12-15-2017

A Spatio-Temporal Approach to Mitigate Automotive Radar Spoofing Attacks

Prateek Kapoor
Binghamton University--SUNY, prateek.kapoor83@gmail.com

A SPATIO-TEMPORAL APPROACH TO MITIGATE AUTOMOTIVE RADAR SPOOFING ATTACKS

BY

PRATEEK KAPOOR

MS, State University of New York at Binghamton, 2017

THESIS

Submitted in partial fulfillment of the requirements for
the degree of Master of Science in Computer Science
in the Graduate School of
Binghamton University
State University of New York
2017

Accepted in partial fulfillment of the requirements for
the degree of Master of Science in Computer Science
in the Graduate School of
Binghamton University
State University of New York
Fall-2017

Dec 15, 2017

Dr. Kyong Dong Kang, Advisor
Department of Computer Science, Binghamton University
Dr. Leslie Lander, Committee Member
Department of Computer Science, Binghamton University

ABSTRACT

Cyber-physical system (CPS) has become an integral part of human life, ranging from aircraft to health care systems. The security of these critical components ensures its wider acceptability [1]–[4]. Traditionally, many works [5]–[9] to secure cyber-physical system (CPS) has been done in the cyber domain, like securing inter/intra CPS communication, securing the exposed software, rebuilding control input derived from sensor data post-digitization, using sensor fusion. All of this security software suffers from a basic attack wherein an attacker compromises the physical/analog sensing system. Researchers have made some progress in mitigating such attacks on physical/analog signals of CPS, the current state of the art methodology proposed in PyCRA [10] uses temporal random signals for physical challenge-response authentication. Though this approach immensely enhances the capability of identifying the sensor attacks, it fails to provide any recovery mechanism to the system. Recent work like Dutta et al., 2017 [11] tries to address this by introducing recursive least squares (RLS) based recovery mechanisms over PyCRA. Although these systems provide some recovery in trivial scenarios, they fail during longer attacks and also result in loss of control because of longer/frequent random no-signal periods. Which could be catastrophic in real-time systems. This work presents Spatio-Temporal Challenge-Response (STCR), an authentication scheme designed to protect active sensing systems against physical attacks occurring in the analog domain. This system utilizes multiple beam-forming [12] and provides physical challenge-response authentication (CRA) in both spatial and temporal domain. Thus providing a much more resilient authentication mechanism that not only detects malicious attacks, but also provides recovery from them. We demonstrate the resilience and effectiveness of STCR over the state of the art in detecting and mitigating attacks through several experiments using a car following (CF) model. This model deploys CPS in the follower car to sense the lead car's relative position and maintain a safe distance by manipulating acceleration.

ACKNOWLEDGEMENTS

I would like to thank my advisor Dr. Kyong Dong Kang. He has guided me in my research and encouraged me to ask fundamental questions. I'm really thankful to him for giving me an opportunity to become a member of the RTESL lab. He really cares about his students and I'm grateful for the opportunity to work with him. I would like to thank Dr. Leslie Lander for serving on my thesis committee.

This work has been supported, in part, by NSF Project CNS-1526932.

Contents

List of Tables

List of Figures

List of Abbreviations

FMCW frequency-modulated continuous-wave radar

CPS cyber-physical system

MUSIC multiple signal classification

CRA challenge-response authentication

RLS recursive least squares

RCA radar cross-section

ULA uniform linear array

x2 chi-squared

TX transmit

RX receive

VM vector mod

CF car following

ADC analog-to-digital converter

DAC digital-to-analog converters

ADAS advanced driver assistance systems

ACC adaptive cruise control

FDMB frequency-division multiple tx-beamforming

MIMO multiple-input multiple-output

DOA direction of arrival

SSB single side-band

VM vector modulator

RF radio frequency

MPC model predictive control

CRMSE continuous root mean square error

RMSE root mean square error

STCR Spatio-Temporal Challenge-Response

IoT internet of things

M2M machine-to-machine

Chapter 1

Introduction

The term CPS was coined in 2016 by Helen Gill at the National Science Foundation in the US [13]. Since then it has gone on to be associated with popular terms like the internet of things (IoT), industry 4.0, the industrial internet, machine-to-machine (M2M), the internet of everything, smart cities, and smart vehicles. All of which underscores the vision of a technology, to be deeply integrated into the fabric of our lives, motivating a lot of research into securing these systems. Although most of the work in securing these systems has been done at a system-level, like securing inter/intra CPS communication, securing the exposed software, rebuilding control input derived from sensor data post-digitization, using sensor fusion. It has been found that if an attacker is able to spoof the physical sensors of the system, they can render the system level security solutions useless [14], [15].

A CPS consists of three main component sensors, control system and actuators affecting each other via feedback loops. Sensors of these systems can be classified as passive or active. Passive sensors like temperature sensors just measure the ambient energy around them, where active sensors send out energy probes and measure their impact on the environment, like measuring the reflected signal as shown in Fig. 1.1. These sensors are susceptible to spoofing attacks, where an attacker captures and measures the original probe and sends out a malicious response, which may be a function of the expected response signal keeping the response under bounds set by a system level security scheme, hence by-passing them. To address this, recently a lot of research on securing active sensors of autonomous CPSs has been done and several attack detection schemes have been proposed such as PyCRA which highlight that the underlying physics governing the sensor can be used to come up with a security mechanism.

And there has been some research done on estimating correct sensor measurements during these attacks like in Dutta et al. 2017, which uses RLS based regression techniques, implying the need for these systems to report measurements under acceptable bounds even when under attack.

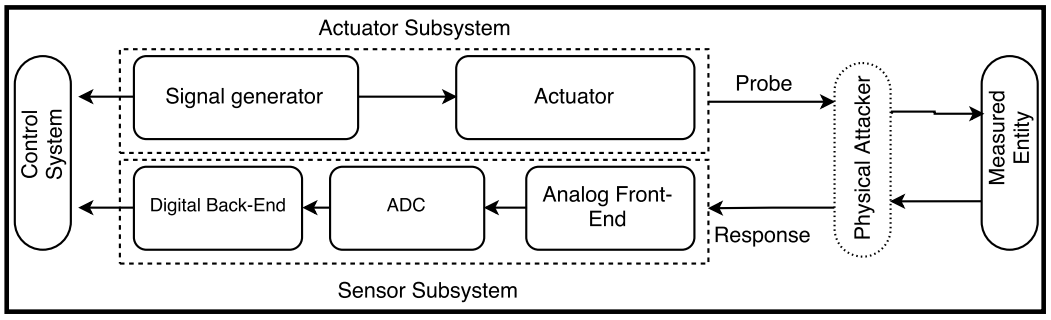

Figure 1.1: A typical CPS system with its sensors under physical attack: The actuator subsystem generates a signal and probes the measured entity. The response from the measured entity is captured by the analog front-end, undergoes analog to digital conversion and is sent to the control system for further processing. Here is a physical entity that captures the probe and sends a spoofed response to the sensor subsystem, altering the observed reality of the whole CPS subsystem.

A CPS system with PyCra uses a CRA modulator built into its actuator and sensor. The signal is modulated via pseudo-random binary modulation $\{0,1\}$, which means the system at random time intervals does not send out any signal, called as the challenge period. A naive attacker might continue to emit attack signal even during the challenge periods and can be detected. Although PyCRA schemes provide fundamental guarantees based on the physics of the sensors. It has to shut off the sensor during the challenge periods which can result in temporary loss of control and also it does not provide any recovery mechanisms. To address the 2_{nd} limitation of PyCRA, Dutta et al. 2017 proposed an RLS based recovery mechanism which relies on the energy sensed during the challenge period and uses it as an initial error input to an RLS estimator. Which predicts the sensor readings during the attack period. The issue with this approach is that it requires longer and more frequent challenge periods which results in longer duration of loss of control, also RLS estimator only has the initial challenge period readings where it can observe the true error introduced by the attacker, and after which it just assumes that the observed signal has the same error component which might not be true and the calculated sensor readings may no longer be accurate. We in this paper propose STCR which deploys CRA in both spatial and temporal domains. This is achieved by using multi-

ple beams forming with all beams separated in angle and base frequency and then applying synchronized CRA. Synchronized CRA makes sure that no two frequencies are in the challenge phase at the same time, hence the system can measure its environment all the time. Also, we don't assume the target to be a point target, i.e. we assume it has a non-zero radar cross-section (RCA). This allows us to send multiple physical CRA at different angles to the measured entity. We also propose a way to compartmentalize and isolate the attacked sensor elements and prune them from our observed signal matrix, enabling the system to operate even under attack.

