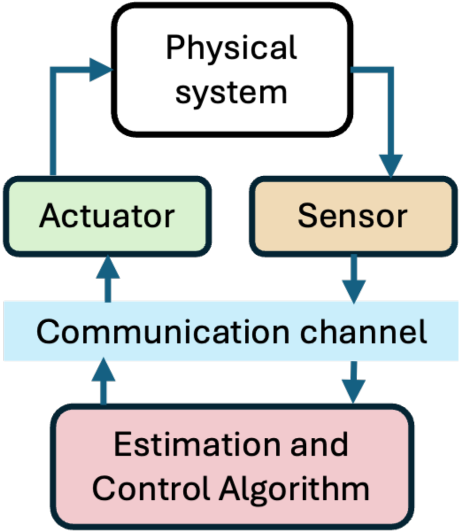
Cyber-physical systems can learn to be secure

CDC workshop on 'Data-Driven Control of Autonomous Systems with Provable Guarantees'.
9th December 2025.

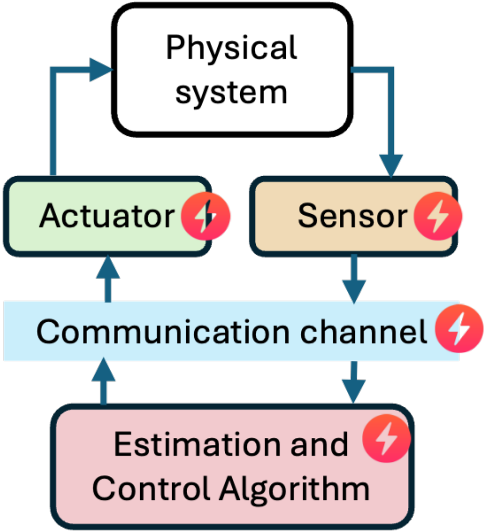
Michelle S. Chong
m.s.t.chong@tue.nl
https://www.michellestchong.com

