Foundations of Cyber Physical Systems

Prof. Soumyajit Dey

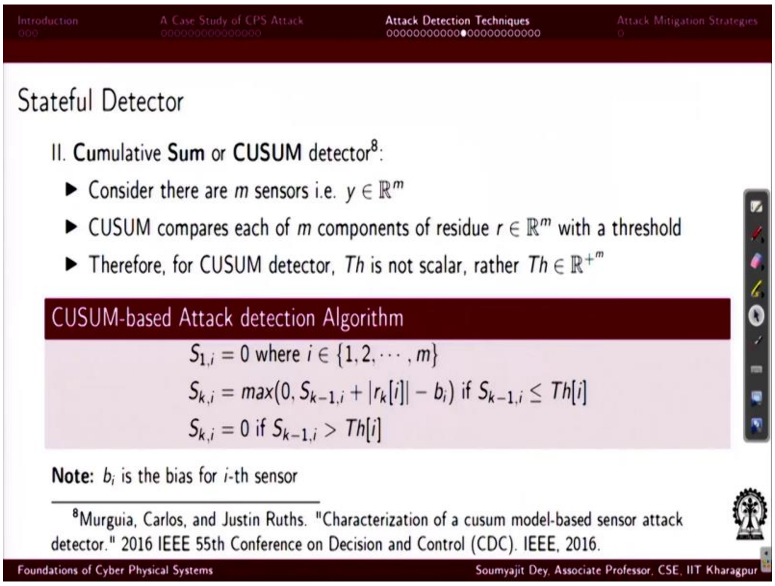
Department of Computer Science and Engineering

Indian Institute of Technology – Kharagpur

Lecture – 56
Attack Detection and Mitigation in CPS (Continued)

Hello and welcome back to this lecture series on Foundations of Cyber Physical Systems. So, in the previous lecture we have been talking about different kinds of detectors.

(Refer Slide Time: 00:36)


Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Stateful Detector

II. **Cumulative Sum** or **CUSUM** detector[8]:

- Consider there are m sensors i.e. $y \in \mathbb{R}^m$
- CUSUM compares each of m components of residue $r \in \mathbb{R}^m$ with a threshold
- Therefore, for CUSUM detector, Th is not scalar, rather $Th \in \mathbb{R}^{+^m}$

CUSUM-based Attack detection Algorithm

$$S_{1,i} = 0 \text{ where } i \in \{1, 2, \cdots, m\}$$
$$S_{k,i} = max(0, S_{k-1,i} + |r_k[i]| - b_i) \text{ if } S_{k-1,i} \leq Th[i]$$
$$S_{k,i} = 0 \text{ if } S_{k-1,i} > Th[i]$$

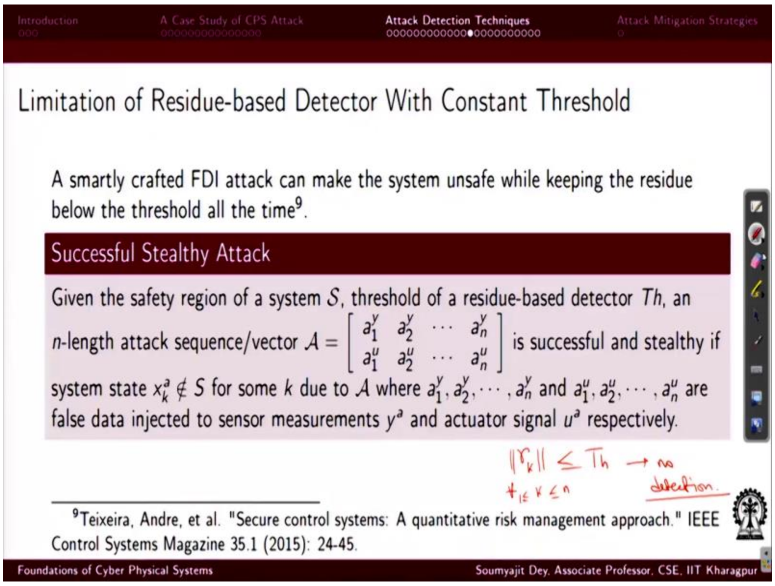
Note: b_i is the bias for i-th sensor

[8] Murguia, Carlos, and Justin Ruths. "Characterization of a cusum model-based sensor attack detector." 2016 IEEE 55th Conference on Decision and Control (CDC). IEEE, 2016.

Foundations of Cyber Physical Systems   Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

And we introduce stateless and stateful detectors like chi square windows chi square simple threshold-based detectors and also the CUSUM detector.

(Refer Slide Time: 00:43)


Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Limitation of Residue-based Detector With Constant Threshold

A smartly crafted FDI attack can make the system unsafe while keeping the residue below the threshold all the time[9].

Successful Stealthy Attack

Given the safety region of a system $\mathcal{S}$, threshold of a residue-based detector Th, an n-length attack sequence/vector $\mathcal{A} = \begin{bmatrix} a_1^y & a_2^y & \cdots & a_n^y \\ a_1^u & a_2^u & \cdots & a_n^u \end{bmatrix}$ is successful and stealthy if system state $x_k^a \notin \mathcal{S}$ for some k due to $\mathcal{A}$ where $a_1^y, a_2^y, \cdots, a_n^y$ and $a_1^u, a_2^u, \cdots, a_n^u$ are false data injected to sensor measurements y^a and actuator signal u^a respectively.

$\|r_k\| \le Th \rightarrow$ no detection.

$\forall_{1 \le k \le n}$

[9] Teixeira, Andre, et al. "Secure control systems: A quantitative risk management approach." IEEE Control Systems Magazine 35.1 (2015): 24-45.