With modern vehicular systems heavily relying on CPS systems, [16] identifies frequency-modulated continuous-wave radar (FMCW) sensors used in driving assist can be jammed and even worse spoofed using off-the-shelf equipment. The feasibility of such contact-less attacks were also highlighted in works by [17], [18]. This threat coupled with the sensitivity of the automobile industry to cost, there is a need to come up with a solution that is both cost-effective and secure. We in this paper take adaptive cruise control (ACC) used in vehicular systems as our case study. And propose a novel method to secure the physical FMCW sensor using currently deployed phased array radars coupled with synchronized CRA to come up with a cost-effective and reliable countermeasure system for vehicular radar sensors. We also simulated PyCRA and Datta et al. based approaches in the ACC system to provide a comparative performance analysis for our approach with respect to the current state of the art. We used the sensitivity of the system to detect attacks and accuracy of the system to provide measurements close to ground truth during attacks as the matrices to compare these approaches and found that STCR bettered the accuracy of the system by 6 times and was more sensitive to detect attacks.

The remainder of this paper is organized as follows. Section 2 lists the current state of the art and compares it to STCR. In Section 3, we list down the techniques used such as FMCW, active beam forming, the angle of arrival detection using multiple signal classification (MUSIC) and CF model. Then we explain STCR in Section 4 followed by performance evaluation in Section 5 and conclude in 6.

Chapter 2

Related Work

PyCRA is based on physical challenge response, which uses randomly spaced physical stimulation and subsequent behavior analysis to determine if the system is under attack. The randomly spaced physical stimulation $p(\tau)$ at time τ is provided by binary modulation $u(\tau)$ of the sensor probe $s(\tau)$ as in eq (2.1)

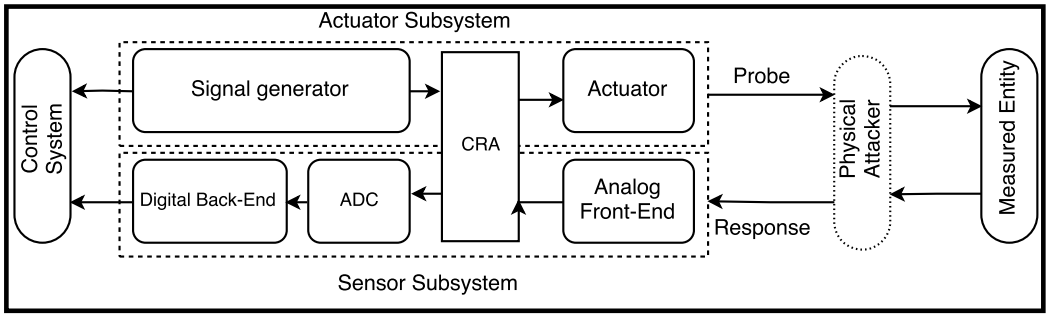

Figure 2.1: PyCRA model: A CPS system with CRA modulator built into its actuator and sensor. The signal is modulated via pseudo-random binary modulation 0,1, which means the system at random time intervals does not send out any signal. A naive attacker might continue to emit attack signal even during these random periods and can be detected.

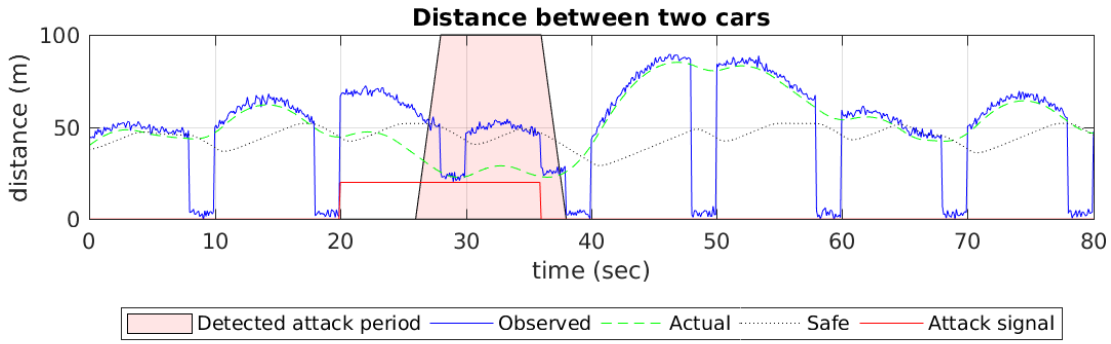

Figure 2.2: PyCRA in ACC output: Red line represents an attack vector and red region the period in which the algorithms detected the attack. Green line represents the ground truth and the blue distance calculated using received signal. Black spaced line is the safe distance based on the current relative speeds between the two vehicles

$$p(\tau) = u(\tau)s(\tau), u(\tau) \in \{0, 1\} \tag{2.1}$$

PyCRA uses residual energy $r(\tau)$ eq (2.2) which is the difference between the predicted output $\hat{p}(\tau)$ and the observed output $o(\tau)$ during $u(\tau) == 0$ (challenge period) to detect if the system is under attack. Here $\hat{p}(\tau)$ is some function over $p(\tau)$ which is defined by the underlying physics of the sensor system. PyCRA uses χ^2 method eq (2.3) to overcome the noise in the system by accumulating the squares of $r(\tau)$ during challenge period T, if the accumulated value is greater than a certain set threshold (θ) an alarm is raised.$\hat{p}(\tau)$ is equal to θ when χ^2 method is used eq (2.2).

$$r(\tau) = o(\tau) - \theta, u(\tau) = 0 \tag{2.2}$$

$$\chi 2 = 1/T(\sum_{\tau=t-T+1}^{t} r^2(\tau)) > \theta \tag{2.3}$$

To analyze PyCRA algorithm, we modeled it in a vehicle's ACC system Fig.2.1 with $\theta = 5$. We can observe the output $o(\tau)$(observed distance) of the active sensor in Fig.2.2, during no attack when $u(\tau == 10) < \theta$ which is as expected. Now, at $\tau = 30$ $r(\tau) \approx 20$ which is $> \theta$, hence the system triggers an alarm.

Although PyCRA provides good theoretical guarantees based on fundamental properties of physics, however, as we can see from the experiment output in Fig.2.2 it:

- Results in loss of control: As we can see during the challenge period the sensor ceases

measurements, this could be of serious concerns to real-time CPS, which rely heavily on their sensors to have a very high uptime. STCR does not suffer from such limitation as it uses multiple beam forming with synchronized CRA, theoretically ensuring 100% uptime.

- Provides no recovery: As we see from the Fig.2.2 that during an attack the actual distance between vehicles drops below the safe distance, there is no effort made by the system to recover even after the attack is detected. This might not be desirable in scenarios where switching off the CPS system relying on these attacked sensors is not an option e.g. medical equipment, automobiles, etc. STCR uses in-sensor fusion to safely estimate the sensor readings hence providing good recovery.

Then there is the solution proposed by Dutta et al., 2017. This solution tries to address the 2^{nd} drawback of PyCRA system and proposes a mechanism to provide recovery. They use RLS to predict sensor values when the attack is detected. The "RLSEstimate" function proposed in this paper depends on the $r(\tau)$ eq (2.2) value as the initial error input when an attack is detected. RLS also uses a forgetting factor λ which reduces the weight-age of older inputs, more essentially $r(\tau)$ calculated in challenge period. We know that Regression Algorithm's such as RLS have a very fast convergence speed [19] to the mean of the running variable, in this case running variable is $r(\tau)$ which tends to 0 until the next challenge period. This mechanism has 2 limitations:

- Firstly: $r(\tau)$ calculated during the challenge period is a critical feedback to "RLSEstimate" i.e. more challenge period readings mean better performance, resulting in longer periods of loss of control, which is not good for real-time CPS that rely heavily on their sensors to have a very high uptime.

- Secondly: due to fast convergence of RLS, as attack persists the system will trust that the error between observed and actual sensor readings(running variable) is ≈ 0 hence recoveries will stop.

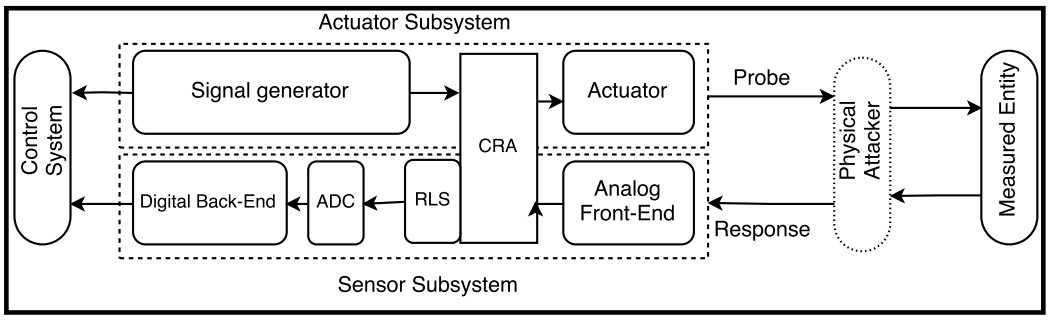

Figure 2.3: Dutta et al. 2017: A CPS system with CRA modulator built into its actuator and an RLS recovery module in the sensor. The module triggers recovery when an attack is detected,future values of the active sensor are estimated by using the recursive least square estimation method.

