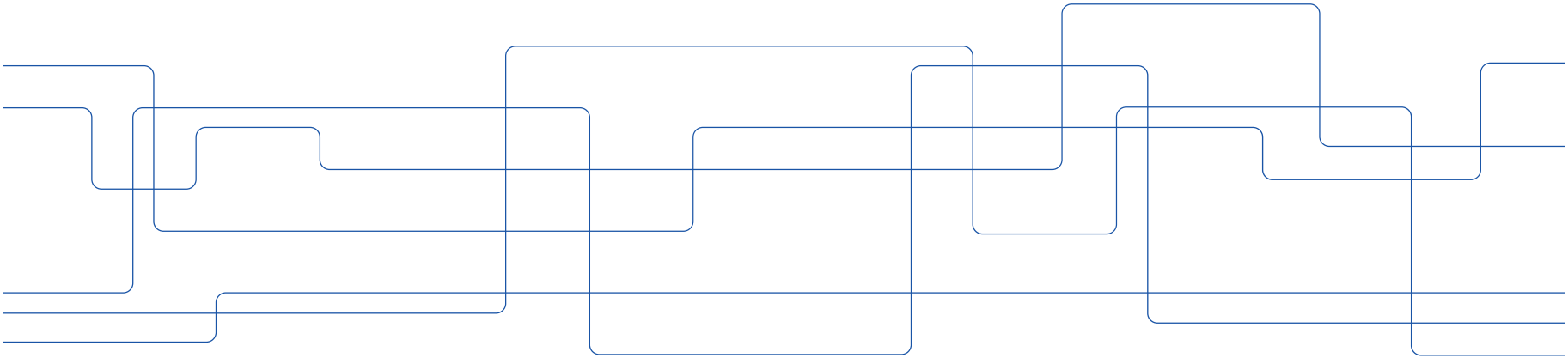
KTH ROYAL INSTITUTE OF TECHNOLOGY

Cyber-Physical Security in Energy Systems

DTU PES Summer School 2026

Henrik Sandberg (hsan@kth.se)

KTH EECS, Decision and Control Systems



Outline

- Part I: CPS Security
  - Security challenges for control systems
  - Cyber-secure control and risk management
  - Case studies

- Part II: Attack Modeling and Detection Methods
  - DoS attack modeling and dynamic instability
  - FDI attack modeling and detection
    - > Unknown state and input estimation
    - > CUSUM
    - > PCA

Acknowledgments



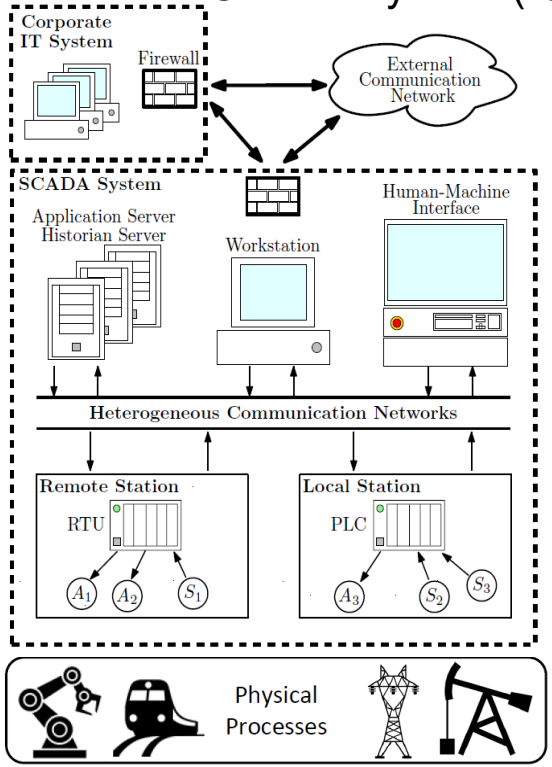




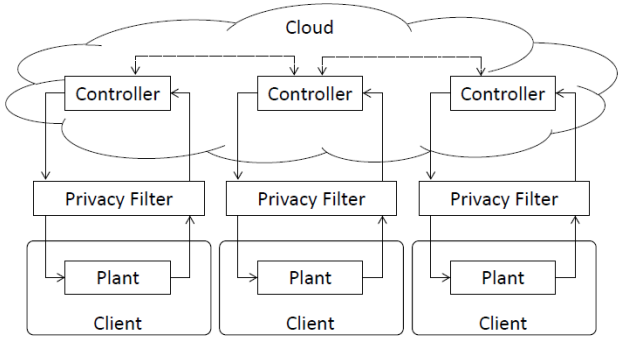
- André Teixeria (Uppsala Univ.)
- Iman Shames (Univ. of Melbourne)
- Kin Cheong Sou (National Sun Yat-sen Univ.)
- Michelle Chong (TU/e)
- Hampei Sasahara (Univ. of Tokyo)
- Alvaro Cardenas (UCSC)
- Shreyas Sundaram (Purdue Univ.)
- Karl Henrik Johansson, György Dán, Jezdimir Milosevic, David Umsonst, Kaveh Paridari, Rijad Alisic, Kamil Hassan, Enno Breukelman, Jacopo Porzio, Takumi Shinohara (KTH)

Cyber-Physical Systems

Industrial Control System (ICS)



Cloud-based Control and IoT



Example 1: The Stuxnet Worm (2010)

https://en.wikipedia.org/wiki/Zero_Days

Synopsis

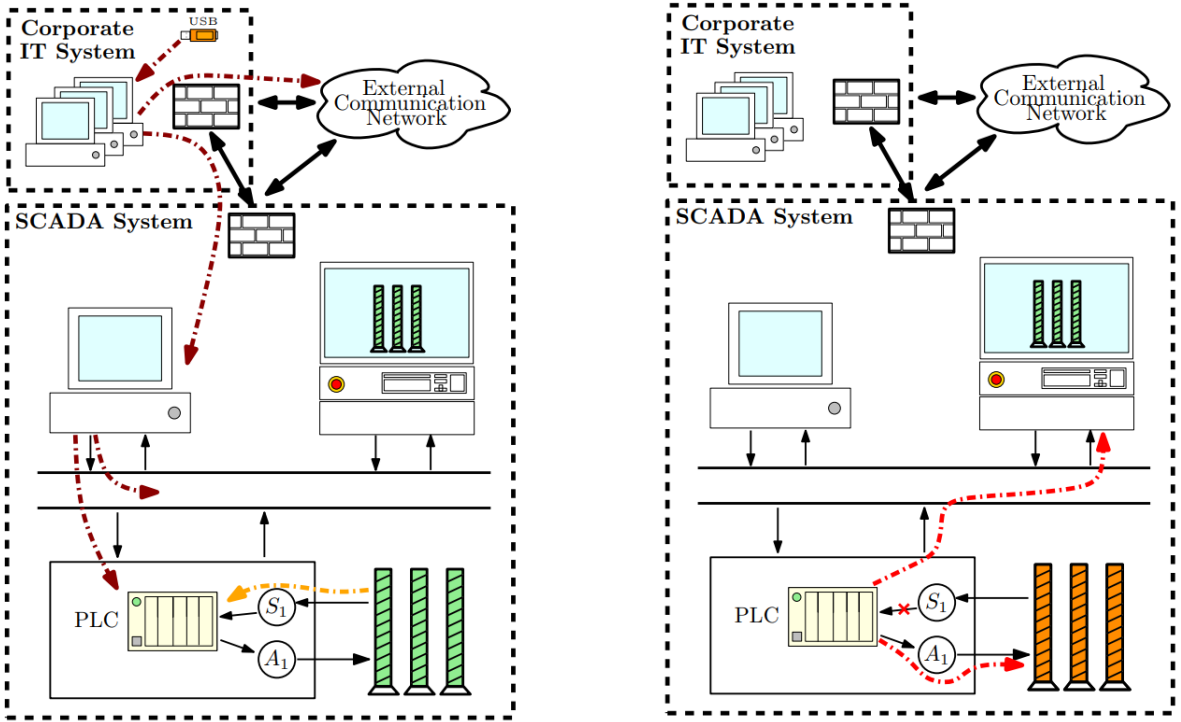
“Zero Days covers the phenomenon surrounding the Stuxnet computer virus [*sic*] and the development of the malware software known as "Olympic Games." It concludes with discussion over follow-up cyber plan Nitro Zeus and the Iran Nuclear Deal.”

Zero Days

Theatrical release poster

Directed by	Alex Gibney
Written by	Alex Gibney
Production companies	Participant Media Showtime Documentary Films Global Produce Jigsaw Productions
Distributed by	Magnolia Pictures
Release dates	February 11, 2016 (Berlin) July 8, 2016 (US)
Running time	116 minutes
Country	United States
Language	English

Example 1: The Stuxnet Worm (2010)



(a) Infection and data recording.

(b) Covert sabotage.

[Teixeira, “Towards cyber-secure and resilient networked control systems,” PhD thesis, KTH, 2014]

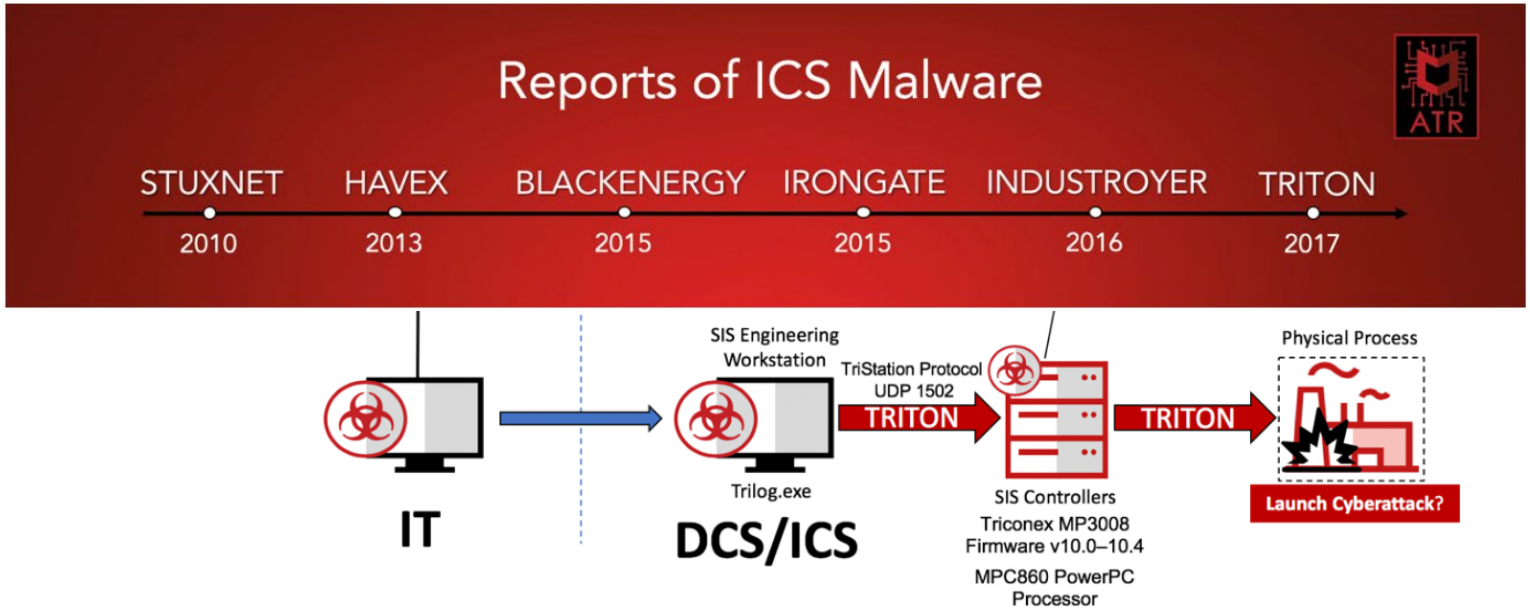
[https://www.mcafee.com/blogs/other-blogs/mcafee-labs/triton-malware-spearheads-latest-generation-of-attacks-on-industrial-systems/]

Example 2: Triton Malware (2017)

Triton framework

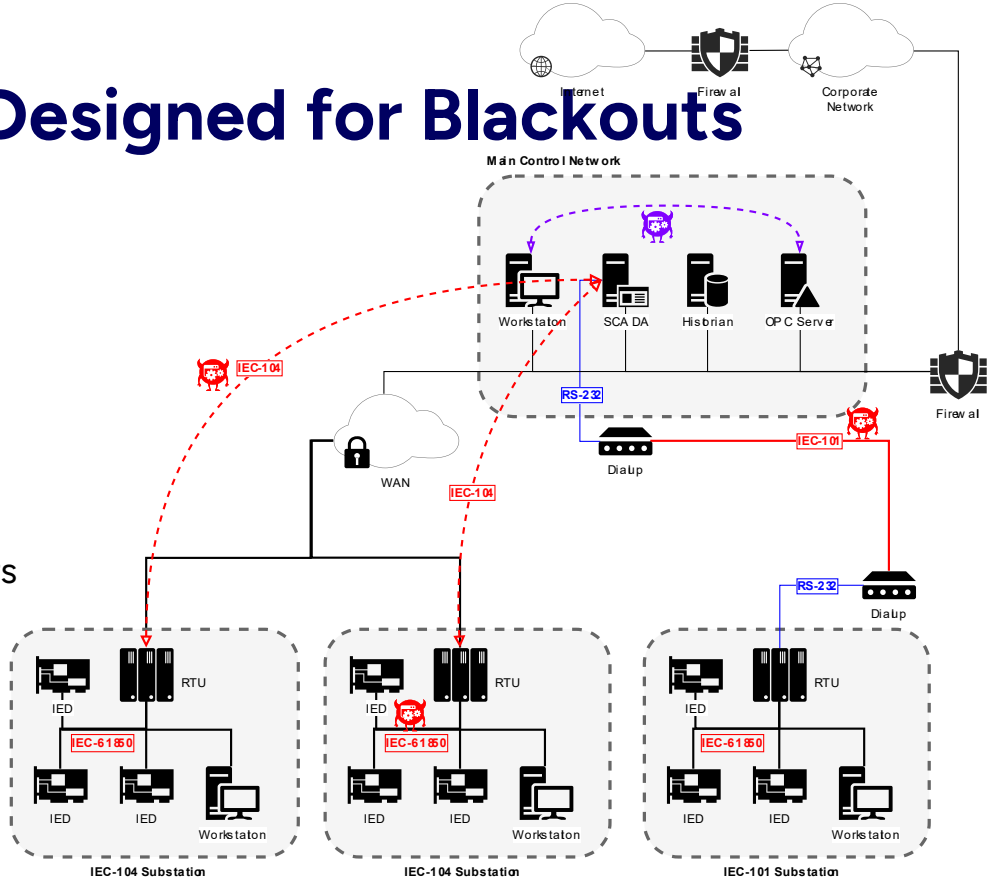
Triton targeted the Triconex safety controller, distributed by Schneider Electric. Triconex safety controllers are used in 18,000 plants (nuclear, oil and gas refineries, chemical plants, etc.), according to the company. Attacks on SIS require a high level of process comprehension (by analyzing acquired documents, diagrams, device configurations, and network traffic). SIS are the last protection against a physical incident.

The attackers gained access to the network probably via spear phishing, according to an investigation. After the initial infection, the attackers moved onto the main network to reach the ICS network and target SIS controllers.



Example 3: Malware Designed for Blackouts

- BlackEnergy (2015)
  - Remote access to the control room
  - Manual opening of circuit breakers
- Industroyer 1 (2016)
- Industroyer 2 (2022)
  - Automatic opening of circuit breakers
  - Deployment in air-gapped systems
- Sandworm group (GRU)



[Salazar *et al.*, "A Tale of Two Industroyers: It was the Season of Darkness," IEEE S&P, 2024]

What Is the Problem?

Safety?

“Protect the human from the system”

Security?

“Protect the system from the human”

[Cardenas, “Cyber-Physical Systems Security Knowledge Area,” CyBOK, 2021]

Why is CPS Security Different?

CyBOK

Security of Control Systems?

Nothing new!
Use normal IT security tools!

Not my job!
It’s the IT security guy’s job!

Security

Control

If attacker has partial control of system, it can drive it to unsafe states.

Not my job!
It’s the control engineers job!

Nothing new!
Safety and fault tolerance will save the day!

Attacks != Failures

12

CyBOK

https://www.cybok.org/

Introductory Concepts

Introduction to CyBOK
Introduction to CyBOK - Version 1.1.0

Human, Organisational & Regulatory Aspects

Risk Management & Governance
Risk Management & Governance - Version 1.1.1

Law & Regulation
Law & Regulation - Version 1.0.2

Human Factors
Human Factors - Version 1.0.1

Privacy & Online Rights
Privacy & Online Rights - Version 1.0.2

Attacks & Defences

Malware & Attack Technologies
Malware & Attack Technologies - Version 1.0.1

Adversarial Behaviours
Adversarial Behaviours - Version 1.0.1

Security Operations & Incident Management
Security Operations & Incident Management - Version 1.0.2

Forensics
Forensics - Version 1.0.1

Software and Platform Security

Software Security
Software Security - Version 1.0.1

Web & Mobile Security
Web & Mobile Security - Version 1.0.1

Secure Software Lifecycle
Secure Software Lifecycle - Version 1.0.2

Infrastructure Security

Applied Cryptography
Applied Cryptography - Version 1.0.0

Network Security
Network Security - Version 2.0.0

Hardware Security
Hardware Security - Version 1.0.1

Cyber Physical Systems
Cyber Physical Systems - Version 1.0.1

Physical Layer and Telecommunications Security
Physical Layer and Telecommunications Security - Version 1.0.1

CyBOK

Cyber-Physical Systems Security Knowledge Area
Version 1.0.1

Alvaro Cardenas | University of California Santa Cruz

EDITOR
Emil Lupu | Imperial College, London

REVIEWERS
Henrik Sandberg | KTH Royal Institute of Technology
Marina Krotofil | Hamburg University of Technology
Mauro Conti | University of Padua
Nils Ole Tippenhauer | CSIPA Helmholtz Center for Information
Rakesh Bobba | Oregon State University

University of BRISTOL

CyBOK

Cyber-Physical Systems Security

bristol.ac.uk

MITRE ATT&CK ICS (Industrial Control Systems)

https://attack.mitre.org/techniques/ics/

ICS

- Initial Access
- Execution
- Persistence
- Privilege Escalation
- Evasion
- Discovery
- Lateral Movement
- Collection
- Command and Control
- Inhibit Response Function
- Impair Process Control
- Impact

ICS Techniques

Techniques represent 'how' an adversary achieves a tactical goal by performing an action. For example, an adversary may dump credentials to achieve credential access.

