparator receives high signals from both the filters and is able to make a positive attack detection decision. We also note here that, similar to previous cases, the residuals of both the filters remain below their respective thresholds before the injection of attack at 100 s. Notably, this scenario can occur under two possible situations: 1) the magnitude of the attack is large such that it has a wide spatiotemporal impact zone or 2) irrespective of the size of the attack, both physical and social sensors are present in the zone of impact of the attack.
where positive constant \(0 < \gamma < k_3\). We also note here that \(\hat{W}(0) = \delta_2\). Next, we define a vector \(\xi = e^{-\delta_2 t/2}\hat{W}, \hat{V}\)^T, matrix \(\mathcal{P}\) as in (29) which yields

\[
\dot{\mathcal{E}}(t) \leq \sum_{i=0}^{N+1} \xi_i^T \mathcal{P} \xi + \frac{1}{2\gamma} \|\delta_i(t)\|^2 + \frac{k_1}{2} \delta_i^2(t).
\]

Subsequently, we choose a matrix \(\mathcal{P}\) such that the maximum singular value of \(\mathcal{P}\) is greater than the maximum singular value of \(\mathcal{P}_y \forall i, \text{i.e., } \tilde{\sigma}^2(\mathcal{P}_y) \leq \tilde{\sigma}^2(\mathcal{P}) \forall i\). Next, we can choose \(\lambda_i\) such that it satisfies the LMI \((\mathcal{P} + \lambda_i \mathcal{I}) < 0\). This in turn yields

\[
\dot{\mathcal{E}}(t) \leq -\lambda_i \mathcal{E}(t) + \frac{1}{2\gamma} \|\delta_i(t)\|^2 + \frac{k_1}{2} \delta_i^2(t).
\]

Consequently, using Grönwall’s inequality [46] we can write \(\dot{\mathcal{E}}(t) \leq e^{-\lambda_i t} \mathcal{E}(0) + \frac{m_1}{\gamma} \text{sup}_t(\|\delta_i(t)\|^2 + \delta_i^2(t)), \) where \(m_1 = \text{max}(1, k_1/\gamma))/2\).

Next, we attempt to prove two inequalities: (I) \(m_2(r_t) \leq \mathcal{E}(t)\) and (II) \(\mathcal{E}(0) \leq m_3 r_2(t)\) for \(m_2, m_3 > 0\). Using Young’s Inequality one more time in (18) we obtain \(r_2(t) \leq m_4 \sum_{i} [\hat{W}^2(z_i, t) + \hat{V}^2(z_i, t)], \) for some \(m_4 = \text{max}(1 + [1/m_3]) \) and \(m_5 > 0\). Subsequently, it is trivial to note that \(\sum_{i} [\hat{W}^2(z_i, t) + \hat{V}^2(z_i, t)] \leq 2e^2 \mathcal{E}(t). \) This yields our required inequality (I) for \(m_2 = e^{-\lambda_i t}/(2m_4).\)

To prove the next inequality (II), we consider bounded initial conditions for the error system, i.e., \(0 < |\hat{W}(x, 0)|, |\hat{V}(x, 0)| < \infty.\) Using this assumption we can obtain (II) where \(m_3 = (L \text{max}_x(|\hat{W}(x, 0)| + |\hat{V}(x, 0)|)^2)/[\text{min}_x(\hat{W}^2(x, 0), \hat{V}^2(x, 0))].\) Finally, choosing \(c_1 = m_3/m_2\) and \(c_2 = m_1/m_2,\) we obtain (28). Moreover, when there is no attack, i.e., \(\delta_i \equiv 0\) and \(\delta_2 \equiv 0,\) we can obtain (27).

## Appendix B

### Convergence of Residual Dynamics for Physical Data-Based Filter

In order to show the asymptotic stability (under no attack) and input-to-state stability (under attack) of the physical data-based residual signal \(r_p(t)\) (26), we transform the error residual dynamics (23)–(25) using backstepping transformation [18]. The transforms used here are: \(\tilde{V} = \psi + \int_{0}^{t} G(x, y) \psi(y, t) dy + \int_{0}^{t} \mathcal{H}(x, y) \phi(y, t) \) dy and \(\hat{W} = \phi - \int_{0}^{t} \mathcal{F}(x, y) \phi(y, t) \) dy. These transformation maps the coupled PDE (23)–(25) to a boundary-condition coupled target PDE given as follows:

\[
\phi = -k_1 \phi_x - k_2 \phi, \quad \psi_t = -k_3 \psi_x + \delta_1
\]

\[
\phi(L, t) = \psi(L, t), \quad \phi(0, t) = \delta_2
\]

where \(\delta_1 = \tilde{\delta}_1 - \int_{0}^{t} \mathcal{F}(x, y) \tilde{\delta}_1(y, t) \) dy.

