Towards data-driven security for cyber-physical systems

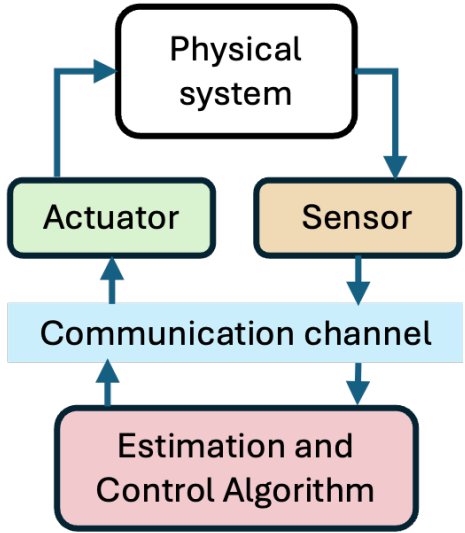
CNRS Grand Séminaire MACS Edition 2026 - Human-Cyber-Physical-Systems. 7th-8th October 2026.

Michelle S. Chong
m.s.t.chong@tue.nl
https://www.michellestchong.com

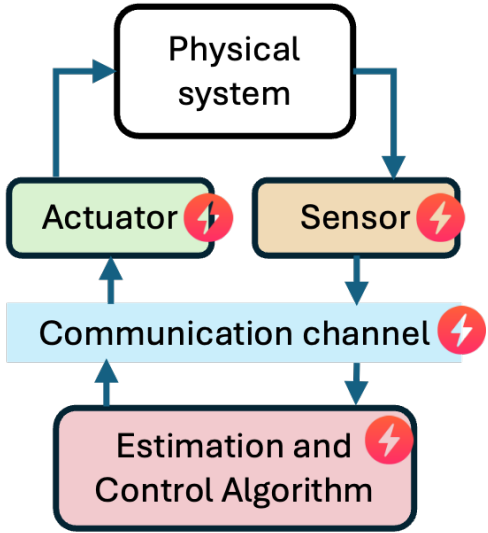


TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Cyber-Physical Systems are vulnerable



Cyber-Physical Systems are vulnerable

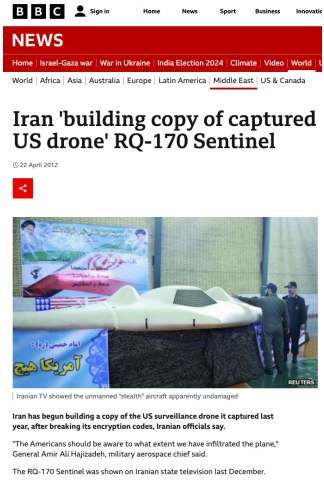


Examples of cyber-attacks in the news

- **Stuxnet** (2009): malware targeting a commercially available Programmable Logic Controller (PLC). Most well-known incident: Iran's nuclear power plants.

Examples of cyber-attacks in the news

- **Stuxnet** (2009): malware targeting a commercially available Programmable Logic Controller (PLC). Most well-known incident: Iran's nuclear power plants.
- **RQ-170** (2011): US operators lost control of the RQ-170, an unmanned aerial vehicle, near Iranian airspace, which then landed in Iran. Speculation is that Iran launched a jamming attack on the UAV, thereby blocking external control signals to the UAV, so that it will fall back onto GPS-based autopilot. Next, GPS-spoofing attack made the autopilot think it has returned to home base.



BBC Sign in Home News Sport Business Innovatic

NEWS

Home | Israel-Gaza war | War in Ukraine | India Election 2024 | Climate | Video | World

World | Africa | Asia | Australia | Europe | Latin America | Middle East | US & Canada

Iran 'building copy of captured US drone' RQ-170 Sentinel

22 April 2012

Iranian TV showed the unmanned "stealth" aircraft apparently undamaged

Iran has begun building a copy of the US surveillance drone it captured last year, after breaking its encryption codes, Iranian officials say.

"The Americans should be aware to what extent we have infiltrated the plane," General Amir Ali Hajizadeh, military aerospace chief said.

The RQ-170 Sentinel was shown on Iranian state television last December.

More examples of cyber-attacks in the news

- **Ukraine power grids** (2015-16): Phishing emails containing the BlackEnergy malware. This allowed the attacker to infiltrate the system to steal critical data and monitor the system, followed by access to critical controls which enabled the spoofing of control commands.



CNN Politics SCOTUS Congress Facts First 2024 Elections

First on CNN: U.S. investigators find proof of cyberattack on Ukraine power grid

By Evan Perez, CNN Justice Reporter
2 minute read · Updated 8:00 PM EST, Wed February 3, 2016

Video Ad Feedback

Koppel: Cyberattack on the grid could devastate U.S.

0:02 - Source: CNN

Washington (CNN) — U.S. investigators have found evidence to confirm what is believed to be the first-of-its-kind cyberattack on a power grid that caused a blackout for hundreds of thousands of people in Ukraine in December.

A U.S. official close to the investigation said the power outage was caused by a sophisticated attack using destructive malware that wrecked computers and wiped out sensitive control systems for parts of the Ukrainian power grid.

More examples of cyber-attacks in the news

- **Ukraine power grids** (2015-16): Phishing emails containing the BlackEnergy malware. This allowed the attacker to infiltrate the system to steal critical data and monitor the system, followed by access to critical controls which enabled the spoofing of control commands.
- **Marorochy water sewage** (2000): A disgruntled employee accessed the SCADA network of 150 sewage pumping stations and manipulated the control signals.



CNN Politics SCOTUS Congress Facts First 2024 Elections

First on CNN: U.S. investigators find proof of cyberattack on Ukraine power grid

By Evan Perez, CNN Justice Reporter
2 minute read · Updated 8:00 PM EST, Wed February 3, 2016

Video Ad Feedback

Koppel: Cyberattack on the grid could devastate U.S.

0:02 · Source: CNN

Washington (CNN) — U.S. investigators have found evidence to confirm what is believed to be the first-of-its-kind cyberattack on a power grid that caused a blackout for hundreds of thousands of people in Ukraine in December.

A U.S. official close to the investigation said the power outage was caused by a sophisticated attack using destructive malware that wrecked computers and wiped out sensitive control systems for parts of the Ukrainian power grid.

Some more examples of cyber-attacks in the news

- **Jeep hack** (2015): Hackers hijacked a Jeep remotely through a mobile connection in the vehicle when it was driving at 70mph on a highway in St. Louis.



BBC Sign in Home News Sport Business Innovatio

NEWS

Home | Israel-Gaza war | War in Ukraine | India Election 2024 | Climate | Video | World | U

Tech

Fiat Chrysler recalls 1.4 million cars after Jeep hack

24 July 2015

FIAT-CHRYSLER

FIAT-CHRYSLER

Jeep's Grand Cherokee is one of the recalled cars

Fiat Chrysler has issued a safety recall affecting 1.4m vehicles in the US, after security researchers showed that one of its cars could be hacked.

On Tuesday, tech magazine Wired **reported that hackers had taken control of a Jeep Cherokee** via its internet-connected entertainment system.

Some more examples of cyber-attacks in the news

- **Jeep hack** (2015): Hackers hijacked a Jeep remotely through a mobile connection in the vehicle when it was driving at 70mph on a highway in St. Louis.
- **Pwn2Own Automotive** (2024): A hacking competition, saw Tesla cars being compromised through its modem and infotainment system.



BBC Sign in Home News Sport Business Innovatio

NEWS

Home | Israel-Gaza war | War in Ukraine | India Election 2024 | Climate | Video | World

Tech

Fiat Chrysler recalls 1.4 million cars after Jeep hack

24 July 2015

FIAT-CHRYSLER

Jeep's Grand Cherokee is one of the recalled cars

Fiat Chrysler has issued a safety recall affecting 1.4m vehicles in the US, after security researchers showed that one of its cars could be hacked.

On Tuesday, tech magazine Wired **reported that hackers had taken control of a Jeep Cherokee** via its internet-connected entertainment system.

Some more examples of cyber-attacks in the news

- **Jeep hack** (2015): Hackers hijacked a Jeep remotely through a mobile connection in the vehicle when it was driving at 70mph on a highway in St. Louis.
- **Pwn2Own Automotive** (2024): A hacking competition, saw Tesla cars being compromised through its modem and infotainment system. No details have been released for security reasons.



BBC Sign in Home News Sport Business Innovatic

NEWS

Home | Israel-Gaza war | War in Ukraine | India Election 2024 | Climate | Video | World | U

Tech

Fiat Chrysler recalls 1.4 million cars after Jeep hack

24 July 2015

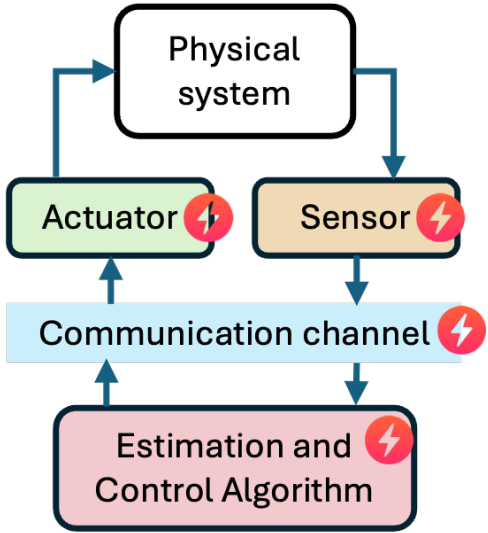
FIAT-CHRYSLER FIAT-CHRYSLER

Jeep's Grand Cherokee is one of the recalled cars

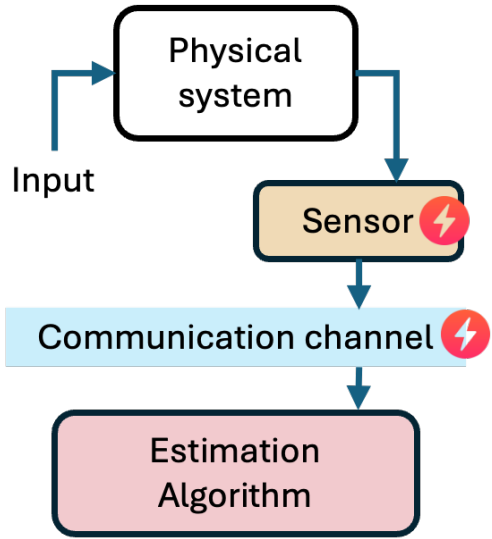
Fiat Chrysler has issued a safety recall affecting 1.4m vehicles in the US, after security researchers showed that one of its cars could be hacked.