Techniques: 8
Sub-techniques:

ID	Name	Description
T0800	Activate Firmware Update Mode	Adversaries may activate firmware update mode on devices to prevent expected response functions from engaging in reaction to an emergency or process malfunction. For example, devices such as protection relays may have an operation mode designed for firmware installation. This mode may halt process monitoring and related functions to allow new firmware to be loaded. A device left in update mode may be placed in an inactive holding state if no firmware is provided to it. By entering and leaving a device in this mode, the adversary may deny its usual functionalities.
T0830	Adversary-in-the-Middle	Adversaries with privileged network access may seek to modify network traffic in real time using adversary-in-the-middle (AiTM) attacks. This type of attack allows the adversary to intercept traffic to and/or from a particular device on the network. If a AiTM attack is established, then the adversary has the ability to block, log, modify, or inject traffic into the communication stream. There are several ways to accomplish this attack, but some of the most-common are Address Resolution Protocol (ARP) poisoning and the use of a proxy.
T0878	Alarm Suppression	Adversaries may target protection function alarms to prevent them from notifying operators of critical conditions. Alarm messages may be a part of an overall reporting system and of particular interest for adversaries. Disruption of the alarm system does not imply the disruption of the reporting system as a whole.
T0802	Automated Collection	Adversaries may automate collection of industrial environment information using tools or scripts. This automated collection may leverage native control protocols and tools available in the control systems environment. For example, the OPC protocol may be used to enumerate and gather information. Access to a system or interface with these native protocols may allow collection and enumeration of other attached, communicating servers and devices.

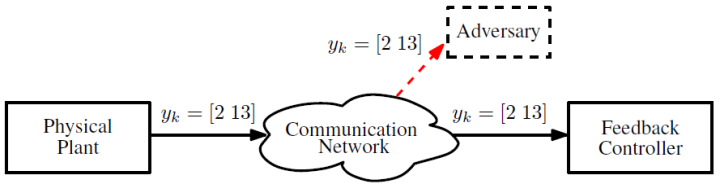
Outline

- Part I: CPS Security
  - Security challenges for control systems
  - **Cyber-secure control and risk management**
  - Case studies

- Part II: Attack Modeling and Detection Methods
  - DoS attack modeling and dynamic instability
  - FDI attack modeling and detection
    - > Unknown state and input estimation
    - > CUSUM
    - > PCA

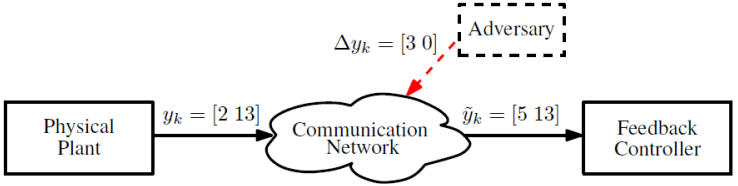
CIA in IT Security

- **C – Confidentiality**



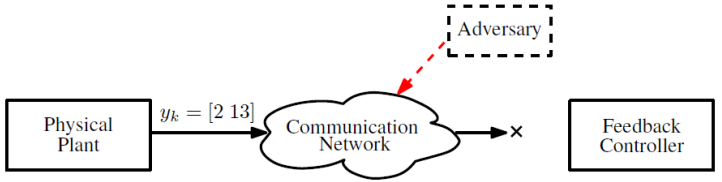
(a) Data confidentiality violation by a disclosure attack.

- **I – Integrity**



(b) Data integrity violation by a false-data injection attack.

- **A – Availability**



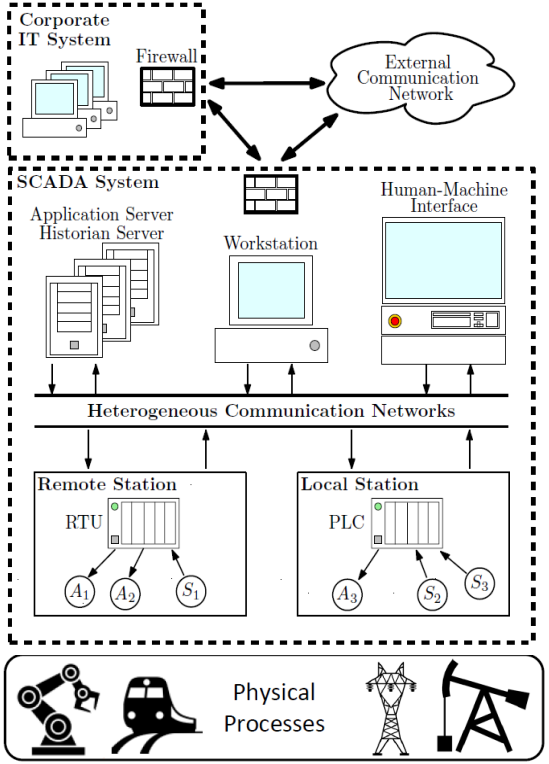
(c) Data availability violation by a denial-of-service attack.

[Teixeira, “Towards cyber-secure and resilient networked control systems,” PhD thesis, KTH, 2014]

Typical ICS Security Vulnerabilities

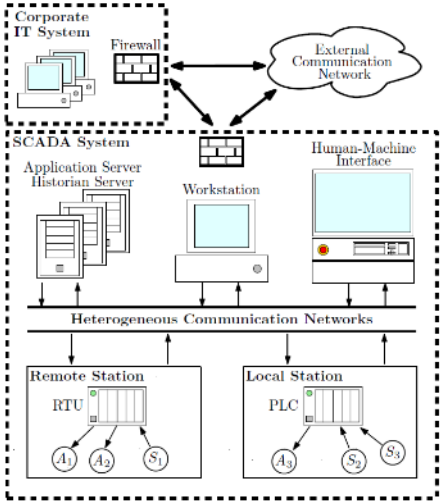
- Computers do not have adequate protection
  - No anti-virus or intrusion detection, USB-ports accessible
- Communication links lack basic security features
  - No encryption, no authentication
- Lack of physical protection
  - PLCs and RTUs accessible
  - Zero-day flaws

Systems designed for an “old” threat landscape!



[Cardenas *et al*., “Research challenges for the security of control systems”, HotSec, 2008]

Security Challenges in ICS



Differences to traditional IT systems:

– **Patching and frequent updates are not well suited for control systems**

– **Real-time availability** (Strict operational environment)

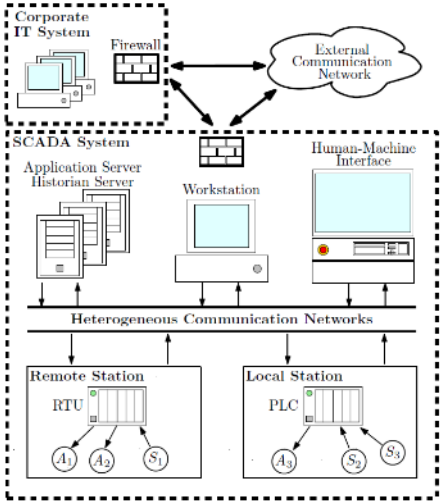
– **Legacy systems** (Often no authentication, no encryption)

– **Protection of information and physical world** (Estimation and control algorithms)

\+ **Simpler network dynamics** (Fixed topology, regular communication (?), limited number of protocols,…)

[Cardenas *et al.*, “Research challenges for the security of control systems”, HotSec, 2008]

Security Challenges in ICS



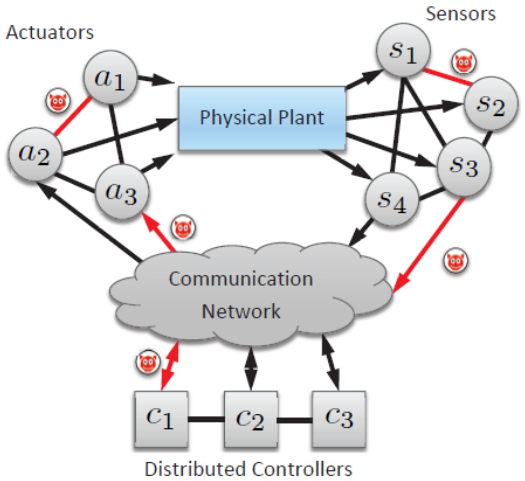
“New” vulnerabilities and “new” threats:

- Controllers are computers (Relays → Microprocessors)
- Networked (Access from corporate network)
- Commodity IT solutions (Windows, TCP/IP,…)
- Open design (Protocols known)
- Increasing size and functionality (New services, wireless,...)
- Large and highly skilled IT global workforce (More IT knowledge)
- Cybercrime (Attack tools available)

[Cardenas *et al.*, “Research challenges for the security of control systems”, HotSec, 2008]

More Than IT Security and Fault Tolerance Needed?

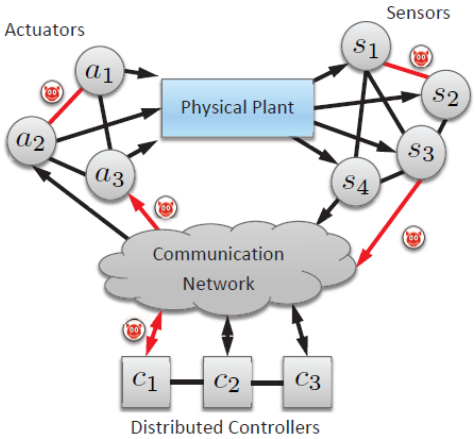
Clearly IT security and fault tolerance are needed:
Authentication, encryption, firewalls, error correction, etc.



But not sufficient…

- **Interaction between physical and cyber systems** make control systems different from normal IT systems
- **Malicious actions can enter anywhere** in the closed loop and cause harm, whether channels secured or not
- **Malicious attackers** have an **intent,** as opposed to faults, and can act strategically
- **Can we trust** the interfaces and channels are really secured? (see **OpenSSL** Heartbleed bug…)

Cyber-Secure Control



Networked control systems

- are being **integrated with business/corporate networks**
- have many potential points of **cyber-physical attack**

Need tools and strategies to understand and mitigate attacks:

- **Which threats** should we care about?
- **What impact** can we expect from attacks?
- **Which resources** should we **protect** (more), and how?
- **Answer: Risk management**

2019 18th European Control Conference (ECC)
Napoli, Italy, June 25-28, 2019

https://ieeexplore.ieee.org/document/8795652

A Tutorial Introduction to Security and Privacy for Cyber-Physical Systems

Michelle S. Chong, Henrik Sandberg, André M.H. Teixeira

***Abstract*— This tutorial provides a high-level introduction to novel control-theoretic approaches for the security and privacy of cyber-physical systems (CPS). It takes a risk-based approach to the problem and develops a model framework that allows us to introduce and relate many of the recent contributions to the area. In particular, we explore the concept of risk in the context of CPS under cyber-attacks, paying special attention to the characterization of attack scenarios and to the interpretation of impact and likelihood for CPS. The risk management framework is then used to give an overview of and map different contributions in the area to three core parts of the framework: attack scenario description, quantification of impact and likelihood, and mitigation strategies. The overview is by no means complete, but it illustrates the breadth of the problems considered and the control-theoretic solutions proposed so far.**

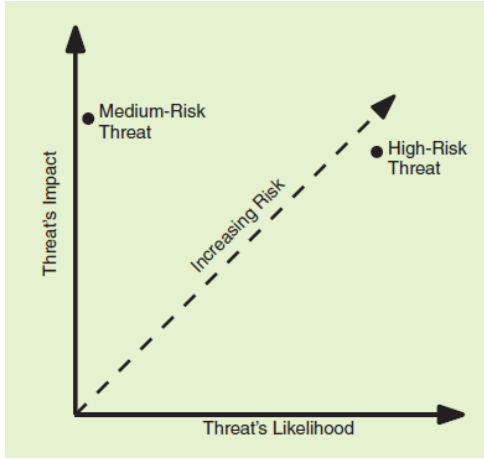
I. Introduction

Cyber-physical systems (CPS) represent a class of networked control systems with vast and promising applications, such as smart cities, distributed sensing and control based on Internet-of-Things (IoT) devices, or ground-breaking transyears have passed since the special issue, and in this tutorial introduction we aim to also introduce some of the more recent work. However, before turning to the CPS security and privacy problems, we should remind ourselves of the basic security properties analyzed in IT systems.

Information is a key asset in knowledge-driven societies, which require a reliable and continuous availability of data and services. Redundant and fault-tolerant architectures are thus required to build IT systems resilient to faults and disturbances [2]. Additionally, IT systems must also be defended against malicious adversaries whose aim is in disrupting or gaining access to the information flow.

Three fundamental properties of information and services in IT systems are mentioned in the computer security literature [3] using the acronym CIA: *confidentiality, integrity*, and *availability*. Confidentiality concerns the concealment of data, ensuring it remains known to the authorized parties alone. Integrity relates to the trustworthiness of data, meaning there is no unauthorized change to the information between the source and destination. Availability considers

Defining Risk



Risk = (Scenario, Likelihood, Impact)

- **Scenario**
  - How to describe the system under attack?

- **Likelihood**
  - Interpretations:

  a) Probability that a given event will occur.

  b) Probability of attack in progress being successful (experts' assessment)

  c) 1

  d) ~1/effort to conduct attack

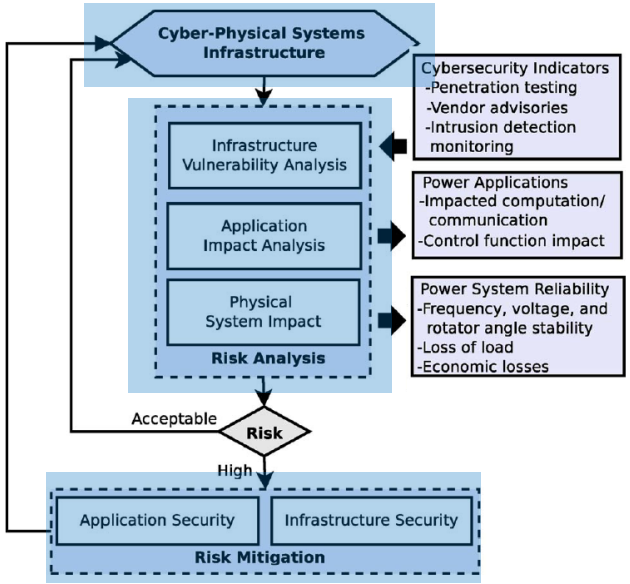
- **Impact**
  - What are the cyber-physical consequences of an attack?

[Kaplan & Garrick, 1981], [Bishop, 2002]

Risk Management Cycle

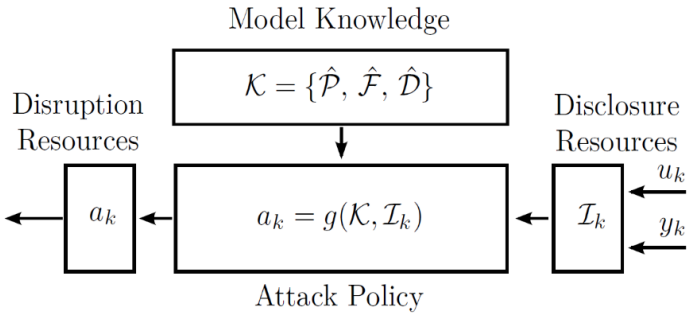
Main steps in risk management

- Scenario characterization
  - Models, Scenarios, Objectives
- Risk Analysis
  - Likelihood Assessment
  - Impact Assessment
- Risk Mitigation
  - Prevention, Detection, Treatment



[Sridhar *et al.*, Proc. IEEE, 2012]

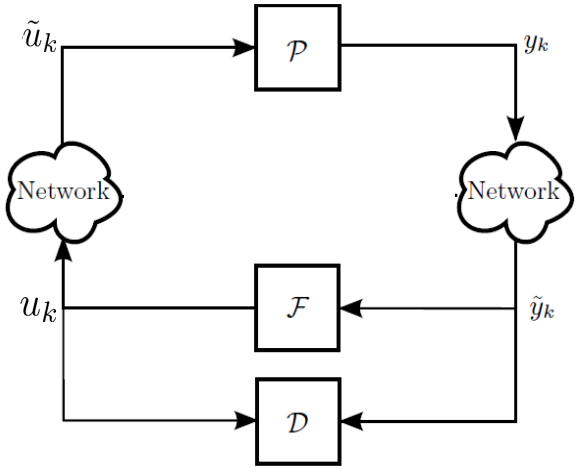
Secure Control Adversary Model

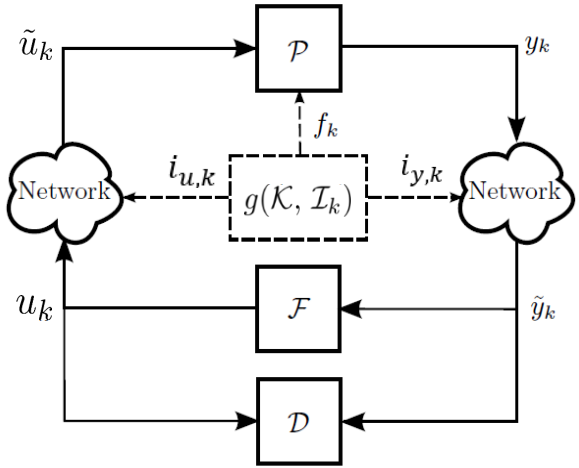


- **Attack policy:** Goal of the attack? Destroy equipment, increase costs, remain undetected,…
- **CPS model knowledge:** Adversary knows models of plant and controller? Possibility for stealthy attacks…
- **Disruption/disclosure resources:** Which channels can the adversary access?