EINDHOVEN UNIVERSITY OF TECHNOLOGY

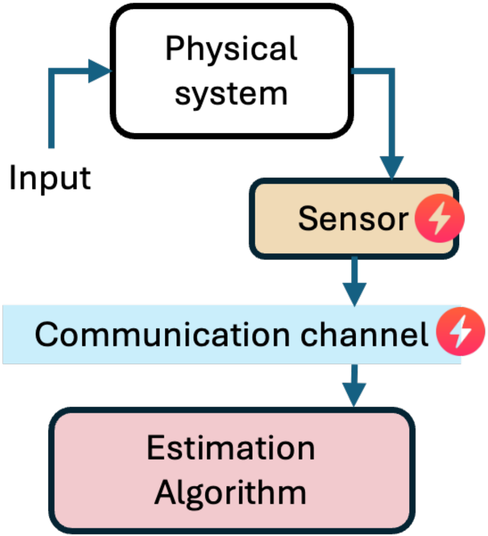
Cyber-Physical Systems are vulnerable



Cyber-Physical Systems are vulnerable



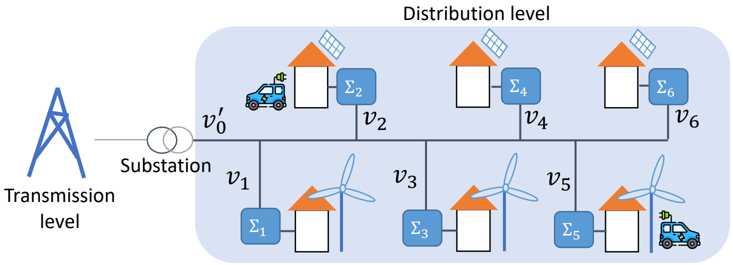
Cyber-Physical Systems are vulnerable



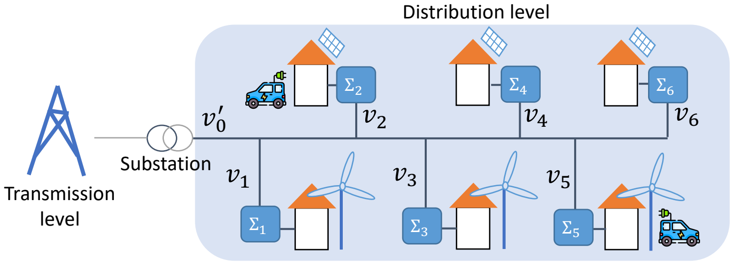
Sensor networks are vulnerable

Source: `toolsense.io`

Sensor networks in power grids are vulnerable

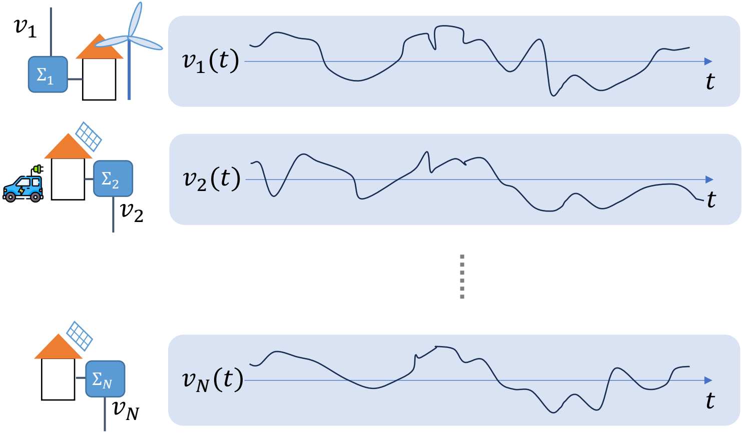


Sensor networks in power grids are vulnerable



The health of the grid is monitored at the substation level.

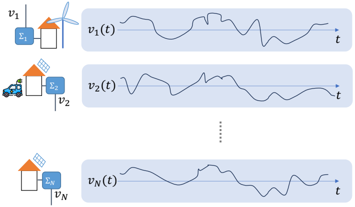
Data may be corrupted



The secure state estimation problem formulation

System with N sensors:

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

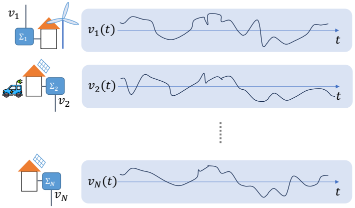


The secure state estimation problem formulation

System with N sensors:

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

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

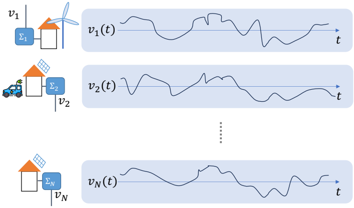


Standing assumptions

- M out of N sensors can be corrupted.

The secure state estimation problem formulation

System with N sensors:



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

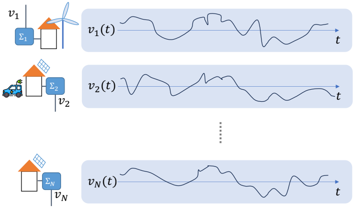
$$\text{DT}: \begin{cases} x(k+1) = f(x(k), u(k)), & k \in \mathbb{N}_{\geq 0}, \\ y_i(k) = h_i(x(k)) + a_i(k), & i \in [N]. \end{cases}$$

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).

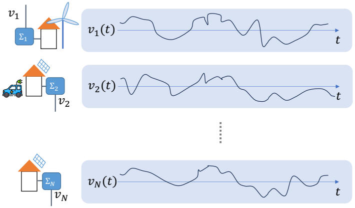
Design an estimator such that the state estimate $\hat{x}$ converges to the true state x

The secure state estimation problem formulation

System with N sensors:

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

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



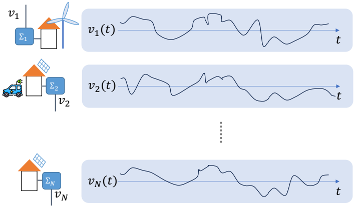
Standing assumptions

- 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).

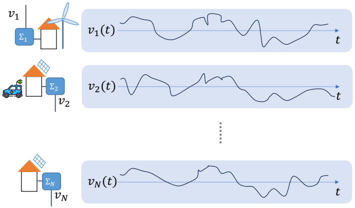
Design an estimator such that the state estimate $\hat{x}$ converges to the true state x with an error bound that is independent of the attack signals a_i.

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).

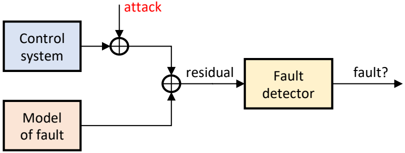
Design an estimator such that the state estimate $\hat{x}$ converges to the true state x with an error bound that is independent of the attack signals a_i.