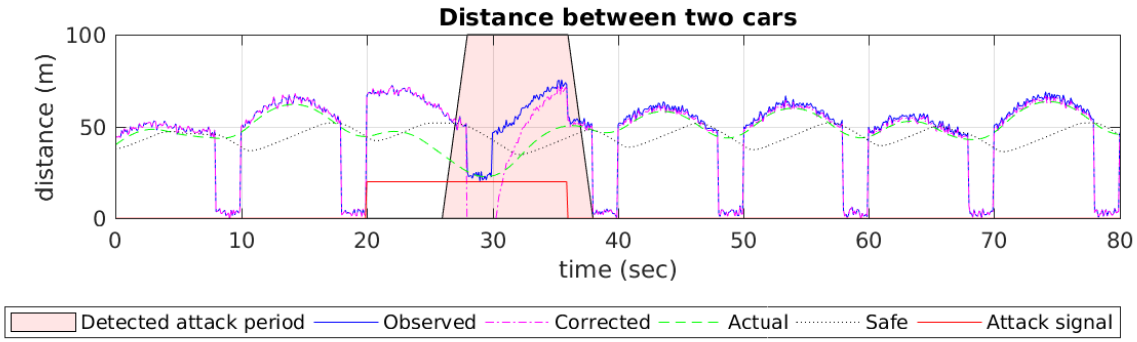

Figure 2.4: Dutta et al. 2017: Red line represents an attack vector and red region the period in which the algorithms detected the attack. The green line represents the ground truth and the blue distance calculated using received signal. The magenta line is the corrected reported distance. Black spaced line is the safe distance based on the current relative speeds between the two vehicles

To analyze this algorithm, we modeled it in a vehicle's ACC system Fig.2.3 with $\theta = 5$. We can observe the output $o(\tau)$(observed distance) of the active sensor in Fig.2.4, during no attack when $u(\tau == 10) < \theta$ which is as expected. Now, at $\tau = 30$ $r(\tau) \approx 20$ which is $> \theta$, hence the system triggers an alarm. Also, if we observe from $\tau = 30$ to 40 we can see the magenta line which is the corrected output by the RLS system converges to the distance reported by the error signal (observed signal) rather than actual signal.

We have also analyzed systems like FIRED [20] which frequently reset the system and depend on its own inertia to compensate for the loss of signal. And found they tend to suffer from temporary loss of control and fails in CF models where the lead car dynamics are not in con-

trol of the CPS and can vary over time. Although they acknowledge the same in their paper and provide solutions using multiple sensors which perform interleaved resets. STCR inherently achieves this interleaving using multiple beam forming and synchronized CRA, which are randomly distributed and grouped to provide a more resilient system.

There are also systems such as [9] which leverage heterogeneous sensors to model an attack vector which will be hard for the attacker to achieve. In their model, it is safely assumed that due to the heterogeneity of the system, not all sensors can be attacked at the same time. As we will be showing in our work, that we can achieve similar attack vectors by grouping random frequencies which cannot all be attacked at the same time.

And there are systems such as [21]–[23] which show how sensor fusion/voting based systems fail when more than 50% sensors are compromised and propose different mechanisms such as "a secure local control loop" can improve the resilience of the system, we achieve this by bucketing Multiple beams thereby curtailing the impact of an attack.

Chapter 3

Background

Now we take a look at the enabling technologies for STCR.

3.1 FMCW

Frequency-modulated,continuous-wave based radar Fig. (3.1) is widely used in automobile industry due to its low cost and form factor.

In FMCW a signal is transmitted, which increases or decreases in the frequency as a function of time. The difference between the transmit (TX) f_1 and receive (RX) f_2 frequencies known as a beat frequency f_b is used to calculate the time delay $\triangle t$. Which can be used to calculate the $Range$.

$$f_b = f_2 - f_1 \tag{3.1}$$

$$\triangle t/T_s = f_b/B_s \tag{3.2}$$

$$Range = cT_s f_b/2B_s \tag{3.3}$$

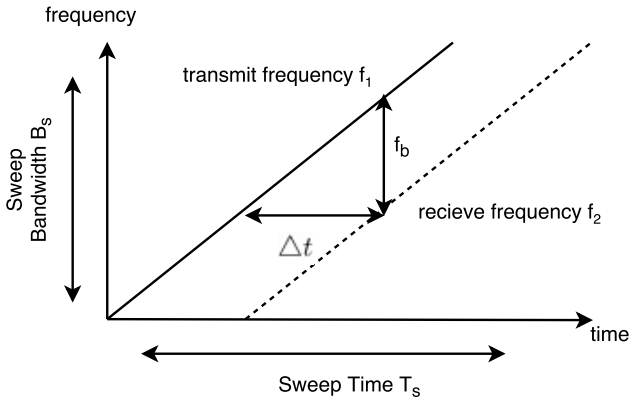

Figure 3.1: FMCW Principle

Where c is the speed of light $3 \times 10^8 m/s$, $T_s >> \triangle t$ is the sweep time, that is the maximum round trip duration of the signal which is a function of the maximum range of the system. B_s is the bandwidth response range of the signal.

The above principle can be used to calculate the relative radial velocity v_r by using doppler frequency shift $\triangle f$ for a given radar wavelength λ.

$$\triangle f = 2v_r/\lambda \tag{3.4}$$

3.2 MUSIC

MUSIC by Schmidt, 1986 [24] has been a fundamental technique to estimate the direction of arrival (DOA) and number of distinct wavefronts in the received signal using m element radar(phase array).

A narrow band signal source s(t) for frequency ρ,angular frequency ω can be represented by:

$$s(t) = \rho e^{j\omega t} \tag{3.5}$$

Now, we can define m narrow band signals as:

$$s_1(t) = \rho_1 e^{j\omega t}, s_2(t) = \rho_2 e^{j\omega t}, ..., s_m(t) = \rho_m e^{j\omega t}, \tag{3.6}$$

The assumption here is that all frequencies are different and the amplitude/angle σ_i^2 uncorre-

lated.

$$E\{\rho_i \rho_j\} = \begin{cases} \sigma_i^2 & \rho_i <> \rho_j \\ 0 & \rho_i == \rho_j \end{cases} \tag{3.7}$$

Now, we model the sensor with m elements separated by distance d Fig. (3.2).

$$\triangle_i = \frac{(i-1)d\sin\theta}{c} \tag{3.8}$$

$$x_i(t) = e^{-j\omega\triangle_i} s(t) \tag{3.9}$$

$x_i(t)$ is an RX signal of a sensor i at an angle θ and delayed by $\triangle_i$.

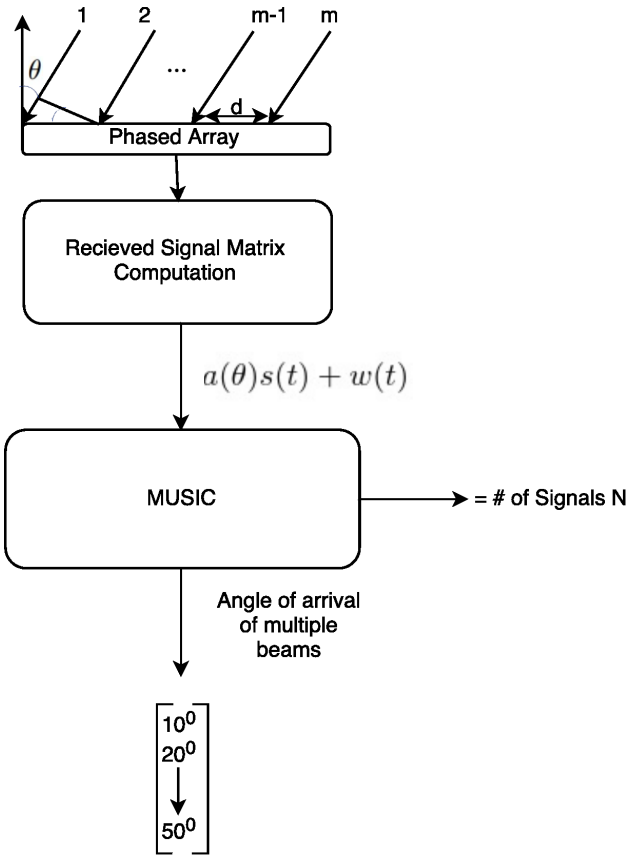

Figure 3.2: Block Diagram for MUSIC

Now putting all m sensor elements together eq (3.10).

$$\begin{bmatrix} X_1 \\ X_2 \\ \downarrow \\ X_m \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j\omega\triangle_1} \\ e^{-j\omega\triangle_2} \\ \downarrow \\ e^{-j\omega\triangle_m} \end{bmatrix} s(t) + \begin{bmatrix} W_1 \\ W_2 \\ \downarrow \\ W_m \end{bmatrix} = a(\theta)s(t) + w(t) \tag{3.10}$$

Where W is the complex vector, representing noise.

We get $a(\theta)$ the steering vector used to calculate the DOA using eq (3.8) and eq (3.9) for the i_{th} signal.

