A Secure Control Learning Framework for Cyber-Physical Systems under Sensor Attacks

Yuanqiang Zhou[1] Kyriakos G. Vamvoudakis[2]
Wassim M. Haddad[2] Zhong-Ping Jiang[3]

[1]Department of Automation, Shanghai Jiao Tong University

[2]School of Aerospace Engineering, Georgia Institute of Technology

[3]Department of Electrical and Computer Engineering, New York University

Jul 12, 2019 / ACC2019

Background

Examples of Cyber Physical System (CPS)

Robots in manufacturing and factories

Transportation networks

Energy systems

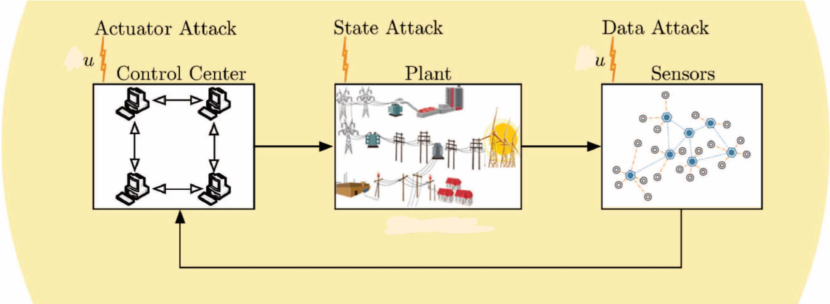
Wearable devices

Augmented reality and Google glasses

Self-driving cars

Motivation

CPSs integrate physical processes, computational resources, and communication capabilities. However, CPSs suffer from specific vulnerabilities [1] .



[1] F. Pasqualetti, F. Dorfler, and F. Bullo, *IEEE Control Systems Magazine*, 2015.

Problem Formulation

System Description

Plant:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0, \quad t \geq 0,$$
$$y_i(t) = C_i x(t), \quad i = 1, \ldots, q.$$

Controller:

$$u(t) = -Ky(t), \quad y \triangleq \left[y_1^{\mathrm{T}}, \ldots, y_q^{\mathrm{T}}\right]^{\mathrm{T}}.$$

Performance measure:

$$J(x_0, u(\cdot)) = \int_0^{\infty} \left[x^{\mathrm{T}}(t)Qx(t) + u^{\mathrm{T}}(t)Ru(t)\right] \mathrm{d}t.$$

Problem Formulation

Sensor Attacks Description

Sensor attack:

$$\tilde{y}_i(t) = y_i(t) + \nu_i(t, y_i(t)), \quad i = 1, \ldots, q,$$

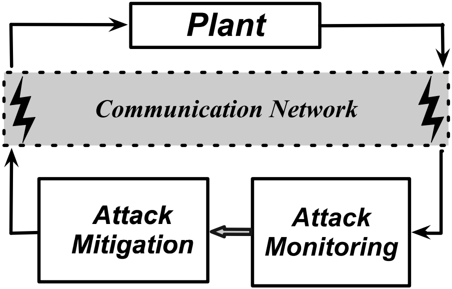
where $\nu_i(t, y_i(t)) \in \mathbb{R}^{p_i}$, $t \geq 0$, denotes an additive attack signal against the ith sensor.

Then the attacker may

- destabilize the system.
- significantly deteriorate the system performance.

Problem Formulation

The Control Architecture



Problem Formulation

Objectives and Contributions

Objectives:

1. check the reliance of the measured outputs in real-time using our attack monitoring process;
2. given the attacked system, design a suboptimal controller which is
   - resilient to sensor attacks;
   - minimizing system performance.

Contributions:

- New technique to select the attack-resilient sensor subset;
- Use of reinforcement learning (RL) to solve the attack mitigation problem.