Foundations of Cyber Physical Systems   Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

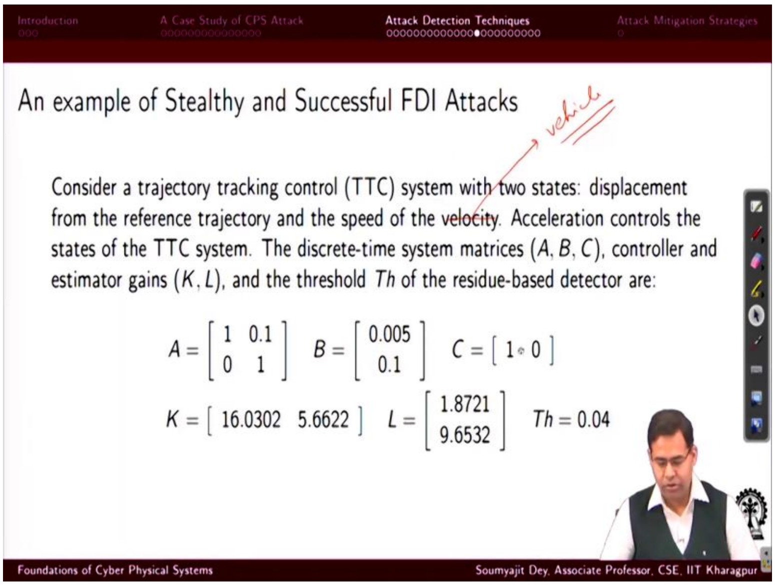
So, ah the we will like to sum sum over that ah with this point that there are some limitations that exist with respect to this residue-based detectors with constant threshold. For example, a smartly crafted FDI attack can make a system unsafe, while also keeping the residue below the threshold. That means the attacker knows the system model or has an estimate of the system model.

And what the attacker is doing is they are injecting just enough impurity, ah so that the residues never do cross. ok So, this raises this issue of stealthy attack which have been highlighted in this cited paper here. So, stealthy attack, what it will do is suppose you are given a safety region of a system S. And there is a threshold of a detector system Th and there is an n length attack sequence.

So, let us say, ah in the step 1 you have an attack amount a_1^y on the on the measurement and an attack amount a_1^u on the on the controller. And you keep on continue the continuing this attack in this sequence, on both the measurements and the controllers Ok for n number of steps. And this can be successful and stealthy if at some point for at some k, kth step the system state ah under attack that is x_k^a is no more inside this safety set S.

And however, due to all these attacks that you did ah what happened is the residue that was measured in all these steps was inside the threshold. Then ah I would really call it that Ah this this is stealthy attack. So that is the additional point which one would have to act Um that while this is happening, this r_k that you are computing ah for all. So, this is less than the threshold, so, there is no detection.

(Refer Slide Time: 02:56)


Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies
vehicle
Foundations of Cyber Physical Systems
Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

An example of Stealthy and Successful FDI Attacks

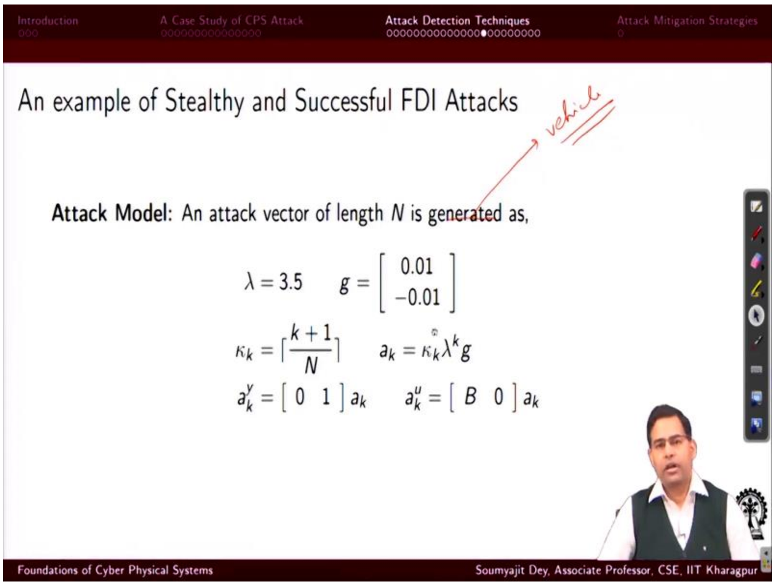
Consider a trajectory tracking control (TTC) system with two states: displacement from the reference trajectory and the speed of the velocity. Acceleration controls the states of the TTC system. The discrete-time system matrices (A, B, C), controller and estimator gains (K, L), and the threshold Th of the residue-based detector are:

$$A = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

$$K = \begin{bmatrix} 16.0302 & 5.6622 \end{bmatrix} \quad L = \begin{bmatrix} 1.8721 \\ 9.6532 \end{bmatrix} \quad Th = 0.04$$

So, we will, we will conclude this part with an example of stealthy and successful FDI attacks. So, let us consider again the trajectory contracting control system. And there are two states displacement from the reference trajectory and the speed of velocity and ah and and the speed of the, let us say the vehicle. So, this is, of course, not velocity. um So, you have acceleration controls, ah over this Ah TTC that means you can using suitable control input you can update the acceleration command of the system. Ok And let us say these are your discrete matrices, A B C and this is your controller gain Ah and this is your estimator gain. And you have set a threshold of Th equal to 0.04 a very small threshold for the residue.