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

Why are traditional approaches not applicable for security?

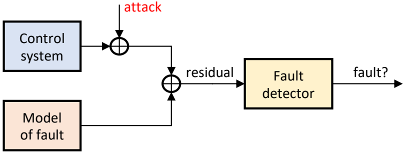
Traditional approaches

1. Fault detection and isolation



Traditional approaches

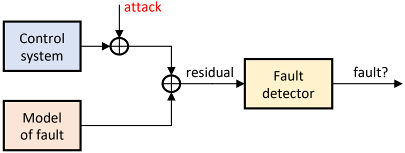
1. Fault detection and isolation



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

Traditional approaches

1. Fault detection and isolation

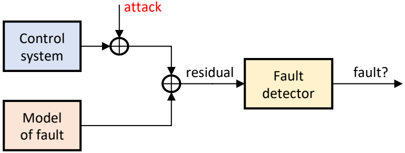


Drawback: Needs a model for each failure mode, which can be many!

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



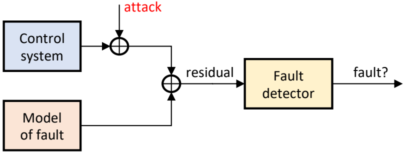
Drawback: Needs a model for each failure mode, which can be many!

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!

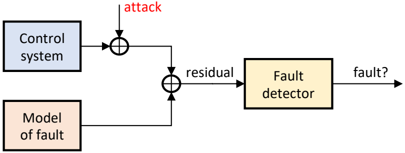
2. Robust control
   - Design system to be robust w.r.t. attacks, which are often treated as *bounded* signals.

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.

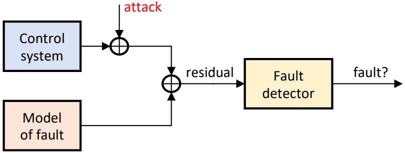
Drawback: Adversarial signals may be unbounded.

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**.

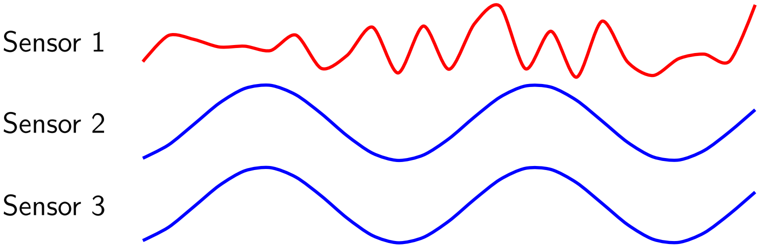
The intuition

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

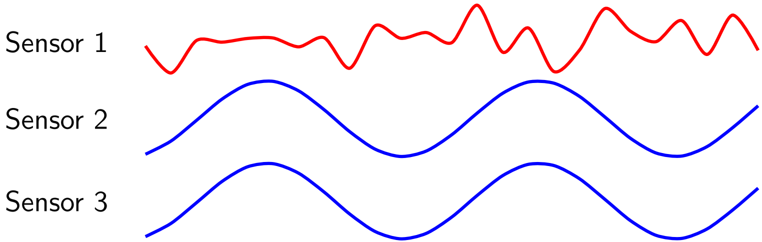
Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 1: One sensor has been corrupted.



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

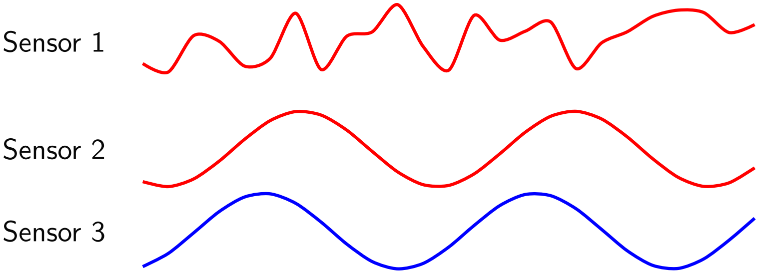
Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 2: Two sensors have been corrupted.