On Tuesday, tech magazine Wired **reported that hackers had taken control of a Jeep Cherokee** via its internet-connected entertainment system.

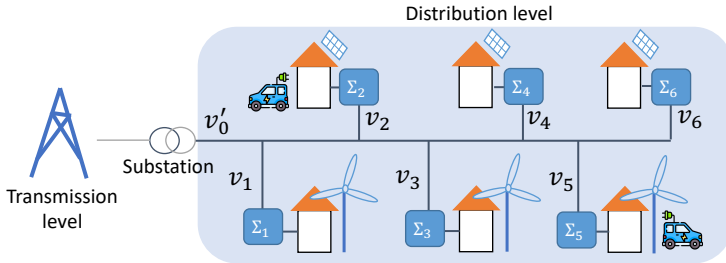
Cyber-Physical Systems are vulnerable



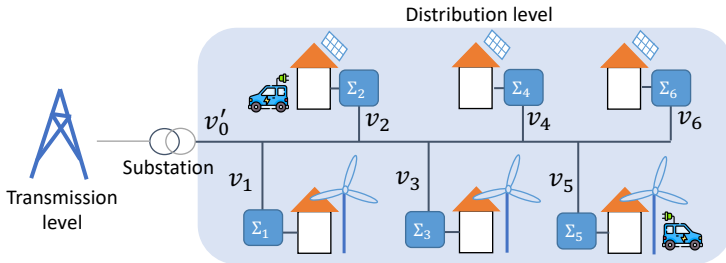
Sensor networks are vulnerable



Sensor networks in power grids are vulnerable

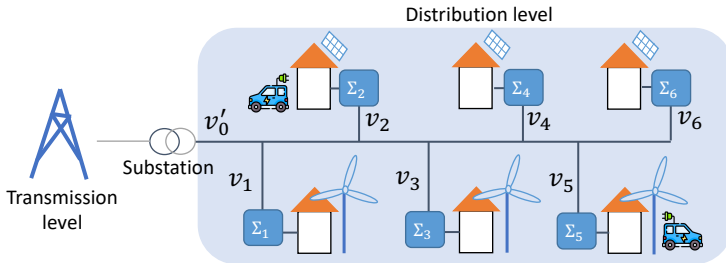


Sensor networks in power grids are vulnerable



Grid operator monitors the health (voltages) of the grid.

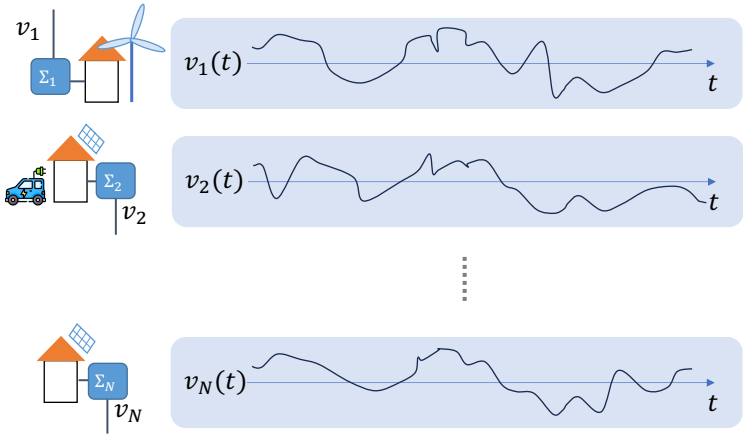
Sensor networks in power grids are vulnerable



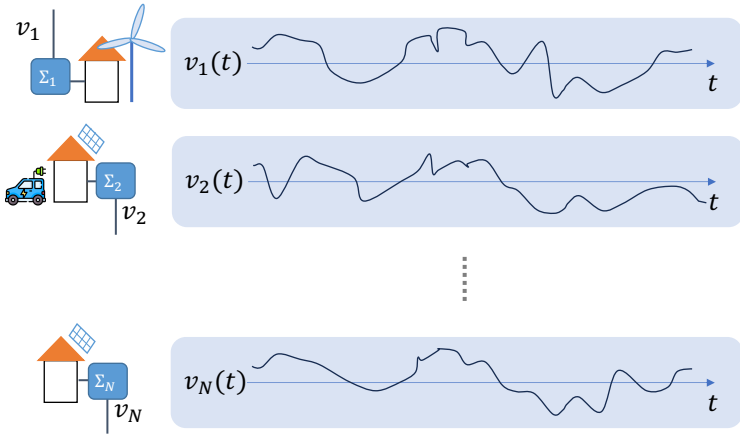
Grid operator monitors the health (voltages) of the grid.

However, the measurement data may be corrupted.

Data may be corrupted

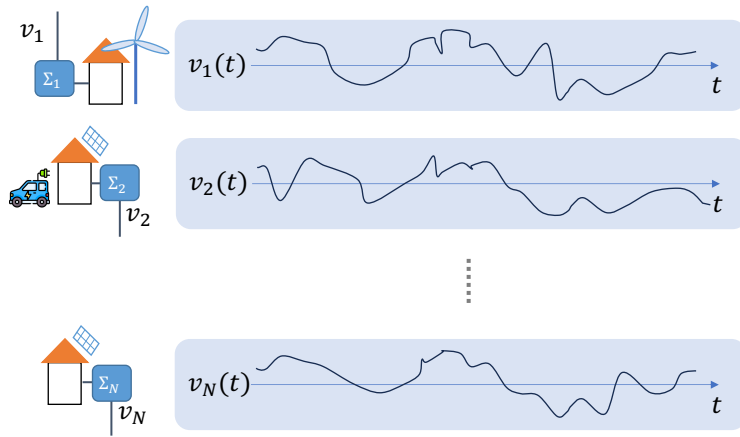


Data may be corrupted



However, grid operators want to monitor the conditions in the grid accurately.

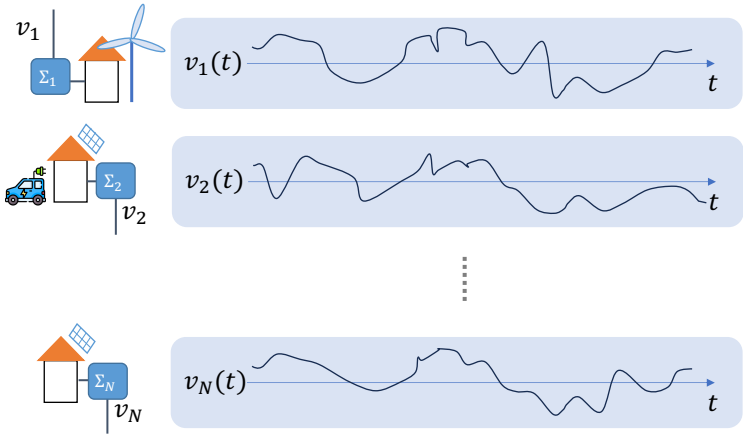
Data may be corrupted



However, grid operators want to monitor the conditions in the grid accurately.

Can v_i be estimated with accuracy that is **independent** of the sensor attacks?

Data may be corrupted



However, grid operators want to monitor the conditions in the grid accurately.

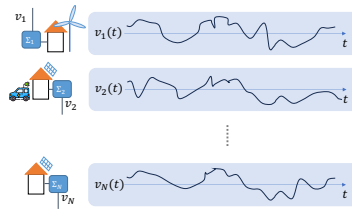
Can v_i be estimated with accuracy that is **independent** of the sensor attacks?

This is the **secure state estimation** problem.

The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

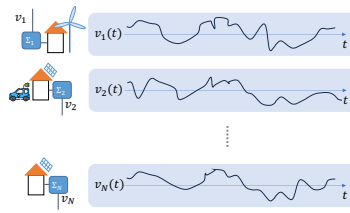


The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

$$\text{DT}: \begin{cases} x(k+1) = f(x(k), u(k)), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = h_i(x(k)) + a_i(k), & i \in [N]. \end{cases}$$

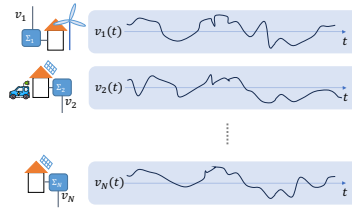


The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

$$\text{DT}: \begin{cases} x(k+1) = f(x(k), u(k)), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = h_i(x(k)) + a_i(k), & i \in [N]. \end{cases}$$



Standing assumptions

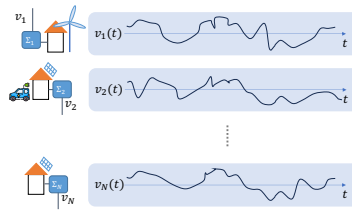
- M out of N sensors can be corrupted.
- No assumption on the attack model (statistical properties nor boundedness).