Problem Formulation

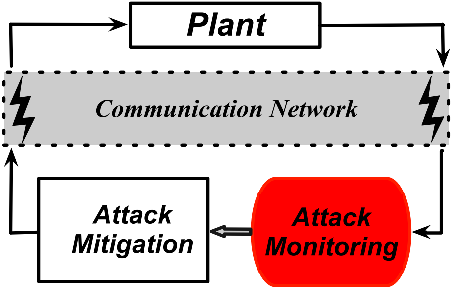
The Method

- Attack Monitoring:
  1. Introducing the threat detection level function.
  2. No need to know the structure of these estimators.
- Attack Mitigation:
  1. Formulate the attack mitigation problem as a two-player, zero-sum differential game.
  2. Use reinforcement learning (RL) to solve the game.
  3. Use data samples to implement the attack mitigation.

Table of Contents

Attack Monitoring

The Attack Monitoring Structure



Attack Monitoring

Basic Assumption

Assumption 1

The system is observable under s attacks [2].

Note that s is the maximum of attacks.

[2] Chong, Michelle S., Masashi Wakaiki, and Joao P. Hespanha, *Proceedings of the American Control Conference*, 2015.

Attack Monitoring

Observer-based Estimator

1. Select $\mathcal{J} \subset \{1, \ldots, q\}$ where

$$\mathrm{Card}(\mathcal{J}) \geq \quad \text{output dimension} - 2 \times \text{maximum of attacks.}$$

2. Design a bank of observers with a common initial condition to obtain

$$\hat{x}_{\mathcal{J}}(t) \ \text{ and } \ \hat{y}_{\mathcal{J}}(t), \quad t \geq 0.$$

3. Associated residual characterized by the subset $\mathcal{J}$ as

$$r_{\mathcal{J}}(t) \triangleq \tilde{y}(t) - C\hat{x}_{\mathcal{J}}(t), \quad t \geq 0,$$

where $C = \left[C_1^{\mathrm{T}}, \ldots, C_q^{\mathrm{T}}\right]^{\mathrm{T}}$.

Attack Monitoring

An Example of Observer Design

Here, we present an example of observer design for $\mathcal{J} \subset \{1, \ldots, q\}$ as

$$\dot{\hat{x}}_{\mathcal{J}}(t) = A\hat{x}_{\mathcal{J}}(t) + Bu(t) + L_{\mathcal{J}}\big(\tilde{y}_{\mathcal{J}}(t) - \hat{y}_{\mathcal{J}}(t)\big),$$
$$\hat{x}_{\mathcal{J}}(0) = \hat{x}(0), \quad t \geq 0,$$
$$\hat{y}_{\mathcal{J}}(t) = C_{\mathcal{J}}\hat{x}_{\mathcal{J}}(t).$$

Defining $\tilde{x}_{\mathcal{J}}(t) \triangleq x(t) - \hat{x}_{\mathcal{J}}(t)$, it follows that

$$\dot{\tilde{x}}_{\mathcal{J}}(t) = A_{\mathcal{J}}\tilde{x}_{\mathcal{J}}(t) - L_{\mathcal{J}}\nu_{\mathcal{J}}(t, y_{\mathcal{J}}), \quad \tilde{x}_{\mathcal{J}}(0) = \tilde{x}(0), \quad t \geq 0.$$

Attack Monitoring

Definition of Threat Detection Level

Recall that $r_{\mathcal{J}}(t) = \tilde{y}(t) - \hat{x}_{\mathcal{J}}(t)$, $t \geq 0$, and then, define the *threat detection level* function

$$\Upsilon_{\mathcal{J}}(t) \triangleq r_{\mathcal{J}}^{\mathrm{T}}(t) r_{\mathcal{J}}(t),$$

for each estimate $\tilde{x}_{\mathcal{J}}(t), t \geq 0$, that uses the subset $\mathcal{J} \subset \{1, \ldots, q\}$.

Given an upper bound

$$\bar{\Upsilon} \in \mathbb{R}_+,$$

a threshold for the attack-free case with unknown initial condition, then we have the following two cases:

- $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon} \quad \Longrightarrow \quad$ a violation of the threat level;
- $\Upsilon_{\mathcal{J}}(t) < \bar{\Upsilon} \quad \Longrightarrow \quad$ the nominal condition.