[Teixeira *et al.*, “A secure control framework for resource-limited adversaries”, Automatica, 2015]

Control System under Attack



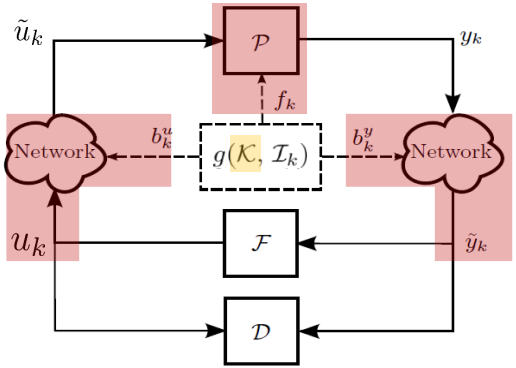


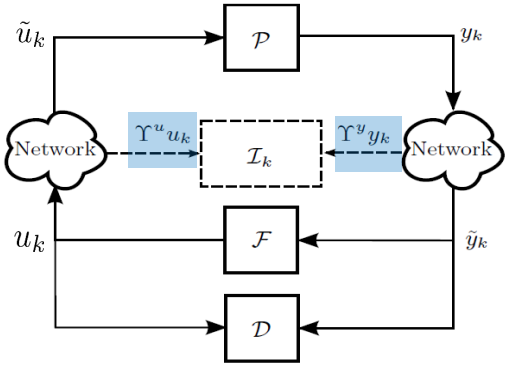
- Physical plant/system ($\mathcal{P}$)
- Feedback controller ($\mathcal{F}$)
- Anomaly detector ($\mathcal{D}$)

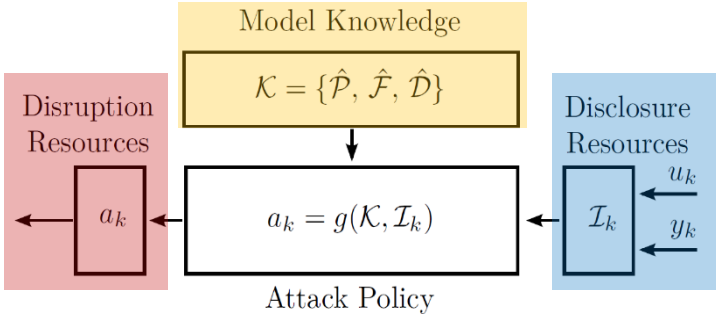
- Disclosure Attacks
- Physical Attacks f_k
- Data Injection Attacks $i_{y,k}, i_{u,k}$

[Teixeira *et al.*, “A secure control framework for resource-limited adversaries”, Automatica, 2015]

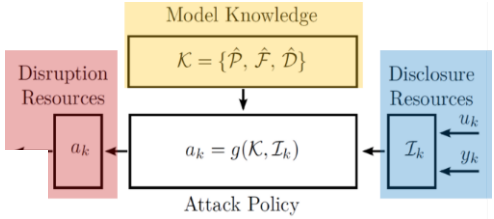
Control System with Adversary Model

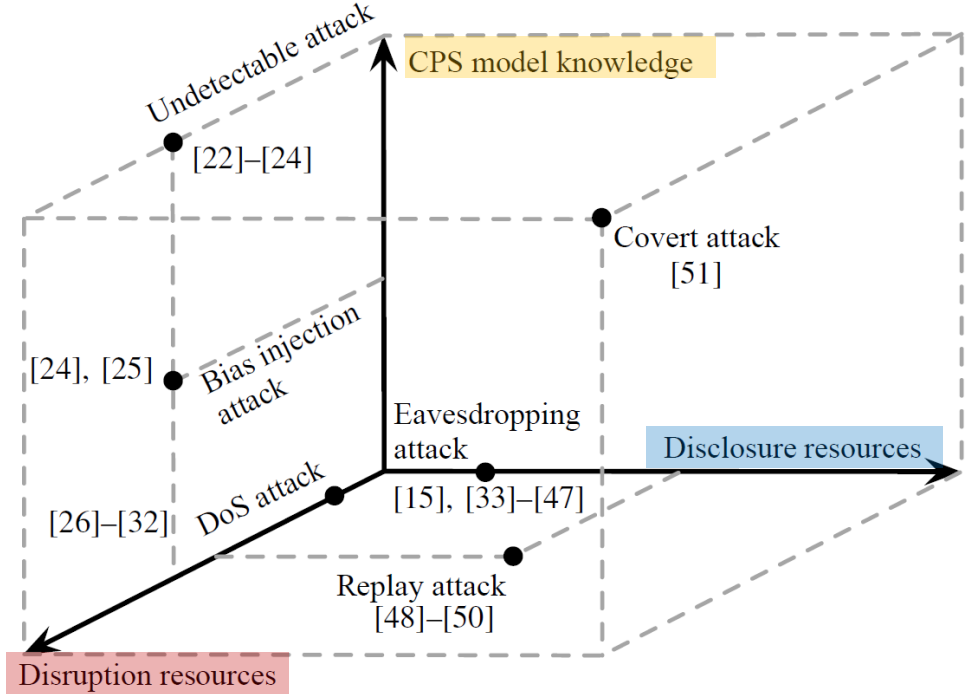






Control Systems Attack Space

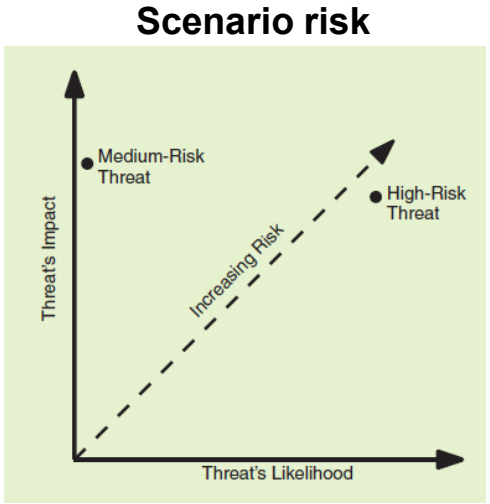




[Chong, Teixeira, Sandberg, "A Tutorial Introduction to Security and Privacy for Cyber-Physical Systems," ECC, 2019]

Tools for Risk Mitigation

- **Prevention** (decrease likelihood by reducing vulnerability)
  - Watermarking and Moving Target Defense
  - Coding and Encryption Strategies
  - Rational Security Allocation
  - Confidentiality Protection by Noise Injection
- **Detection** (continuous anomaly monitoring)
  - Tuning of Detector Thresholds **(Part II)**
  - Secure State Estimation **(Part II)**
  - Watermarking and Moving Target Defense
  - Robust Statistics
- **Treatment** (compensate for, or neutralize, detected attack)
  - Secure State Estimation **(Part II)**
  - Countering DoS Attacks
  - Robust Statistics
  - Controller update



[Chong, Teixeira, Sandberg, "A Tutorial Introduction to Security and Privacy for Cyber-Physical Systems," ECC, 2019]

Outline

- Part I: CPS Security
  - Security challenges for control systems
  - Cyber-secure control and risk management
  - **Case studies**

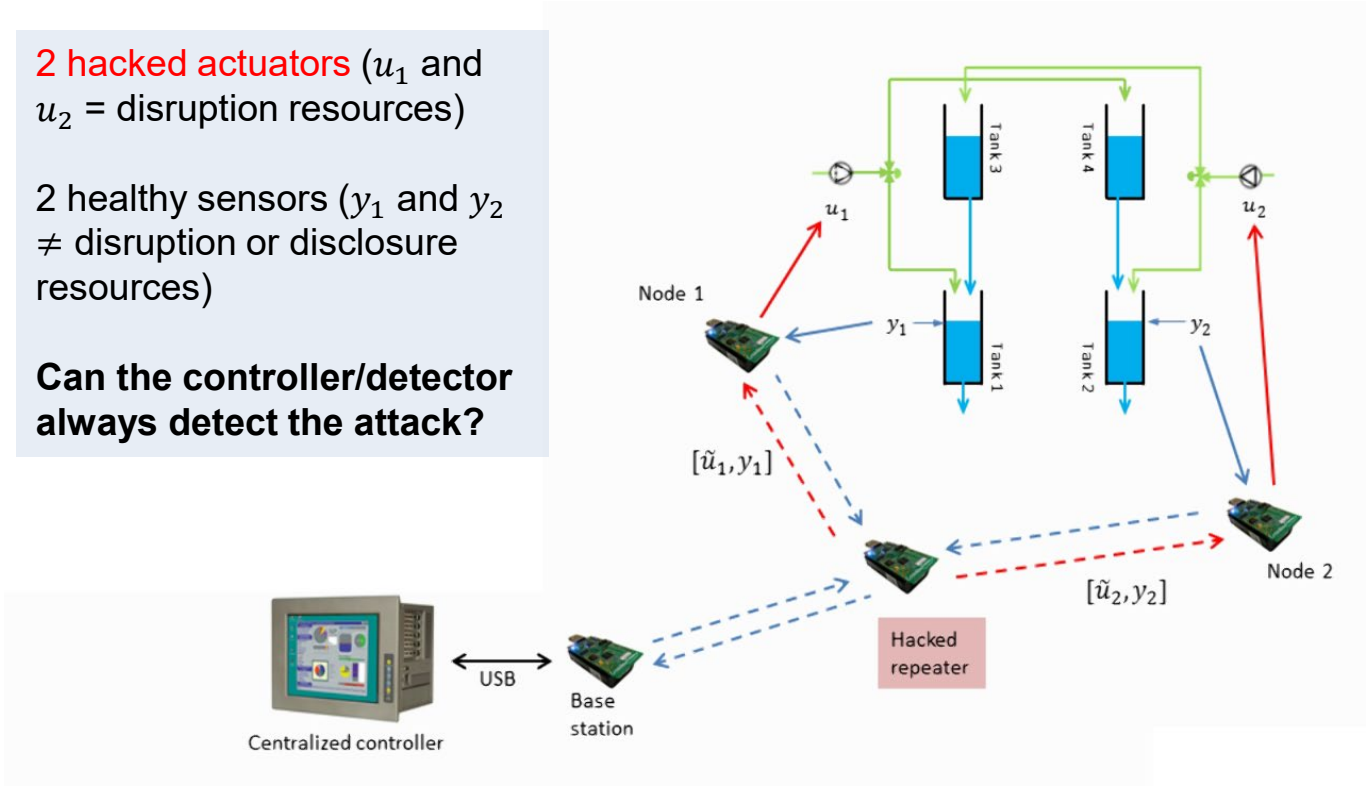
- Part II: Attack Modeling and Detection Methods
  - DoS attack modeling and dynamic instability
  - FDI attack modeling and detection
    - > Unknown state and input estimation
    - > CUSUM
    - > PCA

Case Study 1: Undetectable Water Tank Attack

2 hacked actuators (u_1 and u_2 = disruption resources)

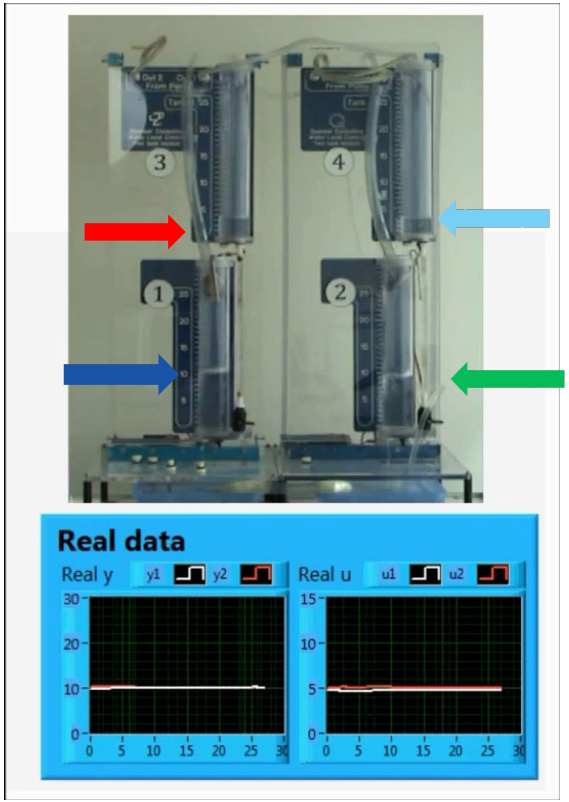
2 healthy sensors (y_1 and y_2 ≠ disruption or disclosure resources)

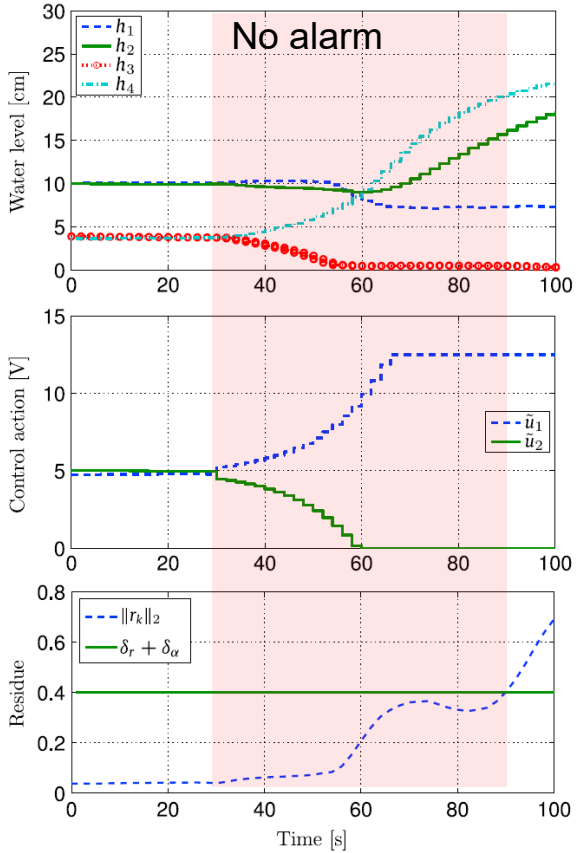
Can the controller/detector always detect the attack?



[Teixeira *et al.*, "A Secure Control Framework for Resource-Limited Adversaries," Automatica, 2015]

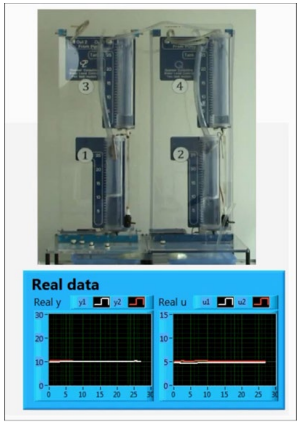
Undetectable Water Tank Attack [Movie]





[Teixeira *et al.*, "A Secure Control Framework for Resource-Limited Adversaries," Automatica, 2015]

Water Tank Model Analysis



- Transfer function matrix from attack to sensor signals

$$G_a(z) = C(zI - A)^{-1}B = \begin{pmatrix} \frac{0.0289}{z-0.8076} & \frac{(1.277z+1.182)\cdot 10^{-3}}{z^2-1.784z+0.7928} \\ \frac{(1.356z+1.24)\cdot 10^{-3}}{z^2-1.754z+0.7643} & \frac{0.02954}{z-0.8347} \end{pmatrix}$$

- Poles = $\{0.8076, 0.8347, 0.9464, 0.9498\}$
- Invariant zeros = $\{0.8675, 1.0362\}$ ⇒ Non-minimum phase system
- Applied attack signal (small ϵ)

$$a(k) = 1.0362^k \begin{pmatrix} 0.2281\epsilon \\ -0.2281\epsilon \end{pmatrix}, \quad x_0 = \begin{pmatrix} 0 & 0 & -0.6521\epsilon & 0.6876\epsilon \end{pmatrix}^T$$

satisfies **zero dynamics** and is **masked by** system transient:

$$0 = y(k) = CA^k x_0 + (g_a * a)(k), \quad k \geq 0$$

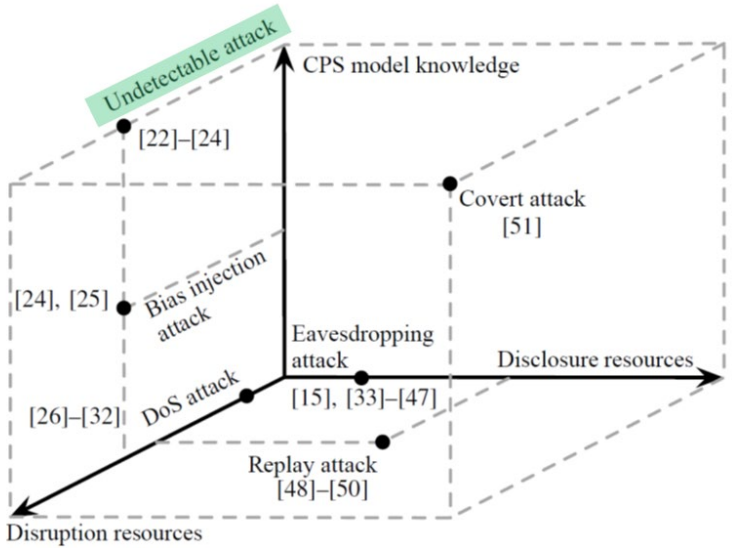
Undetectable Water Tank Attack

2 hacked actuators (u_1 and u_2 = disruption resources)

2 healthy sensors (y_1 and $y_2 \neq$ disruption or disclosure resources)

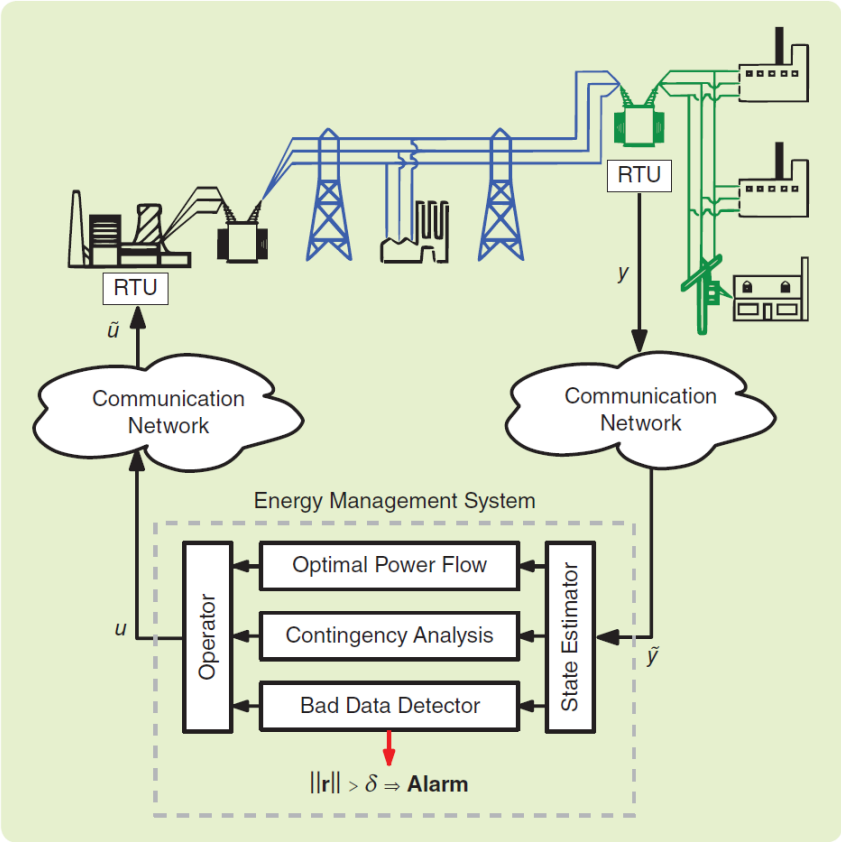
Can the controller/detector always detect the attack?