The intuition

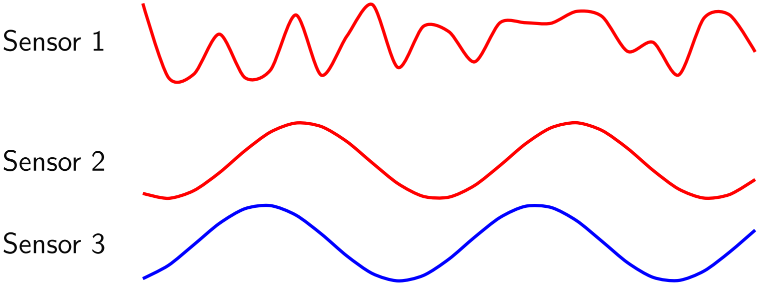
Suppose there are 3 sensors measuring the same system. We know the *number* of sensors which have been corrupted, but not which ones.

Scenario 2: Two sensors have been corrupted.



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.

$\downarrow$

M-attack observability

Theorem: M-attack observability

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

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

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

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

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: 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: 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:

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

From M-attack observability to an algorithm

System with N sensors:

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

Theorem

Suppose the following holds

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

From M-attack observability to an algorithm

System with N sensors:

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

Theorem

Suppose 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 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:

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

Theorem

Suppose the 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 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 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:

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

Theorem

Suppose the 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:

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

Theorem

Suppose the 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

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

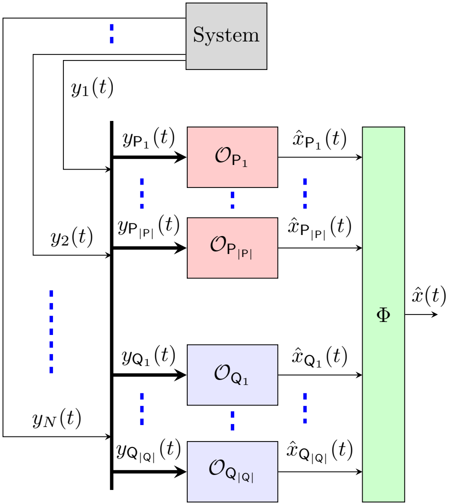
$\forall k \geq 0$, init. cond. $\hat{x}(0)$ and $x(0)$.

From theorem to a

model-based SSE algorithm

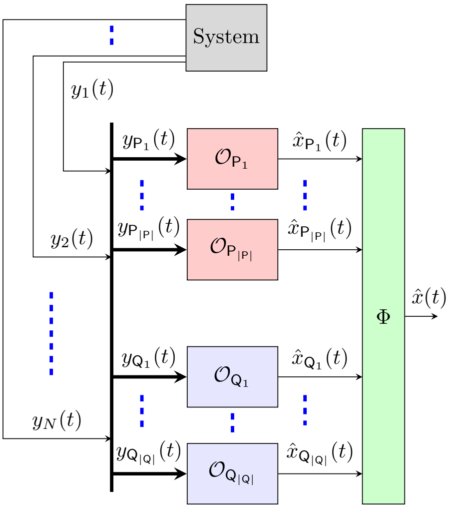
A model-based SSE algorithm

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.



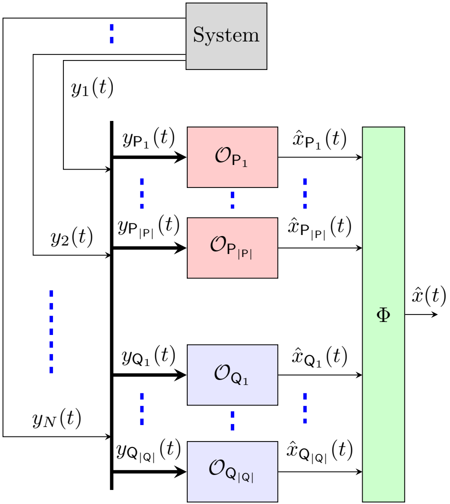
A model-based SSE algorithm

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.



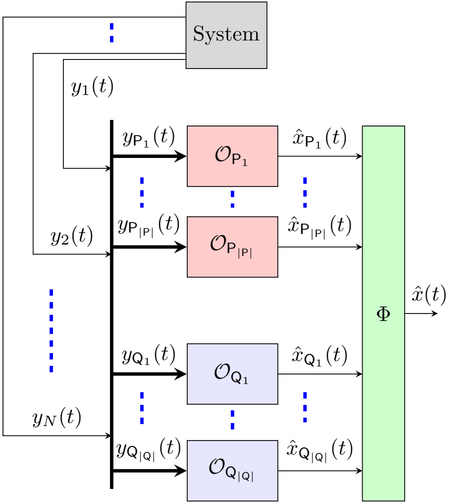
A model-based SSE algorithm

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.