Attack Monitoring

Determining Attack-Resilient Set

Next, consider the optimization problem

$$\color{red}{\mathcal{O}(t)} = \underset{\mathcal{J}\subset\{1,2,\ldots,q\}:\ \mathrm{Card}(\mathcal{J})=l\leq q}{\arg\min} \Delta_{\mathcal{J}}(t),$$

where

$$\Delta_{\mathcal{J}}(t) = \max\left\{ \left\| \Upsilon_{\mathcal{J}}(t) - \Upsilon_{\mathcal{P}}(t) \right\| \text{ with } \mathcal{P}\subset\mathcal{J} \text{ and } \mathrm{Card}(\mathcal{P}) = q - 2s \right\}.$$

Assumption 2

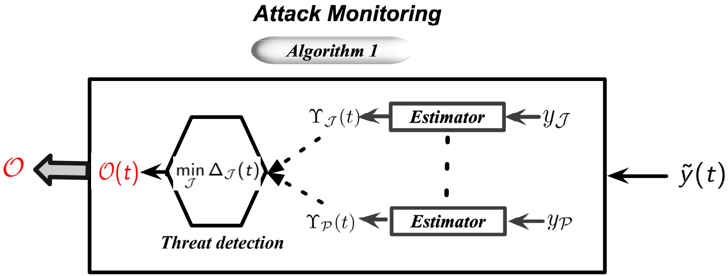
The attack signals $\nu(t, y(t))$, $t \geq 0$, only alter a fixed, albeit unknown, subset of the sensors.

Using Assumption 2, it follows that

$$\color{red}{\mathcal{O} = \mathcal{O}(t)}.$$

Attack Monitoring

Determining Attack-Resilient Set



Attack Monitoring

Attack-Resilient State Estimation

Proposition 1.

When the threat detection level is triggered at some time t, the attack-resilient state estimation $\hat{x}(t), t \geq 0$, is given by

$$\hat{x}(t) = \hat{x}_{\mathcal{O}}(t),$$

where $\hat{x}_{\mathcal{O}}(t), t \geq 0$, is the attack-resilient state estimate generated by the estimator with the set $\mathcal{O}$.

Attack Monitoring

Convergence

Theorem 1.

For every unknown initial condition $x_0 \in \mathbb{R}^n$ and input $u(t), t \geq 0$, the estimate of the attack vector $\hat{\nu}(\cdot)$ is given by

$$\hat{\nu}(t) = \tilde{y}(t) - \hat{x}(t), \quad t \geq 0,$$

and satisfies

$$\|\hat{\nu}(t) - \nu(t, y(t))\| \leq \kappa e^{-\alpha t} \|\tilde{x}(0)\|, \quad t \geq 0,$$

where $\tilde{x}(0) = x_0 - \hat{x}(0)$ and $\kappa, \alpha \in \mathbb{R}_+$.

Attack Monitoring

To Summarize

1 Verify that the system is observable under s attacks.

2 **If** $\Upsilon_{\mathcal{J}}(t) < \bar{\Upsilon}, t \geq 0$, **do**

3 *Physical layer* runs with the nominal output feedback control law.

4 *Control layer* generates all the $\binom{q}{l}$ estimators for every set $\mathcal{J}$ with $\mathrm{Card}(\mathcal{J}) = l \geq q - 2s$.

5 *Control layer* checks the threat detection level for each set $\mathcal{J}$.

6 **Until** One set $\mathcal{J}$ triggers an alarm with $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ at time t.

7 Determine the attack-resilient set $\mathcal{O}$ and reconstruct an estimate of the attack by $\hat{\nu}(t) = \tilde{y}(t) - \hat{x}(t)$, $t \geq 0$.