Not against an adversary with CPS model knowledge



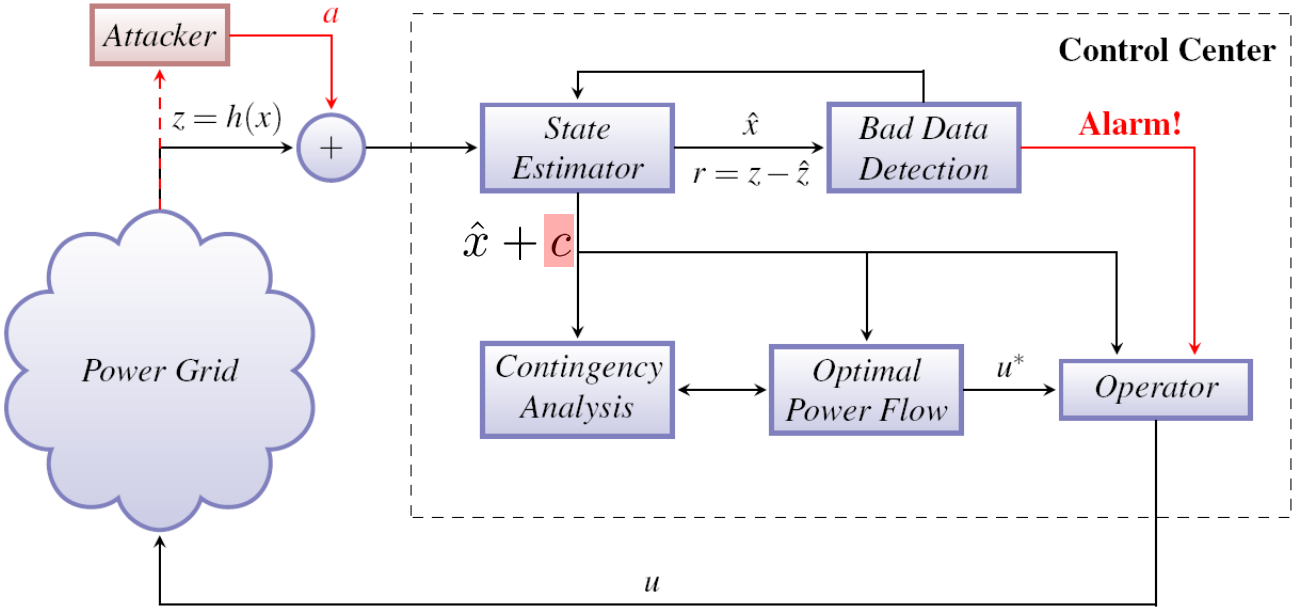
[Teixeira *et al*., “A Secure Control Framework for Resource-Limited Adversaries,” Automatica, 2015]

Case Study 2: Transmission Power System State Estimator



[Teixeira *et al.*, “Secure control systems: A quantitative risk management approach”, IEEE CSM, 2015]

Attacker Model and Bad Data Detection in Control Center



- **Scenario:** Attacker injects **malicious data** a to induce bias c in state estimate
- Typically $\dim(\text{measurement } z) \gg \dim(\text{state } x)$. *Does attacker have to corrupt all sensors to remain undetected?*

Security Index

- Adversary launches undetectable attack against sensor channel i

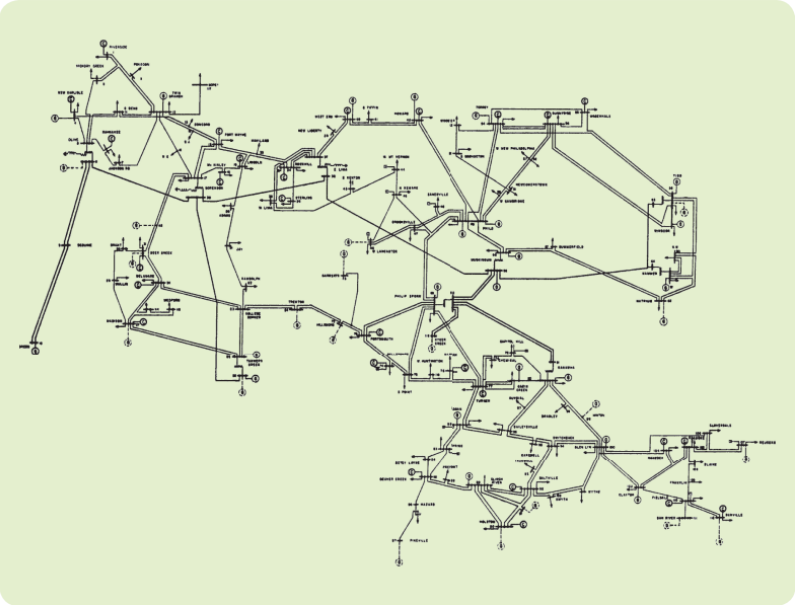
$$\alpha_i := \min_c \|a\|_0 \quad \text{(sparsest possible attack)}$$
$$\text{s.t.} \quad a = Hc \quad \text{(undetectable)}$$
$$a_i = 1 \quad \text{(targets sensor } i\text{)}$$
$$\|a\|_0 := |\{a_k;\ a_k \neq 0\}|$$

[Sandberg *et al.*, "On Security Indices…", SCS, 2010]

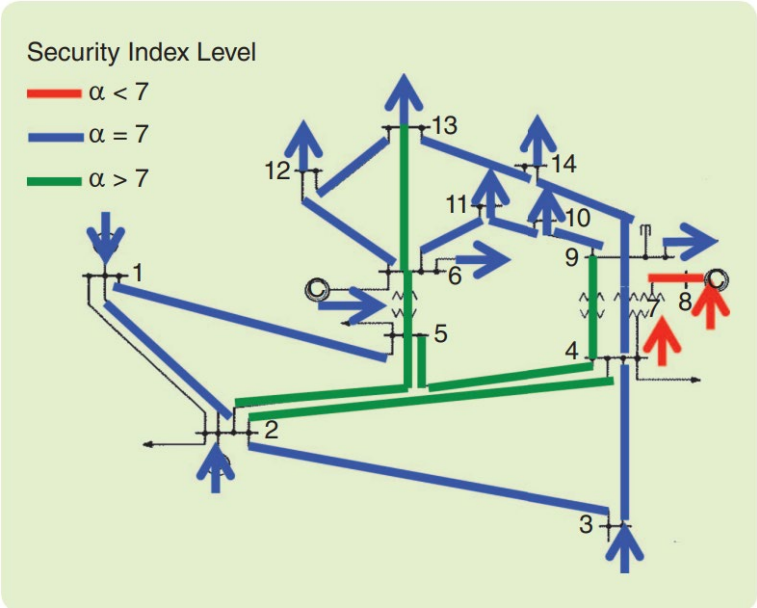
- Quantifies complexity of "least-effort undetectable attack" on sensor i.
- **Example:** $\alpha_1 = 2$ undetectable attack against sensor 1 involves *at least two* sensors in total
- Efficient min-cut/max-flow algorithm for computation exists

[Hendrickx *et al.*, "Efficient computations of a security index…", IEEE TAC, 2013]

IEEE Benchmark Systems



IEEE 118-bus system



IEEE 14-bus system

[Teixeira *et al.*, “Secure control systems: A quantitative risk management approach”, IEEE CSM, 2015]

Undetectable and Identifiable (Correctable) Sensor Attacks

Theorem: Suppose that the attacker can manipulate at most q sensors simultaneously ($\|a\|_0 \leq q$).

i. There exists undetectable attacks against sensor $i \Leftrightarrow \ q \geq \alpha_i$

ii. All attacks are i-identifiable (sensor i correctable) $\Leftrightarrow q \ < \alpha_i/2$

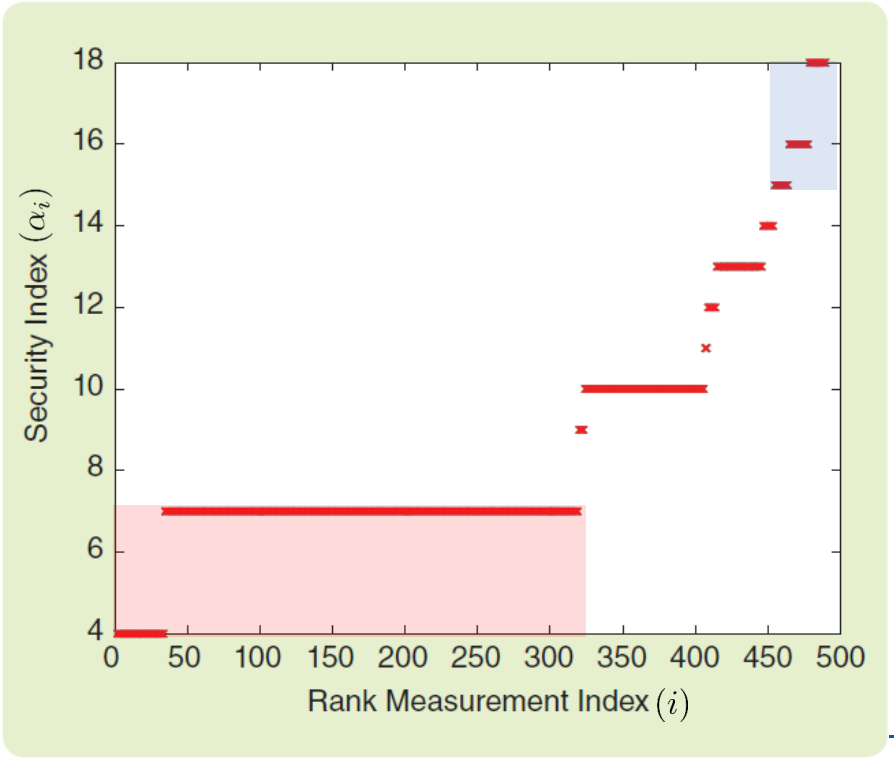
iii. All attacks are identifiable (all sensors correctable) $\Leftrightarrow q \ < \min_i \alpha_i/2$

- Proofs based on compressed-sensing and coding-theory type arguments

[Sandberg and Teixeira, “From Control System Security Indices to Attack Identifiability”, SoSCYPS, 2016]

Example: Power System State Estimator for IEEE 118-bus System

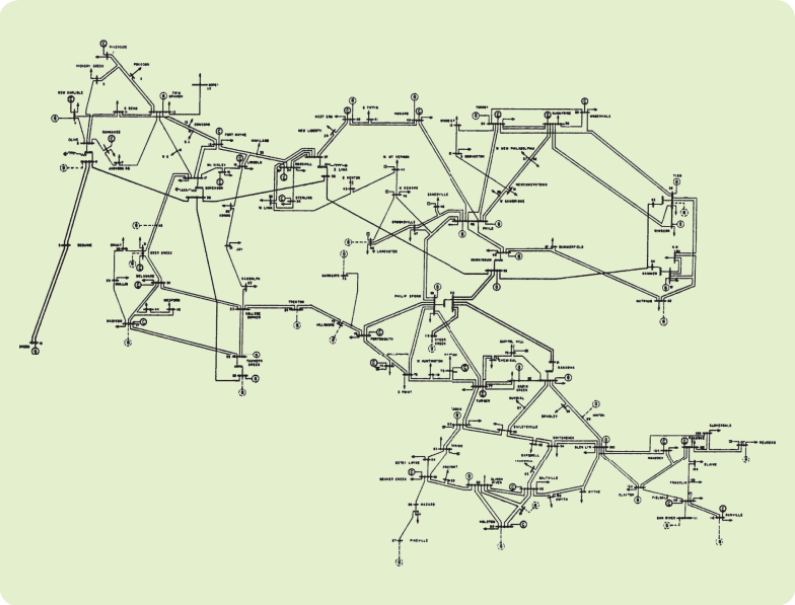
Suppose number of attacked elements is $q \leq 7$. Theorem yields:



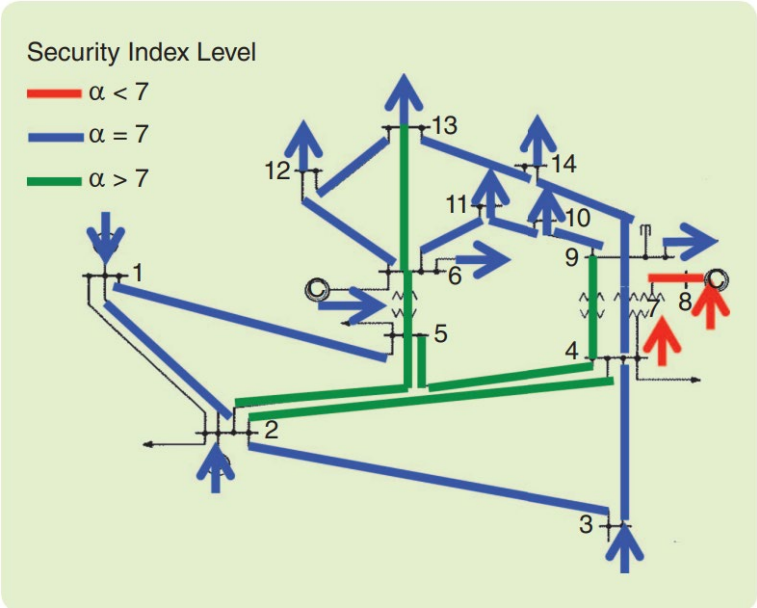
- Sensors susceptible to undetectable attacks
- Sensors where attacks are correctable
- Other sensor attacks are (in principle) detectable, but not correctable

[Teixeira *et al.*, "Secure control systems: A quantitative risk management approach", IEEE CSM, 2015]

IEEE Benchmark Systems



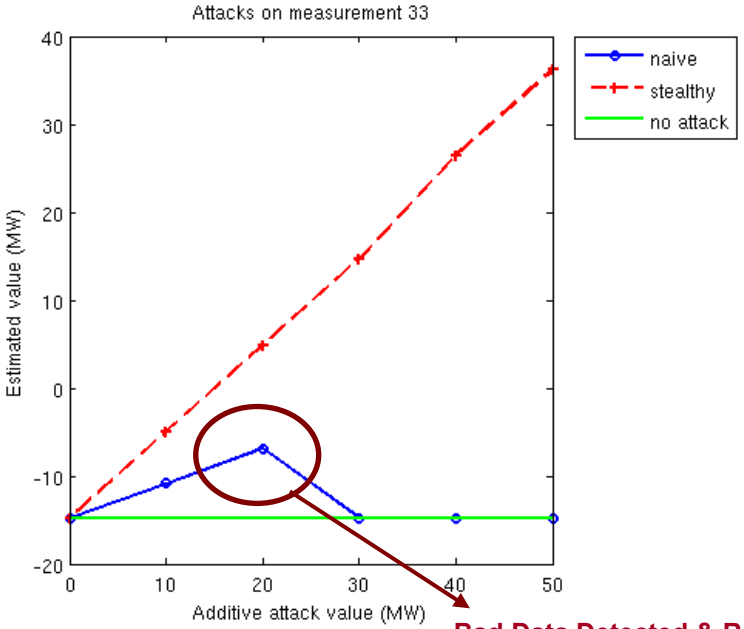
IEEE 118-bus system



IEEE 14-bus system

[Teixeira *et al.*, “Secure control systems: A quantitative risk management approach”, IEEE CSM, 2015]

Experiments on SCADA/EMS Testbed



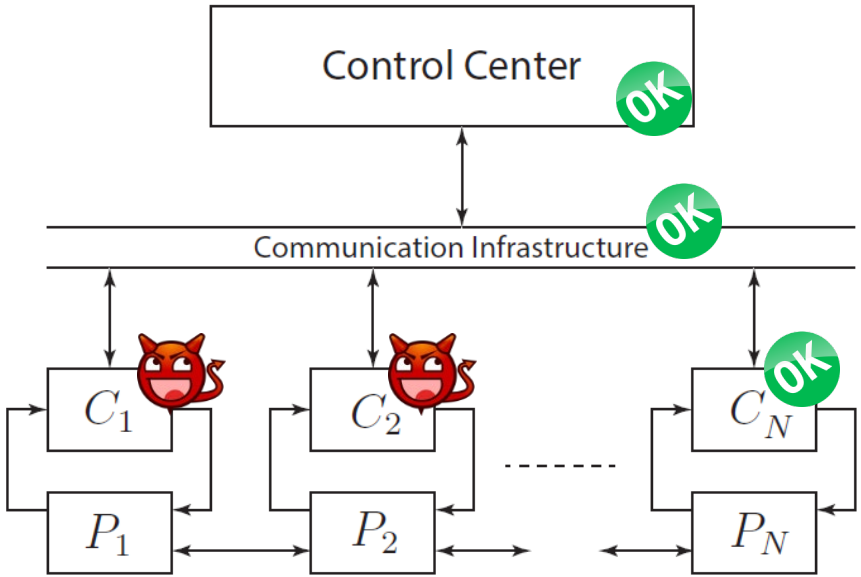
Bad Data Detected & Removed

False value (MW)	Estimated value (MW)	# BDD Alarms
-14.8	-14.8	0
35.2	36.2	0
85.2	86.7	0
135.2	137.5	0
185.2	Non convergent	-

- Attacks computed using linear model but evaluated on “true” nonlinear system
- Attacks of 150 MW (≈55% of nominal value) pass undetected

[Teixeira, *et al*., “A cyber security study of a SCADA energy management system: Stealthy deception attacks on the state estimator”, IFAC WC, 2011]

Case Study 3: Low-level Attacks Against Local Controllers



- Some, but not all, of the local controllers $(C_1, C_2, \dots)$ are arbitrarily corrupted
- Communication Infrastructure, Control Center, and one Local Controller (C_N), are trusted
- Technical assumption: Infrastructure $(P_1, P_2, \dots, P_N)$ *observable* from C_N

[Paridari, *et al.*, "A Framework for Attack-resilient Industrial Control Systems", Proc. IEEE, 2018]
In collaboration with UTRC and Dell-EMC Corporation (Ireland)

NIMBUS Microgrid, Cork, Ireland

Electrical components

10kW wind turbine

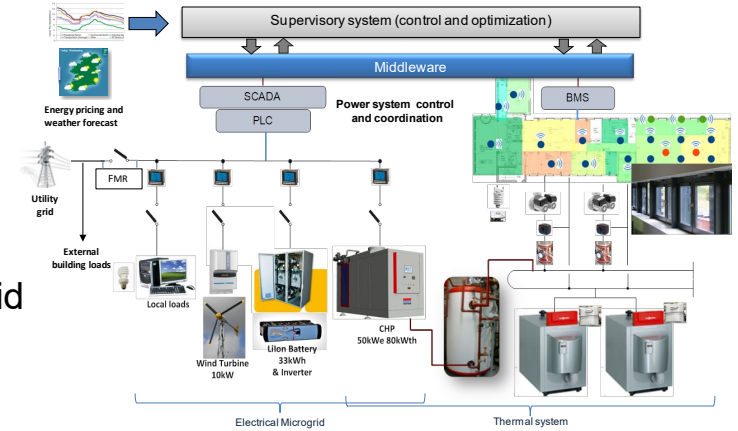
35kWh (85kW peak) Li-ion battery

50kW electrical/82kW thermal combined heat and power unit (CHP) and

Feeder management relay to manage the point of coupling between the microgrid and the rest of the building, and a set of local loads.

