

1 Modeling of Cyber-Physical Systems

Mathematical modeling for CPS

$$\text{CPSs} \xrightarrow{\text{set of devices that}} \begin{cases} \text{inter-communicate} \\ \text{compute} \\ \text{interact with the physical-world} \end{cases} \xrightarrow{\text{math models}} \begin{cases} \text{Hybrid systems (dynamic switching)} \\ \text{CPSs under attacks (\textbf{Part I})} \\ \text{Multi-agent systems (\textbf{Part II})} \end{cases}$$

$$\text{CPS under attacks (on the sensors)} \longrightarrow \begin{cases} x(k+1) = Ax(k) \\ y(k) = Cx(k) + a(k) \end{cases} \quad \text{no } u(k), \text{ no } b(k) \text{ (actuators)}$$

$$A \in \mathbb{R}^{n,n}, \quad C \in \mathbb{R}^{q,n} \quad a(k) \in \mathbb{R}^q, \quad q = \text{number of sensors}$$

Secure State Estimation of CPSs [FUSION CENTER]

Without attacks Observability $\iff \text{rank}(\mathcal{O}_n) = n$, $\mathcal{O}_n = [C \ CA \ \dots \ CA^{n-1}]^T$ (observability matrix) $\Rightarrow x(0)$ found by pseudo-inversion of the system $y = \mathcal{O}_n x(0) \longrightarrow x(0) = \mathcal{O}_n^\dagger y$

$$\text{Presence of attacks no model for the attacks} \implies \begin{cases} y(0) = Cx(0) + a(0) \\ y(1) = CAx(0) + a(1) \\ \vdots \\ y(T) = CA^{T-1}x(0) + a(T) \end{cases} \quad (\infty \text{ solutions})$$

Assumption (sparsity) $\|a\|_0 \leq h \ll q$ (by adding this the problem may have a **unique solution**)

Under which conditions can we solve the problem? \rightarrow Simplification of the problem

$$A = I_n \text{ (the system is static)} \longrightarrow x(k+1) = x(k) = x \xrightarrow{\text{problem becomes}} y = Cx + a \quad \text{s.t. } \|a\|_0 \leq h$$

Prop(Correctability) (resilience to h attacks) $\iff \forall z \in \mathbb{R}^n, \|\mathcal{O}_T z\|_0 > 2h$

A necessary condition for correctability... $q \geq 2h + n$ $\begin{cases} \text{large } q \text{ is not sufficient} \\ \text{a minimum } q \text{ is required} \end{cases}$

How can I solve the problem? \rightarrow Reformulation of the problem

$$\begin{aligned} 0. & \text{ if } \textbf{Correctable}, \text{ the Decoder } \mathcal{D}_0 \doteq \min_{x \in \mathbb{R}^n, a \in \mathbb{R}^q} \|a\|_0 \text{ s.t. } y = Cx + a \text{ corrects } h \text{ attacks;} \\ 1. & y = Cx + a + \text{noise} \longrightarrow y \sim Cx + a \longrightarrow \min_{x \in \mathbb{R}^n} \frac{1}{2} \|y - Cx\|_2^2 \text{ (Least-Squares problem)} \\ 2. & \ell_0\text{-norm} \Rightarrow \text{bad function} \xRightarrow{\text{convex relaxation}} \ell_1\text{-norm (best convex approximation)} \leftarrow \begin{cases} \text{promotes sparsity} \\ \text{convex and continuous} \end{cases} \end{aligned}$$

$$(0)+(1)+(2) \implies z = \underset{x \in \mathbb{R}^n, a \in \mathbb{R}^q}{\text{argmin}} \frac{1}{2} \|y - Gz\|_2^2 + \lambda \|a\|_1 \quad G = \begin{pmatrix} C & I \end{pmatrix}, z = \begin{pmatrix} x \\ a \end{pmatrix} \iff \begin{cases} \text{unconstrained} \\ \text{non differentiable in 0} \end{cases}$$

(Partial) LASSO $\xrightarrow{\text{solved by using}} x(k+1) = \mathbb{S}_{\lambda\tau}[x(k) + \tau C^T(y - Cx(k))]$ (**IST** algorithm), $k \rightarrow$ iteration
 \iff derived from the **alternating minimization of the surrogate functional** $\mathcal{R}(x, b)$

In general for CPSs x is not sparse $\longrightarrow \Lambda = (0, 0, \dots, 0, \lambda, \lambda, \dots, \lambda)$, $\lambda_i = 0, i = 1, \dots, n$

(Indoor) Localization by RSS-fingerprinting [FUSION CENTER]

0. **Init:** cell-grid discretization ($p = \#$ of cells, $q = \#$ of sensors, $N_t = \#$ of targets)
1. **Training phase:** Dictionary $D \in \mathbb{R}^{q,n}$ is built. Target in each cell + measurement
2. **Runtime phase:** each sensor takes $y_i \longrightarrow$ **Where are the targets?** $y = Dx + \eta$, $x_i \in \{0, 1\}$
 $x_i = 1 \rightarrow$ in the i -th cell there is a target $\implies \min_{x \in \{0,1\}^p} \|y - Dx\|_2^2$ (mixed-integer \rightarrow NP-Hard)
 - (a) $x \in \{0, 1\}^p \xrightarrow{\text{relaxation}} x \in \mathbb{R}^p$ so that the problem is feasible;
 - (b) $N_t \ll p \Rightarrow$ seeking of a **sparse solution** \implies LASSO

(#1) Localization without attacks $x^* = \min_{x \in \mathbb{R}^p} \frac{1}{2} \|y - Dx\|_2^2 + \lambda \|x\|_1$ (ISTA can be used)

(#2) Localization under sensor attacks $x^* = \min_{z \in \mathbb{R}^p} \frac{1}{2} \|y - Gz\|_2^2 + \lambda \|z\|_1$, $G = \begin{pmatrix} D & I \end{pmatrix}$, $z = \begin{pmatrix} x \\ a \end{pmatrix}^T$

Dynamic Secure State Estimation of CPSs

[FUSION CENTER]

In general $A \neq I_n \implies$ The system is moving and the static-batch approach is too slow

How to dynamically estimate the state (attack-free)?