(Refer Slide Time: 03:48)


Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies
vehicle
Foundations of Cyber Physical Systems
Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

An example of Stealthy and Successful FDI Attacks

Attack Model: An attack vector of length N is generated as,

$$\lambda = 3.5 \quad g = \begin{bmatrix} 0.01 \\ -0.01 \end{bmatrix}$$

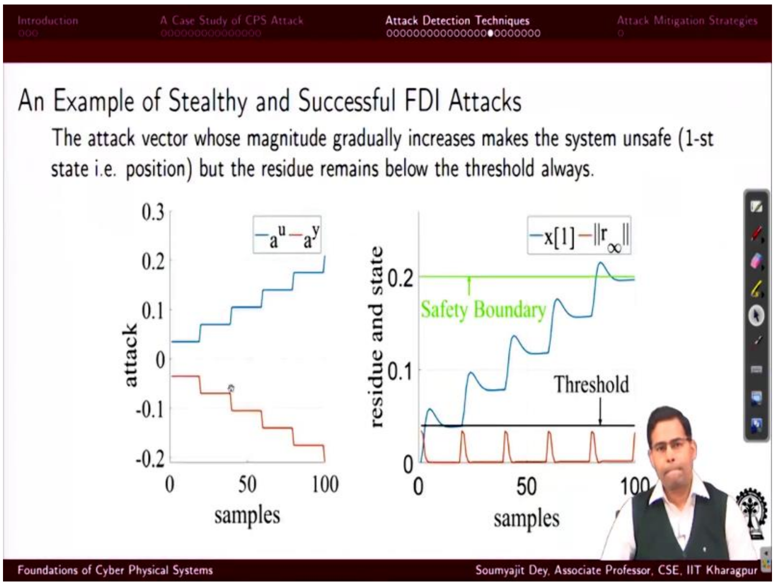
$$\kappa_k = \lceil \frac{k+1}{N} \rceil \quad a_k = \kappa_k \lambda^k g$$

$$a_k^y = \begin{bmatrix} 0 & 1 \end{bmatrix} a_k \quad a_k^u = \begin{bmatrix} B & 0 \end{bmatrix} a_k$$

And let us say you are generating attack vectors using this following model. So, you have some parameters lambda g and this kappa. OK and using So, I mean so, the parameters values Ah so, lambda is a constant g is a constant and this kappa value is changing with this k. right And an attack you are and you have an attack vector of length n. ok So, let us say n equal to 10 and you are running k equal to 1 to 10, so, n is 10, k is 1 and 10.

So, all this for for this different parameter values of k, you will get different values of kappa k and accordingly, you will generate different values of a k. right And then using this a k you can actually create you are creating two different attack vectors one for y because y is ah I mean ah a sensor reading of size 2. And let us say the control input that goes is also of size 2. ok So, all we are using this kind of a generator mechanism through which we are introducing this much of attack exactly to the y variable and the u variable.

(Refer Slide Time: 04:57)



Introduction

A Case Study of CPS Attack

Attack Detection Techniques

Attack Mitigation Strategies

An Example of Stealthy and Successful FDI Attacks

The attack vector whose magnitude gradually increases makes the system unsafe (1-st state i.e. position) but the residue remains below the threshold always.

Foundations of Cyber Physical Systems

Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

So that is what we are doing here. So, let us understand why we did it like this because of course, for generating the attack in an automated manner. ah we would need ah this kind of ah mechanisms because the so because the attack will be inserted in a programmatic manner. So, we will like to give this kind of a relation or something like that through which the attack values can be generated automatically in the runtime. Right

So, we are we have chosen this kind of a pattern and using this formula, the attack vectors are getting generated. Now, ah of course, you see that there is some, there is some way in which the values are chosen. So, what will happen is within with each update in k ah this numerator will increase, so that means this coefficient will increase. That means this coefficient is kind of controlling slowly that how the how the multiplication factor of the attack vector is increasing slowly, slowly.

Not only that you also have k as a power on lambda. So, essentially this This is also like an exponential graph increase that is happening. So, you are incrementing two parameters over multiple steps in a nice automated way, using a control over this increasing coefficient, monotonically increasing coefficient. And also a control over this monotonically increasing ah power power value of this of this constant lambda. Right

So that is why, ah the attack value you see ah will slowly have an have an increase right a step, step-based approximation of an exponential curve here. right So that is how your attack value of y will increase. So, here we are typically showing ah although y is multi carrier there are two points. Here, we are plotting, ah the the corresponding norm here. And similarly, you have an attack value of these. ah ok sorry ah

For that vector we are actually making the system unsafe and we are just plotting for the first state ah that is the position vector Ah so that is not a norm. You are just making the attack on the position vector. So, just as a recap, this is your TTC system we are talking about. So, there are two states here. One is the position that is the displacement from the reference trajectory and also the other, is the speed of the velocity, the speed of this vehicle. Right

So, what we are really attacking is kind of ah the the position value that is really there. We are modifying the position, value with larger ah deviation slowly and the deviation values are increasing slowly. Not at a step I am increasing the attack value by a large deviation but I am increasing it slowly. And I am not only doing that in parallel I am also changing the control input value slowly ah with this kind of deviations.

The deviation values are changing like this. Ok Now, it can be shown that if these attack steps are chosen, what will happen is ah that for the system you see, ah the residue is always, so, these are the residue values. This is the threshold that has been chosen. And this is the safety boundary of the separation that is the displacement ah or the position variable. Ok And you can see that that variables value ah is slowly increasing at each step of control.

And eventually it is going beyond the safety boundary here. But nobody will detect this attack because at each point the residues that are getting computed. All these residues are less than the threshold slow. ah what will happen is that the trajectory is control system it is separation from the trajectory will slowly diverge without the residue base detector. If this kind of a simple detector even detecting it.