Battery and wind turbine interfaced through power electronics converters

CHP with synchronous machine



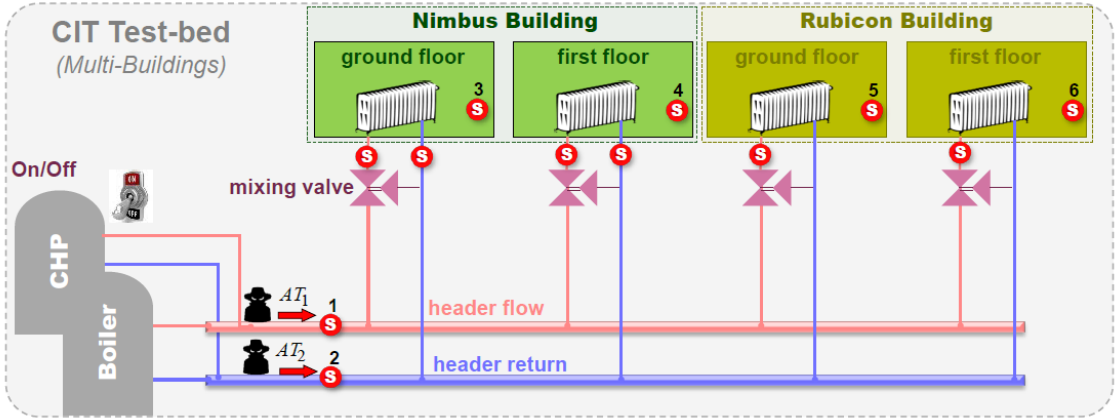
IT System

Interlinked Building Management System and Microgrid SCADA

Three-layer control systems

UTRC Middleware

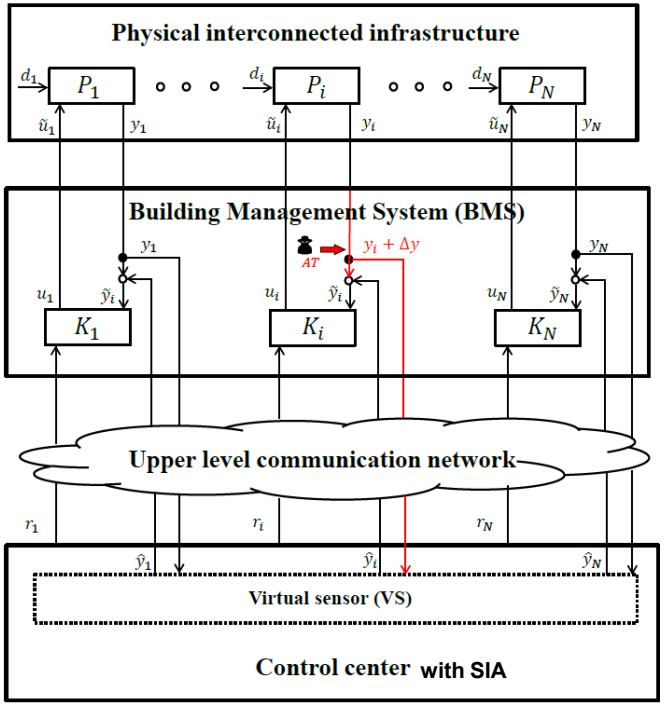
Attack Scenario: NIMBUS Microgrid



- **Adversary:** Infect some field devices with malware (á la Stuxnet) corrupting measurements sent to PLCs (Here: AT_1 and AT_2)

- **Defender:** Access to remote correlated measurements and a physical model (here temp. measurements and modeling by system identification)

Resilient Monitoring and Control



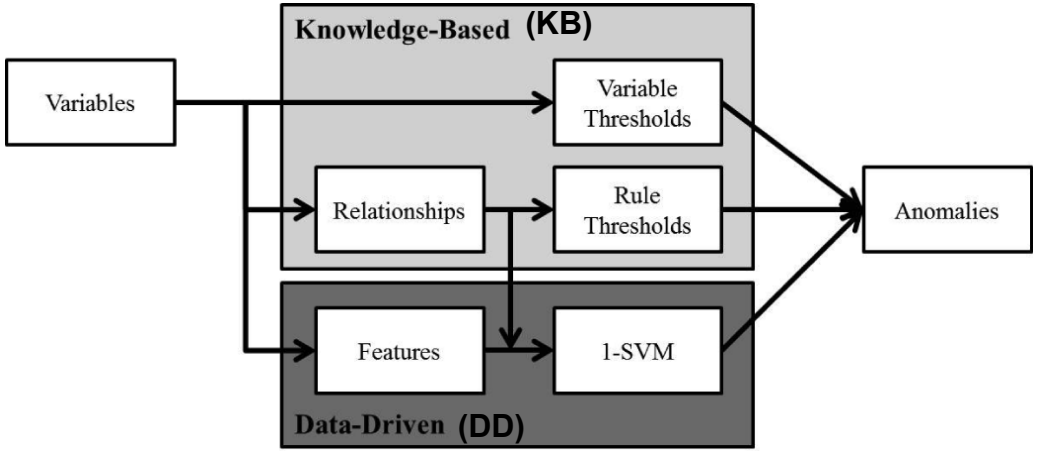
1) Anomaly detector in control center detects attacked measurement $y_i + \Delta y$

2) Optimal physics-based prediction $\hat{y}_i$ from **un-attacked** measurements $y_1, .., y_N$ (VS)

3) Feed $\hat{y}_i$ back to PLCs

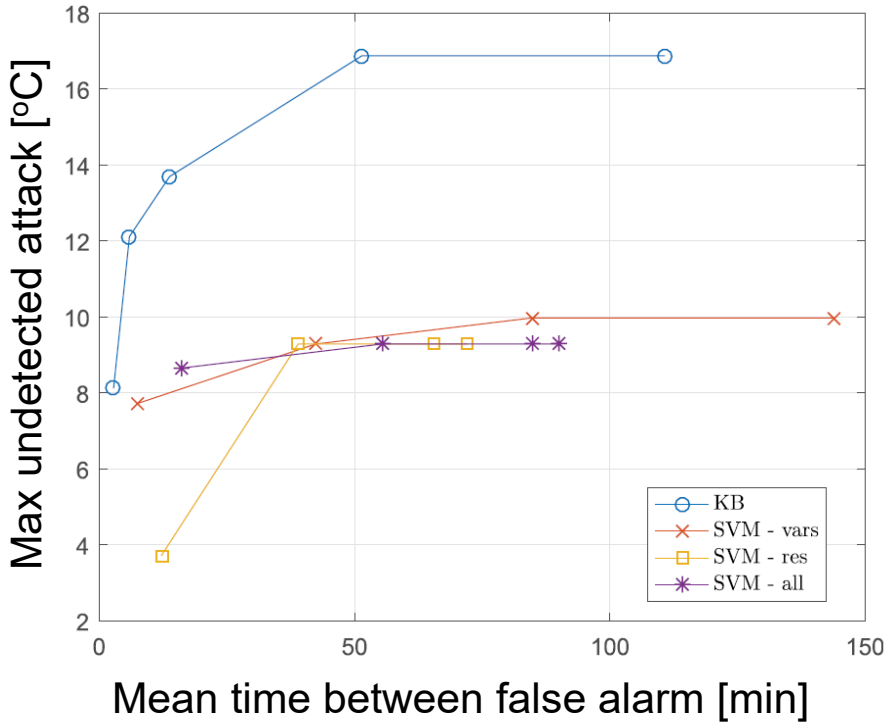
[Paridari, *et al.,* "A Framework for Attack-resilient Industrial Control Systems", Proc. IEEE, 2018]

1) Model- and Data-Based Anomaly Detector



- KB Relationships: Physics-based model predictions
- DD Features: 1) Raw data, 2) KB residues, 3) Windowed mean and standard deviations
- Healthy data used to train 1-SVM

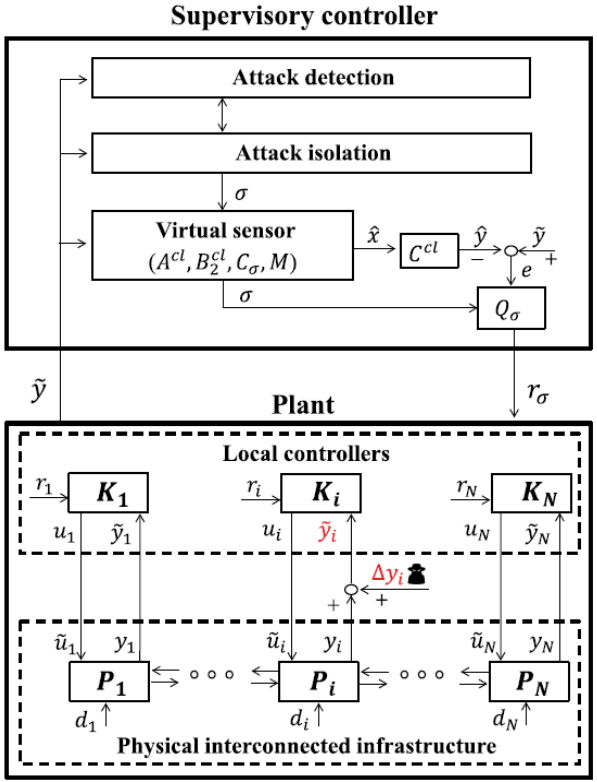
Test Results: Attack Detection



- DD (data-based) detector restricts attacker more
- KB (model-based) detector only checks “physicality” of time series
- DD (data-based) detector also checks for unusual operation

Metric proposed in [Urbina *et al.*, “Limiting the Limiting the Impact of Stealthy Attacks on Industrial Control Systems”, ACM CCS’16, 2016]

2-3) Reconfigured Control System



- Virtual sensor: Model-based, switched Kalman filter

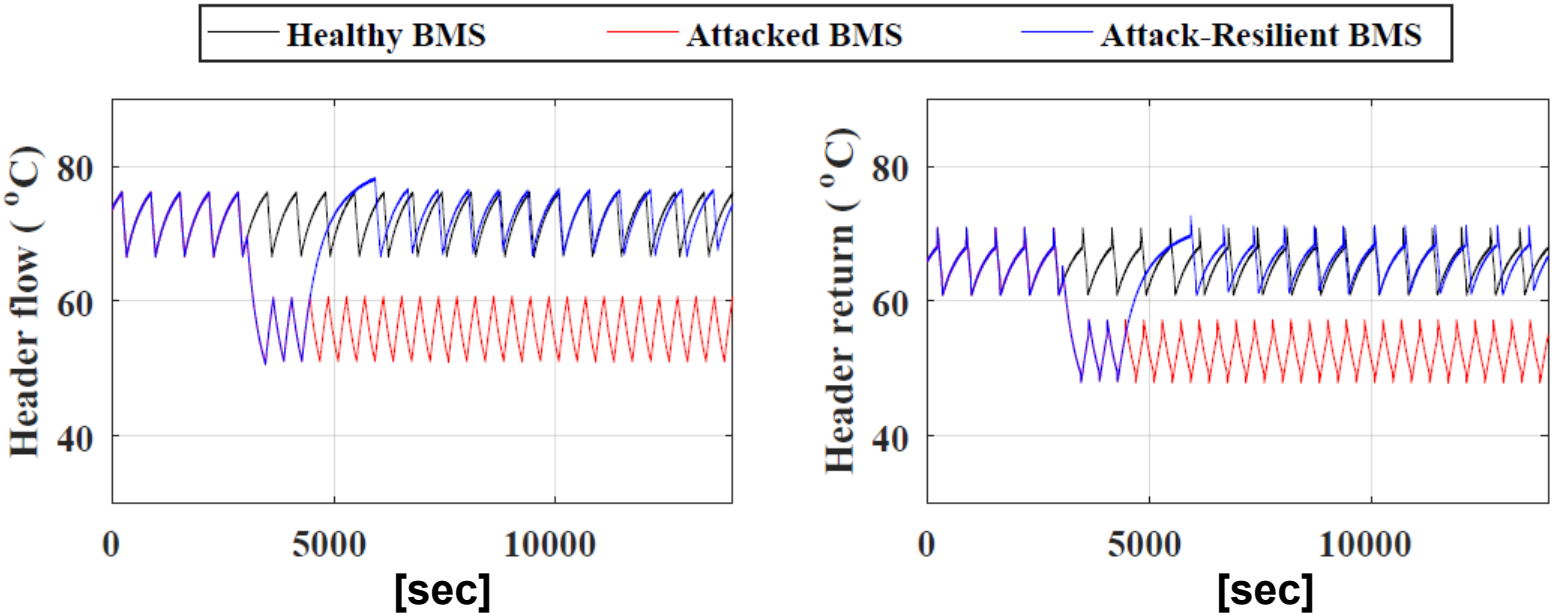
$$\hat{x}(k+1|k) = A^{cl}\hat{x}(k|k-1) + K_\sigma(k)\underbrace{\left[y_\sigma(k) - C_\sigma\hat{x}(k|k-1)\right]}_{\varepsilon(k)}$$

- Attack isolation chooses system mode $\sigma(k) \in \{1,2,\dots,M\}$
  - $\sigma = 1$: All sensors OK
  - $\sigma = 2$: Sensor 1 malfunction
  - ⋮
  - $\sigma = M$: Only trusted Sensor(s) OK

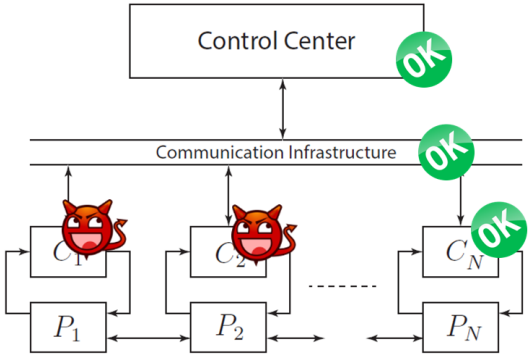
- Healthy sensors used to optimally correct unhealthy sensors, and signal the correction $r_\sigma(k)$ to affected Local Controllers

Test Results: Control Performance

24 min delay in anomaly detector (“attacker free time”):



Case Study 3: Summary



- Data-Driven and Knowledge-Based models, and trusted sensor(s) used for
  - Attack/fault detection and correction in untrusted low-level controllers
  - Gracefully degraded real-time control performance under identified fault/attack conditions → Resilience
  - Degraded performance due to increased time-delay and noise in feedback loops

- Requirements
  - Trusted control center and communication infrastructure
  - Control center has authority to replace local sensor measurements
  - Access to accurate healthy data for SVM training

Takeaways Part I

- Security increasingly important in control and monitoring systems
  - Dramatic attacks reported in media: power, water, gas, process industry
  - Critical infrastructures, legacy components, heterogeneous infrastructures
  - Dedicated malware since more than 10 years, advanced persistent threats
  - *IT security necessary, not sufficient. Think defense in depth!*
- Careful and repeated risk analysis identifying the most relevant attacks is good starting point for secure control design
- Careful attacker and operator modeling necessary – Many tools and solutions very sensitive to changes in the agents' resources
- Three case studies presented and illustrated relevant aspects
- Dynamics matter: can be exploited by both attacker and defender
- Part II goes into some depth regarding attack modeling and anomaly detection

Cyber-Physical Security in Energy Systems

Part II: Attack Modeling and Detection Methods

DTU PES Summer School 2026

Henrik Sandberg (hsan@kth.se)

KTH EECS, Decision and Control Systems

Outline

- DoS attack modeling and dynamic instability
- FDI attack modeling and detection
  - Unknown state and input estimation
  - CUSUM
  - PCA

- Main references:

  - [S. Sundaram, “Fault-Tolerant and Secure Control Systems,” Lecture Notes, 2010: https://engineering.purdue.edu/~sundara2/misc/ft_control_lecture_notes.pdf]