The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

$$\text{DT}: \begin{cases} x(k+1) = f(x(k), u(k)), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = h_i(x(k)) + a_i(k), & i \in [N]. \end{cases}$$



Standing assumptions

- M out of N sensors can be corrupted.
- No assumption on the attack model (statistical properties nor boundedness).

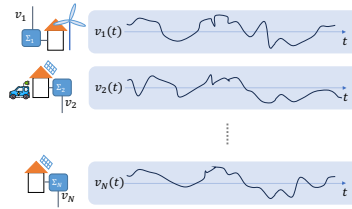
Design an estimator such that the state estimate $\hat{x}$ converges to the true state x with an error bound that is independent of the attack signals a_i.

The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

$$\text{DT}: \begin{cases} x(k+1) = f(x(k), u(k)), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = h_i(x(k)) + a_i(k), & i \in [N]. \end{cases}$$



Standing assumptions

- M out of N sensors can be corrupted.
- No assumption on the attack model (statistical properties nor boundedness).

Design an estimator such that the state estimate $\hat{x}$ converges to the true state x with an error bound that is independent of the attack signals a_i.

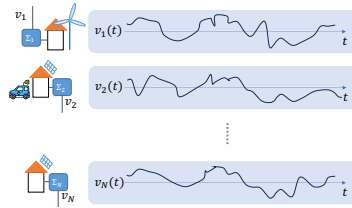
In this talk, model-based $\rightarrow$ data-based.

The secure state estimation problem formulation

System with N sensors:

$$\text{CT}: \begin{cases} \dot{x}(t) = f(x(t), u(t)), & t \in \mathbb{R}_{\geq 0}, \\ y_i(t) = h_i(x(t)) + a_i(t), & i \in [N]. \end{cases}$$

$$\text{DT}: \begin{cases} x(k+1) = Ax(k) + Bu(k), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = C_i x(k) + a_i(k), & i \in [N]. \end{cases}$$



Standing assumptions

- M out of N sensors can be corrupted.
- No assumption on the attack model (statistical properties nor boundedness).

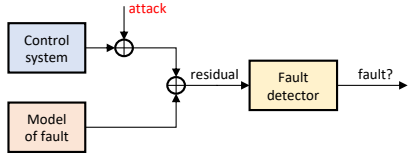
Design an estimator such that the state estimate $\hat{x}$ converges to the true state x with an error bound that is independent of the attack signals a_i.

In this talk, model-based $\rightarrow$ data-based.

Why are traditional approaches not applicable for security?

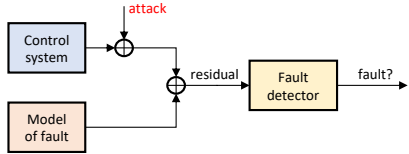
Traditional approaches

1. Fault detection and isolation



Traditional approaches

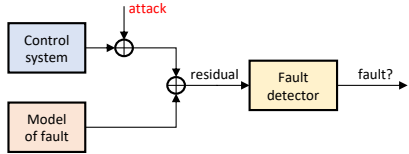
1. Fault detection and isolation



Drawback: Needs a model for each failure mode, which can be many!

Traditional approaches

1. Fault detection and isolation

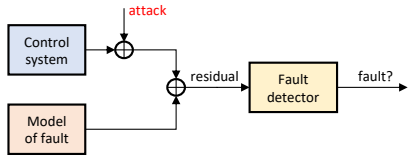


Drawback: Needs a model for each failure mode, which can be many!

2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

Traditional approaches

1. Fault detection and isolation



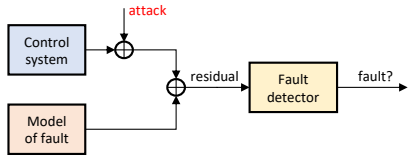
Drawback: Needs a model for each failure mode, which can be many!

2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

Drawback: Adversarial signals may be unbounded.

Traditional approaches

1. Fault detection and isolation



Drawback: Needs a model for each failure mode, which can be many!

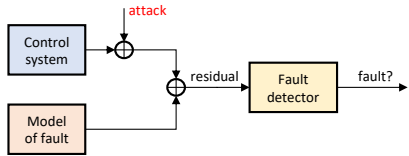
2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

Drawback: Adversarial signals may be unbounded.

3. Stochastic estimation and control
   - Assume that attacks follow a probabilistic model.

Traditional approaches

1. Fault detection and isolation



Drawback: Needs a model for each failure mode, which can be many!

2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

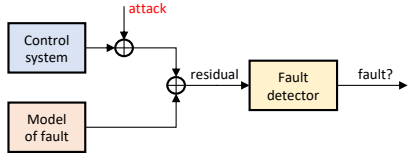
Drawback: Adversarial signals may be unbounded.

3. Stochastic estimation and control
   - Assume that attacks follow a probabilistic model.

Drawback: Does not necessarily model the adversary's behaviour.

Traditional approaches

1. Fault detection and isolation



Drawback: Needs a model for each failure mode, which can be many!

2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

Drawback: Adversarial signals may be unbounded.

3. Stochastic estimation and control
   - Assume that attacks follow a probabilistic model.

Drawback: Does not necessarily model the adversary's behaviour.

Secure state estimation aims to achieve an estimation accuracy that is **independent of the attack**.

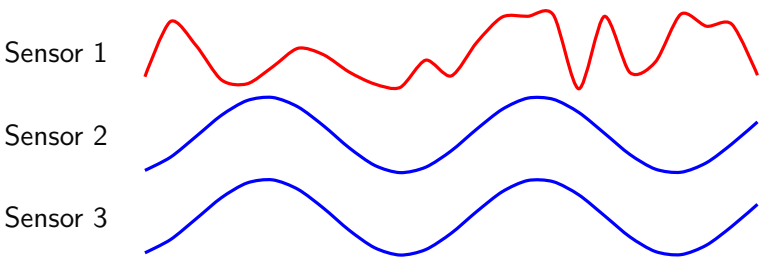
The intuition

Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

The intuition

Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

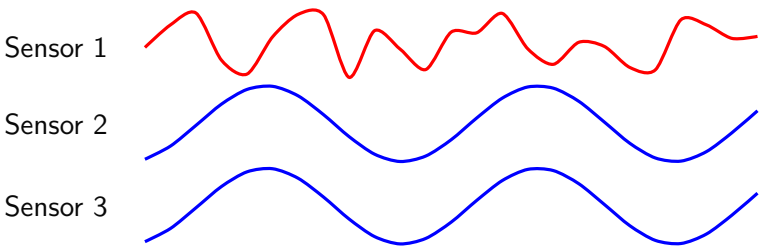
Scenario 1: One sensor has been corrupted.



The intuition

Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 1: One sensor has been corrupted.



By inspection of the signals, easy to tell that Sensor 1 has been corrupted.

The intuition

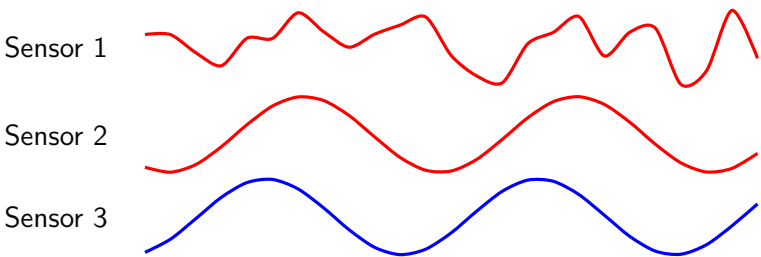
Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 2: Two sensors have been corrupted.

The intuition

Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

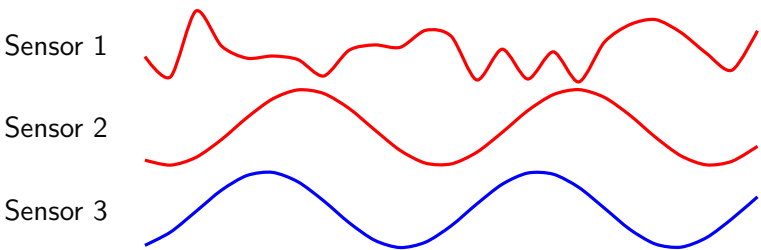
Scenario 2: Two sensors have been corrupted.



The intuition

Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 2: Two sensors have been corrupted.



Difficult to tell by inspection of the signals. One might infer that Sensor 2 and 3 have been corrupted. Untrue!

Sensor redundancy is needed.

Sensor redundancy is needed.



M-attack observability

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u,

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u, index sets $\mathcal{I}, \mathcal{I}' \subseteq [N]$ with M elements,

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u, index sets $\mathcal{I}, \mathcal{I}' \subseteq [N]$ with M elements, attack vec. $a \in \mathcal{A}_{\mathcal{I}}$ and $a' \in \mathcal{A}_{\mathcal{I}'}$,

($\mathcal{A}_{\mathcal{I}}$ denotes the set of all vectors $(a_1, a_2, \ldots, a_N)$ where $a_j \equiv 0$, $j \in [N] \backslash \mathcal{I}$)

Theorem

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u, index sets $\mathcal{I}, \mathcal{I}' \subseteq [N]$ with M elements, attack vec. $a \in \mathcal{A}_{\mathcal{I}}$ and $a' \in \mathcal{A}_{\mathcal{I}'}$, init. cond. $x(0)$, $x'(0)$, the following holds

Theorem

1. $N > 2M$, where M is the number of compromised sensors,

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u, index sets $\mathcal{I}, \mathcal{I}' \subseteq [N]$ with M elements, attack vec. $a \in \mathcal{A}_{\mathcal{I}}$ and $a' \in \mathcal{A}_{\mathcal{I}'}$, init. cond. $x(0)$, $x'(0)$, the following holds

$$y_i(t, x(0), u, a) = y_i(t, x'(0), u, a'), \forall t \in \{0, 1, \ldots, T\}, \forall i \in [N] \implies x(0) = x'(0).$$

Theorem

The system is M-**attack observable**, if and only if

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is **observable** via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements.