3.3 Multiple beams forming

Multiple beams forming as described in detail by Pfeffer et al., 2013[25] uses frequency-division multiple tx-beamforming (FDMB) and multiple-input multiple-output (MIMO) radar systems. It is based on the FMCW principle and provides multiple TX beams using phase array radar. The key component of their work is a vector modulator (VM) Fig. (3.3) in each TX path of the phased array, which is used as a single side-band (SSB) mixer and is controlled by digital-to-analog converters (DAC). This implementation provides us with some nice properties such as:

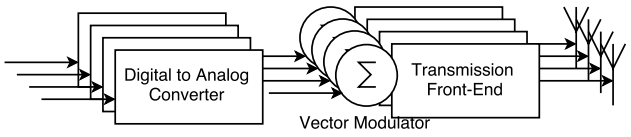

Figure 3.3: FDMB system with 4 TRX Front-End

$$k\tau = B_s/T_s \tag{3.11}$$

$$f_{RF} = f_0 + k\tau \tag{3.12}$$

- Simultaneous TX: Different beams are transmitted simultaneously at different frequencies f_{RF}.
- Per beam steer-ability: Steering a beam is achieved by changing the phase relations between the TX cells. In case of multiple beams, the complex control signals are calculated by simply summing up of the required control signals of each TX beam for each

VM.

- One more beneficial property is the use of input frequencies f_{RF} to each beam derived using the ramp slope $k\tau$ eq (3.11). The resultant beams are separated in frequency domain satisfying the

FMCW principle.

3.4 Car-Following Model

ACC used in vehicles is an example of a typical CPS, which uses radars as physical sensors that measure the distance to the lead car Fig. (3.4) and give feedback to a CF model-based computation system which maintains the desired state of the vehicle by manipulating throttle/acceleration. We will be using it for modeling STCR. ACC in vehicles uses FMCW based phase array radars, which are capable of multiple beam forming and tracking different objects using DOA estimation. This information is used as a feedback to a CF model which assumes a two-vehicle system consisting of a following and a lead vehicle. Lead vehicle's dynamics are independent of the following vehicle. The behavior of the following vehicle is adjusted as per distance, speed relative to the lead vehicle, reaction time of the driver and physics of the vehicle. The CF model works in two modes:

- Speed control: The following car travels at a predefined (cruise) speed.
- Spacing control: The following car maintains a safe distance from the lead car.

In CF model which mode to be used is decided based on the real-time distance to the lead car reported by the radar system, if the distance to lead car is less than safe distance Fig. (3.4) it switches to Spacing Control else it runs in Speed Control. And this is achieved by controlling throttle/acceleration of the following car.

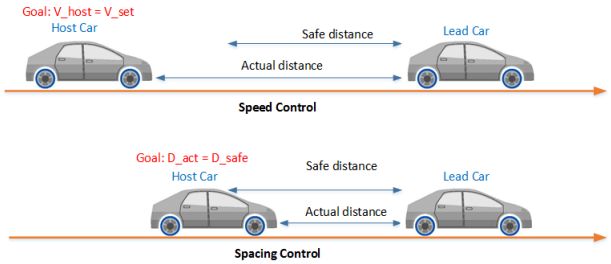

Figure 3.4: CF Model

We have used matlab's "Adaptive Cruise Control System Using Model Predictive Control" implementation of CF model.

Chapter 4

STCR: Spatio-Temporal Challenge-Response

The basic idea behind our approach is to have multiple points of authentication. This is achieved by splitting the radar signal into many narrowband beams. Each beam is capable of performing physical CRA. Then sending those beams at multiple angles towards the leading car (angle are randomly distributed bounded by the width of the lane). At every time interval, One of these beams will perform physical CRA. ensuring that rest of the beams stay active and the system does not stop sensing. This approach is further enhanced by bucketing these frequencies into autonomous groups which based on the quality of the response received for there individual beams provide a computed distance with a confidence level. Buckets with a confidence level less than a certain threshold are pruned and the rest are used to select the distance based on closest bond or mean. In STCR we employ the existing MIMO beam-forming techniques as mentioned in [25] to deploy a radar system with K independent beams separated in time, space and frequency. a total number of radar elements M, N number of elements per beam we get $K = M/N$. From now on we will be explaining the algorithm taking K independent beams.

STCR creates K independent beams using multiple beam forming:

- A $\lambda/2$ uniformly spaced linear array (uniform linear array (ULA)) with M the number of TX and RX antennas, where λ is the wavelength of the signal.
- Linearly increasing radio frequency (RF) signal frequency $f_{RF} = f_0 + k_r t$ where $k_r =$

B_s/T_s is the ramp slope with B_s th sweep bandwidth and T_s the sweep duration eq 3.11,3.12

- Steering this beacon is achieved by changing the phase relations between TX elements. Which is performed in baseband applying phase offsets to the complex control signals of each VM Fig.3.3.

- In case of multiple beams, the complex control signals are calculated by simply summing up the required control signals of each TX beam for each VM.

These beams are now grouped together in buckets B_i. We make sure that these frequencies are randomly selected. These buckets work independently of each other, forming autonomous sensor systems.

Below we will be defining 2 algorithms one implemented within buckets and other to perform data fusion over these buckets to provide a reliable reading.

Algorithm1: Attack detection within Buckets

- **Input:** 1) angle α bounded by the width of the lane w and minimum safe distance d.
  2) a vector $\overrightarrow{f}$ of frequencies.

- **Output:** 1) confidence value V_c initially set to $\| \overrightarrow{f} \|$
  2) measured distance d

- **Step 1** We can provide a vector $A_i \in R^{\| \overrightarrow{f} \|}$, of provided angle a

- **Step 2** Synchronized CRA: then we apply the physical CRA Algorithm, by using a pseudo-random generator to generate a vector $\overrightarrow{u} \in R^K$ such that it has all ones except one and take a dot product of it with A_i to get a vector A_t. This way we make sure that the system supports interleaving by not having more than one element of $\overrightarrow{u}$ as 0.

- **Step 3** We will be generating a reflection vector $A_r \in R^K$ and compare it with the actual received vector $A_a \in R^K$ to find the edit distance ed, and if $ed > \epsilon$ will raise the alarm that system is under attack.

- **Step 4** In case attack is detected in **Step 3** we will be reducing the confidence value V_c of the bucket by the number of susceptible frequencies.

- **Step 5** Also computes the distance and store it in d ignoring frequencies for which attack was detected or are in challenge period.

Algorithm2: Data fusion using confidence index

- **Input:** 1) a vector $\overrightarrow{d}$ of distances calculated by i_{th} angle bucket
  2) with a vector $\overrightarrow{V_c}$ indicating their confidence index.
- **Output:** measured distance D
- **Step 1** Prune the buckets based on $\overrightarrow{V_c} : V_{ci} < [somethreshold]$
- **Step 2** Use bound checking on remaining values in d_i and select the distance with the tightest bound and save it in D

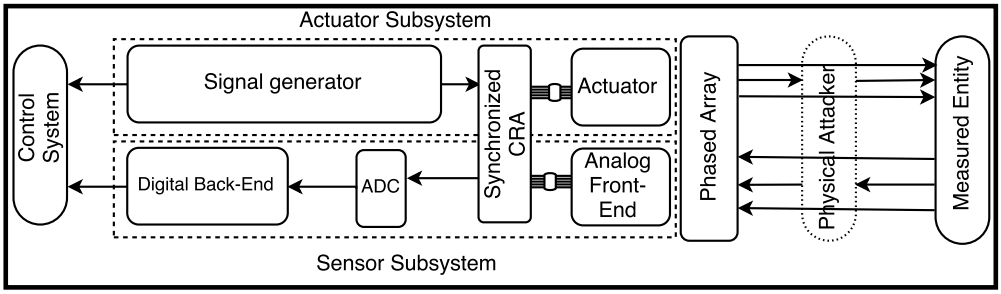

Figure 4.1: STCR model: A CPS system with a synchronized CRA module which generates pseudo random 0,1 modulation for M beams making sure only one beam per bucket will be in the challenge phase at any time τ.

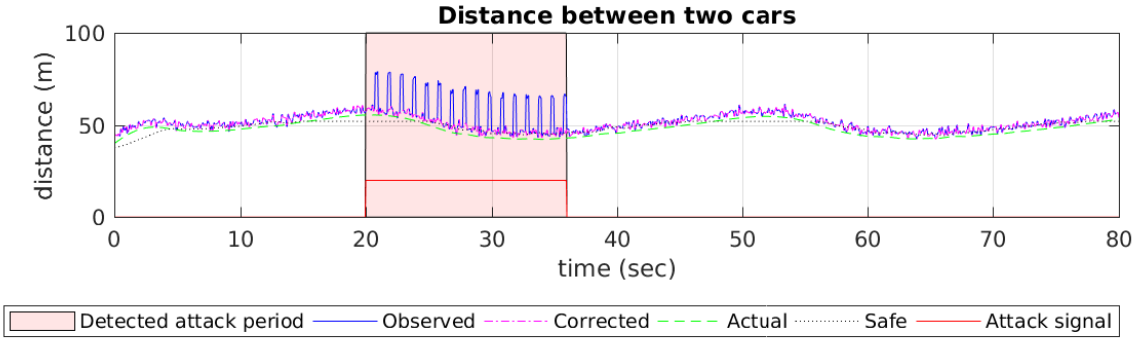

Figure 4.2: STCR algorithm in ACC output: Red line represents an attack vector and red region the period in which the algorithms detected the attack. The green line represents the ground truth and the blue distance calculated using received signal. The magenta line is the corrected reported distance. Black spaced line is the safe distance based on the current relative speeds between the two vehicles

To analyze this algorithm, we modeled it in a vehicle's ACC system Fig.4.1 with $\theta = 5$. We can observe the output $o(\tau)$(observed distance) of the active sensor in Fig.4.2, during absence

of attack when $u(\tau == 10) < \theta$ which is as expected. Now, at $\tau = 20$ $r(\tau) \approx 20$ which is $> \theta$, hence the system triggers an alarm. With STCR we were able to detect the attack early due to synchronized CRA mentioned in Algorithm1: Step2. The system was resilient to attack as the attack was curtailed to few buckets and using Algorithm 2 we were able to select the buckets with the highest confidence index and selected D with tightest bound.