Luemberger Observer $\xRightarrow{\text{def}}$ $\begin{cases} \hat{x}(k+1) = A\hat{x}(k) - L[\hat{y}(k) - y(k)] \\ \hat{y}(k) = C\hat{x}(k) \end{cases} \xrightarrow{e(k)=\hat{x}-x} e(k+1) = (A-LC)e(k)$
 (copy of the system + correction)

L is the **observer-gain matrix**, chosen such that $e(k+1)$ is asymptotically stable $\xrightarrow{k \rightarrow \infty} \hat{x}(k) = x(k)$

Online Gradient Descent (OGD) $L_g = \tau AC^T$, $\tau \leq \|C\|_2^{-2}$

What about the attacks?

In general CPSs under attacks are **not observable** \longrightarrow NO Luemberger Observer, NO OGD

SPARSE OBSERVER $\xrightarrow[\text{given } \hat{z} \text{ and } y=Gz]{k \text{ is the time}} \begin{cases} \hat{z}^+(k) = A\hat{z}(k) - \tau AG^T[G\hat{z}(k) - y(k)] & \text{estimation (OGD)} \\ \hat{x}(k+1) = A\hat{x}^+(k) & \text{prediction} \\ \hat{a}(k+1) = \mathbb{S}_{\lambda\tau}[\hat{a}^+(k)] & \text{sparsify the attacks} \\ \hat{y}(k) = G\hat{z}(k) & \text{measurement} \end{cases}$

APPLICATION: LOCALIZATION UNDER SENSOR ATTACKS (WITH MOVING TARGETS)

Distributed Secure State Estimation of CPSs

[NO FUSION CENTER]

Multi-agent system set of systems which cooperate in order to achieve a **common goal**.

Agents $\xRightarrow{\text{exchange}} \text{local estimate } x^{(i)}(k) \xrightarrow{\text{to the}} \text{neighbourhood}$

Graph $\mathcal{G} = (\mathcal{N}, \mathcal{E}) \rightarrow$ math tool to model the inter-communication. It is made up of:

- A set \mathcal{N} of agents or nodes or vertices;
- A set $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ of links or edges

Neighbourhood $\mathcal{N}_i \rightarrow$ set of the nodes from which receives information.

Local mean $\bar{x}^{(i)}(k) = \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} x^{(j)}(k)$

CONSENSUS $x(k+1) = Qx(k) \longrightarrow$ the system organized how Q dictates, achieves a **global common decision** if Q has got some properties.

Stochastic matrix Q is stochastic $\iff \forall i = 1, \dots, q \quad \sum_{j=1}^q Q_{ij} = 1$ (**row stochastic**)

Doubly stochastic matrix Q is doubly stochastic \iff is symmetric (undirected graphs)

Some important results:

$\longrightarrow \lambda_1 = \lambda_{PF} > |\lambda_2| > \dots > |\lambda_q| \implies \lim_{k \rightarrow \infty} x(k) = \alpha \mathbf{1}$ (suff. conditions for Consensus)

$\longrightarrow \lambda_{PF} > |\lambda_2| > \dots > |\lambda_q|$, Q doubly stochastic $\xRightarrow{\text{average consensus}} \lim_{k \rightarrow \infty} x(k) = \alpha \mathbf{1}$, $\alpha = \frac{1}{q} \sum_{j=1}^q x^{(j)}(0)$

\longrightarrow **Convergence rate** of the consensus algorithm $\xrightarrow{\text{is related to}} \text{esr}(Q) = \lambda_2$

\longrightarrow If $Q^T = \text{Adjacency matrix}$ describes a strongly connected graph $\implies Q$ achieves consensus.

APPLICATIONS of CONSENSUS ALGORITHM

Distributed Least-Squares $\xrightarrow{\text{minimize}} F(x) = \|y - Ax\|_2^2$, each sensor has $\begin{cases} y_i & \text{its own measurement} \\ A_i & i\text{-th row of the matrix } A \end{cases}$

\implies **Distributed Gradient Descent (DGD)** $\xrightarrow[\forall i \in \{1, \dots, q\}]{\text{the solution can be found by using...}} \begin{cases} \bar{x}^{(i)}(k) = \sum_{j=1}^q Q_{ij} x^{(j)}(k) \\ x^{(i)}(k+1) = \underbrace{\bar{x}^{(i)}(k)}_{\text{consensus step}} - \underbrace{\tau \nabla F(x^{(i)}(k))}_{\text{gradient step}} \end{cases}$

Distributed LASSO $\xrightarrow{\text{minimize}} F(x) = \|y - Ax\|_2^2 + \lambda \|x\|_1$

\implies **Distributed ISTA (DISTA)** $\xrightarrow[\forall i \in \{1, \dots, q\}]{\text{the solution can be found by using...}} x^{(i)}(k+1) = \mathbb{S}_{\lambda\tau}[\bar{x}^{(i)}(k) + \tau A_i^T (y - A_i x^{(i)}(k))]$

Note that... for these algorithms proofs of convergence have been provided.

2 Control of Cyber Physical Systems

Agents' experimental modeling (System Identification)

Modeling of the communication network

State Variable FeedBack control (SVFB)

Formation control