Theorem: M-attack observability

System with N sensors: $\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$

M-attack observability

The system is M-attack observable on $\{0, 1, \ldots, T\}$, $T < \infty$ if for every input u, index sets $\mathcal{I}, \mathcal{I}' \subseteq [N]$ with M elements, attack vec. $a \in \mathcal{A}_{\mathcal{I}}$ and $a' \in \mathcal{A}_{\mathcal{I}'}$, init. cond. $x(0)$, $x'(0)$, the following holds

$$y_i(t, x(0), u, a) = y_i(t, x'(0), u, a'), \forall t \in \{0, 1, \ldots, T\}, \forall i \in [N] \implies x(0) = x'(0).$$

Theorem

The system is M-**attack observable**, if and only if

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is **observable** via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ \quad y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements.

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

Then there exists an estimator to provide estimate $\hat{x}$ such that

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

Then there exists an estimator to provide estimate $\hat{x}$ such that there exists a function $\beta \in \mathcal{KL}$ satisfying

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is M-**attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

Then there exists an estimator to provide estimate $\hat{x}$ such that there exists a function $\beta \in \mathcal{KL}$ satisfying

$$|\hat{x}(k) - x(k)| \leq \beta(|\hat{x}(0) - x(0)|, k),$$

$\forall k \geq 0$, init. cond. $\hat{x}(0)$ and $x(0)$.

From M-attack observability to an algorithm

System with N sensors:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k), \\ y_i(k) = C_i x(k) + a_i(k), \ i \in [N], \quad k \in \mathbb{N}_{\geq 0}. \end{cases}$$

Theorem

Suppose the system is **M-attack observable**, i.e., the following holds

1. $N > 2M$, where M is the number of compromised sensors,
2. the system is observable via every $y_{\mathcal{J}} := (y_i)_{i \in \mathcal{J}}$ sensors, where $\mathcal{J} \subset [N]$ with $N - 2M$ elements. (every $(A, C_{\mathcal{J}})$ pair is observable).

Then there exists an estimator to provide estimate $\hat{x}$ such that there exists a function $\beta \in \mathcal{KL}$ satisfying

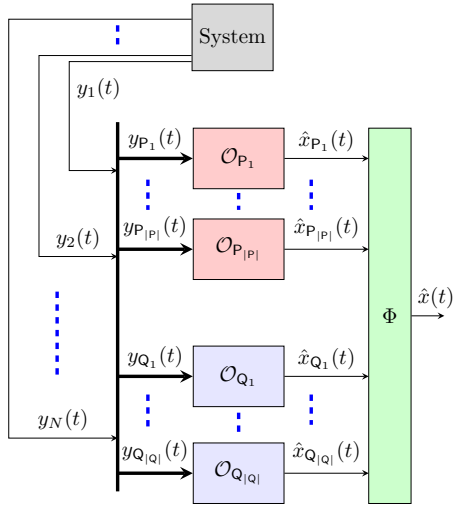
$$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a} ,$$

$\forall k \geq 0$, init. cond. $\hat{x}(0)$ and $x(0)$.

From theorem to a
model-based SSE algorithm

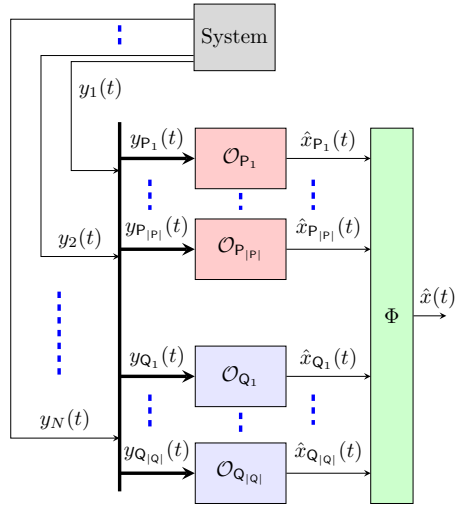
A model-based SSE algorithm

1. For each combination of $N - M$ ($\geq N - 2M$) sensors, construct an estimator $\mathcal{O}_{\mathsf{P}}$ based on those sensors, that is robust (input-to-state stable) w.r.t. attack a.



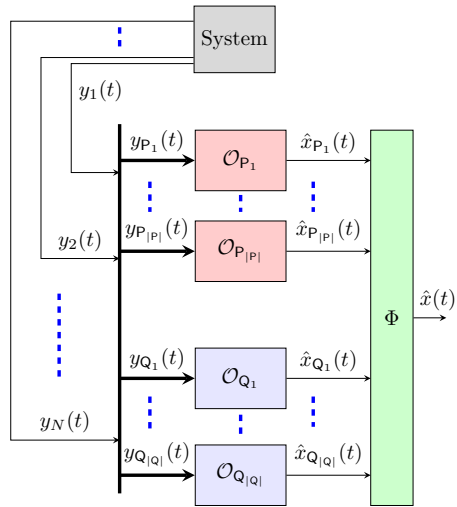
A model-based SSE algorithm

1. For each combination of $N - M$ ($\geq N - 2M$) sensors, construct an estimator $\mathcal{O}_{\mathsf{P}}$ based on those sensors, that is robust (input-to-state stable) w.r.t. attack a.
2. For each combination of $N - 2M$ sensors, construct a robust estimator $\mathcal{O}_{\mathsf{Q}}$ w.r.t. a.



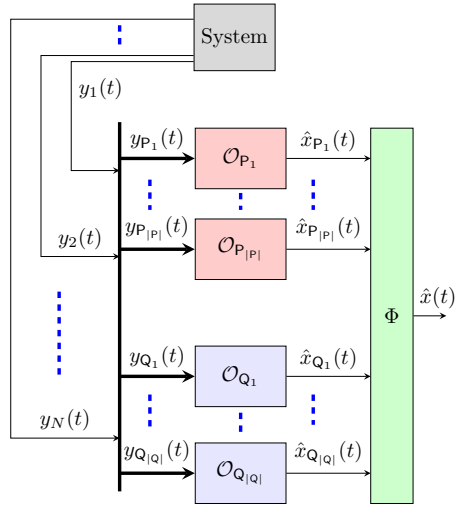
A model-based SSE algorithm

1. For each combination of $N - M$ ($\geq N - 2M$) sensors, construct an estimator $\mathcal{O}_{\mathsf{P}}$ based on those sensors, that is robust (input-to-state stable) w.r.t. attack a.
2. For each combination of $N - 2M$ sensors, construct a robust estimator $\mathcal{O}_{\mathsf{Q}}$ w.r.t. a.
3. One set of $N - M$ sensors is attack-free. For this set, all combinations of $N - 2M$ sensors will also be attack-free.

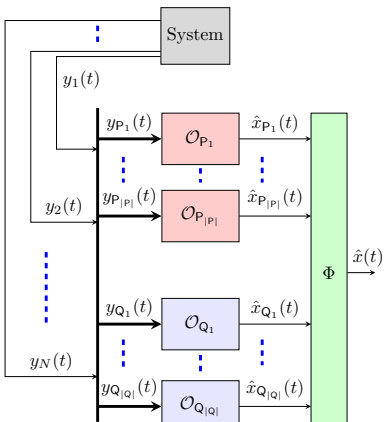


A model-based SSE algorithm

1. For each combination of $N - M$ ($\geq N - 2M$) sensors, construct an estimator $\mathcal{O}_{\mathsf{P}}$ based on those sensors, that is robust (input-to-state stable) w.r.t. attack a.
2. For each combination of $N - 2M$ sensors, construct a robust estimator $\mathcal{O}_{\mathsf{Q}}$ w.r.t. a.
3. One set of $N - M$ sensors is attack-free. For this set, all combinations of $N - 2M$ sensors will also be attack-free.
   The choice of the state estimate $\hat{x}$ is handled by the consistency mapping Φ.