Chapter 5

Performance Evaluation

We have modified "Adaptive Cruise Control System Using Model Predictive Control" [26] implementation in matlab to model our experiments. This simulation implements ACC, which uses a CF model having the following properties:
Inputs:

- Driver-set velocity V_{set}
- Velocity of the host car V_{host}
- Actual distance to the lead car D_{act} (from radar)
- Velocity of the lead car V_{lead} (from radar)

Outputs:

- Acceleration

The dynamics between acceleration and velocity are modeled as:

$$\frac{1}{s(0.5s+1)} \tag{5.1}$$

which approximates the dynamics of the throttle body and vehicle inertia. The same transfer function applies to both the host car and lead car.
The safe distance between the lead car and the host car is a function of the velocity of the host car, V_{host}:

$$D_{safe} = 10 + 1.4 \times V_{host} \tag{5.2}$$

where 10 (m) is the standstill distance and 1.4 (sec) is the time gap.
The following rules are used to determine the ACC system operating mode:

- If $D_{act} \geq D_{safe}$, then speed control mode is active. The control goal is to track the driver-set velocity, V_{set}.
- If $D_{act} < D_{safe}$, then spacing control mode is active. The control goal is to maintain the safe distance, D_{safe}.

Three major components of this design are lead car, sensed data and host car. We modified the sensed data component by adding an "attack and noise subsystem" to introduce noise and attack vectors into the sensor readings. Then we implemented three algorithms PyCRA, Dutta et al. and STCR for detecting and recovering from the attack in the sensed data component. We also modified the Host car component to create measurement check points.