  - [H. Sandberg, Lecture notes, EL2850, KTH, 2025]

Example 1: DoS Attack in F-8 Longitudinal Dynamics

2.3.2 Longitudinal Dynamics of an F-8 Aircraft

Consider a model for the (sampled) linearized longitudinal dynamics of an F-8 aircraft [87]:

$$\underbrace{\begin{bmatrix} V[k+1] \\ \gamma[k+1] \\ \alpha[k+1] \\ q[k+1] \end{bmatrix}}_{\mathbf{x}[k+1]} = \begin{bmatrix} 0.9987 & -3.2178 & -4.4793 & -0.2220 \\ 0 & 1 & 0.1126 & 0.0057 \\ 0 & 0 & 0.8454 & 0.0897 \\ 0.0001 & -0.0001 & -0.8080 & 0.8942 \end{bmatrix} \underbrace{\begin{bmatrix} V[k] \\ \gamma[k] \\ \alpha[k] \\ q[k] \end{bmatrix}}_{\mathbf{x}[k]} + \begin{bmatrix} -0.0329 \\ 0.0131 \\ -0.0137 \\ -0.0092 \end{bmatrix} \mathbf{u}[k], \quad (2.4)$$

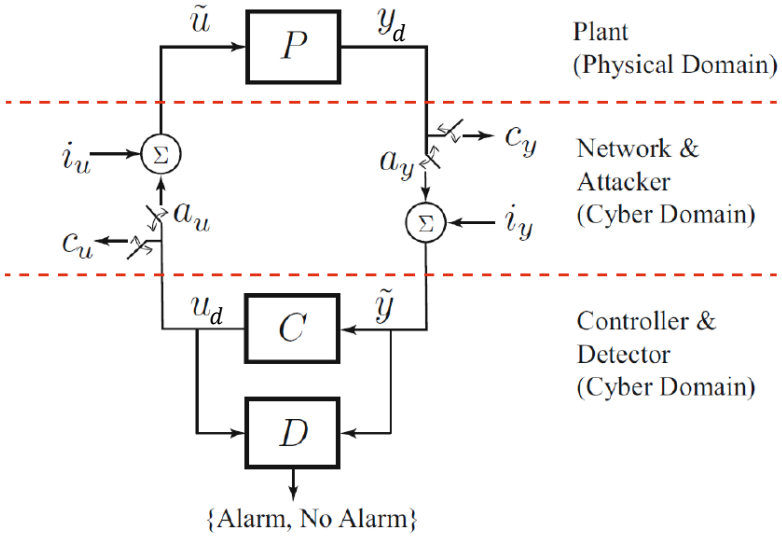
where $V[k]$ is the velocity of the aircraft, $\gamma[k]$ is the flight-path angle, $\alpha[k]$ is the angle-of-attack, and $q[k]$ is the pitch rate. The input to the aircraft is taken to be the deflection of the elevator flaps.

The output of the system will depend on the sensors that are installed on the aircraft. For example, if there is a sensor to measure the velocity and the pitch rate, the output would be given by

$$\mathbf{y}[k] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}[k].$$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Attack Model: Denial-of-Service (DoS) on Actuator Channel



[H. Sandberg, "Cyber-physical security," in Encyclopedia of Systems and Control, 2020]
[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Corollary 6.1)]

Confidentiality attack: disclosure (c_u, c_y)
Integrity attack: false data injection (i_u, i_y)
Availability attack: DoS attack (a_u, a_y)

Assume plant P can be modeled by linear discrete-time state-space model (time index $k \geq 0$):

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k]$$

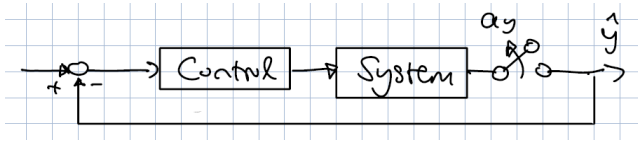
$$\mathbf{u}[k] = \begin{cases} -\mathbf{K}\mathbf{x}[k], & \text{no drop with probability } p, \\ 0, & \text{drop with probability } q := 1-p \end{cases}$$

Theorem: Suppose $\mathbf{B}$ is square and has full rank. The expected value of the state $\mathbf{x}[k]$ is bounded, iff

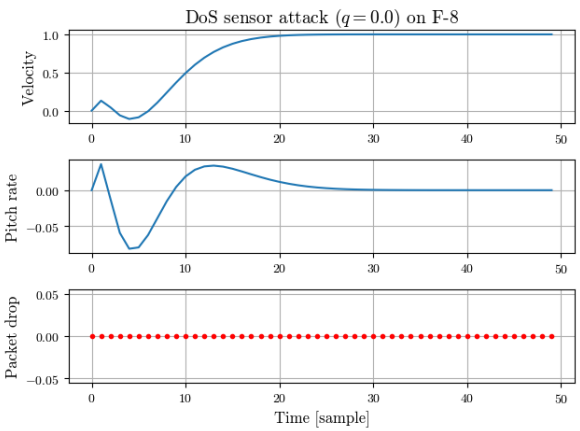
$$\rho(\mathbf{A})^2 q < 1$$

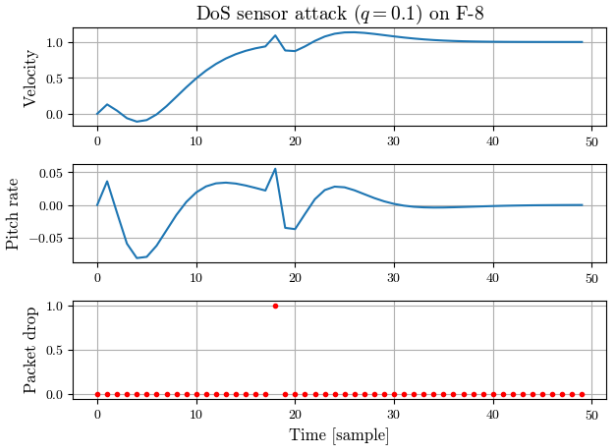
($\rho(\mathbf{A})$ is largest magnitude of eigenvalues of $\mathbf{A}$)

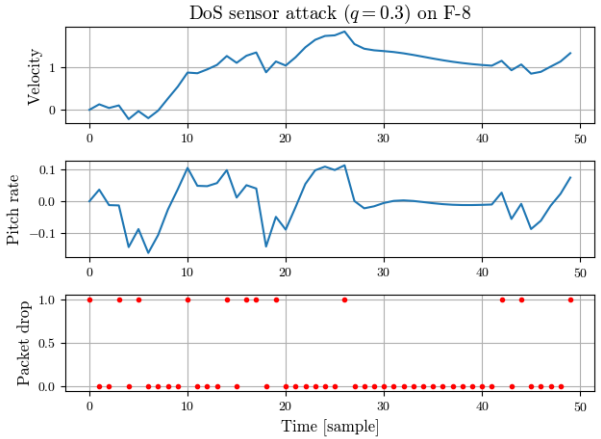
Inherently unstable plants P are much more sensitive to a few dropped control commands. Stable plants lose performance, but *not* stability



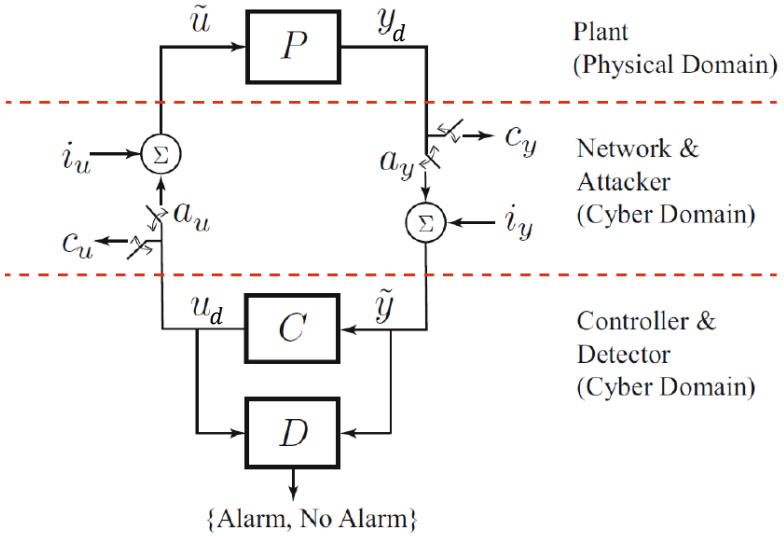
Example 1: DoS Attack in F-8 Longitudinal Dynamics







Attack Model: False Data Injection (FDI)



Confidentiality attack: disclosure (c_u, c_y)
Integrity attack: false data injection (i_u, i_y)
Availability attack: DoS attack (a_u, a_y)

Assume plant P can be modeled by linear discrete-time state-space model (time index $k \geq 0$):

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}'\tilde{\mathbf{u}}[k], \qquad \mathbf{x}[k] \in \mathbb{R}^n,\ \tilde{\mathbf{u}}[k] \in \mathbb{R}^{m'}$$
$$\tilde{\mathbf{y}}[k] = \mathbf{C}\mathbf{x}[k] + \mathbf{D}'\tilde{\mathbf{u}}[k] + \mathbf{i}_\mathbf{y}[k], \qquad \tilde{\mathbf{y}}[k],\ \mathbf{i}_\mathbf{y}[k] \in \mathbb{R}^p$$

Sensors and actuators can be corrupted by (i_u, i_y)

Problem: How, and when, can we design a Detector that alarms when $(i_u, i_y) \neq 0$?

[H. Sandberg, "Cyber-physical security," in Encyclopedia of Systems and Control, 2020]
[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

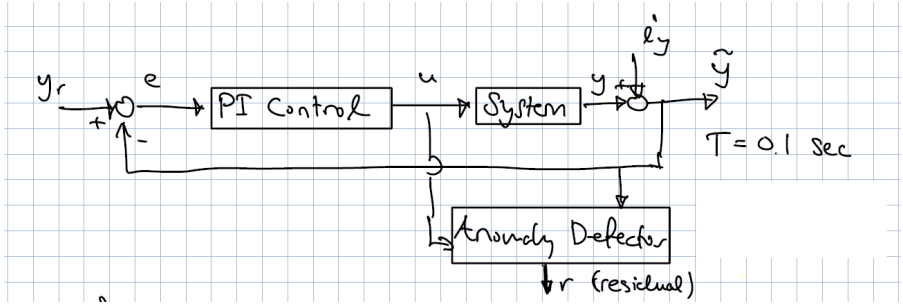
Example 2: FDI Attack in Car Speed Control

2.3.1 A Simple Model for a Car

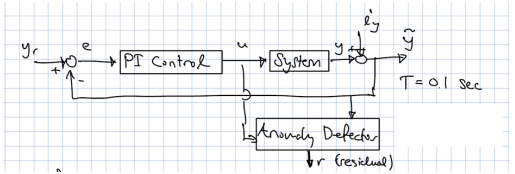
Consider again the model of the car from Chapter 1, with speed $v(t)$ at any time t, and acceleration input $a(t)$. Along the lines of Example 1.1, suppose that we *sample* the velocity of the car every T seconds, and that the acceleration is held constant between sampling times. We can then write

$$v[k+1] = v[k] + Ta[k], \tag{2.1}$$

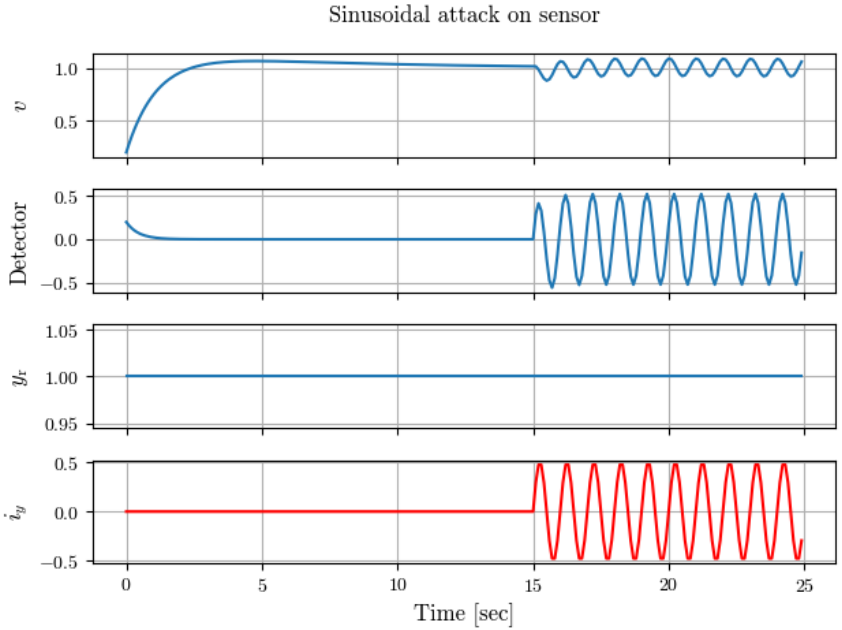
where $v[k]$ is shorthand for $v(kT), k \in \mathbb{N}$, and $a[k]$ is the acceleration that is applied at time $t = kT$. Now, suppose that we wish to also consider the amount of fuel in the car at

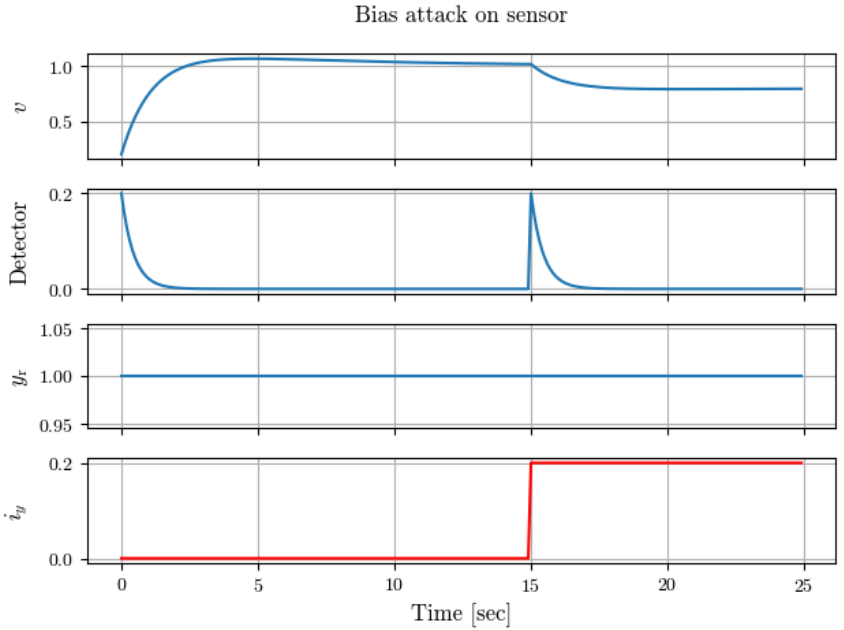


[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]



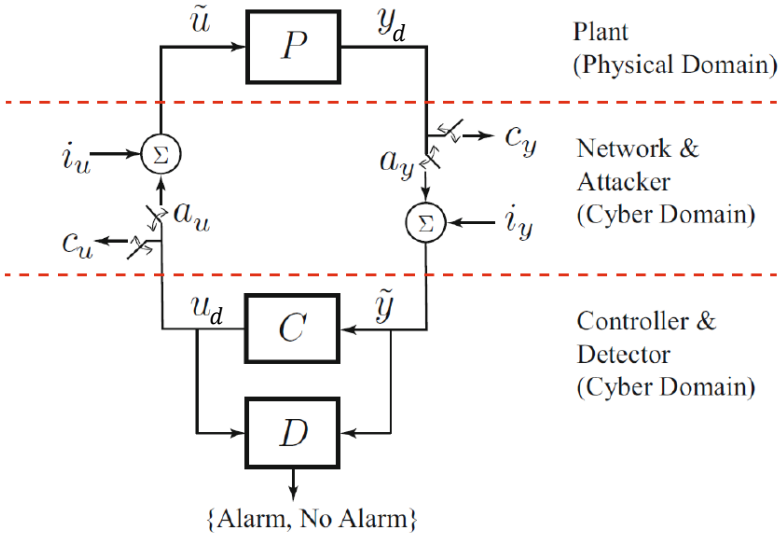
Example 2: FDI Attack in Car Speed Control





When can Detector estimate i_y?

Attack Model: False Data Injection (FDI)



Confidentiality attack: disclosure (c_u, c_y)
Integrity attack: false data injection (i_u, i_y)
Availability attack: DoS attack (a_u, a_y)

Assume plant P can be modeled by linear discrete-time state-space model (time index $k \geq 0$):

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}'\tilde{\mathbf{u}}[k], \qquad \mathbf{x}[k] \in \mathbb{R}^n,\ \tilde{\mathbf{u}}[k] \in \mathbb{R}^{m'}$$
$$\tilde{\mathbf{y}}[k] = \mathbf{C}\mathbf{x}[k] + \mathbf{D}'\tilde{\mathbf{u}}[k] + \mathbf{i_y}[k], \qquad \tilde{\mathbf{y}}[k],\ \mathbf{i_y}[k] \in \mathbb{R}^p$$

Sensors and actuators can be corrupted by (i_u, i_y)

Problem: How, and when, can we design a Detector that alarms when $(i_u, i_y) \neq 0$?