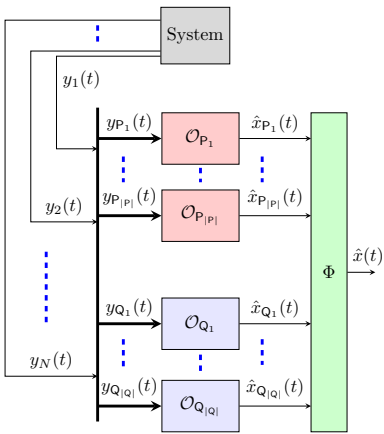
The consistency mapping Φ



Consistency mapping Φ to choose an estimate $\hat{x}$ from the bank of state estimates $\hat{x}_{\mathsf{P}}$, $\hat{x}_{\mathsf{Q}}$:

$$\begin{aligned} \pi_{\mathsf{P}}(k) &:= \max_{\mathsf{Q}\subset\mathsf{P},|\mathsf{Q}|=N-2M} \left|\hat{x}_{\mathsf{Q}}(k) - \hat{x}_{\mathsf{P}}(k)\right|, k \geq 0. \\ \hat{x}(k) &= \hat{x}_{\sigma(k)}(k), \quad \sigma(k) := \operatorname*{arg\,min}_{\mathsf{P}\subset[N],|\mathsf{P}|=N-M} \pi_{\mathsf{P}}(k). \end{aligned}$$

The consistency mapping Φ



Consistency mapping Φ to choose an estimate $\hat{x}$ from the bank of state estimates $\hat{x}_{\mathrm{P}}$, $\hat{x}_{\mathrm{Q}}$:

$$\begin{aligned} \pi_{\mathrm{P}}(k) &:= \max_{\mathrm{Q}\subset\mathrm{P},|\mathrm{Q}|=N-2M} |\hat{x}_{\mathrm{Q}}(k) - \hat{x}_{\mathrm{P}}(k)|\,, k \geq 0. \\ \hat{x}(k) &= \hat{x}_{\sigma(k)}(k), \quad \sigma(k) := \operatorname*{arg\,min}_{\mathrm{P}\subset[N],|\mathrm{P}|=N-M} \pi_{\mathrm{P}}(k). \end{aligned}$$

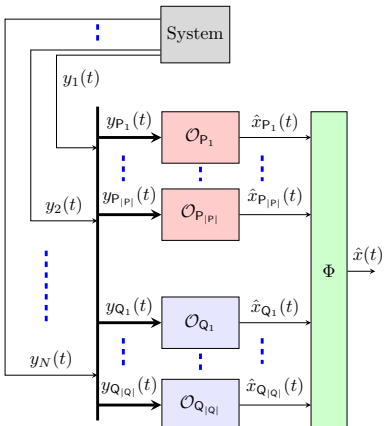
Theorem

Suppose system is *M*-attack observable, then

$$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a} \qquad k \geq 0,$$

where $\beta \in \mathcal{KL}$, for all $x(0)$, $\hat{x}_{\mathrm{P}}(0)$, $\hat{x}_{\mathrm{Q}}(0) \in \mathbb{R}^n$.

The consistency mapping Φ



Consistency mapping Φ to choose an estimate $\hat{x}$ from the bank of state estimates $\hat{x}_{\mathsf{P}}$, $\hat{x}_{\mathsf{Q}}$:

$$\begin{aligned} \pi_{\mathsf{P}}(k) &:= \max_{\mathsf{Q}\subset\mathsf{P},|\mathsf{Q}|=N-2M} |\hat{x}_{\mathsf{Q}}(k) - \hat{x}_{\mathsf{P}}(k)|\,, k \geq 0. \\ \hat{x}(k) &= \hat{x}_{\sigma(k)}(k), \quad \sigma(k) := \operatorname*{arg\,min}_{\mathsf{P}\subset[N],|\mathsf{P}|=N-M} \pi_{\mathsf{P}}(k). \end{aligned}$$

Theorem

Suppose system is M-attack observable, then

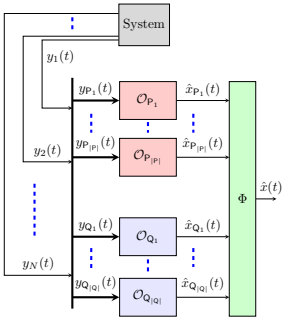
$$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a} \qquad k \geq 0,$$

where $\beta \in \mathcal{KL}$, for all $x(0)$, $\hat{x}_{\mathsf{P}}(0)$, $\hat{x}_{\mathsf{Q}}(0) \in \mathbb{R}^n$.

Corollary: As $k \to \infty$, $\sigma(k)$ chooses the attack-free set.

M. Chong ⟨m.s.t.chong@tue.nl⟩

Takeaway 1: Model-based SSE



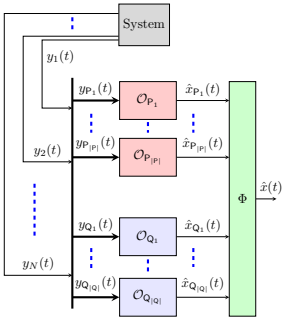
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI

Takeaway 1: Model-based SSE



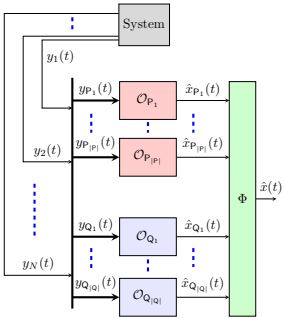
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. Sufficient conditions only for nonlinear systems.

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.

Takeaway 1: Model-based SSE



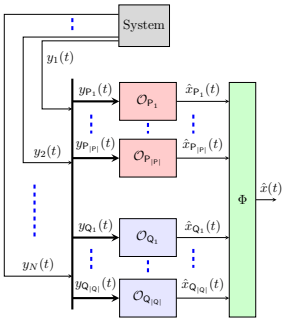
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. Sufficient conditions only for nonlinear systems.
3. When M sensors are under attack, we need
   (i) $N > 2M$ sensors for $|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}$, $\forall k$.

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.

Takeaway 1: Model-based SSE



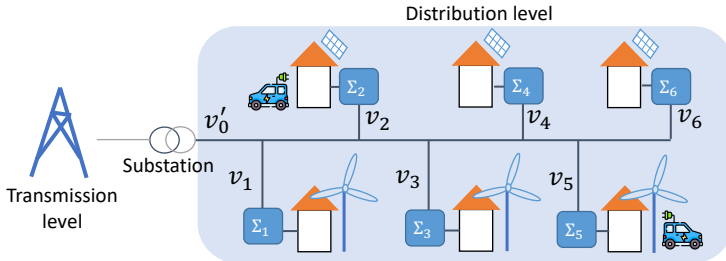
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. Sufficient conditions only for nonlinear systems.
3. When M sensors are under attack, we need
   (i) $N > 2M$ sensors for $|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}$, $\forall k$.

References:

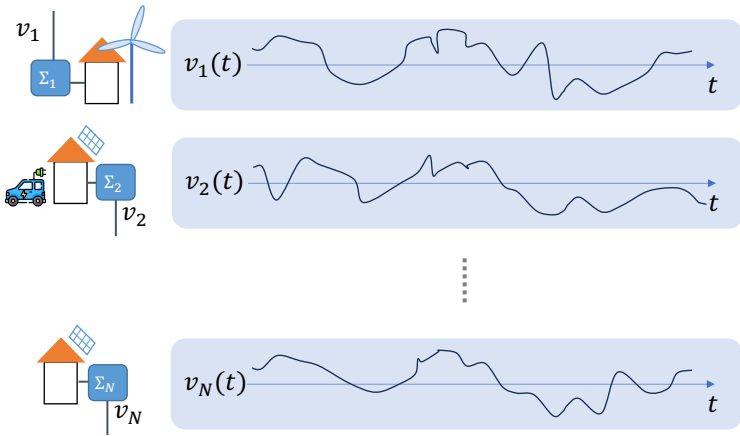
- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously and non-uniformly time-sampled measurements.

Revisit the power grid

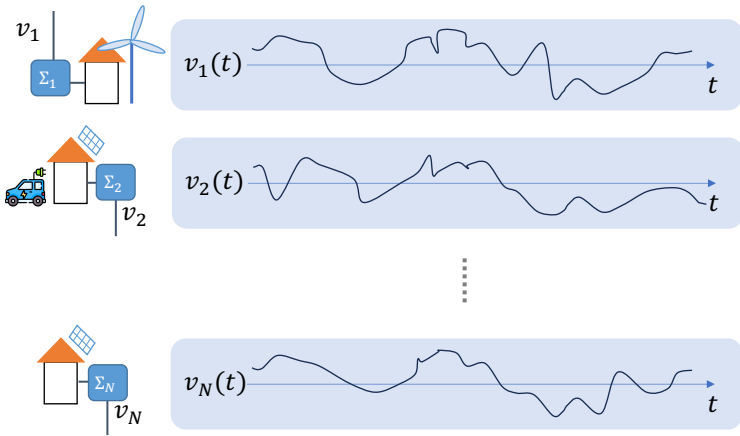


Grid operator monitors the health (voltages v_i) of the grid.

Data may be corrupted

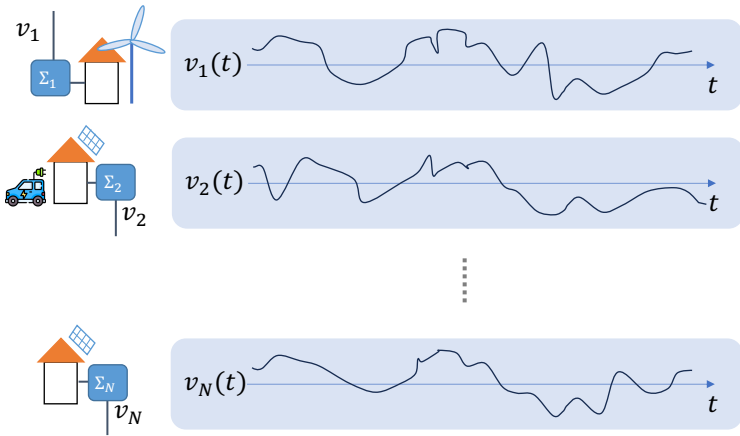


Data may be corrupted



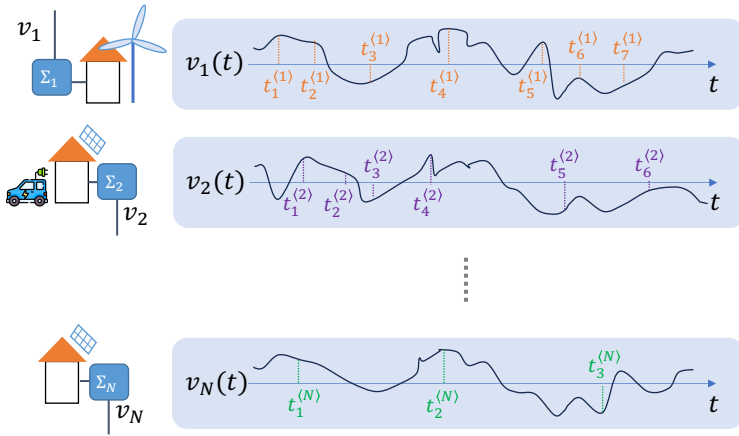
However, grid operators want to monitor the conditions in the grid accurately.