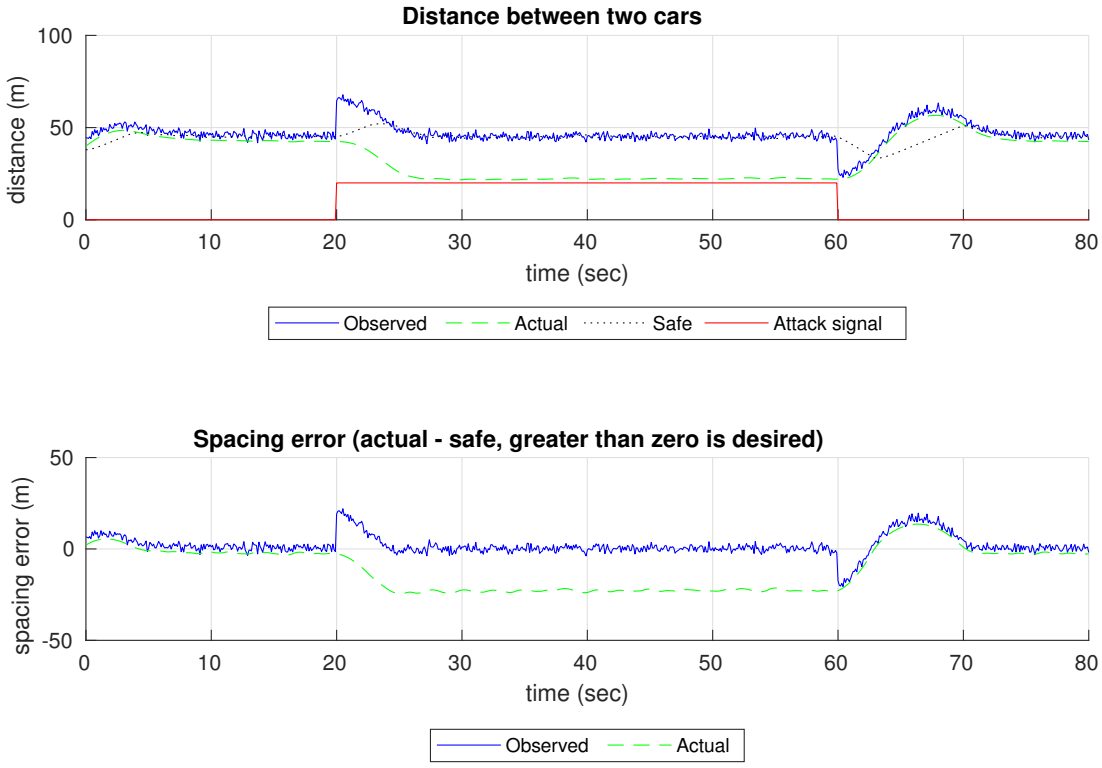

Figure 5.1: No detection and mitigation: Long duration static attack scenario

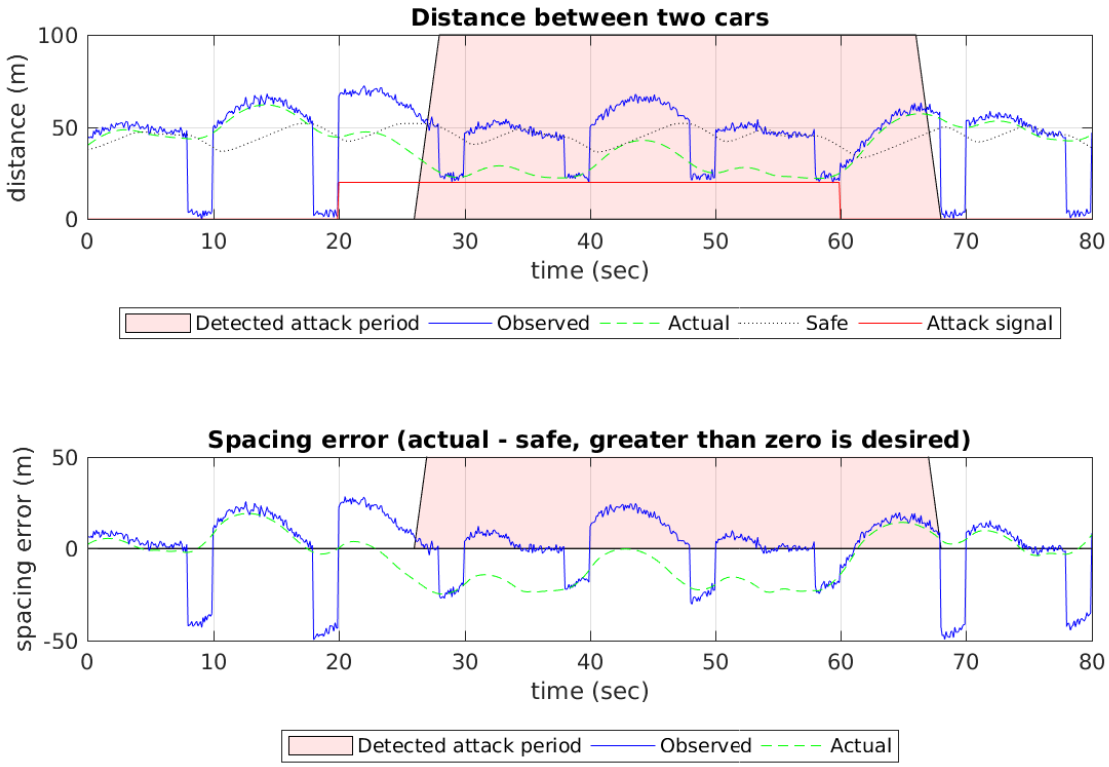

Figure 5.2: PyCRA: Long duration static attack scenario

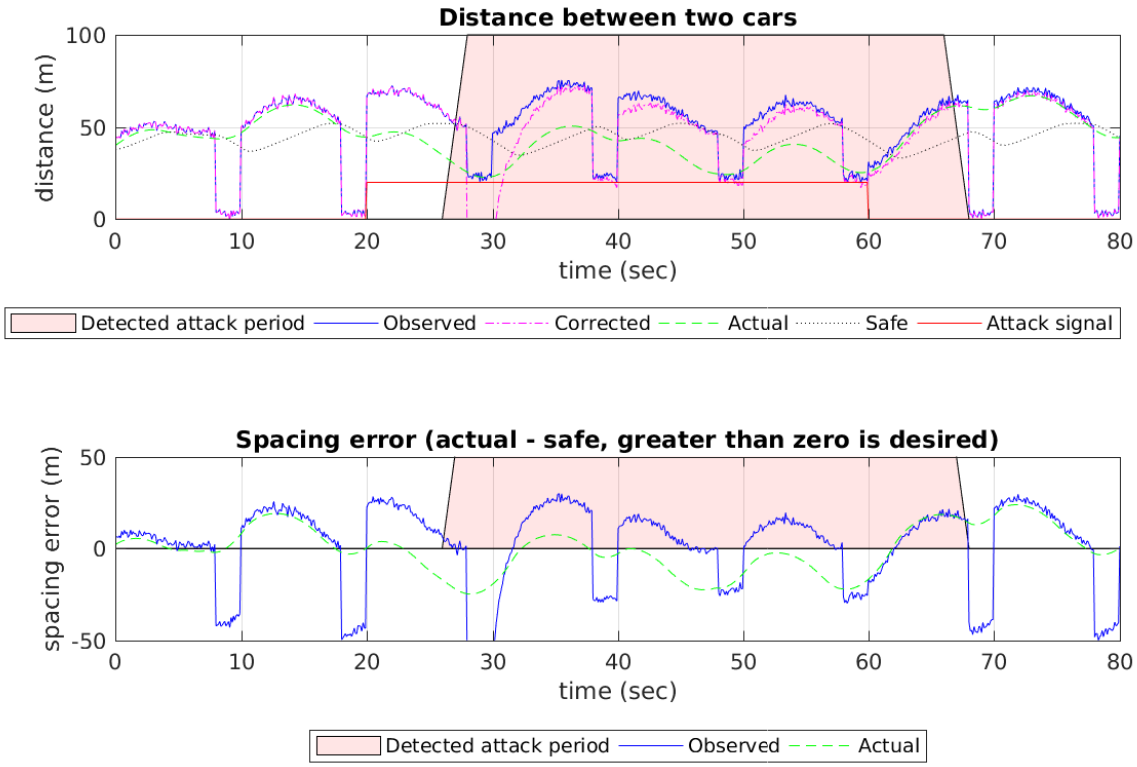

Figure 5.3: Dutta et al. 2017: Long duration static attack scenario

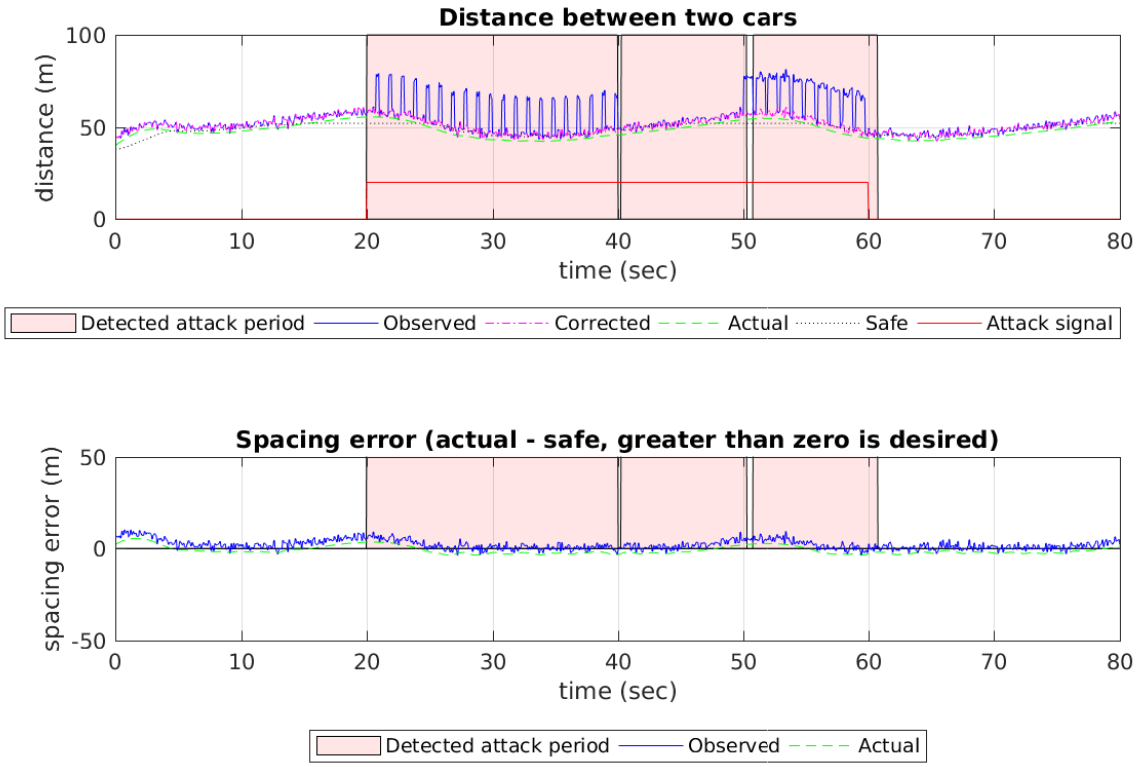

Figure 5.4: STCR: Long duration static attack scenario

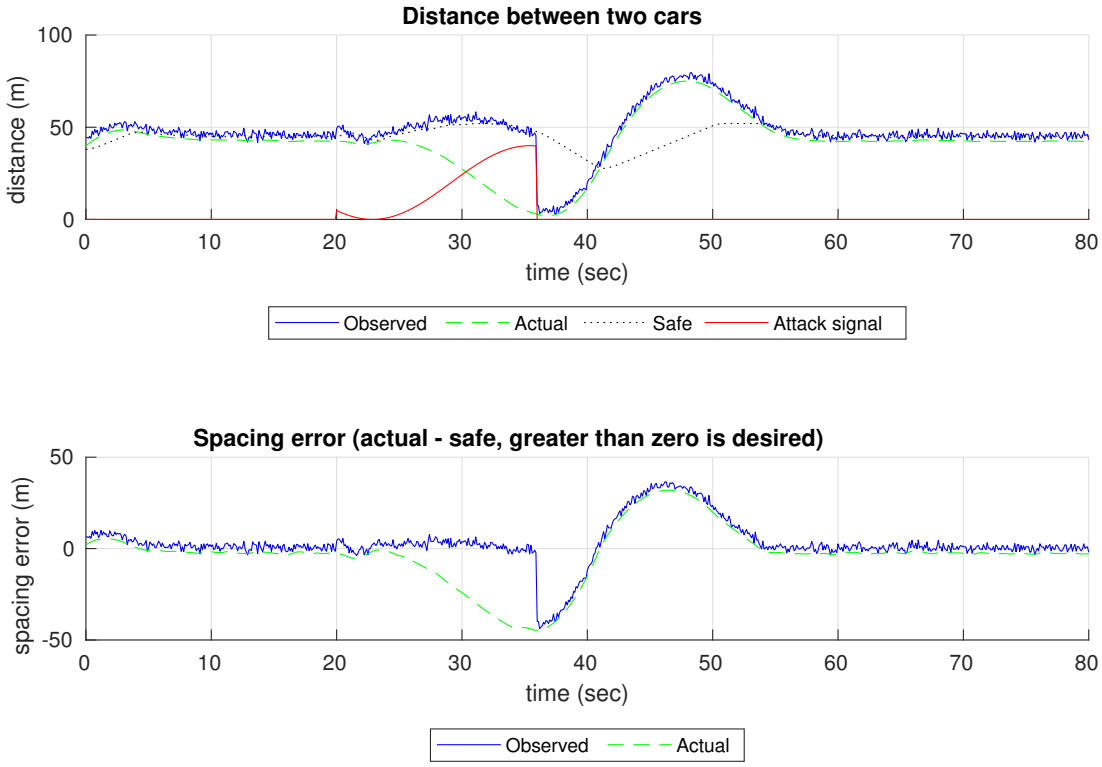

Figure 5.5: No detection and mitigation:Sinusoidal attack scenario

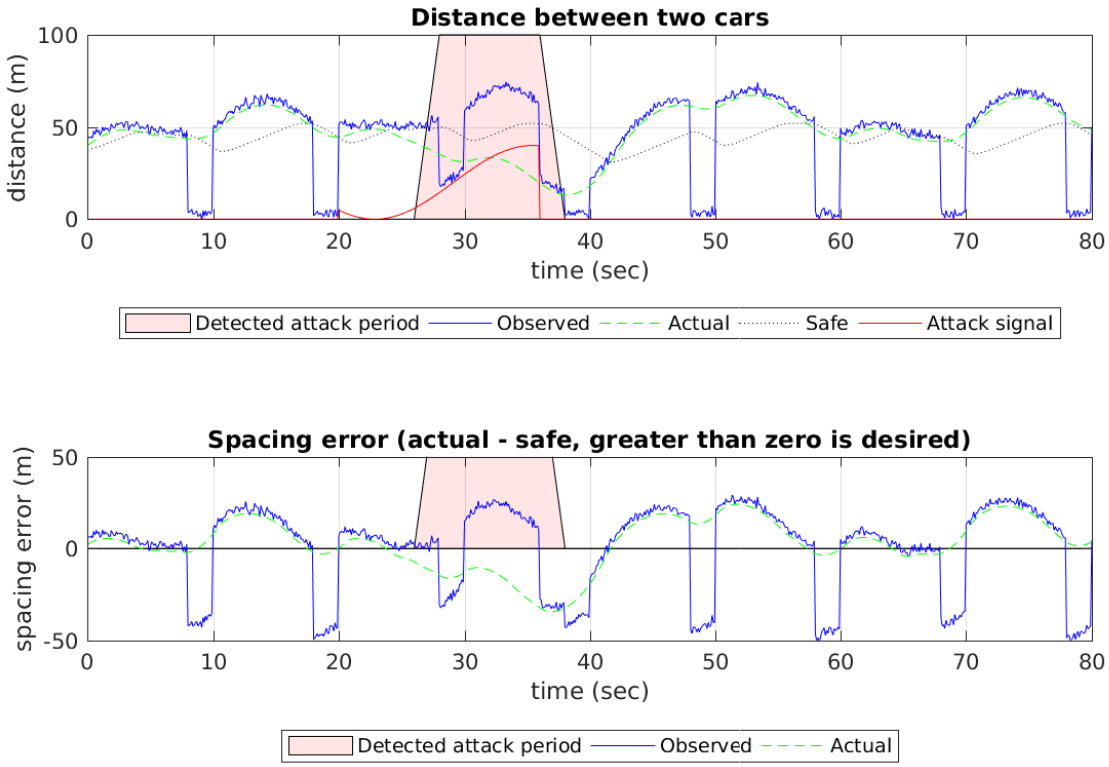

Figure 5.6: PyCRA:Sinusoidal attack scenario

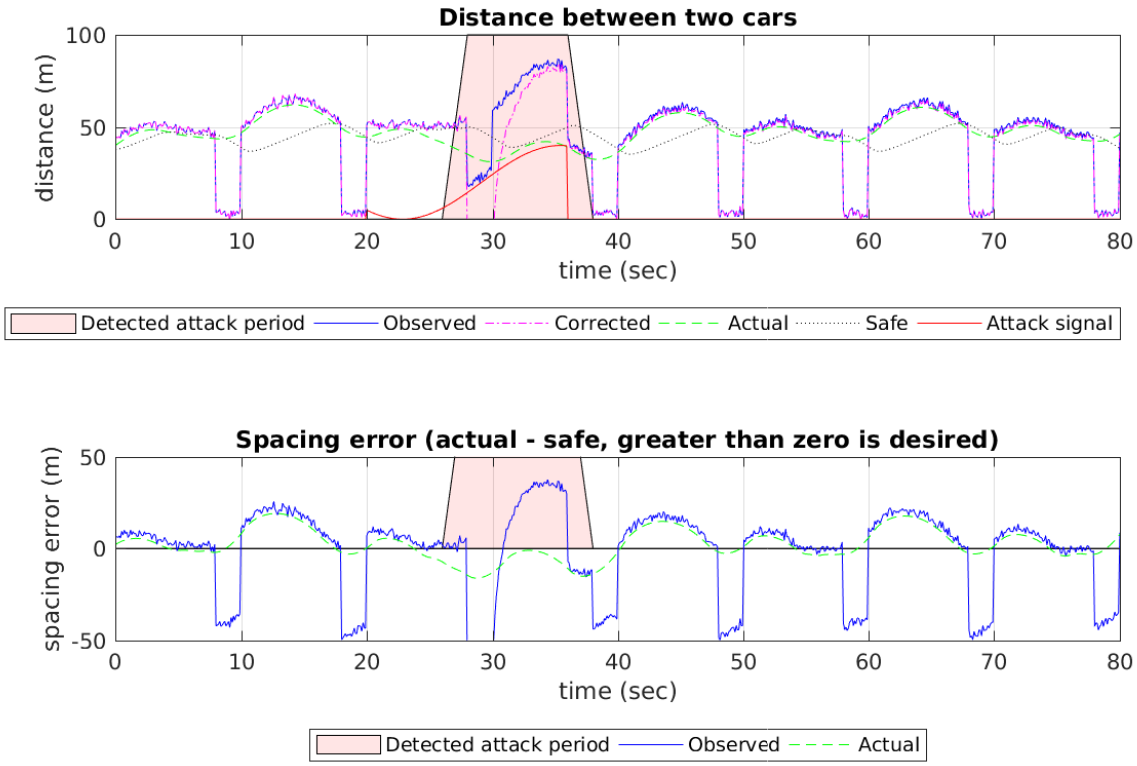

Figure 5.7: Dutta et al.: Sinusoidal attack scenario

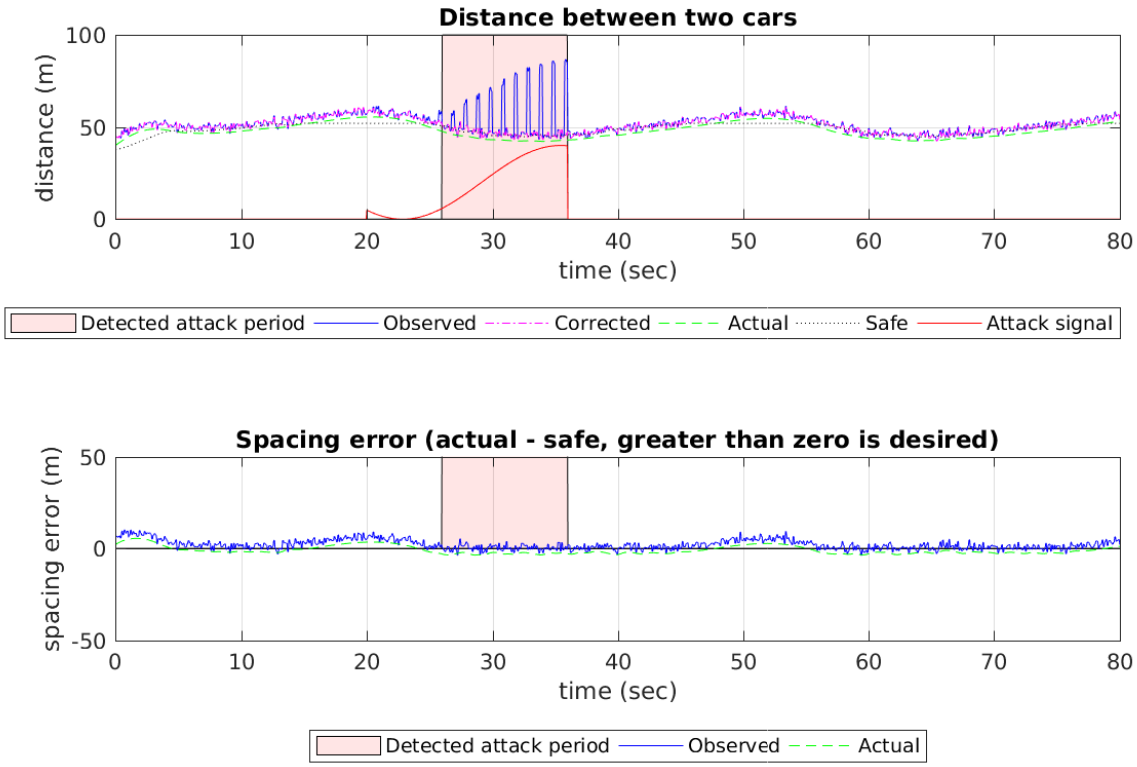

Figure 5.8: STCR: Sinusoidal attack scenario

To evaluate STCR we have taken 2 spoofing test scenarios: constant and sinusoidal. We have implemented and compared STCR against [11] which implements regression based recovery and have also taken as benchmark [10] implementation to compare results in case there is no recovery. A vanilla implementation is taken with no detection/recovery mechanism to show system behavior under attack when there is no recovery.

The charts from Fig.5.1 - 5.8 capture outcome of the results. It has 2 graphs each, where the top one is modeling the impact of attack/recovery on the vehicle dynamics of the following vehicle with respect to the lead vehicle and the bottom one represents the spacing error observed by the ACC system of the following car, which helps the vehicle to switch between speed control or spacing control modes.

The red solid line in the charts represents distance calculated via attack signal and red region the period in which algorithm detected the attack. The green '-' line represents the ground truth and blue solid line distance calculated using received signal. The magenta "-." line is

the corrected distance based on the signal reported by the algorithms. Black dotted line is the safe distance based on the current relative speeds between the two vehicles, the spacing error in bottom graph is calculated based on the difference of safe distance and actual/observed distance respectively.

As we can see from our experimental results in Fig. 5.3 and 5.7 regression based solutions tend to degrade as the attack goes on. We were able to attribute this behavior to two key factors. First, these algorithms rely on learning done during the challenge period, and we can not have this period too long as it would impact sensitivity of the system and can result in temporary loss of control. Secondly, due to the forgetting factor λ involved in algorithms such as RLS the learning done during the challenge period carries a much lower weight-age and the current attack signal has higher weight-age, hence longer the attack persists we observe that the recovery algorithm starts converging to the attack signal.