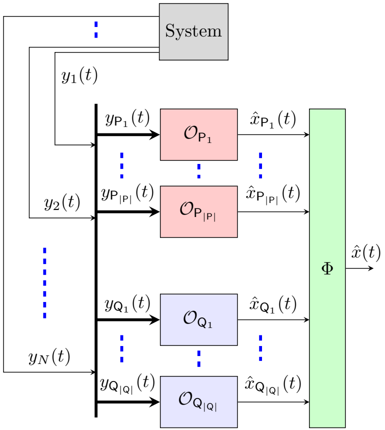


A model-based SSE algorithm

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. Handled by consistency mapping Φ.



The consistency mapping Φ

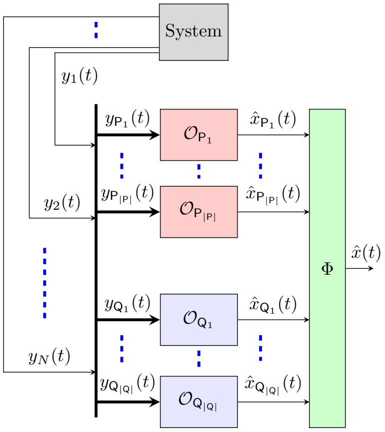


Consistency mapping Φ to choose an estimate $\hat{x}$ from the multi-observer:

$$\pi_P(k) := \max_{Q \subset P, |Q| = N-2M} |\hat{x}_Q(k) - \hat{x}_P(k)|, k \geq 0.$$

$$\hat{x}(k) = \hat{x}_{\sigma(k)}(k), \quad \sigma(k) := \arg\min_{P \subset [N], |P| = N-M} \pi_P(k).$$

The consistency mapping Φ



Consistency mapping Φ to choose an estimate $\hat{x}$ from the multi-observer:

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

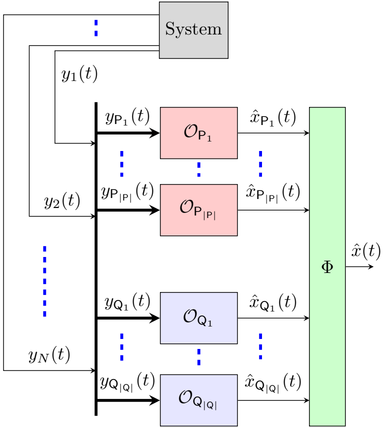
Theorem

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 $x(0)$, $\hat{x}_P(0)$, $\hat{x}_Q(0) \in \mathbb{R}^n$.

The consistency mapping Φ



Consistency mapping Φ to choose an estimate $\hat{x}$ from the multi-observer:

$$\pi_P(k) := \max_{Q \subset P, |Q| = N-2M} |\hat{x}_Q(k) - \hat{x}_P(k)|, \; k \geq 0.$$

$$\hat{x}(k) = \hat{x}_{\sigma(k)}(k), \quad \sigma(k) := \operatorname*{arg\,min}_{P \subset [N], |P| = N-M} \pi_P(k).$$

Theorem

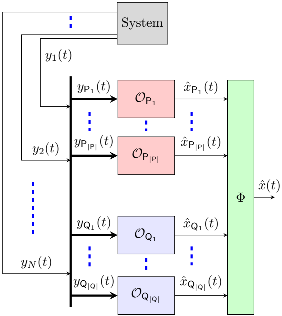
Suppose system is *M-attack observable*, then

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

where $\beta \in \mathcal{KL}$, for all $x(0)$, $\hat{x}_P(0)$, $\hat{x}_Q(0) \in \mathbb{R}^n$.

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

Takeaways: model-based SSE



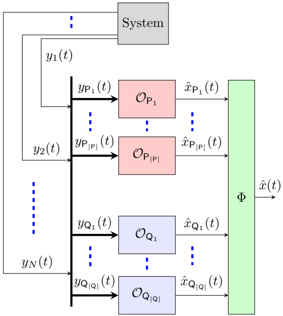
When the model is known, we have

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

References:

- (Chong, Wakaiki, Hespanha; 2015) for LTI

Takeaways: model-based SSE



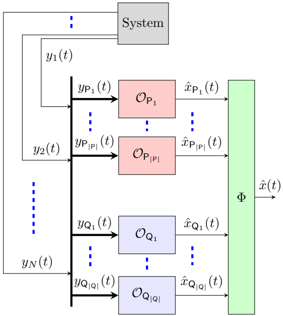
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. When *M* sensors are under attack, we need