Data may be corrupted



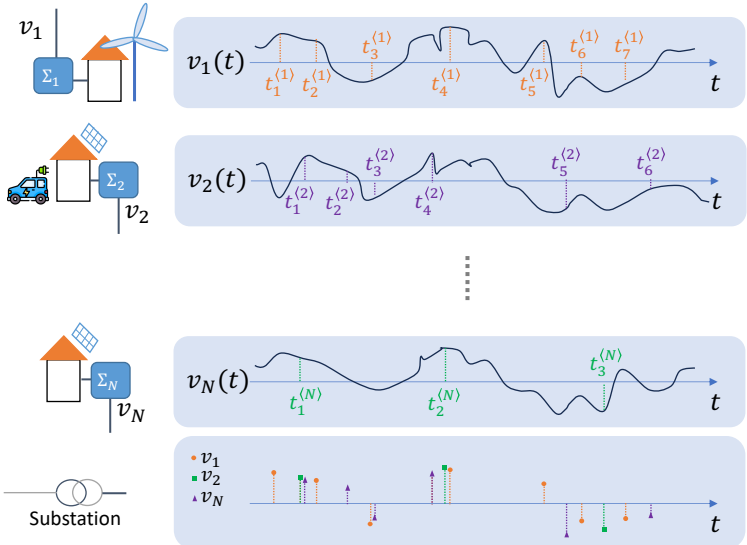
However, grid operators want to monitor the conditions in the grid accurately.
Moreover, the measurements are time-sampled...

Data may be corrupted and time-sampled



However, grid operators want to monitor the conditions in the grid accurately.
Moreover, the measurements are time-sampled...

Data may be corrupted and time-sampled



The SSE solution with a redesign of the observers $\mathcal{O}$

Under nonuniform and synchronous sampling times $t_k,\ k \in \mathbb{N}$, i.e.,

- The *inter-sample times* are upper and lower bounded: $t_{k+1} - t_k \in [\underline{T}, \bar{T}]$.

The SSE solution with a redesign of the observers $\mathcal{O}$

Under nonuniform and synchronous sampling times t_k, $k \in \mathbb{N}$, i.e.,

- The *inter-sample times* are upper and lower bounded: $t_{k+1} - t_k \in [\underline{T}, \bar{T}]$.

Redesign each of the observers $\mathcal{O}$ such that its state estimate $\hat{x}_\mathsf{J}$ is ISS with respect to attacks, noise, disturbances and sampling induced error, i.e.,

The SSE solution with a redesign of the observers $\mathcal{O}$

Under nonuniform and synchronous sampling times t_k, $k \in \mathbb{N}$, i.e.,

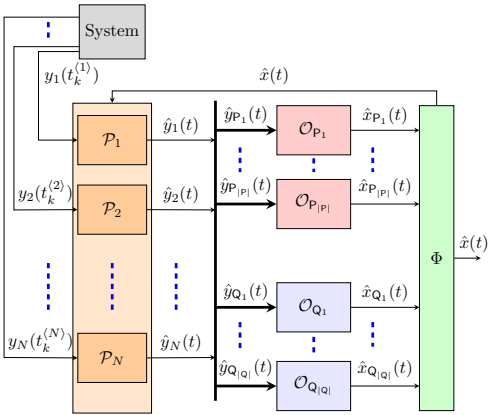
- The *inter-sample times* are upper and lower bounded: $t_{k+1} - t_k \in [\underline{T}, \bar{T}]$.

Redesign each of the observers $\mathcal{O}$ such that its state estimate $\hat{x}_J$ is ISS with respect to attacks, noise, disturbances and sampling induced error, i.e.,

$$|x(t) - \hat{x}_J(t)| \leq \max\{\hat{\beta}(|x(0) - \hat{x}_J(0)|, t),\ \hat{\gamma}(||(\underbrace{\delta(t)}_{\text{sampling effects}}, \overbrace{d(t)}^{\text{disturbance}}, \underbrace{d(t_k) + \eta(t_k) + a(t_k)}_{\text{disturbance, noise, attacks}})||_{[0,t]})\},$$

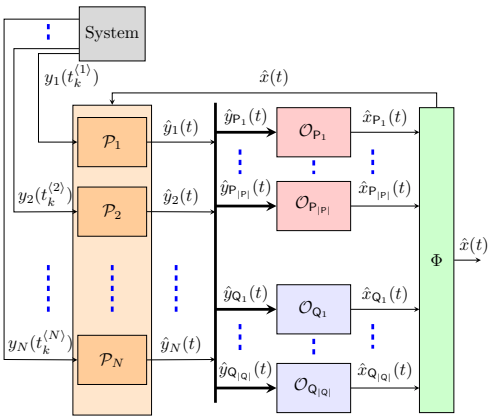
for any initial conditions $x(0)$, $\hat{x}_J(0)$ and where $\hat{\beta} \in \mathcal{KL}$ and $\hat{\gamma} \in \mathcal{K}_\infty$.

SSE with time-sampled measurements



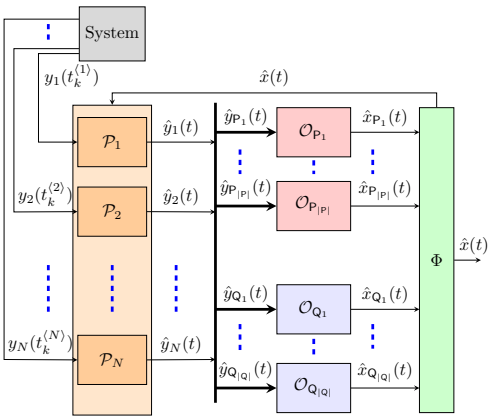
- Introduction of the $\mathcal{P}$ components, to handle inter-sample behavior such that we obtain the desired ISS property.

SSE with time-sampled measurements



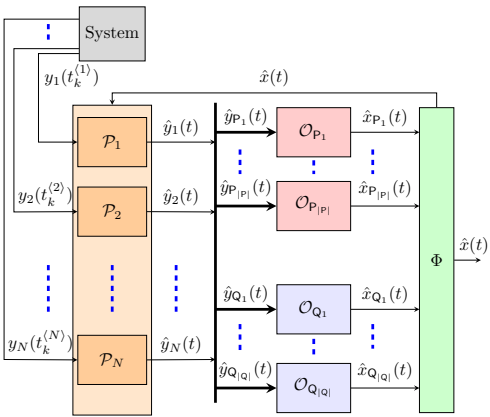
- Introduction of the $\mathcal{P}$ components, to handle inter-sample behavior such that we obtain the desired ISS property.
- Using the same consistency mapping Φ,

SSE with time-sampled measurements



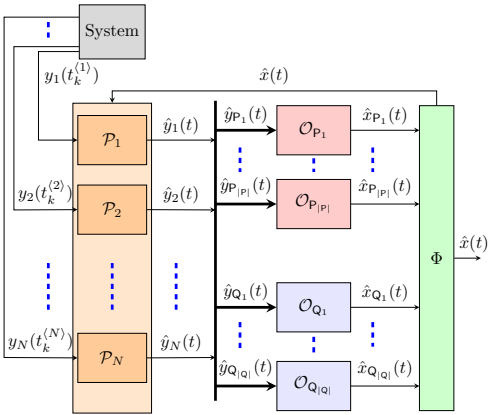
- Introduction of the $\mathcal{P}$ components, to handle inter-sample behavior such that we obtain the desired ISS property.
- Using the same consistency mapping Φ, we remove the effect of the attacks a

SSE with time-sampled measurements



- Introduction of the $\mathcal{P}$ components, to handle inter-sample behavior such that we obtain the desired ISS property.
- Using the same consistency mapping Φ, we remove the effect of the attacks a and achieve secure state estimation.

SSE with time-sampled measurements



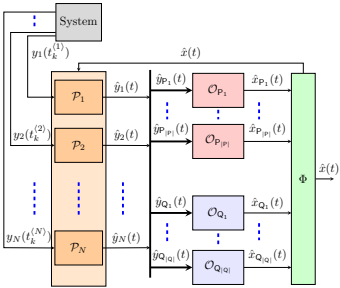
- Introduction of the $\mathcal{P}$ components, to handle inter-sample behavior such that we obtain the desired ISS property.
- Using the same consistency mapping Φ, we remove the effect of the attacks a and achieve secure state estimation.

Details can be found in:

Gootzen, Chong (2026)
Secure estimator design for Lur'e-type systems with nonuniformly and synchronously sampled measurements under attacks



Takeaway 2: Model-based SSE

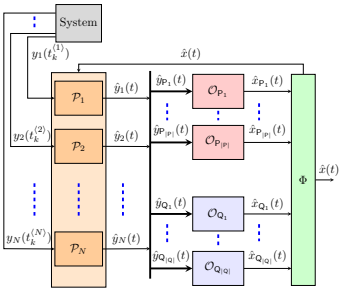


When the model is known

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.