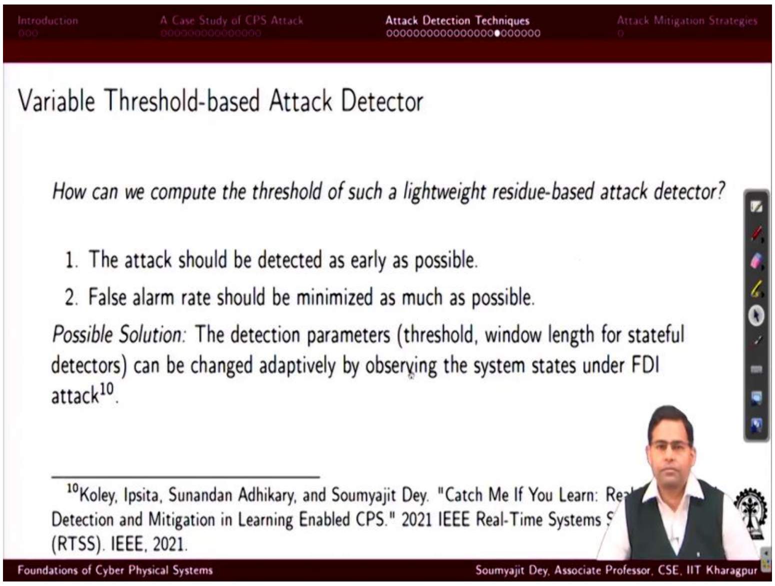
And this is precisely happening because I am deviating both the ah the informations. That is the control input, as well as the as as well as the state information both of them. So ah that is an example of a stealthy and successful FDI attack. So now, one may be wondering that well what does it mean by modifying the state information. See the state information is is kind of fed back to the controller.

So, ah so when the state information is going to the controller, I am changing, its measurement. So, changing the measurement is not really changing the actual state. You need to be careful here. So, when the plant dynamics is changing, it is changing based on the actual state and the plant dynamics is only affected by the modifications in the control input. But the when the control input is getting computed it is what is happening due to this perturbations in a^y is, the control input calculation process is not aware of the original plant dynamics but rather it is getting the original plant dynamics plus this a y attack. Right So, essentially it is using a wrong ah value of state variable to compute the control input and due to that this state variable value is going beyond the safety boundary. So, essentially here you see that the first state that is the position ah that that is what you are able to cross here and when you are doing the attack here.

You are changing Ah this information here. ah sorry I think, ah I told this part wrong. Ah When you are doing the attack here, you are actually here you have a 0 and here you have a_1. So that means you are doing the attack on the second information ah which is the the velocity of the vehicle, the velocity information of the vehicle. So, you are doing the attack on the velocity information of the vehicle and due to that you are computing anyway, ah wrong control input.

And you are further making some attack on the control input using this u k. right And in effect due to both of this what you are getting is ah computed threshold which is below this ah a computer residue which is below this threshold and you are computing, this value of ah state. Ah Sorry, you are, I mean this is the observed state of the system and this observed state is going at some point. It is going beyond the safety boundary that we have said of let us say 0.2 or something.

(Refer Slide Time: 11:44)


Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Variable Threshold-based Attack Detector

How can we compute the threshold of such a lightweight residue-based attack detector?

1. The attack should be detected as early as possible.
2. False alarm rate should be minimized as much as possible.

Possible Solution: The detection parameters (threshold, window length for stateful detectors) can be changed adaptively by observing the system states under FDI attack[10].

[10] Koley, Ipsita, Sunandan Adhikary, and Soumyajit Dey. "Catch Me If You Learn: Re Detection and Mitigation in Learning Enabled CPS." 2021 IEEE Real-Time Systems S (RTSS). IEEE, 2021.

Foundations of Cyber Physical Systems
Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

So, ah we can understand that this is a really an issue that if we keep the threshold value as fixed then it is easy to bypass. right Now, the question is what can be done. So, of course, there are several things we can do. One example is one idea is that well ah the attack should be detected as early as possible. Because if you can detect the attack ah earlier then you can have suitable counter measures in place.

For example, for that point of time you can have encryption on. right You can have encryption one with less number of samples because encryption will increase the bandwidth consumption, so, you decrease the periodicity of the loop. That means you run your control less frequently. So that would mean you are compromising ah with the quality of control. But still you have a guarantee that well whatever I am controlling that packet that is a good packet.

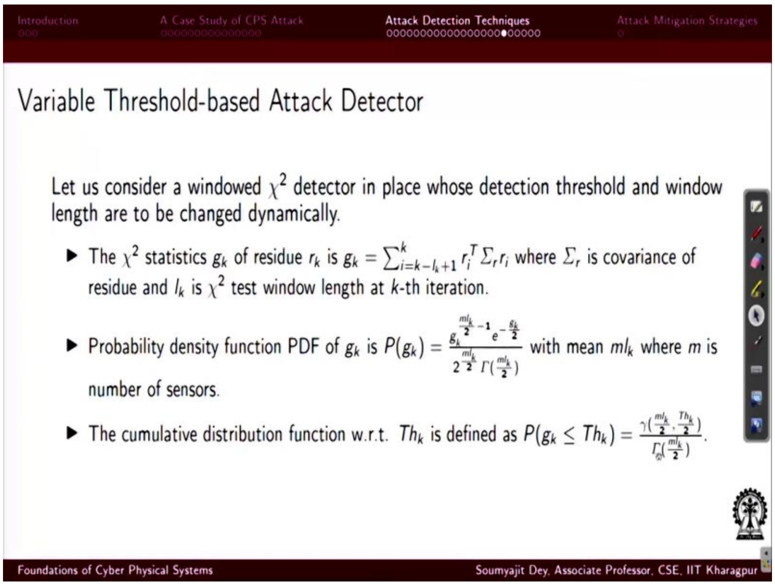
So, you have a trustworthy control but at the expense of performance. ok So that would be an option that I am sensing attacks are happening. Let us not forget about quality of control. Let us just ensure that the system works without being unsafe. ok But at the same time, ah the detectors cannot be too sensitive because if I if I use this as an excuse to reduce my threshold then what will happen is there may be standard ambient noises in the system.

And due to which the measurement errors occurred, there was no attacker really and that would generate a false alarm. right So, what can be done is instead of using a fixed threshold detector one can choose to change these detectors detection threshold over a window length. That means one can choose to adaptively change this reduction threshold by observing the system state under FDI attack.