Attack Monitoring

To Summarize

1. Verify that the system is observable under s attacks.
2. **If** $\Upsilon_{\mathcal{J}}(t) < \bar{\Upsilon}, t \geq 0$, **do**
3. *Physical layer* runs with the nominal output feedback control law.
4. *Control layer* generates all the $\binom{q}{l}$ estimators for every set $\mathcal{J}$ with $\mathrm{Card}(\mathcal{J}) = l \geq q - 2s$.
5. *Control layer* checks the threat detection level for each set $\mathcal{J}$.
6. **Until** One set $\mathcal{J}$ triggers an alarm with $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ at time t.
7. Determine the attack-resilient set $\mathcal{O}$ and reconstruct an estimate of the attack by $\hat{\nu}(t) = \tilde{y}(t) - \hat{x}(t)$, $t \geq 0$.

Attack Monitoring

To Summarize

1. Verify that the system is observable under s attacks.
2. **If** $\Upsilon_{\mathcal{J}}(t) < \bar{\Upsilon}, t \geq 0$, **do**
3. *Physical layer* runs with the nominal output feedback control law.
4. *Control layer* generates all the $\binom{q}{l}$ estimators for every set $\mathcal{J}$ with $\mathrm{Card}(\mathcal{J}) = l \geq q - 2s$.
5. *Control layer* checks the threat detection level for each set $\mathcal{J}$.
6. **Until** One set $\mathcal{J}$ triggers an alarm with $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ at time t.
7. Determine the attack-resilient set $\mathcal{O}$ and reconstruct an estimate of the attack by $\hat{\nu}(t) = \tilde{y}(t) - \hat{x}(t)$, $t \geq 0$.

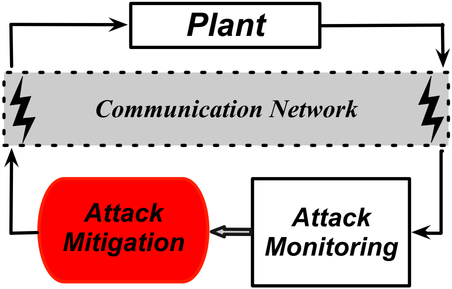
Attack Monitoring

To Summarize

1. Verify that the system is observable under s attacks.
2. **If** $\Upsilon_{\mathcal{J}}(t) < \bar{\Upsilon}, t \geq 0$, **do**
3. *Physical layer* runs with the nominal output feedback control law.
4. *Control layer* generates all the $\binom{q}{l}$ estimators for every set $\mathcal{J}$ with $\mathrm{Card}(\mathcal{J}) = l \geq q - 2s$.
5. *Control layer* checks the threat detection level for each set $\mathcal{J}$.
6. **Until** One set $\mathcal{J}$ triggers an alarm with $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ at time t.
7. Determine the attack-resilient set $\mathcal{O}$ and reconstruct an estimate of the attack by $\hat{\nu}(t) = \tilde{y}(t) - \hat{x}(t),\ t \geq 0$.

Attack Mitigation

The Attack Mitigation Structure



Attack Mitigation

Three Steps of the Attack Mitigation Framework

1. Formulate a two-player, zero-sum differential game with

$$\begin{array}{ccc} \text{the minimizer} & \longleftarrow & \text{the defender} \\ \text{the maximizer} & \longleftarrow & \text{the attacker} \end{array}$$

2. Solve a joint attack-resilient state estimation and attack mitigation problem using RL.
3. Use data samples to implement the RL-based attack mitigation algorithm and analyze the convergence.

Attack Mitigation

Two-Player, Zero-Sum Differential Game

Augmented plant:

$$\dot{\xi}(t) = \mathcal{A}\xi(t) + \mathcal{B}u(t) + \mathcal{D}\nu(t), \quad \xi(t_0) = \xi_0, \quad t \geq 0,$$

where

$$\xi(t) = \begin{bmatrix} \hat{x}_{\mathcal{O}}(t) \\ \tilde{x}_{\mathcal{O}}(t) \end{bmatrix} \in \mathbb{R}^{2n},$$

$\hat{x}_{\mathcal{O}}(t)$ is the attack-resilient state estimation,
$\tilde{x}_{\mathcal{O}}(t)$ is the error between $\hat{x}_{\mathcal{O}}(t)$ and the real state,
$\xi_0 \triangleq \left[\hat{x}^{\mathrm{T}}(0)\ \tilde{x}^{\mathrm{T}}(0)\right]^{\mathrm{T}}$.