Takeaway 2: Model-based SSE

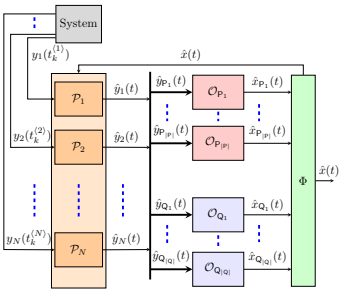


When the model is known, M sensors are under attack

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.

Takeaway 2: Model-based SSE

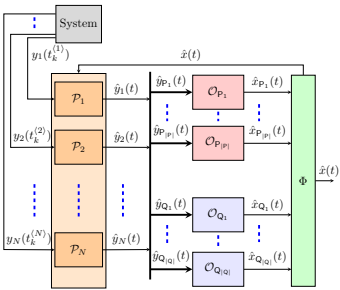


When the model is known, M sensors are under attack and when the data is time-sampled

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.
- (Niazi, Alanwar, Chong, Johansson; '23, '26) on set-based SSE for LTI systems.

Takeaway 2: Model-based SSE

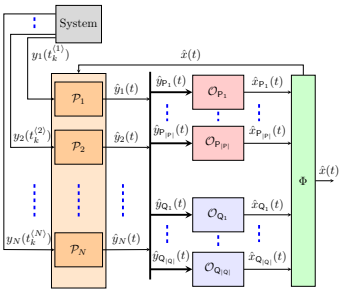


When the model is known, M sensors are under attack and when the data is time-sampled, we need

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.
- (Niazi, Alanwar, Chong, Johansson; '23, '26) on set-based SSE for LTI systems.

Takeaway 2: Model-based SSE



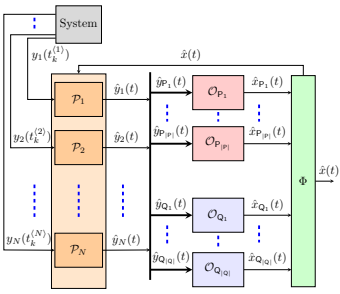
When the model is known, M sensors are under attack and when the data is time-sampled, we need

- $N > 2M$ sensors for
$$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}, \ \forall k.$$

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.
- (Niazi, Alanwar, Chong, Johansson; '23, '26) on set-based SSE for LTI systems.

Takeaway 2: Model-based SSE



When the model is known, M sensors are under attack and when the data is time-sampled, we need

- $N > 2M$ sensors for
$$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}, \ \forall k.$$

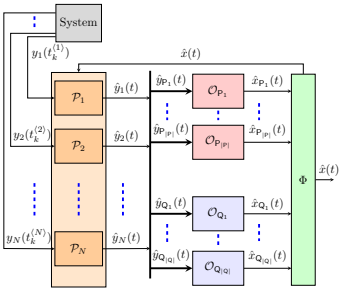
When we account for model uncertainty and relax our estimates from trajectories $\rightarrow$ sets, we need

- $N > M$ sensors for
$$x(k) \in \hat{\mathcal{X}} \subseteq \mathcal{X} \leftarrow \text{known}, \ \forall k.$$

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.
- (Niazi, Alanwar, Chong, Johansson; '23, '26) on set-based SSE for LTI systems.

Takeaway 2: Model-based SSE



When the model is known, M sensors are under attack and when the data is time-sampled, we need

- $N > 2M$ sensors for
  $|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}$, $\forall k$.

When we account for model uncertainty and relax our estimates from trajectories $\rightarrow$ sets, we need

- $N > M$ sensors for $\underbrace{x(k) \in \hat{\mathcal{X}}}_{\text{set-based SSE}} \subseteq \mathcal{X} \leftarrow$ known, $\forall k$.

References:

- (Chong, Wakaiki, Hespanha; '15) for LTI and (Chong, Sandberg, Hespanha; '20) for NL.
- (**Gootzen**, Chong; '26) for synchronously time-sampled measurements.
- (Niazi, Alanwar, Chong, Johansson; '23, '26) on set-based SSE for LTI systems.

What if the model is unknown?

What if the model is unknown?

Identify the attack-free set of sensors.

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

  $\downarrow$

- We don't know which set

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

  $\downarrow$

- We don't know which set $\longrightarrow$ Check every $N - M$ set.

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

  ↓

- We don't know which set $\longrightarrow$ Check every $N - M$ set.

  ↓

- For every $N - M$ set of sensors z_j, we have the following model:

$$x(k+1) = Ax(k) + Bu(k), \quad z_j(k) = C_{\mathcal{J}_j} x(k), \quad k \in \mathbb{N}_{\geq 0},$$

  where $\mathcal{J}_j \subset [N]$ with cardinality $N - M$

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

  ↓

- We don't know which set $\longrightarrow$ Check every $N - M$ set.

  ↓

- For every $N - M$ set of sensors z_j, we have the following model:

$$x(k+1) = Ax(k) + Bu(k), \quad z_j(k) = C_{\mathcal{J}_j} x(k), \quad k \in \mathbb{N}_{\geq 0},$$

where $\mathcal{J}_j \subset [N]$ with cardinality $N - M$ for $j \in [n_\mathsf{J}]$, where $n_\mathsf{J} := \binom{N}{N-M}$.

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

  ↓

- We don't know which set $\longrightarrow$ Check every $N - M$ set.

  ↓

- For every $N - M$ set of sensors z_j, we have the following model:

$$x(k+1) = Ax(k) + Bu(k), \quad z_j(k) = C_{\mathcal{J}_j} x(k), \quad k \in \mathbb{N}_{\geq 0},$$

where $\mathcal{J}_j \subset [N]$ with cardinality $N - M$ for $j \in [n_J]$, where $n_J := \binom{N}{N-M}$.

Since the model (A, B) are unknown,

Our approach

Recall that we consider a (A, B)-unknown LTI system with N sensors, where $M < N$ are under attack:

$$x(k+1) = Ax(k) + Bu(k), \quad y_i(k) = C_i x(k) + a_i(k),\ i \in [N], \quad k \in \mathbb{N}_{\geq 0}.$$

- There must be one set of $N - M$ sensors which is attack-free.

$\downarrow$

- We don't know which set $\longrightarrow$ Check every $N - M$ set.

$\downarrow$

- For every $N - M$ set of sensors z_j, we have the following model:

$$x(k+1) = Ax(k) + Bu(k), \quad z_j(k) = C_{\mathcal{J}_j} x(k), \quad k \in \mathbb{N}_{\geq 0},$$

where $\mathcal{J}_j \subset [N]$ with cardinality $N - M$ for $j \in [n_\mathcal{J}]$, where $n_\mathcal{J} := \binom{N}{N-M}$.

Since the model (A, B) are unknown, we will learn them from data!

Data-based representations for LTI systems

According to Willems et. al.'s Fundamental Lemma,

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

Data-based representations for LTI systems

According to Willems et. al.'s Fundamental Lemma,

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathrm{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathrm{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

See de Persis and Tesi (2019) Formulas for data-driven control...

Data-based representations for LTI systems

According to Willems et. al.'s Fundamental Lemma,

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathrm{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathrm{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

$U_{n,T}$, $\hat{X}_{j,n,T}$ and $\hat{X}_{j,n+1,T}$ are data matrices:

$$\begin{aligned} \hat{X}_{j,n,T} &= \begin{bmatrix} \mathcal{X}_j(n) & \dots & \mathcal{X}_j(n+T-1) \end{bmatrix}, \\ \hat{X}_{j,n+1,T} &= \begin{bmatrix} \mathcal{X}_j(n+1) & \dots & \mathcal{X}_j(n+T) \end{bmatrix}, \\ U_{n,T} &= \begin{bmatrix} u(n) & \dots & u(n+T-1) \end{bmatrix}, \end{aligned}$$

$$\mathcal{X}_j(k) := \begin{bmatrix} z_j(k-n) \\ \vdots \\ z_j(k-1) \\ \hdashline u(k-n) \\ \vdots \\ u(k-1) \end{bmatrix}$$

See de Persis and Tesi (2019) Formulas for data-driven control...

Data-based representations for LTI systems

According to Willems et. al.'s Fundamental Lemma,

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathrm{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathrm{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

Theorem

Recall $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$.

Model-based rep. $=$ Data-based rep.

only if

$$\operatorname{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$

See de Persis and Tesi (2019) Formulas for data-driven control...

A data-based attack detection algorithm : the main idea

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathsf{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

A data-based attack detection algorithm : the main idea

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathsf{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

- Now, consider the online measurements $\tilde{z}_j = z_j + a_{\mathcal{J}_i}$.

A data-based attack detection algorithm : the main idea

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathsf{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

- Now, consider the online measurements $\tilde{z}_j = z_j + a_{\mathcal{J}_i}$.
- Recall that, we have $N - M$ attack-free sensors.

A data-based attack detection algorithm : the main idea

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathsf{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

- Now, consider the online measurements $\tilde{z}_j = z_j + a_{\mathcal{J}_i}$.
- Recall that, we have $N - M$ attack-free sensors. $\implies$ one $\tilde{z}_j$ is attack-free.

A data-based attack detection algorithm : the main idea

Model-based representation

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ z_j(k) &= C_{\mathcal{J}_j} x(k), \qquad k \in \mathbb{N}_{\geq 0}, \end{aligned}$$

for $\mathcal{J}_j \subset [N]$ with $N - M$ elements and $j \in n_{\mathsf{J}}$.

$\rightarrow$

Data-based representation

For $j \in [n_{\mathsf{J}}]$,

$$\begin{aligned} \mathcal{X}_j(k+1) &= \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \\ \Lambda_j &:= \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}. \end{aligned}$$

- Now, consider the online measurements $\tilde{z}_j = z_j + a_{\mathcal{J}_i}$.
- Recall that, we have $N - M$ attack-free sensors. $\implies$ one $\tilde{z}_j$ is attack-free.