So, you keep on observing the system state and it is, and it is and it is trajectory whenever attacks are happening. And based on those observations you you keep on adjusting the threshold says threshold. Because you saw that right now my system is in a sensitive zone. I will like the threshold to be small. Right now, my sensor threshold system is in a zone where it may it is much much far from going to the unsafe situation.

I can increase my threshold. So, using such situation observations can I create an AI based or some other technique which will decide when to select which threshold value. So that I have a system where the false alarm rate is not too high and still have a guarantee that my system will never be answered.

(Refer Slide Time: 14:16)



Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Variable Threshold-based Attack Detector

Let us consider a windowed χ^2 detector in place whose detection threshold and window length are to be changed dynamically.

- The χ^2 statistics g_k of residue r_k is $g_k = \sum_{i=k-l_k+1}^{k} r_i^T \Sigma_r r_i$ where Σ_r is covariance of residue and l_k is χ^2 test window length at k-th iteration.
- Probability density function PDF of g_k is $P(g_k) = \frac{g_k^{\frac{ml_k}{2}-1} e^{-\frac{g_k}{2}}}{2^{\frac{ml_k}{2}} \Gamma(\frac{ml_k}{2})}$ with mean ml_k where m is number of sensors.
- The cumulative distribution function w.r.t. Th_k is defined as $P(g_k \leq Th_k) = \frac{\gamma(\frac{ml_k}{2}, \frac{Th_k}{2})}{\Gamma(\frac{ml_k}{2})}$.

Foundations of Cyber Physical Systems
Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

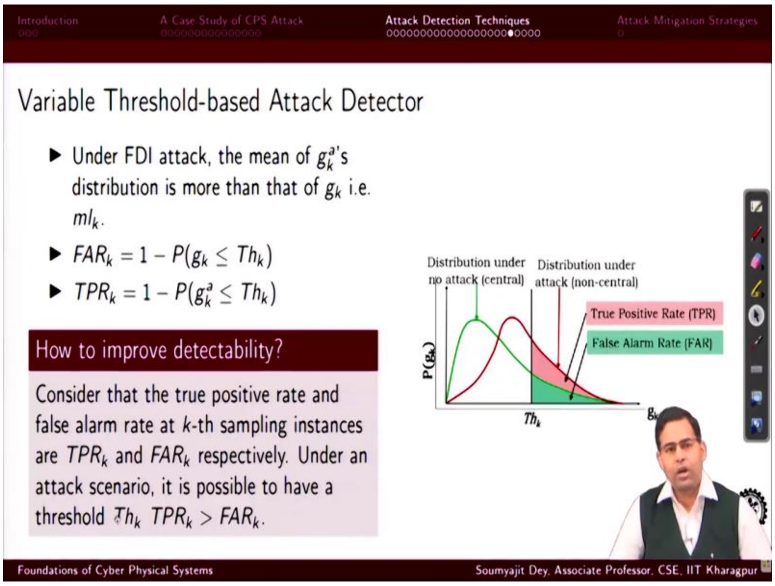
So, let us consider ah windowed chi square detector in place, ah whose detection threshold and window length can be changed dynamically. So, let us understand this. What we are doing is we earlier talked about such windows chi square detectors, so, you will be comparing a chi square static steps with a threshold. And we are saying that well for this threshold base detector, we can change the following parameters.

One is the window length that means how many samples together, I should what is the history of the attack that I should consider for generating the chi square statistics, this length l, and based on this length ah I what what we can do is we can compute ah what is the probability density function of this chi square statistics g_k. And this is the standard mathematical formula that comes out from basic knowledge of chi square statistics that if the mean is this ml_k and the number of sensors considered is m. So, we are generating chi square statistics for m number of sensors and mean chi square statistics is ml_k, then the we will have a probability density function, ah given like this. ah So, this is the probability density function which will be capturing this chi square statistics. Ok And we can actually check ah what is that.

Whether I am less than this threshold or not, will be given by this probability density function, ah by by which will be given by this integration of this to a PDF which will be the CDF or the cumulative distribution function. That means all I am doing is I am well I have this chi square statistics computed. The chi square statistics, ah will be following this distribution as per the statistics of chi square ok ah the the properties of chi square distribution.

So, this is how the chi square statistics PDF will be distributed. And now we need to raise an alarm at certain points. That means, when I go beyond the threshold then what is the probability that I am below the threshold? So that would mean that well I have to kind of integrate this PDF from 0 to up to the threshold value and that would give me this function. So, here gamma this is the gamma function here just note.

(Refer Slide Time: 16:32)



Introduction | A Case Study of CPS Attack | Attack Detection Techniques | Attack Mitigation Strategies

Variable Threshold-based Attack Detector

- Under FDI attack, the mean of g_k^a's distribution is more than that of g_k i.e. ml_k.
- $FAR_k = 1 - P(g_k \leq Th_k)$
- $TPR_k = 1 - P(g_k^a \leq Th_k)$

How to improve detectability?

Consider that the true positive rate and false alarm rate at k-th sampling instances are TPR_k and FAR_k respectively. Under an attack scenario, it is possible to have a threshold Th_k $TPR_k > FAR_k$.

Distribution under no attack (central)
Distribution under attack (non-central)
True Positive Rate (TPR)
False Alarm Rate (FAR)
P(g_k)
Th_k
g_k

Foundations of Cyber Physical Systems | Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

So, this is a this is an idea that well ah so, we are just putting in these things for your knowledge, let us not bother about some of these, some of these comments here. That this is the typical chi square distribution that you will have under no attack so, this is a central distribution. Under attack this distributions will shift so but let us not bother about this statistical properties of chi square detectors. Let us understand what is important for us.