Attack Mitigation

Two-Player, Zero-Sum Differential Game

Introducing an attack attenuation level

$$\gamma \in \mathbb{R}_+,$$

we formulate a two-player, zero-sum differential game problem as

$$J^{\odot}\left(\xi_0, u(\cdot), \nu(\cdot)\right) = \int_t^{\infty}\left[\xi^{\mathrm{T}}(\tau)\mathcal{Q}\xi(\tau) + u^{\mathrm{T}}(\tau)Ru(\tau) - \gamma^2\nu^{\mathrm{T}}(\tau)\nu(\tau)\right]\mathrm{d}\tau.$$

Attack Mitigation

The Equivalent Problems

$$\text{Finding a secure control policy while optimizing the performance.}$$
$$\Updownarrow$$
$$\text{Solving the two}-\text{player, zero}-\text{sum differential game.}$$

Attack Mitigation

The Optimal Solution to The Differential Game

The game is given by the dynamics for all $\xi_0 \in \mathbb{R}^{2n}$ and

$$V^\star(\xi) = \min_{u(\cdot)} \max_{\nu(\cdot)} \int_t^\infty \left[\xi^{\mathrm{T}}(\tau)\mathcal{Q}\xi(\tau) + u^{\mathrm{T}}(\tau)Ru(\tau) - \gamma^2 \nu^{\mathrm{T}}(\tau)\nu(\tau)\right] \mathrm{d}\tau.$$

The saddle point solution to the differential game is given by

$$u^\star(\xi) = -\frac{1}{2}R^{-1}\mathcal{B}^{\mathrm{T}} V_\xi^{\star\mathrm{T}},$$
$$\nu^\star(\xi) = \frac{1}{2\gamma^2}\mathcal{D}^{\mathrm{T}} V_\xi^{\star\mathrm{T}}.$$

Attack Mitigation

Existence of the Saddle Point Solution to The Game

Theorem 2.

If there exists $0 \preceq Z \in \mathbb{R}^{2n \times 2n}$ satisfying

$$\mathcal{A}^{\mathrm{T}} Z + Z\mathcal{A} + \mathcal{Q} - Z\big(\mathcal{B}R^{-1}\mathcal{B}^{\mathrm{T}} - \gamma^{-2}\mathcal{D}\mathcal{D}^{\mathrm{T}}\big)Z = 0,$$

then $(u^\star, \nu^\star)$ provides a saddle point solution to this game in the sense that

$$J^{\odot}(\xi_0, u^\star, \nu) \leq J^{\odot}(\xi_0, u^\star, \nu^\star) \leq J^{\odot}(\xi_0, u, \nu^\star),$$

with an optimal value function $V^\star(\xi) = J^{\odot}(\xi_0, u^\star, \nu^\star) = \xi_0^{\mathrm{T}} Z \xi_0$.

Attack Mitigation

RL-Driven Attack Mitigation Algorithm

Using the RL method [3], we can find V^i by solving

$$\mathcal{H}\left(V_\xi^i, u^i, \nu^i\right) = 0,$$

and update the learning-based secure control and attack policies as

$$u^{i+1} = \arg\min_u \mathcal{H}\big(V_\xi^i, u, \nu^{i+1}\big) = -\frac{1}{2} R^{-1} \mathcal{B}^{\mathrm{T}} V_\xi^{i\mathrm{T}},$$

$$\nu^{i+1} = \arg\max_\nu \mathcal{H}\big(V_\xi^i, u^i, \nu\big) = \frac{1}{2\gamma^2} \mathcal{D}^{\mathrm{T}} V_\xi^{i\mathrm{T}}.$$

[3] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, *IEEE Control Systems Magazine*, 2012.

Attack Mitigation

Convergence of the RL-Driven Attack Mitigation Algorithm

Theorem 3.