Using the backstepping transformation and comparing the target dynamics (34)–(35) with the residual dynamics (23)–(25), we obtain the dynamics of the kernels of the backstepping transformation: \(\mathcal{F}_x + \mathcal{F}_y = 0, \) \(\mathcal{F}(0, y) = 0, \) \(\mathcal{G}_x + \mathcal{G}_y = 0, \) \(\mathcal{G}(0, y) = 0, \) \(k_3 \mathcal{H}_x + k_1 \mathcal{H}_y - k_2 \mathcal{H} = k_2 \mathcal{F}_x, \) \(\mathcal{H}(0, y) = 0.\)

Furthermore, \(\psi(0, t) = 0\) since \(\hat{V}(0, t) = 0\) and the physical filter gains are chosen so that \(\gamma_1(x) = -\mathcal{F}(x, L)k_3/\rho^*\) and \(\gamma_2(x) = -\mathcal{G}(x, L)k_3/\rho^*.\) With this, we present our second theorem that specifies the design criteria for the Physical Data-Based Filter.

**Theorem 2 (Convergence of Residual Dynamics for Physical Data-Based Filter):** Consider the error dynamics (23)–(25) and residual definition (26). The residual signal is asymptotically stable in the sense of: \(r_p(t) \leq c_3 r_0(0) e^{-\lambda_2 t} + c_4 \text{sup}_t(\|\delta_i(t)\|^2 + \delta_i^2(t)),\) for some \(c_4 > 0\) under attack. These conditions are satisfied if the filter gains meet the prescribed conditions mentioned above.

**Proof:** The theorem can be proved following an approach similar to the one presented in [18].

## Acknowledgment

This work was conducted by all authors while they were at The Pennsylvania State University.

## References

[1] J. Reilly, S. Martin, M. Payer, and A. M. Bayen, “Creating complex congestion patterns via multi-objective optimal freeway traffic control with application to cyber-security,” *Transp. Res. B: Methodol.*, vol. 91, pp. 366–382, Sep. 2016.

[2] S. Gowrishankar, E. Stern, and D. B. Work, “Including the social component in smart transportation systems,” in *Proc. Nat. Workshop Transp. Cyber-Phys. Syst.*, 2014, pp. 1–3.

[3] Z. Zheng, C. Wang, P. Wang, Y. Xiong, F. Zhang, and Y. Lv, “Framework for fusing traffic information from social and physical transportation data,” *PloS One*, vol. 13, no. 8, 2018, Art. no. e020151.

[4] J. C. Becker and A. Simon, “Sensor and navigation data fusion for an autonomous vehicle,” in *Proc. IEEE Intell. Veh. Symp.*, 2000, pp. 156–161.

[5] M. Kordestani and M. Saif, “Data fusion for fault diagnosis in smart grid power systems,” in *Proc. IEEE 50th Can. Conf. Elect. Comput. Eng. (CCECE)*, 2017, pp. 1–6.

[6] C. Harris, A. Bailey, and T. Dodd, “Multi-sensor data fusion in defence and aerospace,” *Aeronaut. J.*, vol. 102, no. 1015, pp. 229–244, 1998.

[7] D. P. Mandic et al., “Data fusion for modern engineering applications: An overview,” in *Proc. Int. Conf. Artif. Neural Nets.*, 2005, pp. 715–721.

[8] H. F. Durrant-Whyte, “Sensor models and multisensor integration,” *Int. J. Robot. Res.*, vol. 7, no. 6, pp. 97–113, 1988. [Online]. Available: https://doi.org/10.1177/027836498800700608.

[9] N. Zheng et al., “Big data for social transportation,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 3, pp. 620–630, Mar. 2016.

[10] T. Roy, A. Tarig, and S. Dey, “A socio-technical approach for resilient connected transportation systems in smart cities,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 5019–5028, Jun. 2022.

[11] Z. A. Biron, S. Dey, and P. Psu, “Resilient control strategy under denial of service in connected vehicles,” in *Proc. Amer. Control Conf. (ACC)*, 2017, pp. 4971–4976.

[12] L. He and W. T. Zhu, “Mitigating DoS attacks against signature-based authentication in VANETS,” in *Proc. IEEE Int. Conf. Comput. Sci. Autom. Eng.* (CSAE), vol. 3, 2012, pp. 261–265.

[13] A. Petrilillo, A. Pescapé, and S. Santini, “A collaborative approach for improving the security of vehicular scenarios: The case of platooning,” *Comput. Commun.*, vol. 122, pp. 59–75, Jun. 2018.

[14] R. Mercier, Z. A. Biron, and P. Psu, “Replay attack detection in a platoon of connected vehicles with cooperative adaptive cruise control,” in *Proc. Amer. Control Conf. (ACC)*, 2018, pp. 5582–5587.