For us, what is important is that ah for any such statistical detector, you will have two things one is a false alarm rate and the other is a true positive rate. So, let us understand that what is this false alarm rate and what is this true positive rate. So, false alarm means that well there is a detection that is happening. But actually, there is no ah no attack that really happened. Ok And the true positive rate is that well there was a detection that happened.

And that actually was corresponding to a real attack that happened. right So, ah at some kth sampling instance ah we can compute both of these things using all these formula, ah based on ah statistical properties of chi square distribution that we talked about. So, what we said is suppose we have taken ah l window length Ah detector and it has a threshold ah Th_k. So, for a l window length detector this is how you compute the chi square statistics, g_k.

And by statistical properties it will follow this distribution of PDF following this function, where ml k is the mean of the chi square distribution. Once I have known this PDF, I can compute the CDF with a like this up to the threshold. And once I have computed it up to the threshold, I can check what is the true positive rate by doing this. That means I consider that well I can be under attack and I may not be under attack.

So, when I am not under attack, I will have the one distribution function and when I am under attack, I can calculate the my corresponding distribution function. And when I am not under attack in this case, I figure out well what is the probability that I am not beyond, below the threshold? Right So, this is how I compute that what is the probability I am below the threshold. So, take 1 minus of that then that is telling me that I am not below the threshold.

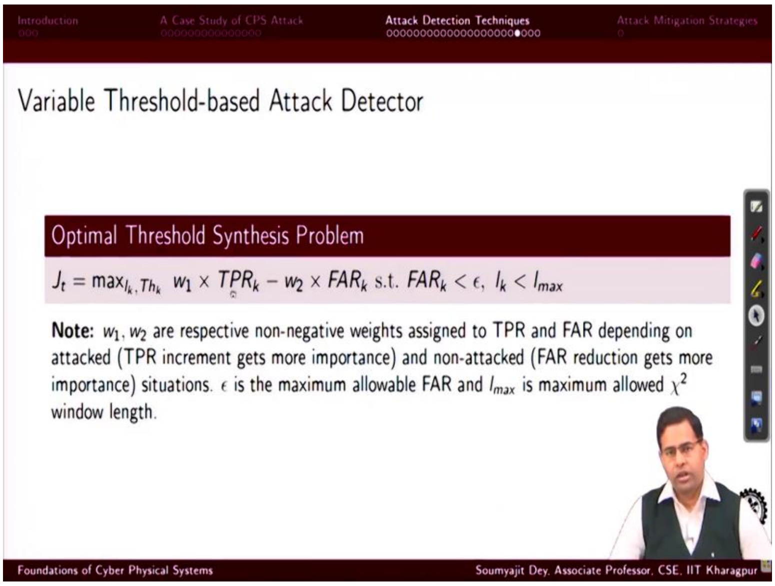
But there is no attack here. Right So, I am not below the threshold but and and the and the hypothesis that there is no attack. So, there is the ground truth. So that is why this corresponds to a false alarm. right And this is how I can compute the false alarm probability in the kth step.

I hope this is clear. right ah Here we have already used the chi square distribution, statistical properties to compute the PDF.

And using this relation that means integrating the PDF I can actually see what is the probability that ah this chi square statistics that I am coming computing based on my current observations that is less than the threshold so, this is how I compute that. And then at the runtime there can be two possibilities I am either under attack or I am under no attack. So, in both these situations, if I am under no attack, I compute the probability ah that well what is the probability I am beyond the threshold. So that is when actually the ground truth is, there is no attack. This probability is that of the false alarm that I am, there is no attack but still the detector is firing. And in the other case we are saying that well there is an attack under the detector has detected it.

So that is a true positive rate. So, my point is, if I am able to compute these distributions under attack and under no attack scenario and there are mathematical methods for computing them ah computing or distribution means identifying the distributions mean and variance under attack and under suitable attack models. Once you are able to do that I can see based on ah this chi square detectors window length and the threshold parameter set that well what is my false alarm rate and what is my true positive rate. Now, in order to improve the detectability what I will do is I will like to have a higher true positive rate. Right So, my the problem then becomes that well how do I choose to have a threshold so that I always achieve this. Because this is what I do I desire. Right

(Refer Slide Time: 21:00)



Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Variable Threshold-based Attack Detector

Optimal Threshold Synthesis Problem

$J_t = \max_{l_k, Th_k} \; w_1 \times TPR_k - w_2 \times FAR_k \text{ s.t. } FAR_k < \epsilon, \; l_k < l_{max}$

Note: w_1, w_2 are respective non-negative weights assigned to TPR and FAR depending on attacked (TPR increment gets more importance) and non-attacked (FAR reduction gets more importance) situations. ϵ is the maximum allowable FAR and l_{max} is maximum allowed χ^2 window length.

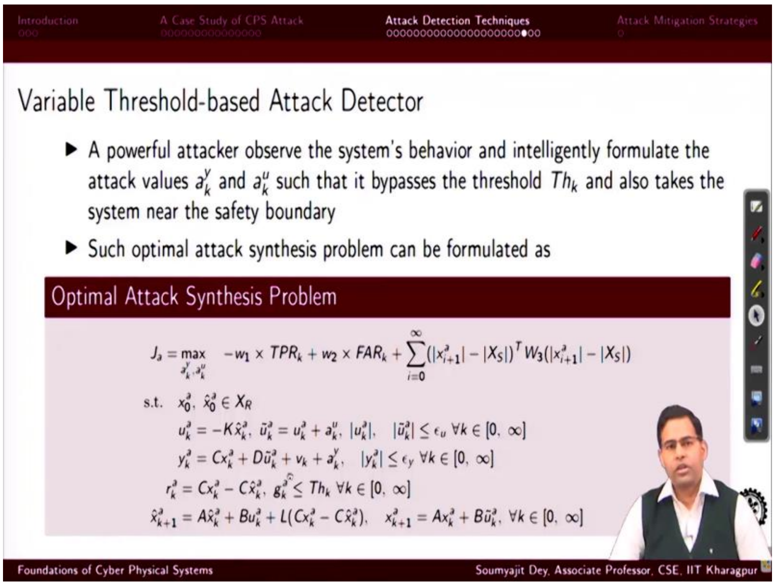
Foundations of Cyber Physical Systems

Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

So, this leads us to an optimal threshold synthesis problem. That means I am trying to select suitable values of this ah chi square window size and the threshold size. So that my true positive rate will increase and my false alarm rate will go down. That means this function that means this subtraction should be as much as possible. right And they have been weighted with with suitable weight parameters W_1 and W_2 depending on the importance ah of the attack and the non-attack variables.