- *Convergence*:

$$u^{i+1}(\xi) \to u^{\star}(\xi), \quad \nu^{i+1}(\xi) \to \nu^{\star}(\xi) \quad \text{as} \quad i \to \infty,$$

  where

$$V^{i}(\xi) \to V^{\star}(\xi) \quad \text{as} \quad i \to \infty.$$

- *Stability*: The closed-loop system with the RL-driven control and attack policies has an asymptotically stable equilibrium point.

Attack Mitigation

Approximation Using One Critic and Two Actors

Construct three approximators consisting of one critic and two actors as

$$\begin{aligned} \hat{V}^i(\xi(t)) &= \hat{W}_1^{\mathrm{T}} \phi(\xi(t)), \\ \hat{u}^{i+1}(\xi(t)) &= \hat{W}_2^{\mathrm{T}} \varphi(\xi(t)), \\ \hat{\nu}^{i+1}(\xi(t)) &= \hat{W}_3^{\mathrm{T}} \psi(\xi(t)), \end{aligned}$$

where $\phi(\xi) = [\phi_1(\xi), \ldots, \phi_{l_1}(\xi)]^{\mathrm{T}} \in \mathbb{R}^{l_1}$, $\varphi(\xi) = [\varphi_1(\xi), \ldots, \varphi_{l_2}(\xi)]^{\mathrm{T}} \in \mathbb{R}^{l_2}$, and $\psi(\xi) = [\psi_1(\xi), \ldots, \psi_{l_3}(\xi)]^{\mathrm{T}} \in \mathbb{R}^{l_3}$ are suitable basis functions.

Attack Mitigation

The approximation of The Value Function

Substituting the three approximators into the iteration of the value function yields

$$\begin{aligned}
&\hat{W}_1^{i\mathrm{T}}\Big[\phi(\xi(t+\delta t))-\phi(\xi(t))\Big] \\
&=\int_t^{t+\delta t}\Big[-\xi^{\mathrm{T}}(\tau)Q\xi(\tau)-\hat{u}^{i\mathrm{T}}(\tau)R\hat{u}^i(\tau)+\gamma^2\hat{\nu}^{i\mathrm{T}}(\tau)\hat{\nu}^i(\tau)\Big]\mathrm{d}\tau \\
&\quad -2\sum_{k=1}^{m} r_k\int_t^{t+\delta t}\hat{W}_{2,k}^{i\mathrm{T}}\varphi(\xi(\tau))\hat{\zeta}_k^i\mathrm{d}\tau \\
&\quad +2\gamma^2\sum_{j=1}^{q}\int_t^{t+\delta t}\hat{W}_{3,j}^{i\mathrm{T}}\psi(\xi(\tau))\hat{\varsigma}_j^i\mathrm{d}\tau+\mu^i(t),\quad t\geq 0.
\end{aligned}$$

Attack Mitigation

Linear Equation and Parameterized Representation

Considering $t_0, t_1, \ldots, t_N$, with $t_i = t_0 + i\delta t$, and by concatenating, we end up with [4]

$$\left(\Theta^{(i)}\Theta^{(i)\mathrm{T}}\right) \hat{W}^i = \Theta^{(i)}\Pi^{(i)\mathrm{T}},$$

where

$$\hat{W}^i = \left[\hat{W}_1^{i\mathrm{T}}, \hat{W}_{2,1}^{i\mathrm{T}}, \ldots, \hat{W}_{2,m}^{i\mathrm{T}}, \hat{W}_{3,1}^{i\mathrm{T}}, \ldots, \hat{W}_{3,q}^{i\mathrm{T}}\right]^{\mathrm{T}} \in \mathbb{R}^{l_1+l_2\times m+l_3\times q}.$$

[4] H. Modares, F. L. Lewis, and Z. P. Jiang, *IEEE Transactions on Neural Networks and Learning Systems*, 2015.

Attack Mitigation

Determining The Weights of The Three Approximators

Assumption 3.