[H. Sandberg, "Cyber-physical security," in Encyclopedia of Systems and Control, 2020]
[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Assumptions on Detector

- Assume desired control u_d known
- Use linearity and subtract the desired control from the model, and collect *all unknown perturbations in a new attack signal* $\mathbf{u}[k] = (\mathbf{i_u}[k], \mathbf{i_y}[k]) \in \mathbb{R}^m$
- New model

$$\begin{aligned} \mathbf{x}[k+1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k], & \mathbf{x}[k] \in \mathbb{R}^n,\ \mathbf{u}[k] \in \mathbb{R}^m \\ \mathbf{y}[k] &= \mathbf{C}\mathbf{x}[k] + \mathbf{D}\mathbf{u}[k], & \mathbf{y}[k] \in \mathbb{R}^p \end{aligned}$$

- **Known (typically):** Model $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ and output $\mathbf{y}[k], k \geq 0$
- **Unknown (typically):** Input $\mathbf{u}[k]$ and state $\mathbf{x}[k]$, $k \geq 0$

Reformulated problem: Alarm as soon as possible when $\mathbf{u}[k] \neq 0$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Prototypical Estimation Problems

- Depending on assumptions on what variables are known and unknown, several estimation problems can be posed, and solved using linear algebra:
  - Problem 1: Estimation of $\mathbf{x}$ from $\mathbf{y}$ and $\mathbf{u}$ (*observability*)
  - Problem 2: Estimation of $\mathbf{u}$ from $\mathbf{y}$ and $\mathbf{x}[0]$ (*invertibility*)
  - Problem 3: Estimation of $\mathbf{u}$ and $\mathbf{x}$ from $\mathbf{y}$ (*strong observability*)
  - Problem 4: Asymptotic Estimation of $\mathbf{u}$ and $\mathbf{x}$ from $\mathbf{y}$ (*strong detectability*)

Useful Result from Linear Algebra

Consider the model ($\mathbf{y}$ received measurement, $\mathbf{x}$ unknown state, $\mathbf{u}$ unknown attack):

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}, \quad \mathbf{y} \in \mathbb{R}^p, \mathbf{x} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m$$

(U1): There is at least one exact solution $\mathbf{x}, \mathbf{u}$ iff $\operatorname{rank}\begin{bmatrix}\mathbf{C} & \mathbf{D} & \mathbf{y}\end{bmatrix} = \operatorname{rank}\begin{bmatrix}\mathbf{C} & \mathbf{D}\end{bmatrix}$

(Compute one solution using $[\mathbf{x}\ \mathbf{u}]^\top = [\mathbf{C}\ \mathbf{D}]^+\mathbf{y}$, where + denotes a pseudoinverse)

(U2): The solution $\mathbf{u}$ is unique iff $\operatorname{rank}\begin{bmatrix}\mathbf{C} & \mathbf{D}\end{bmatrix} = \operatorname{rank}\begin{bmatrix}\mathbf{C}\end{bmatrix} + m$

- Sample applications:
  - (**U1** - anomaly detection): Is $\mathbf{y}$ consistent with the state and attack model?
  - (**U2** - attack reconstruction): Can I uniquely reconstruct the attack $\mathbf{u}$ from $\mathbf{y}$?

From Dynamics to a Linear Equation

$$\boxed{\begin{aligned}\mathbf{x}[k+1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k] \\ \mathbf{y}[k] &= \mathbf{C}\mathbf{x}[k] + \mathbf{D}\mathbf{u}[k]\end{aligned}}$$

$$\mathbf{x}[k+1] = \mathbf{A}^{k+1}\mathbf{x}[0] + \sum_{i=0}^{k} \mathbf{A}^{k-i}\mathbf{B}\mathbf{u}[i]$$

$$= \mathbf{A}^{k+1}\mathbf{x}[0] + \underbrace{\begin{bmatrix}\mathbf{A}^k\mathbf{B} & \mathbf{A}^{k-1}\mathbf{B} & \dots & \mathbf{A}\mathbf{B} & \mathbf{B}\end{bmatrix}}_{\mathcal{C}_k} \underbrace{\begin{bmatrix}\mathbf{u}[0] \\ \mathbf{u}[1] \\ \vdots \\ \mathbf{u}[k-1] \\ \mathbf{u}[k]\end{bmatrix}}_{\mathbf{u}[0:k]}$$

$$\equiv \mathbf{A}^{k+1}\mathbf{x}[0] + \mathcal{C}_k\mathbf{u}[0:k].$$

Problem 1: Estimation of $\mathbf{x}$ from $\mathbf{y}$ and $\mathbf{u}$

- State estimation: estimate unknown state from known input and output
- Stack data over time horizon $[0, L]$:

$$\underbrace{\begin{bmatrix}\mathbf{y}[0]\\ \mathbf{y}[1]\\ \mathbf{y}[2]\\ \vdots\\ \mathbf{y}[L]\end{bmatrix}}_{\mathbf{y}[0:L]} = \underbrace{\begin{bmatrix}\mathbf{C}\\ \mathbf{CA}\\ \mathbf{CA}^2\\ \vdots\\ \mathbf{CA}^L\end{bmatrix}}_{\mathcal{O}_L} \mathbf{x}[0] + \underbrace{\begin{bmatrix}\mathbf{D} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \mathbf{CA}^{L-1}\mathbf{B} & \mathbf{CA}^{L-2}\mathbf{B} & \mathbf{CA}^{L-3}\mathbf{B} & \cdots & \mathbf{D}\end{bmatrix}}_{\mathcal{J}_L} \underbrace{\begin{bmatrix}\mathbf{u}[0]\\ \mathbf{u}[1]\\ \mathbf{u}[2]\\ \vdots\\ \mathbf{u}[L]\end{bmatrix}}_{\mathbf{u}[0:L]}$$

with *observability matrix* $\mathcal{O}_L$ and *invertibility matrix* $\mathcal{J}_L$

- Rearranging we obtain $\mathbf{y}[0:L] - \mathcal{J}_L \mathbf{u}[0:L] = \mathcal{O}_L \mathbf{x}[0]$ and we can uniquely solve for $\mathbf{x}[0]$ iff $\operatorname{rank} \mathcal{O}_L = n$ (system is *observable;* use **(U2)**)
- **Theorem:** System is observable iff $\operatorname{rank} \mathcal{O}_{n-\operatorname{rank}(\mathbf{C})} = n$
- **Remark:** There is a delay $L \leq n - \operatorname{rank}(\mathbf{C})$ before we can estimate the state $\mathbf{x}[0]$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Example 3: Observability

Example 2.4. Consider the pair $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$, $\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix}$ $\mathbf{x}$, with no inputs to the system. The observability matrix for this pair is

$$\mathcal{O}_{n-\text{rank}(\mathbf{C})} = \mathcal{O}_1 = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix},$$

which has rank 2, and thus the pair is observable. The initial state of the system can be recovered as follows:

$$\mathbf{y}[0:1] = \mathcal{O}_1 \mathbf{x}[0] \quad \Rightarrow \mathcal{O}_1^{-1} \mathbf{y}[0:1] = \mathbf{x}[0].$$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Problem 2: Estimation of $\mathbf{u}$ from $\mathbf{y}$ and $\mathbf{x}[0]$

- Input estimation: estimate unknown input from known input and state
- Stack data over time horizon $[0, L]$:

$$\underbrace{\begin{bmatrix}\mathbf{y}[0]\\ \mathbf{y}[1]\\ \mathbf{y}[2]\\ \vdots\\ \mathbf{y}[L]\end{bmatrix}}_{\mathbf{y}[0:L]} = \underbrace{\begin{bmatrix}\mathbf{C}\\ \mathbf{CA}\\ \mathbf{CA}^2\\ \vdots\\ \mathbf{CA}^L\end{bmatrix}}_{\mathcal{O}_L} \mathbf{x}[0] + \underbrace{\begin{bmatrix}\mathbf{D} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \mathbf{CA}^{L-1}\mathbf{B} & \mathbf{CA}^{L-2}\mathbf{B} & \mathbf{CA}^{L-3}\mathbf{B} & \cdots & \mathbf{D}\end{bmatrix}}_{\mathcal{J}_L} \underbrace{\begin{bmatrix}\mathbf{u}[0]\\ \mathbf{u}[1]\\ \mathbf{u}[2]\\ \vdots\\ \mathbf{u}[L]\end{bmatrix}}_{\mathbf{u}[0:L]}$$

$\mathcal{J}_{L-1}$

with *observability matrix* $\mathcal{O}_L$ and *invertibility matrix* $\mathcal{J}_L$

- Rearranging we obtain $\mathbf{y}[0:L] - \mathcal{O}_L\mathbf{x}[0] = \mathcal{J}_L\mathbf{u}[0:L]$ and can uniquely solve for $\mathbf{u}[0]$ iff $\operatorname{rank}\mathcal{J}_L = \operatorname{rank}\mathcal{J}_{L-1} + m$ (system is *invertible* with delay L; use **(U2)**)
- **Theorem:** If system not invertible for $L \leq n - \operatorname{nullity}(D) + 1$, then it is never invertible
- **Remark:** To determine $\mathbf{u}[1]$, shift the horizon $[0, L] \to [1, L+1]$
- **Lemma:** System is invertible (for some L) iff $\operatorname{rank}\left(\begin{bmatrix}\mathbf{A} - z\mathbf{I}_n & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{bmatrix}\right) = n + m$ for *at least one* $z \in \mathbb{C}$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Example 4: Fault/Attack in F-8 Elevator Angle

Consider a model for the (sampled) linearized longitudinal dynamics of an F-8 aircraft [87]:

$$\underbrace{\begin{bmatrix} V[k+1] \\ \gamma[k+1] \\ \alpha[k+1] \\ q[k+1] \end{bmatrix}}_{\mathbf{x}[k+1]} = \begin{bmatrix} 0.9987 & -3.2178 & -4.4793 & -0.2220 \\ 0 & 1 & 0.1126 & 0.0057 \\ 0 & 0 & 0.8454 & 0.0897 \\ 0.0001 & -0.0001 & -0.8080 & 0.8942 \end{bmatrix} \underbrace{\begin{bmatrix} V[k] \\ \gamma[k] \\ \alpha[k] \\ q[k] \end{bmatrix}}_{\mathbf{x}[k]} + \begin{bmatrix} -0.0329 \\ 0.0131 \\ -0.0137 \\ -0.0092 \end{bmatrix} \mathbf{u}[k], \quad (2.4)$$

where $V[k]$ is the velocity of the aircraft, $\gamma[k]$ is the flight-path angle, $\alpha[k]$ is the angle-c attack, and $q[k]$ is the pitch rate. The input to the aircraft is taken to be the deflectic of the elevator flaps.

The output of the system will depend on the sensors that are installed on the aircraft. For example, if there is a sensor to measure the velocity and the pitch rate, the output would be given by

$$\mathbf{y}[k] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}[k].$$

Suppose there is a fault/attack in the elevator angle: $\tilde{\mathbf{u}}[k] = \mathbf{u}_d[k] + \mathbf{i}_\mathbf{u}[k]$.
Can we reconstruct the fault/attack using only $\mathbf{x}[0], \mathbf{y}_2$, and $\mathbf{u}_\mathrm{d}$?

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Problem 3: Estimation of u and x from y

- Input and state estimation: estimate unknown input and state from known output
- Stack data over time horizon $[0, L]$:

$$\underbrace{\begin{bmatrix} \mathbf{y}[0] \\ \mathbf{y}[1] \\ \mathbf{y}[2] \\ \vdots \\ \mathbf{y}[L] \end{bmatrix}}_{\mathbf{y}[0:L]} = \underbrace{\begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^L \end{bmatrix}}_{\mathcal{O}_L} \mathbf{x}[0] + \underbrace{\begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{L-1}\mathbf{B} & \mathbf{CA}^{L-2}\mathbf{B} & \mathbf{CA}^{L-3}\mathbf{B} & \cdots & \mathbf{D} \end{bmatrix}}_{\mathcal{J}_L} \underbrace{\begin{bmatrix} \mathbf{u}[0] \\ \mathbf{u}[1] \\ \mathbf{u}[2] \\ \vdots \\ \mathbf{u}[L] \end{bmatrix}}_{\mathbf{u}[0:L]}$$

with *observability matrix* $\mathcal{O}_L$ and *invertibility matrix* $\mathcal{J}_L$

- We can uniquely solve for $\mathbf{x}[0]$ iff $\operatorname{rank}[\mathcal{O}_L \ \ \mathcal{J}_L] = \operatorname{rank}\mathcal{J}_L + n$ (system is *strongly observable* with delay L; use **(U2)**)
- **Theorem:** If system not strongly observable for $L \leq n$, then it is never strongly observable
- **Remark:** Once $\mathbf{x}[k]$ and $\mathbf{x}[k+1]$ are found, we solve for $\mathbf{u}[k]$ using $\begin{bmatrix} \mathbf{x}[k+1] - \mathbf{A}\mathbf{x}[k] \\ \mathbf{y}[k] \end{bmatrix} = \begin{bmatrix} \mathbf{B} \\ \mathbf{D} \end{bmatrix} \mathbf{u}[k]$
- **Lemma:** System is strongly observable (for some L) iff $\operatorname{rank}\left(\begin{bmatrix} \mathbf{A} - z\mathbf{I}_n & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}\right) = n + m$ for *all* $z \in \mathbb{C}$ (transfer function matrix has *no zeros*, only poles)

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Example 5: Strong Observability of F-8

$$\mathbf{y}[k] = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}[k]$$

Example 2.7. Consider again the F-8 from Example 2.6. To check whether one can recover the fault input $\mathbf{u}[k]$ can be recovered, regardless of the states of the system, we check whether the system is strongly observable. Specifically, for $L = n$, we have

$$\mathcal{O}_n = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0.0001 & -0.0001 & -0.8080 & 0.8942 \\ 0.0001 & -0.0004 & -1.4058 & 0.7271 \\ 0.0002 & -0.0009 & -1.7765 & 0.5240 \\ 0.0002 & -0.0015 & -1.9261 & 0.3092 \end{bmatrix},$$

$$\mathcal{J}_n = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ -0.0092 & 0 & 0 & 0 & 0 \\ 0.0028 & -0.0092 & 0 & 0 & 0 \\ 0.0125 & 0.0028 & -0.0092 & 0 & 0 \\ 0.0195 & 0.0125 & 0.0028 & -0.0092 & 0 \end{bmatrix}.$$

One can verify that $\text{rank}\left(\begin{bmatrix} \mathcal{O}_n & \mathcal{J}_n \end{bmatrix}\right) - \text{rank}(\mathcal{J}_n) = 1$, and thus the system is not strongly observable.

However, suppose that we also have a sensor that measures the velocity of the aircraft (in addition to the pitch rate). The $\mathbf{C}$ matrix would then become

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

and one can verify that the system is strongly observable in this case. Specifically, one can recover the fault input $\mathbf{u}[k]$ from the output of the system $\mathbf{y}[k : k + n]$ without knowing the initial state of the system.

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapters 2-3)]

Problem 4: Asymptotic Estimation of u and x from y

- If system is *not strongly observable*, we cannot exactly find state and input with a fixed delay using only outputs. *When and how can we asymptotically estimate state and input?*
- If system is invertible with delay L, there is a $\mathbf{P}$ such that $\mathbf{P}\mathcal{J}_L = \begin{bmatrix}\mathbf{I}_m & 0 & \dots & 0\end{bmatrix}$
- We can write the (unknown) input as $\mathbf{P}\mathbf{y}[k:k+L] = \mathbf{P}\mathcal{O}_L\mathbf{x}[k] + \mathbf{u}[k]$
- Eliminate input in the system model

$$\begin{aligned}\mathbf{x}[k+1] &= \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k] \\ &= \mathbf{A}\mathbf{x}[k] - \mathbf{B}\mathbf{P}\mathcal{O}_L\mathbf{x}[k] + \mathbf{B}\mathbf{P}\mathbf{y}[k:k+L] \\ &= (\mathbf{A} - \mathbf{B}\mathbf{P}\mathcal{O}_L)\mathbf{x}[k] + \mathbf{B}\mathbf{P}\mathbf{y}[k:k+L]\end{aligned}$$

- Replace $\mathbf{x}$ with a state estimate $\hat{\mathbf{x}}$ leading to an *unknown input observer* (UIO):

$$\hat{\mathbf{x}}[k+1] = (\mathbf{A} - \mathbf{B}\mathbf{P}\mathcal{O}_L)\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{P}\mathbf{y}[k:k+L]$$

- When do we have $\hat{\mathbf{x}}[k] \to \mathbf{x}[k]$ for any $\hat{\mathbf{x}}[0]$? Iff there is a $\mathbf{P}$ such that all eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{P}\mathcal{O}_L$ are inside the unit disc ($\rho(\mathbf{A} - \mathbf{B}\mathbf{P}\mathcal{O}_L) < 1$)! (System is *strongly detectable* with delay L)
- **Lemma:** System is strongly detectable (for some L) iff $\operatorname{rank}\left(\begin{bmatrix}\mathbf{A} - z\mathbf{I}_n & \mathbf{B} \\ \mathbf{C} & \mathbf{D}\end{bmatrix}\right) = n + m$ for *all* $z \in \mathbb{C}$ such that $|z| \geq 1$ (transfer function matrix has *no zeros on or outside of the disc*. The system is *minimum phase)*
- **Remark:** Asymptotically estimate the attack using $\mathbf{P}\mathbf{y}[k:k+L] - \mathbf{P}\mathcal{O}_L\hat{\mathbf{x}}[k] = \hat{\mathbf{u}}[k]$

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapter 3)]

Comparison to Regular (Luenberger) State Observer

- Consider the system model

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k]$$
$$\mathbf{y}[k] = \mathbf{C}\mathbf{x}[k]$$

with *known* input $\mathbf{u}[k]$

- For any filter gain $\mathbf{L}$, add output injection

$$\mathbf{x}[k+1] = \mathbf{A}\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}\underbrace{(\mathbf{y}[k] - \mathbf{C}\mathbf{x}[k])}_{\equiv 0}$$
$$= (\mathbf{A} - \mathbf{L}\mathbf{C})\mathbf{x}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}\mathbf{y}[k]$$

- Replace $\mathbf{x}$ with a state estimate $\hat{\mathbf{x}}$ leading to the state observer

$$\hat{\mathbf{x}}[k+1] = (\mathbf{A} - \mathbf{L}\mathbf{C})\hat{\mathbf{x}}[k] + \mathbf{B}\mathbf{u}[k] + \mathbf{L}\mathbf{y}[k]$$

- When do we have $\hat{\mathbf{x}}[k] \to \mathbf{x}[k]$ for any $\hat{\mathbf{x}}[0]$? Iff there is an $\mathbf{L}$ such that all eigenvalues of $\mathbf{A} - \mathbf{L}\mathbf{C}$ are inside the unit disc ($\rho(\mathbf{A} - \mathbf{L}\mathbf{C}) < 1$)! (Model $(\mathbf{A}, \mathbf{C})$ is *detectable*)