So that is how, depending on their importance, I mean if I am giving TPR increment as more importance, I will give more weight is W_1 here or if I am giving FAR reduction more importance I will give a higher weightage of W_2 here. right So, ah based on this, I can create this kind of for thresholds into this problem and I can compute based on the probability distributions and I can compute what is the TPR rate and what is the FAR and I can create this problem here.

(Refer Slide Time: 22:01)



Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Variable Threshold-based Attack Detector

- A powerful attacker observe the system's behavior and intelligently formulate the attack values a_k^y and a_k^u such that it bypasses the threshold Th_k and also takes the system near the safety boundary
- Such optimal attack synthesis problem can be formulated as

Optimal Attack Synthesis Problem

$$J_a = \max_{a_k^y, a_k^u} \quad -w_1 \times TPR_k + w_2 \times FAR_k + \sum_{i=0}^{\infty}(|x_{i+1}^a| - |X_S|)^T W_3(|x_{i+1}^a| - |X_S|)$$

$$\begin{aligned} \text{s.t.} \quad & x_0^a, \hat{x}_0^a \in X_R \\ & u_k^a = -K\hat{x}_k^a, \ \tilde{u}_k^a = u_k^a + a_k^u, \ |u_k^a|, \quad |\tilde{u}_k^a| \leq \epsilon_u \ \forall k \in [0, \infty] \\ & y_k^a = Cx_k^a + D\tilde{u}_k^a + v_k + a_k^y, \quad |y_k^a| \leq \epsilon_y \ \forall k \in [0, \infty] \\ & r_k^a = Cx_k^a - C\hat{x}_k^a, \ g_k^a \leq Th_k \ \forall k \in [0, \infty] \\ & \hat{x}_{k+1}^a = A\hat{x}_k^a + Bu_k^a + L(Cx_k^a - C\hat{x}_k^a), \quad x_{k+1}^a = Ax_k^a + B\tilde{u}_k^a, \ \forall k \in [0, \infty] \end{aligned}$$

Foundations of Cyber Physical Systems

Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

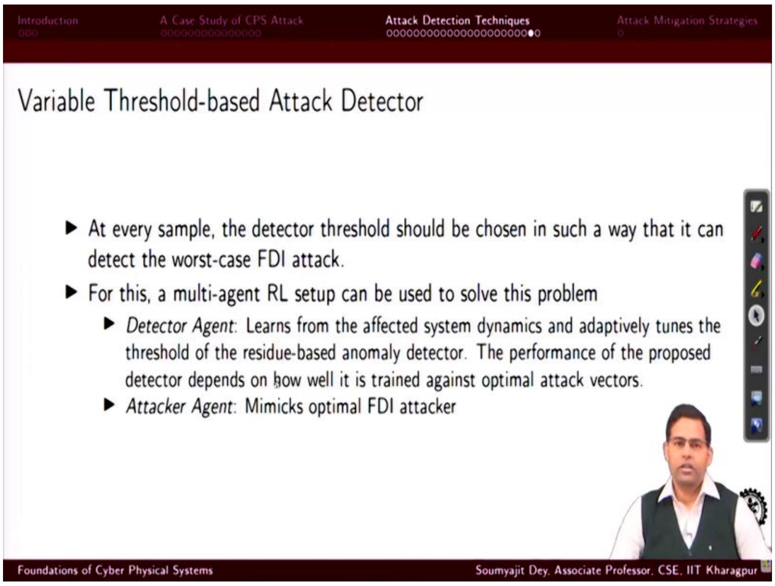
Now, a powerful attacker will observe the system's behaviour and intelligently they will try to formulate attack values. Right I mean that is how an attacker works and it is trying to bypass a threshold. right So, ah I can I I can create an AI agent game. right So, what what can happen is I can learn a powerful attacker and then for the attacker ah I can create an optimal attack synthesis problem. So, what I can do is well, ah what the attacker wants to do is?

The attacker wants to kind of ah do a negation of this thing. That means the attacker does not want my threshold base detector to succeed. Right So, it will have the have the opposite effect. So that means it will try to decrease my true positive ah rate, it will try to increase my false alarm rate. And it will try to ah maximize the error, ah between the state estimate and the ah and the and the actual state right, the estimation error.

So, this is how I can model and optimal attacker that optimal attacker is one who will like to make my detection mechanism bad. So, it will It will just do the opposite of whatever are the properties of a good detector. So, here it is doing the opposite of that. And it will also try to maximize the estimation error at the same time. So, this is how it is achieving stealth and this is how in the in the second component, the attacker is achieving what we call as lethality. Right

So, using these two components the attacker is becoming powerful but again the attacker has to be subjected to all my dynamical equation constraints of ah control of measurement update, of residue update, of state update, etcetera. right So, we can create a complex, ah optimization problem which will create optimal attacks. Right

(Refer Slide Time: 23:52)



Introduction
A Case Study of CPS Attack
Attack Detection Techniques
Attack Mitigation Strategies

Variable Threshold-based Attack Detector

- At every sample, the detector threshold should be chosen in such a way that it can detect the worst-case FDI attack.
- For this, a multi-agent RL setup can be used to solve this problem
  - *Detector Agent*: Learns from the affected system dynamics and adaptively tunes the threshold of the residue-based anomaly detector. The performance of the proposed detector depends on how well it is trained against optimal attack vectors.
  - *Attacker Agent*: Mimicks optimal FDI attacker

Foundations of Cyber Physical Systems
Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

And what we can do is at every sample the threshold detector would choose in such a way that it can detect the worst-case FDI attack. That means ah we are looking at estimating what is the worst-case attack. At every cycle dynamically I am trying to see what can be the attack that can happen? And then into truth word that I will like to choose suitable values, suitable parameters for the detector.