References:

- (Chong, Wakaiki, Hespanha; 2015) for LTI

Takeaways: model-based SSE



When the model is known, we have

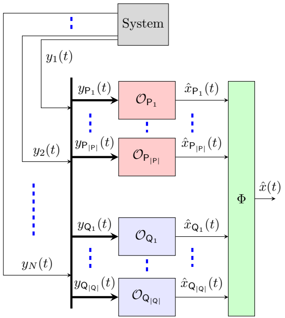
1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. When M sensors are under attack, we need
   (i) $N > 2M$ sensors for

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

References:

- (Chong, Wakaiki, Hespanha; 2015) for LTI

Takeaways: model-based SSE



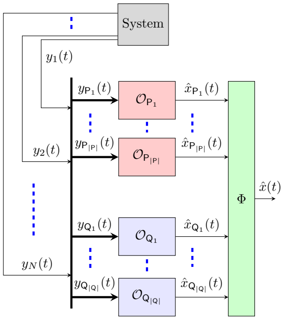
When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. 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; 2015) for LTI and (Chong, Sandberg, Hespanha; 2020) for NL;
- (Chong; 2025) for time-sampled and corrupted measurements;

Takeaways: model-based SSE



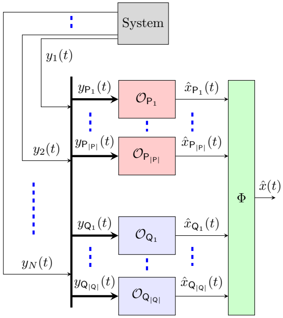
When the model is known, we have

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

References:

- (Chong, Wakaiki, Hespanha; 2015) for LTI and (Chong, Sandberg, Hespanha; 2020) for NL;
- (Chong; 2025) for time-sampled and corrupted measurements;

Takeaways: model-based SSE



When the model is known, we have

1. Necessary and sufficient conditions for **secure state estimation** of LTI systems.
2. When M sensors are under attack, we need
   (i) $N > 2M$ sensors for
   $$|\hat{x}(k) - x(k)| \leq \underbrace{\beta(|\hat{x}(0) - x(0)|, k)}_{\text{independent of attack signals } a}, \forall k.$$
   (ii) $N > M$ sensors for
   $$\underbrace{\hat{x}(k) \in \hat{\mathcal{X}}}_{\text{set-based SSE}} \subseteq \mathcal{X} \leftarrow \text{known}, \forall k.$$

References:

- (Chong, Wakaiki, Hespanha; 2015) for LTI and (Chong, Sandberg, Hespanha; 2020) for NL;
- (Chong; 2025) for time-sampled and corrupted measurements;
- (Niazi, Alanwar, Chong, Johansson; 2023, 2025) on set-based SSE.

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 LTI system with N sensors, where $M < N$ has been attacked:

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

Our approach

Recall that we consider a LTI system with N sensors, where $M < N$ has been attacked:

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

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

Our approach

Recall that we consider a LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 $N - M$

Our approach

Recall that we consider a LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 $j \in [n_{\mathrm{J}}]$, where $n_{\mathrm{J}} := \binom{N}{N-M}$.

Our approach

Recall that we consider a LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 $j \in [n_{\mathrm{J}}]$, where $n_{\mathrm{J}} := \binom{N}{N-M}$.

Since the model (A, B, C_i) are unknown,

Our approach

Recall that we consider a LTI system with N sensors, where $M < N$ has been attacked:

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

- 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 $j \in [n_\mathrm{J}]$, where $n_\mathrm{J} := \binom{N}{N-M}$.

Since the model (A, B, C_i) 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 $j \in n_{\text{J}}$.

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 $j \in n_{\mathrm{J}}$.

$\longrightarrow$

Data-based representation

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

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

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:

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

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

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

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

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 $j \in n_{\mathrm{J}}$.

$\longrightarrow$

Data-based representation

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

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

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 $j \in n_{\text{J}}$.

$\rightarrow$

Data-based representation

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

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

- 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 $j \in n_{\mathrm{J}}$.

$\rightarrow$

Data-based representation

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

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

- 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 $j \in n_{\mathrm{J}}$.

$\rightarrow$

Data-based representation

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

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

- 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 $j \in n_{\mathrm{J}}$.