Problem 4: Asymptotic Estimation of u and x from y

- To find an UIO in practice, first solve

$$\mathbf{P}\mathcal{J}_L = \begin{bmatrix}\mathbf{I}_m & 0 & \dots & 0\end{bmatrix} \quad (\mathbf{P} = \begin{bmatrix}\mathbf{I}_m & 0 & \dots & 0\end{bmatrix}\mathcal{J}_L^+)$$
$$\mathbf{N}\mathcal{J}_L = \begin{bmatrix}0 & 0 & \dots & 0\end{bmatrix}$$

where $\mathbf{N}$ is a basis of the left nullspace of $\mathcal{J}_L$

- Note that for any filter gain $\mathbf{G}$ of appropriate dimensions

$$(\mathbf{BP} + \mathbf{GN})(\mathbf{y}[k:k+L] - \mathcal{O}_L\mathbf{x}[k]) = \mathbf{Bu}[k]$$

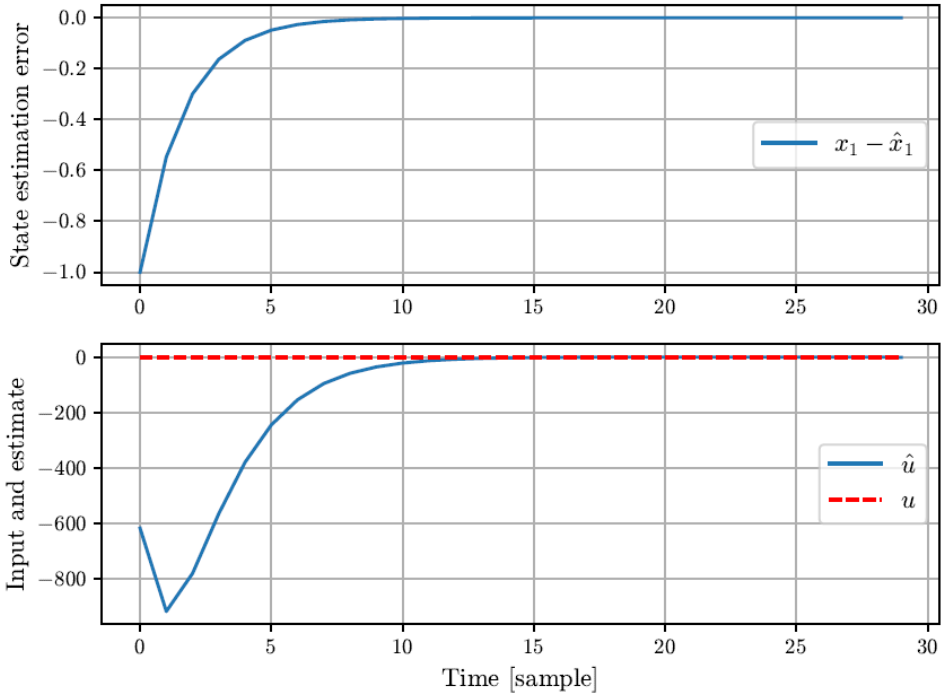
- Then an UIO is given by

$$\hat{\mathbf{x}}[k+1] = (\mathbf{A} - \mathbf{BP}\mathcal{O}_L - \mathbf{GN}\mathcal{O}_L)\hat{\mathbf{x}}[k] + (\mathbf{BP} + \mathbf{GN})\mathbf{y}[k:k+L]$$

- If model $(\mathbf{A} - \mathbf{BP}\mathcal{O}_L, \mathbf{N}\mathcal{O}_L)$ is *detectable* there is a gain $\mathbf{G}$ such that $\rho(\mathbf{A} - \mathbf{BP}\mathcal{O}_L - \mathbf{GN}\mathcal{O}_L) < 1$. Equivalent(!) to *no zeros on or outside of the disc*
- Compare gain $\mathbf{G}$ with $\mathbf{L}$!

[S. Sundaram, "Fault-Tolerant and Secure Control Systems," Lecture Notes, 2010 (see Chapter 3)]

Example 6: Estimating Unknown Input and State in F-8 Airplane from Outputs (with delay $L = 1$)



Outline

- DoS attack modeling and dynamic instability
- FDI attack modeling and detection
  - Unknown state and input estimation
  - **CUSUM**
  - PCA

- Main references:

  - [S. Sundaram, “Fault-Tolerant and Secure Control Systems,” Lecture Notes, 2010: https://engineering.purdue.edu/~sundara2/misc/ft_control_lecture_notes.pdf]

  - [H. Sandberg, Lecture notes, EL2850, KTH, 2025]

What If There Is Stochastic Noise? Use CUSUM!

- What if measurements are corrupted by stochastic noise? Leads to a noisy residual signal

$$\mathbf{r}[k] := \hat{\mathbf{u}}[k] = \mathbf{u}[k] + \epsilon[k]$$

Suppose $\epsilon[k]$ is zero-mean i.i.d. (independent, identically distributed) stochastic noise

- A simple threshold $\eta > 0$ alarm test:

$$\|\mathbf{r}[k]\|_2^2 \underset{\text{No Alarm}}{\overset{\text{Alarm}}{\gtrless}} \eta$$

generally has a *bad trade-off between false and true alarm rates.* [**Stateless test**]

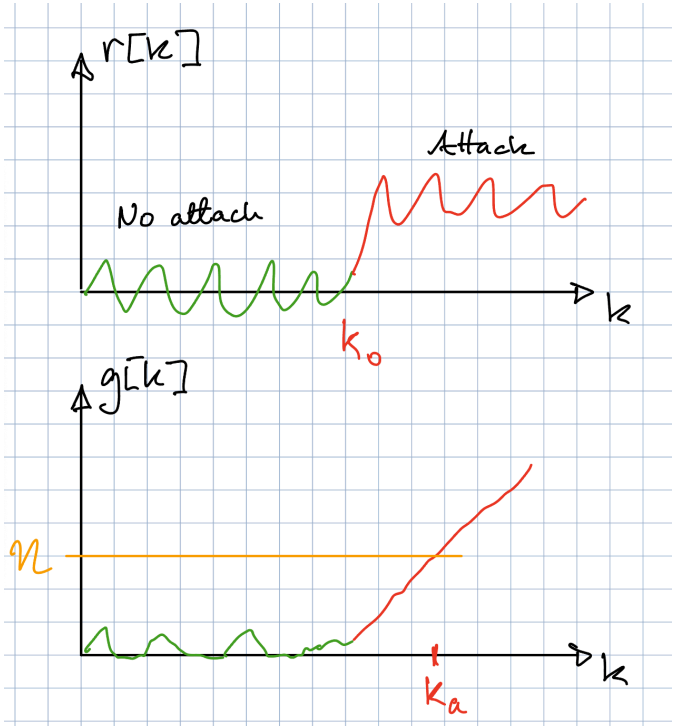
- Cumulative Sum (CUSUM) tests generally generate *far fewer false alarms* and still provide *short delay until true alarm* (sometimes optimal test structure):

$$g[k] = (g[k-1] + \|\mathbf{r}[k]\|_2^2 - \delta)^+ := \max\{0, g[k-1] + \|\mathbf{r}[k]\|_2^2 - \delta\}$$
$$g[-1] = 0$$
$$g[k] \underset{\text{No Alarm}}{\overset{\text{Alarm}}{\gtrless}} \eta$$

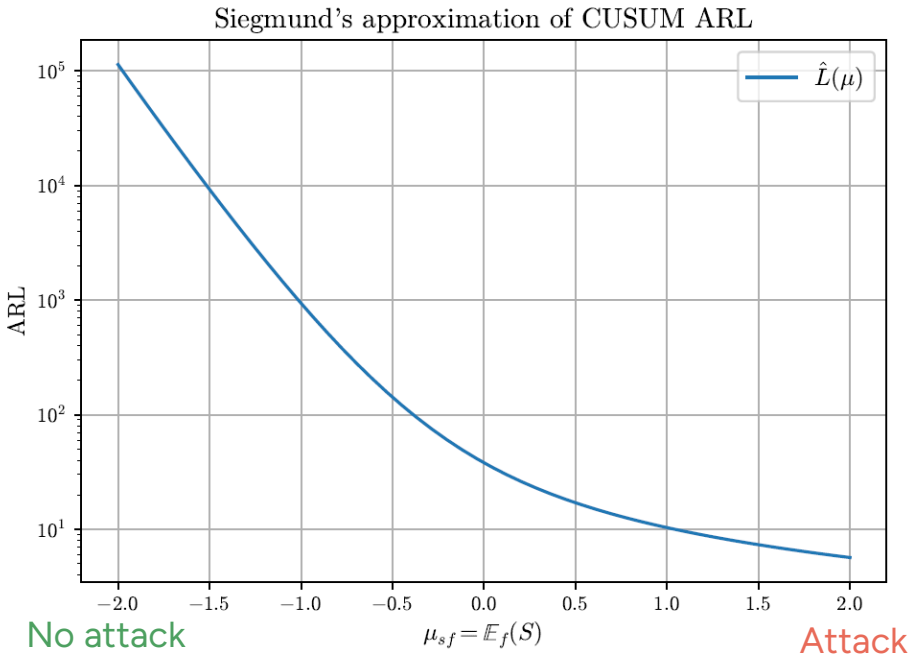
where $\delta > 0$ is a tunable bias. [**Stateful test**]

[J, Giraldo, *et al.*, "A Survey of Physics-Based Attack Detection in Cyber-Physical Systems." ACM Computing Surveys, 2018]
[H. Sandberg, "Lecture 13: Quickest Change Detection and Data-Based Methods," Lecture notes, EL2850, KTH, 2025]

CUSUM Performance



ARL = Average Run Length (mean #samples to alarm)



Mean #samples between false alarms: $\sim 10^5$
Mean #samples alarm delay after onset attack: ~ 5

[H. Sandberg, "Lecture 13: Quickest Change Detection and Data-Based Methods," Lecture notes, EL2850, KTH, 2025]

Outline

- DoS attack modeling and dynamic instability
- FDI attack modeling and detection
  - Unknown state and input estimation
  - CUSUM
  - **PCA**

- Main references:

  - [S. Sundaram, “Fault-Tolerant and Secure Control Systems,” Lecture Notes, 2010: https://engineering.purdue.edu/~sundara2/misc/ft_control_lecture_notes.pdf]

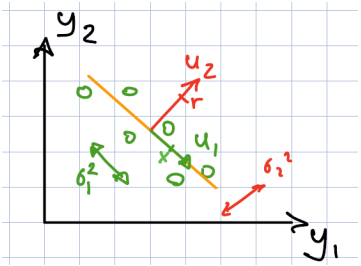
  - [H. Sandberg, Lecture notes, EL2850, KTH, 2025]

What If There Is No Model? Consider PCA!

- Results so far have assumed model $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is known. What if we only have collected input/output data?
- **Indirect approach:** Estimate $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ first using system identification (subspace, ML, etc.)
- **Direct approach:** Large number of data-driven methods available (autoencoders, **PCA**, etc.)
- **Principal Component Analysis (PCA)** is a classical method from multivariate statistics
- Suppose high-dimensional sample data sequence $\mathbf{y}[0], \mathbf{y}[1], \ldots, \mathbf{y}[L]$ (L and p large) collected
- Suppose nominal data (with no attack/anomalies) lie on a low-dimensional subspace in $\mathbb{R}^p$:
  1. Estimate nominal subspace of dimension $n < p$ that captures the maximum possible variance of the collected data
  2. As new data samples arrive, $\mathbf{y}[k], k > L$, compute distance to nominal subspace and *alarm if distance to subspace is large enough* (use stateless or stateful test)
- **Autoencoders:** Nonlinear generalization of PCA. Assume nominal data lies on low-dimensional (nonlinear) manifold instead of (linear) subspace. Less theory, lots of code available

[H. Sandberg, "Lecture 13: Quickest Change Detection and Data-Based Methods," Lecture notes, EL2850, KTH, 2025]

PCA Formulas



- From samples $\mathbf{y}[0], \mathbf{y}[1], \ldots, \mathbf{y}[L]$, first compute sample mean and center the data

$$\bar{\mathbf{y}} := \frac{1}{N}\sum_{k=0}^{L}\mathbf{y}[k] \qquad \mathbf{z}[k] := \mathbf{y}[k] - \bar{\mathbf{y}}$$

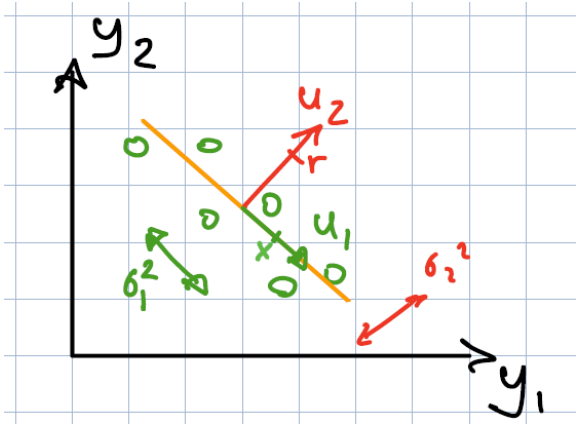
- Compute sample covariance of data

$$\mathbf{Z} := \begin{bmatrix}\mathbf{z}[0] & \mathbf{z}[1] & \ldots & \mathbf{z}[L]\end{bmatrix} \in \mathbb{R}^{p\times N} \qquad \mathbf{\Sigma} := \frac{1}{L}\sum_{k=0}^{L}\mathbf{z}[k]\mathbf{z}[k]^\top = \frac{1}{L}\mathbf{Z}\mathbf{Z}^\top \in \mathbb{R}^{p\times p}$$

- Find the largest directions of variations in data ("principal components") using eigenvectors

$$\mathbf{\Sigma} = \begin{bmatrix}\mathbf{u}_1 & \mathbf{u}_2 & \ldots & \mathbf{u}_p\end{bmatrix}\begin{bmatrix}\sigma_1^2 & & & 0\\ & \sigma_2^2 & & \\ & & \ddots & \\ 0 & & & \sigma_p^2\end{bmatrix}\begin{bmatrix}\mathbf{u}_1^\top\\ \mathbf{u}_2^\top\\ \vdots\\ \mathbf{u}_p^\top\end{bmatrix} \qquad \sigma_1^2 \geq \sigma_2^2 \geq \cdots \geq \sigma_p^2$$

PCA Formulas



- If data mostly lies in a subspace of dimension n, then

$$\sigma_1^2 \geq \cdots \geq \sigma_n^2 \gg \sigma_{n+1}^2 \geq \cdots \geq \sigma_p^2$$

- Projected coordinates in subspace

$$\mathbf{x}[k] := \mathbf{U}_{1:n}^\top \mathbf{z}[k] = \mathbf{U}_{1:n}^\top (\mathbf{y}[k] - \bar{\mathbf{y}}) \in \mathbb{R}^n \qquad \mathbf{U}_{1:n}^\top := \begin{bmatrix} \mathbf{u}_1^\top \\ \vdots \\ \mathbf{u}_n^\top \end{bmatrix} \in \mathbb{R}^{n \times p}$$

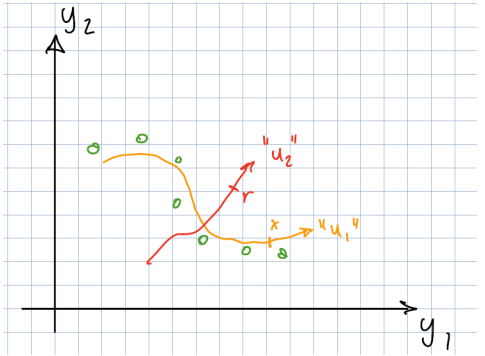
- For new data point, compute residual to check distance to subspace and alarm if large

$$\mathbf{r}[k] := \mathbf{U}_{n+1:p}^\top \mathbf{z}[k] = \mathbf{U}_{n+1:p}^\top (\mathbf{y}[k] - \bar{\mathbf{y}}) \in \mathbb{R}^{p-n}, \quad k > L \qquad \mathbf{U}_{n+1:p}^\top := \begin{bmatrix} \mathbf{u}_{n+1}^\top \\ \vdots \\ \mathbf{u}_p^\top . \end{bmatrix}$$

- Use CUSUM test to curtail number of false alarms

$$g[k] = (g[k-1] + \|\mathbf{r}[k]\|_2^2 - \delta)^+, \quad g[L] := 0$$

Autoencoders and Nonlinear PCA



- **Encoder:** $\mathbf{x}[k] = f_\theta(\mathbf{y}[k])$

where f_θ is (deep) Neural Network (NN) with many trained parameters θ

- **Decoder:** $\hat{\mathbf{y}}[k] = g_\phi(\mathbf{x}[k])$

where g_ϕ is (deep) NN with many trained parameters ϕ

- Parameters trained to minimize reconstruction loss

$$\min_{\theta,\phi} \frac{1}{N}\sum_{k=0}^{L} \text{loss}(\mathbf{y}[k], \hat{\mathbf{y}}[k]) \qquad \text{loss}(\mathbf{y}, \hat{\mathbf{y}}) = \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2$$

- For new data point, alarm if $\text{loss}(y[k], \hat{y}[k]) > \eta$, $k > L$, or use CUSUM test as earlier

- If maps f_θ, g_ϕ linear and loss quadratic: PCA is the optimal solution. *For nonlinear maps, few theoretical guarantees available*

Summary (Part II)

- DoS attack modeling and instability in control systems
- FDI attacks and detection methods
  - Linear algebra reveals which sensors are necessary for detecting specified attack/faults
  - *Dynamics helps* in estimating unknown state and attacks/faults!
  - Linear algebra provides constructive tools for reconstructing attacks/faults
  - Useful tools both for *risk management (offline)* and *operations (online)*
  - Four prototypical estimation/detection problems
- Noisy data: Use threshold tests. Stateful tests (CUSUM) provide (much) better design trade-offs than stateless tests (χ^2-test, for example)
- Indirect vs. direct data-based methods: Powerful direct data-based *anomaly detection* methods (PCA, autoencoders, etc.) available. To *reconstruct* attacks/faults, some model knowledge generally necessary, and indirect methods are then preferable

Cyber-Physical Security in Energy Systems

Part II: Attack Modeling and Detection Methods

Henrik Sandberg (hsan@kth.se)

KTH EECS, Decision and Control Systems