Of course, this is a complex problem to solve in the runtime, so, we can bring in an AI agent. So, I can have a reinforcement learning setup where I have a detected agent. And the detector agent is learning from the affected system dynamics and it is adaptively tuning the threshold of this residue-based anomaly detector. And the performance of the detector is depending on how well it is trained against the optimal attack vectors.

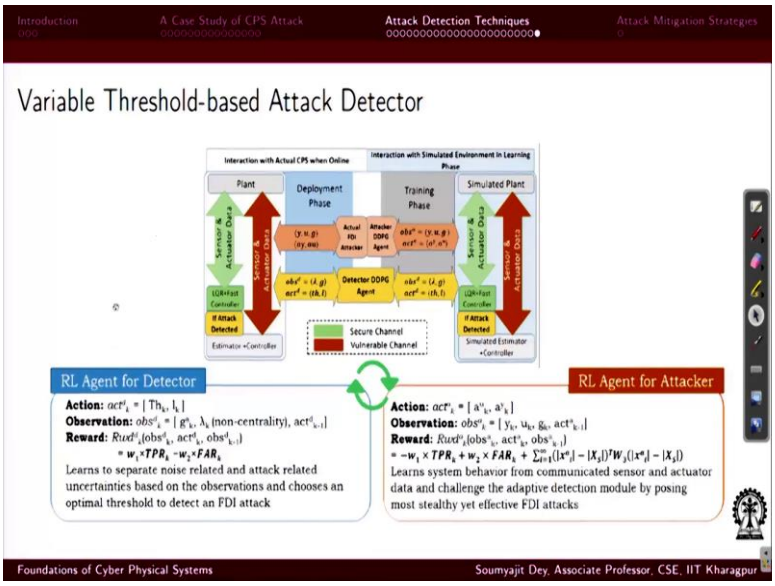
And also, you can have the attacker agent which is mimicking an optimal FDI attacker. So, let us understand how to build an attacker agent. The attacker agent will try to solve this problem. And the detector agent will try to figure out that well what are the suitable parameters for the threshold based detector so that this objective function is kind of maximized. Right

And ah once we learn both these agents Ah deploying them together, I mean in in presence of the detector the attacker agent ah who is mimicking an optimal FDI attacker, I can learn detection strategies for setting suitable thresholds at different points of the the state space. right So that is how the performance of the detector will be depending on how well it is trained against the optimal attack vector.

So, just in to summarize, if I am trying to choose optimal threshold parameters, they should be chosen such that this thing happens, this thing is maximized. Right The true positive rate is high and the false alarm rate is down. But if I have to do that I have to do that against good attacks and the way to generate good attacks is by solving this optimization problem. You can try to solve this using some analytical method or you can have some AI based agent which is mimicking this kind of optimal attacks.

And you can have a detection agent which is trying to subvert this optimal attacks Ah by by maximizing this function in presence of such attacks.

(Refer Slide Time: 26:09)



Variable Threshold-based Attack Detector

RL Agent for Detector

Action: $act^d_k = [\,Th_k, l_k\,]$
Observation: $obs^d_k = [\,g^a_k, \lambda_k$ (non-centrality), $act^d_{k-1}]$
Reward: $Rwd^d_k(obs^d_k, act^d_k, obs^d_{k-1})$
$= w_1 \times TPR_k - w_2 \times FAR_k$
Learns to separate noise related and attack related uncertainties based on the observations and chooses an optimal threshold to detect an FDI attack

RL Agent for Attacker

Action: $act^a_k = [\,a^u_k, a^y_k\,]$
Observation: $obs^a_k = [\,y_k, u_k, g_k, act^a_{k-1}]$
Reward: $Rwd^a_k(obs^a_k, act^a_k, obs^a_{k-1})$
$= -w_1 \times TPR_k + w_2 \times FAR_k + \sum_{i=1}^{\infty}(|x^a_i| - |X_S|)^T W_3 (|x^a_i| - |X_S|)$
Learns system behavior from communicated sensor and actuator data and challenge the adaptive detection module by posing most stealthy yet effective FDI attacks

Foundations of Cyber Physical Systems

Soumyajit Dey, Associate Professor, CSE, IIT Kharagpur

So that is how you can create a multi-agent RL problem OK ah where you have this attacker agent who is getting rewards based on, so you have this attacker agent who is getting rewards based on this cost function on the right hand side. So, it is trying to learn behaviours Ah between I mean the system behaviour and accordingly, it is trying to give suitable attacks on the control And the ah and the system measurement so that this Ah this thing happens this objective function is attained.

And at the same time, ah the detector agent will try to figure out what can be suitable window length and other parameter, so that this attack can be subverted. So, we are not asking you to get deeper into this. But what we want you to understand is what is the motivation behind this cost function, how this is modelling, how this equation is modelling a lethal attack vector. And how this equation is modelling a suitable detector?

Because what this can help is, if I know that what is a lethal attack vector. For that I can choose suitable detector parameter. So that the detector works more often than now ok so, ah that is how ah this variable threshold idea can work by deploying suitable AI agents in a system. And with this we will like to end this lecture. Thank you for your attention.