There exist $\bar{N} \in \mathbb{N}_+$ and $\lambda > 0$ such that, for all $N \geq \bar{N}$,

$$\Theta^{(i)}\Theta^{(i)\mathrm{T}} \succeq \lambda I_{l_1+l_2\times m+l_3\times q},$$

where $\Theta^{(i)} = \left[\theta^i(t_1), \ldots, \theta^i(t_N)\right] \in \mathbb{R}^{(l_1+l_2\times m+l_3\times q)\times N}$.

Using Assumption 3, the weight $\hat{W}^i$ is given by

$$\hat{W}^i = \left(\Theta^{(i)}\Theta^{(i)\mathrm{T}}\right)^{-1}\Theta^{(i)}\Pi^{(i)\mathrm{T}},$$

where

$$\Pi = \left[\pi(t_1), \ldots, \pi(t_N)\right] \in \mathbb{R}^{1\times N}.$$

Attack Mitigation

Convergence of the Data-Based Implementation

Theorem 4.

For $\varepsilon > 0$, there exist integers $i^\star > 0$ and $l^\star > 0$, such that, for $i > i^\star$ and $\min\{l_1, l_2, l_3\} > l^\star$,

$$|\hat{V}^i(\xi) - V^\star(\xi)| < \varepsilon,$$
$$\|\hat{u}^{i+1}(\xi) - u^\star(\xi)\| < \varepsilon,$$
$$\|\hat{\nu}^{i+1}(\xi) - \nu^\star(\xi)\| < \varepsilon.$$

where ξ belongs to a compact set $\Omega \in \mathbb{R}^{2n}$.

Attack Mitigation

To Summarize

1. Run the attack monitoring process of Algorithm 1 until the condition $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ is triggered at some time $t \geq 0$. Select a sufficiently small constant $\varepsilon > 0$ and an integer N satisfying the rank condition.
2. **Repeat**
3. *Data collection:* Define the N different samples as $t_j = j\delta t$, $j = 1, \ldots, N$, and form $\Theta^{(i)}$ and $\Pi^{(i)}$ over the time interval $[0, t]$.
4. *Policy search:* Determine the weights of the three approximators.
5. **Until** $\|\hat{u}^{i+1} - \hat{u}^{i}\| \leq \varepsilon$.
6. Apply $u(t) = \hat{u}^{i+1}(t)$ to the attacked system and use $\hat{\nu}^{i+1}(t)$ as an output amendment to the attacked output.

Attack Mitigation

To Summarize

1 Run the attack monitoring process of Algorithm 1 until the condition $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ is triggered at some time $t \geq 0$. Select a sufficiently small constant $\varepsilon > 0$ and an integer N satisfying the rank condition.

2 **Repeat**

3 *Data collection:* Define the N different samples as $t_j = j\delta t$, $j = 1, \ldots, N$, and form $\Theta^{(i)}$ and $\Pi^{(i)}$ over the time interval $[0, t]$.

4 *Policy search:* Determine the weights of the three approximators.

5 **Until** $\|\hat{u}^{i+1} - \hat{u}^{i}\| \leq \varepsilon$.

6 Apply $u(t) = \hat{u}^{i+1}(t)$ to the attacked system and use $\hat{\nu}^{i+1}(t)$ as an output amendment to the attacked output.

Attack Mitigation

To Summarize

1 Run the attack monitoring process of Algorithm 1 until the condition $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ is triggered at some time $t \geq 0$. Select a sufficiently small constant $\varepsilon > 0$ and an integer N satisfying the rank condition.

2 **Repeat**

3 *Data collection:* Define the N different samples as $t_j = j\delta t$, $j = 1, \ldots, N$, and form $\Theta^{(i)}$ and $\Pi^{(i)}$ over the time interval $[0, t]$.

4 *Policy search:* Determine the weights of the three approximators.

5 **Until** $\|\hat{u}^{i+1} - \hat{u}^i\| \leq \varepsilon$.

6 Apply $u(t) = \hat{u}^{i+1}(t)$ to the attacked system and use $\hat{\nu}^{i+1}(t)$ as an output amendment to the attacked output.