$\longrightarrow$

Data-based representation

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

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

- 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

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

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

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

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

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

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

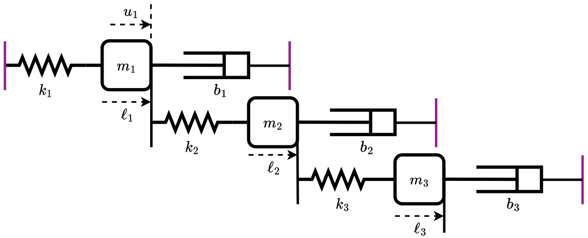
Data-based representation

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

Algorithm

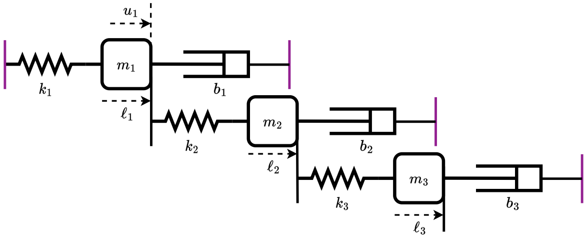
1. **(Offline)** Construct Λ_j, $\forall j \in [n_J]$.
2. **(Offline)** Apply input u of length n, collect online measurements of length T until
$$\text{rank} \begin{bmatrix} U_{n,T} \\ \hat{X}_{j,n,T} \end{bmatrix} = m(n+1) + (N-M)n \text{ with } T \geq (m+1)((m+N-M)n+1).$$
3. **(Online)** Apply *test*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

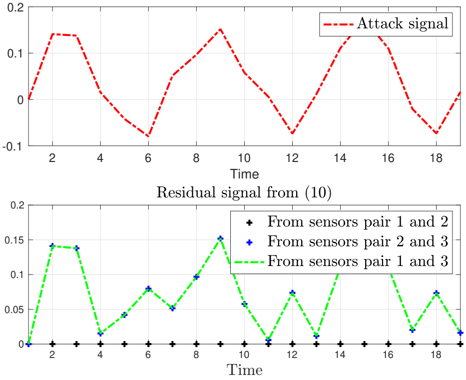
An example



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

An example





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

Takeaways: Data-based identification of attacked sensors

Based on

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

Takeaways: 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

Takeaways: 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 +

Takeaways: 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

Takeaways: 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:

Takeaways: 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

Takeaways: 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

- What about set-based approaches?

Takeaways: 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

- What about set-based approaches? Preliminary work presented at this CDC:

Z. Zhang, M. Niazi, M. Chong, K. Johansson, A. Alanwar
Data-driven Nonconvex Rechability Analysis using Exact Multiplication
`Thursday. C03. 1715-1730. Oceania III.`

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 convergence.

Closing remarks

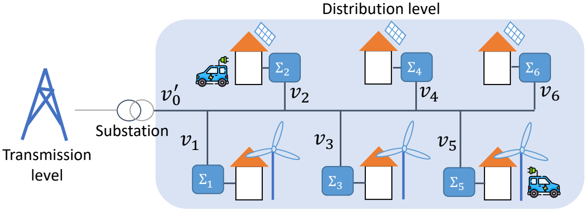
- Security is crucial for CPS.
- Model-based secure state estimation when M sensors are under attack achieved through sensor redundancy:
  1. Need $N > 2M$ for trajectory-based convergence.
  2. Need $N > M$ for set-based convergence.
- Data-driven techniques are very useful, if we can overcome key challenges...

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 convergence.
  2. Need $N > M$ for set-based convergence.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First steps: 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 convergence.
  2. Need $N > M$ for set-based convergence.
- Data-driven techniques are very useful, if we can overcome key challenges...
- First steps: A data-driven sensor attack identification algorithm for *linear* networked control systems.
- Towards nonlinear and networked (hybrid) control systems!



Acknowledgments

RESli consortium,
supported by European Union's Horizon 2020 research and innovation programme under grant agreement No. 883973.

I am hiring!

- Postdoc (3 years)
- PhD (4 years)

Looking for candidates with a strong background and interest in hybrid dynamical systems, control, estimation and optimization.

Join my group at the Eindhoven University of Technology (TU/e) in the Netherlands!

- Proximity and close ties to the high-tech industry in the region.
- TU/e has a vibrant group of active researchers in the area of systems and control.

Get in touch: m.s.t.chong@tue.nl or https://www.michellestchong.com