A data-based attack detection algorithm

Data-based representation

$$\text{For } j \in [n_J], \quad \mathcal{X}_j(k+1) = \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \qquad \Lambda_j := \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}.$$

Algorithm

1. **(Offline)** Construct Λ_j, $\forall j \in [n_J]$.

A data-based attack detection algorithm

Data-based representation

$$\text{For } j \in [n_\mathrm{J}], \quad \mathcal{X}_j(k+1) = \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \qquad \Lambda_j := \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^\dagger.$$

Algorithm

1. **(Offline)** Construct Λ_j, $\forall j \in [n_\mathrm{J}]$.
2. **(Offline)** Apply input u of length n, collect online measurements of length T until

$$\text{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$

A data-based attack detection algorithm

Data-based representation

$$\text{For } j \in [n_\text{J}], \quad \mathcal{X}_j(k+1) = \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \qquad \Lambda_j := \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^\dagger.$$

Algorithm

1. **(Offline)** Construct Λ_j, $\forall j \in [n_\text{J}]$.
2. **(Offline)** Apply input u of length n, collect online measurements of length T until
$$\text{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$
3. **(Online)** Apply *test* input data of length 1, collect the online data,

A data-based attack detection algorithm

Data-based representation

$$\text{For } j \in [n_J], \quad \mathcal{X}_j(k+1) = \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \qquad \Lambda_j := \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}.$$

Algorithm

1. **(Offline)** Construct Λ_j, $\forall j \in [n_J]$.
2. **(Offline)** Apply input u of length n, collect online measurements of length T until
$$\text{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$
3. **(Online)** Apply *test* input data of length 1, collect the online data, and construct the matrices $\underline{U}_{k,1}, \underline{\hat{X}}_{j,k,1}, \underline{\hat{X}}_{j,k+1,1}$, $\forall j \in [n_J]$.

A data-based attack detection algorithm

Data-based representation

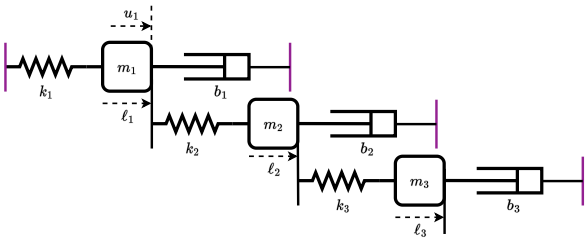
$$\text{For } j \in [n_J], \quad \mathcal{X}_j(k+1) = \Lambda_j \begin{bmatrix} u(k) \\ \mathcal{X}_j(k) \end{bmatrix}, \qquad \Lambda_j := \hat{X}_{j,n+1,T} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix}^{\dagger}.$$

Algorithm

1. **(Offline)** Construct Λ_j, $\forall j \in [n_J]$.
2. **(Offline)** Apply input u of length n, collect online measurements of length T until
$$\text{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$
3. **(Online)** Apply *test* input data of length 1, collect the online data, and construct the matrices $\underline{U}_{k,1}, \underline{\hat{X}}_{j,k,1}, \underline{\hat{X}}_{j,k+1,1}$, $\forall j \in [n_J]$.
4. **(Online)** Compute set of attack-free sensors
$$j^*[k+1] \in \arg\min_{j \in [n_J]} \left\| \underline{\hat{X}}_{j,k+1,1} - \Lambda_j \begin{bmatrix} \underline{U}_{k,1} \\ \underline{\hat{X}}_{j,k,1} \end{bmatrix} \right\|_2.$$

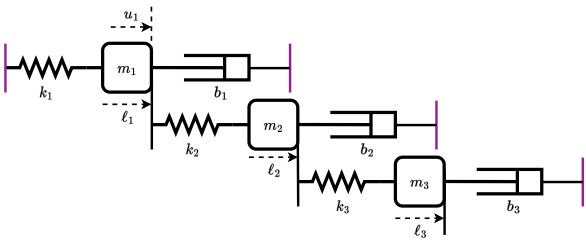
M. Chong ⟨m.s.t.chong@tue.nl⟩

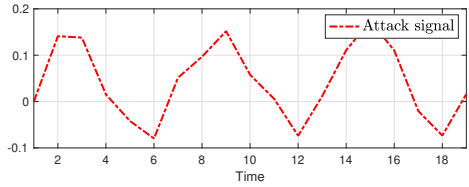
An example



Let $x := \begin{bmatrix} l_1 \\ \dot{l}_1 \\ l_2 \\ \dot{l}_2 \\ l_3 \\ \dot{l}_3 \end{bmatrix}$. $\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-k_1}{m_1} & \frac{-b_1}{m_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{1}{m_2} & 0 & \frac{-k_2}{m_2} & \frac{-b_2}{m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{m_3} & 0 & \frac{-k_3}{m_3} & \frac{-b_3}{m_3} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$, $\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \left[\begin{array}{cc|cc|cc} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] x$

An example





$$\text{Let } x := \begin{bmatrix} l_1 \\ \dot{l}_1 \\ l_2 \\ \dot{l}_2 \\ l_3 \\ \dot{l}_3 \end{bmatrix}. \quad \dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-k_1}{m_1} & \frac{-b_1}{m_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \frac{1}{m_2} & 0 & \frac{-k_2}{m_2} & \frac{-b_2}{m_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{1}{m_3} & 0 & \frac{-k_3}{m_3} & \frac{-b_3}{m_3} \end{bmatrix} x + \begin{bmatrix} 0 \\ \frac{1}{m_1} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u, \quad \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x$$

Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



- A data-based attack identification algorithm with

Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



- A data-based attack identification algorithm with

  offline learning +

Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



- A data-based attack identification algorithm with

  offline learning + **online** data-based attacked sensor identification

Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



- A data-based attack identification algorithm with

  offline learning + **online** data-based attacked sensor identification

- A **fully online algorithm** for certain attack types:

Takeaway 3: Data-based identification of attacked sensors

Based on

Sribalaji Anand, M. Chong, A. Texeira (2025)
Data-driven attack detection for networked control systems.



- A data-based attack identification algorithm with

  offline learning + **online** data-based attacked sensor identification

- A **fully online algorithm** for certain attack types:

  replay attacks and network delay attacks

Closing remarks

Closing remarks

Security is crucial for CPS

Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:

Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.

Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.

Closing remarks: Security is crucial for CPS

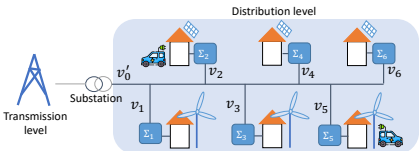
- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.
- Data-driven techniques are very useful, if we can overcome key challenges...

Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First step: A data-driven sensor attack identification algorithm for *linear* networked control systems.

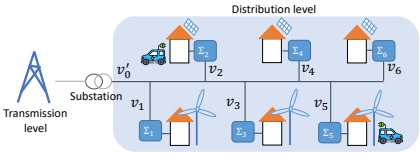
Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First step: A data-driven sensor attack identification algorithm for *linear* networked control systems.
- Towards nonlinear and networked (hybrid) control systems!



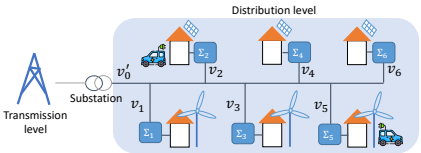
Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First step: A data-driven sensor attack identification algorithm for *linear* networked control systems.
- Intermediate step: Data-driven reachable set computation using potentially corrupted measurements.
- Towards nonlinear and networked (hybrid) control systems!



Closing remarks: Security is crucial for CPS

- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based estimation.
  2. Need $N > M$ for set-based estimation.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First step: A data-driven sensor attack identification algorithm for *linear* networked control systems.
- Intermediate step: Data-driven reachable set computation using potentially corrupted measurements. See our paper in the upcoming CDC!
- Towards nonlinear and networked (hybrid) control systems!

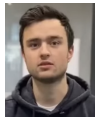
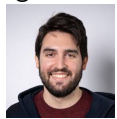


Pantazis, Chong (2026)
Data-driven Reachable Set Estimation with Tunable Adversarial and Wasserstein Distributional Guarantees



Acknowledgments

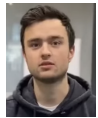
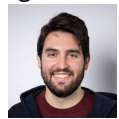
- Current team: Julian Gootzen, Georgios Pantazis, Zeyad Manaa, Viet Nguyen



Acknowledgments

- Current team: Julian Gootzen, Georgios Pantazis, Zeyad Manaa, Viet Nguyen

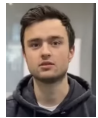
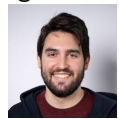





- Collaborators: Nathan van de Wouw, Joao Hespanha, Masashi Wakaiki, Henrik Sandberg, Junsoo Kim, Umar Niazi, Amr Alanwar, Karl Henrik Johansson.

Acknowledgments

- Current team: Julian Gootzen, Georgios Pantazis, Zeyad Manaa, Viet Nguyen



- Collaborators: Nathan van de Wouw, Joao Hespanha, Masashi Wakaiki, Henrik Sandberg, Junsoo Kim, Umar Niazi, Amr Alanwar, Karl Henrik Johansson.

- ProRES consortium, supported by the European Commission (Grant Agreement No. 101069750) and the Dutch Research Council NWO (File No. EP.1602.24.001).

- Marie Skłodowska-Curie Actions (MSCA) Doctoral Networks: SecReSy4You - Security and Resilience of Critical Cyber-Physical Systems.