Attack Mitigation

To Summarize

1 Run the attack monitoring process of Algorithm 1 until the condition $\Upsilon_{\mathcal{J}}(t) \geq \bar{\Upsilon}$ is triggered at some time $t \geq 0$. Select a sufficiently small constant $\varepsilon > 0$ and an integer N satisfying the rank condition.

2 **Repeat**

3 *Data collection:* Define the N different samples as $t_j = j\delta t$, $j = 1, \ldots, N$, and form $\Theta^{(i)}$ and $\Pi^{(i)}$ over the time interval $[0, t]$.

4 *Policy search:* Determine the weights of the three approximators.

5 **Until** $\|\hat{u}^{i+1} - \hat{u}^i\| \leq \varepsilon$.

6 Apply $u(t) = \hat{u}^{i+1}(t)$ to the attacked system and use $\hat{\nu}^{i+1}(t)$ as an output amendment to the attacked output.

Illustrative Numerical Example

Consider a system given by

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -4 & 4 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t), \quad t \geq 0,$$

$$y_i(t) = x_1(t), \quad i = 1, 2, 3;$$

$$y_i(t) = x_2(t), \quad i = 4, 5, 6.$$

Note that

- $q = 6$ outputs.
- 1-attack observable.
- $\binom{q}{q-2s} = 15$ sets with $q - 2s = 4$ sensors, 6 sets with 5 sensors, and 1 set with all 6 sensors.

Illustrative Numerical Example

Suppose that at time $t = 2$ s, the system is subjected to the adversarial sensor attack

$$\nu(t, y(t)) = \left[5e^{(2-0.2t)} \cos(2t) + 5\sin(y(t)), 0, 0, 0, 0, 0\right]^{\mathrm{T}}, t \geq 2.$$

Illustrative Numerical Example

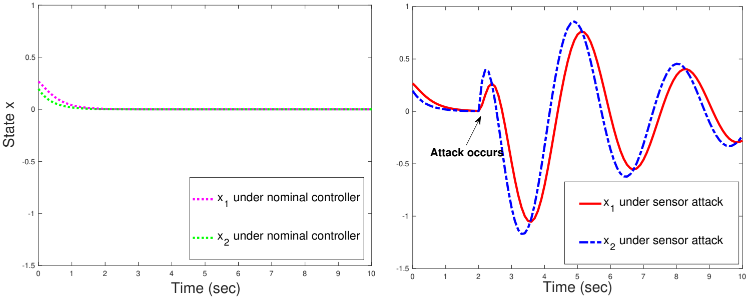


Figure: System performance with nominal controller and sensor attack.

Illustrative Numerical Example

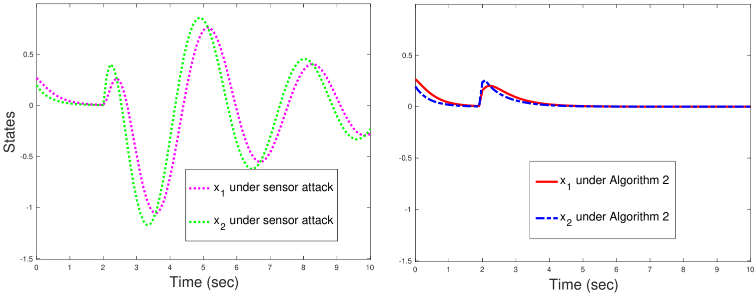


Figure: System performance in the face of adversarial sensor attacks and with Algorithm 2 engaged.

Conclusion and Extensions

Conclusion:

1. A learning-based secure control framework in the presence of sensor attacks.
2. The attack mitigation problem addressed using a secure estimation approach and a game-theoretic architecture.
3. The implementation algorithm based on a RL-driven attack mitigating architecture.

Extensions:

1. Actuator attacks[5].
2. Possibility of the defender and the attacker adapting to their respective control and attack policies.

[5] Y. Zhou, K. G. Vamvoudakis, W. M. Haddad, and Z. P. Jiang, "A secure control learning framework for cyber-physical systems under sensor and actuator attacks, " submitted and under review, 2019.