We came up with two criteria for comparing the different approaches.

First is sensitivity S_{algo} of the algorithm used, which is given by:

$$S_{algo} = \frac{1}{(A_e - A_d)} \sum_{1}^{n} (D_{ei} - D_{si}) \tag{5.3}$$

$$D_{ei} = \begin{cases} D_{ei} & D_{ei} < A_e \\ A_e & D_{ei} >= A_e \end{cases} \quad D_{si} = \begin{cases} D_{si} & D_{si} > A_s \\ A_s & D_{si} <= A_s \end{cases} \tag{5.4}$$

In all cases, the attack starts at time $A_s = 20$ and ends at $A_e = 35$ for short attacks and $A_e = 60$ and the red regions in these graphs are the detection periods D. We found that STCR outperformed the regression-based algorithms in all cases

Algo	Short Attack	Long attack	Sinusoidal attack
STCR	1	0.9575	0.625
Dutta et al.	0.625	0.85	0.625
PyCRA	0.625	0.85	0.625

Table 5.1: Sensitivity Comparison: This table represents the Sensitivity of the system given by eq. 5.3

Second is accuracy, we used continuous root mean square error (CRMSE) to calculate Ac_{algo} which is given by:

$$Ac_{algo} = \sqrt{\frac{1}{T} \int_0^T (s_{err}^2) dt} \tag{5.5}$$

After comparing different algorithms we found that regression-based algorithms have a really high base root mean square error (RMSE) due to there long learning phase, even after normalizing for this we found that STCR performed better in all cases to regression-based approaches.

Algo	Base error	Short Attack	Long attack	Sin attack
STCR	3.96	4.44	5.27	4.44
Dutta et al.	19.35	21.29	21.18	21.29
PyCRA	18.97	23.03	22.16	21.15
Vanilla	4.05	10.77	16.48	11.81
Algo	Base error normalised	Short Attack	Long attack	Sin attack
STCR	0.0	0.48	1.31	0.48
Dutta et al.	0.0	1.94	1.83	1.94
PyCRA	0.0	4.06	3.19	2.18
Vanilla	0.0	6.72	12.43	7.76

Table 5.2: RMSE comparison: This table represents the Accuracy of the system in meters given by eq. 5.5

Chapter 6

Conclusions and Future Work

We have presented STCR a spatial and temporal challenged response based spoofing attack mitigation method for active sensors. One of the key features of STCR is that it uses currently available radar technologies (FMCW + phased array) deployed in vehicles and improves detection and resilience of the system by grouping a random set of frequencies in a bucket, and then treating them as different sources as they are both distinct in space (angle) and frequency, allowing us to create a framework based on belief representation and hence achieve benefits of sensor fusion. In our approach buckets are interleaved together in time domain, hence avoiding temporary loss of control which other solutions suffer from. Then we simulated and compared our approach with the current state of the art using Matlab. We performed several experiments and showed that our approach was more sensitive to attacks and had 6 times better accuracy than the state of the art. However, our approach will fail if the attacker is able to compromise all the sensor buckets, and is also very specific to sensors which support multiple beam-forming. Our future research will try to address these limitations and we will try to extend this to enhance Sensor Fusion based schemes by increasing the confidence level in radar based techniques.

Chapter 7

References

[1] Alvaro A Cárdenas, Saurabh Amin, and Shankar Sastry. "Research Challenges for the Security of Control Systems." In: *HotSec*. 2008.

[2] Jairo Giraldo, Esha Sarkar, Alvaro Cardenas, et al. "Security and Privacy in Cyber-Physical Systems: A Survey of Surveys". In: *IEEE Design & Test* (2017).

[3] Ragunathan Raj Rajkumar, Insup Lee, Lui Sha, et al. "Cyber-physical systems: the next computing revolution". In: *Proceedings of the 47th Design Automation Conference*. ACM. 2010, pp. 731–736.

[4] António Lima, Francisco Rocha, Marcus Völp, et al. "Towards Safe and Secure Autonomous and Cooperative Vehicle Ecosystems". In: *Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy*. ACM. 2016, pp. 59–70.

[5] Robert Mitchell and Ing-Ray Chen. "A survey of intrusion detection techniques for cyber-physical systems". In: *ACM Computing Surveys (CSUR)* 46.4 (2014), p. 55.

[6] Song Han, Miao Xie, Hsiao-Hwa Chen, et al. "Intrusion detection in cyber-physical systems: Techniques and challenges". In: *IEEE Systems Journal* 8.4 (2014), pp. 1052–1062.

[7] Saif Al-Sultan, Moath M Al-Doori, Ali H Al-Bayatti, et al. "A comprehensive survey on vehicular ad hoc network". In: *Journal of network and computer applications* 37 (2014), pp. 380–392.

[8] Renju Liu and Mani Srivastava. "PROTC: PROTeCting Drone's Peripherals through ARM TrustZone". In: *Proceedings of the 3rd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*. ACM. 2017, pp. 1–6.

[9] Pinyao Guo, Hunmin Kim, Nurali Virani, et al. "Exploiting Physical Dynamics to Detect Actuator and Sensor Attacks in Mobile Robots". In: *arXiv preprint arXiv:1708.01834* (2017).

[10] Yasser Shoukry, Paul Martin, Yair Yona, et al. "PyCRA: Physical challenge-response authentication for active sensors under spoofing attacks". In: *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM. 2015, pp. 1004–1015.

[11] Raj Gautam Dutta, Xiaolong Guo, Teng Zhang, et al. "Estimation of Safe Sensor Measurements of Autonomous System Under Attack". In: *Proceedings of the 54th Annual Design Automation Conference 2017*. ACM. 2017, p. 46.

[12] Bon-Hyun Ku, Paul Schmalenberg, Ozgur Inac, et al. "A 77–81-GHz 16-Element Phased-Array Receiver". In: *IEEE Transactions on Microwave Theory and Techniques* 62.11 (2014), pp. 2823–2832.

[13] Edward Ashford Lee and Sanjit A Seshia. *Introduction to embedded systems: A cyber-physical systems approach*. MIT Press, 2016.

[14] Yasser Shoukry, Paul Martin, Paulo Tabuada, et al. "Non-invasive spoofing attacks for anti-lock braking systems". In: *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer. 2013, pp. 55–72.

[15] Denis Foo Kune, John Backes, Shane S Clark, et al. "Ghost talk: Mitigating EMI signal injection attacks against analog sensors". In: *Security and Privacy (SP), 2013 IEEE Symposium on*. IEEE. 2013, pp. 145–159.

[16] Chen Yan, Wenyuan Xu, and Jianhao Liu. "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle". In: *DEF CON* 24 (2016).

[17] Charlie Miller and Chris Valasek. "Remote exploitation of an unaltered passenger vehicle". In: *Black Hat USA* 2015 (2015).

[18] Tencent Cohen laboratory. *Tesla's physical contact without long-range attack*. 2016. URL: http://keenlab.tencent.com/zh/2016/09/19/Keen-Security-Lab-of-Tencent-Car-Hacking-Research-Remote-Attack-to-Tesla-Cars/ (visited on 09/19/2016).

[19] Eweda Eweda and Odile Macchi. "Convergence of the RLS and LMS adaptive filters". In: *IEEE Transactions on Circuits and Systems* 34.7 (1987), pp. 799–803.

[20] Miguel Arroyo, Hidenori Kobayashi, Simha Sethumadhavan, et al. "FIRED: Frequent Inertial Resets with Diversification for Emerging Commodity Cyber-Physical Systems". In: *arXiv preprint arXiv:1702.06595* (2017).

[21] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. "Secure estimation and control for cyber-physical systems under adversarial attacks". In: *IEEE Transactions on Automatic Control* 59.6 (2014), pp. 1454–1467.

[22] Donggang Liu, Peng Ning, and Wenliang Kevin Du. "Attack-resistant location estimation in sensor networks". In: *Proceedings of the 4th international symposium on Information processing in sensor networks*. IEEE Press. 2005, p. 13.

[23] Fabio Pasqualetti, Florian Dörfler, and Francesco Bullo. "Attack detection and identification in cyber-physical systems". In: *IEEE Transactions on Automatic Control* 58.11 (2013), pp. 2715–2729.

[24] Ralph Schmidt. "Multiple emitter location and signal parameter estimation". In: *IEEE transactions on antennas and propagation* 34.3 (1986), pp. 276–280.

[25] Clemens Pfeffer, Reinhard Feger, Christoph Wagner, et al. "FMCW MIMO radar system for frequency-division multiple TX-beamforming". In: *IEEE Transactions on Microwave Theory and Techniques* 61.12 (2013), pp. 4262–4274.

[26] MathWorks. *Adaptive Cruise Control System Using Model Predictive Control*. 2017. URL: https://www.mathworks.com/help/mpc/examples/design-an-adaptive-cruise-control-system-using-model-predictive-control.html (visited on 10/11/2017).