# Compensation Estimation Algorithm for Multi-Sensor Attack Detection in Autonomous Driving

CHEN Xingzhou

HKUST[1],

2023/03/23

# System Model

- multi-sensor system model

$$x_{t+1} = Ax_t + Bu_t + w_t$$

$$\begin{bmatrix} y_t^1 \\ y_t^2 \\ \vdots \\ y_t^n \end{bmatrix} = \begin{bmatrix} C^1 \\ C^2 \\ \vdots \\ C^n \end{bmatrix} x_t + \begin{bmatrix} v_t^1 \\ v_t^2 \\ \vdots \\ v_t^n \end{bmatrix}$$

- Kalman filter (multi-sensor fusion algorithm)

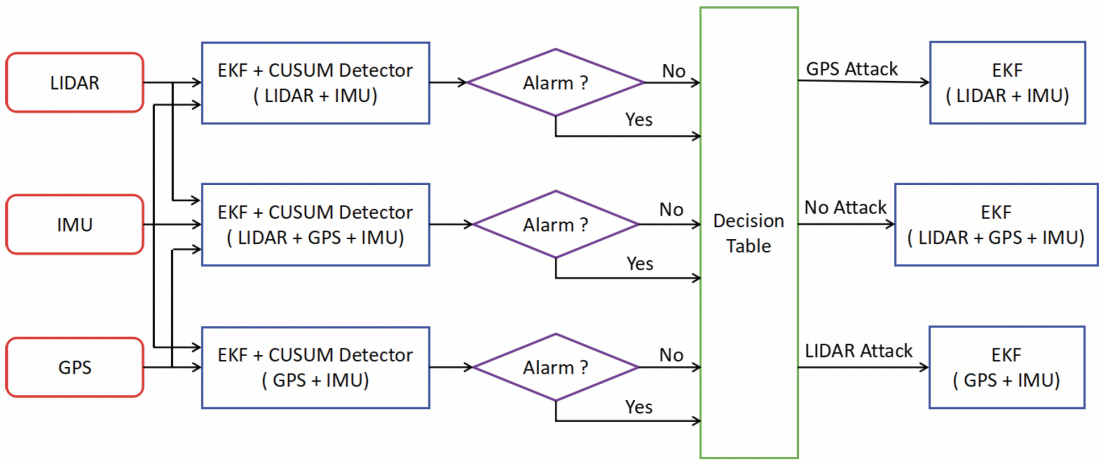
$$\widehat{x}_{t+1|t} = A\widehat{x}_{t|t} + Bu_t$$

$$\begin{bmatrix} r_t^1 \\ r_t^2 \\ \vdots \\ r_t^n \end{bmatrix} = \begin{bmatrix} y_t^1 \\ y_t^2 \\ \vdots \\ y_t^n \end{bmatrix} - \begin{bmatrix} C^1 \\ C^2 \\ \vdots \\ C^n \end{bmatrix} \widehat{x}_{t+1|t} = \begin{bmatrix} C^1 \\ C^2 \\ \vdots \\ C^n \end{bmatrix} (x_{t+1} - \widehat{x}_{t+1|t}) + \begin{bmatrix} v_t^1 \\ v_t^2 \\ \vdots \\ v_t^n \end{bmatrix}$$

$$\widehat{x}_{t+1|t+1} = \widehat{x}_{t+1|t} + \begin{bmatrix} K^1 & K^2 & \cdots & K^n \end{bmatrix} \begin{bmatrix} r_t^1 \\ r_t^2 \\ \vdots \\ r_t^n \end{bmatrix}$$

# Background

## Mo's paper [1]




- main idea: detect the $\{r_t^i\}$ separately

$$\begin{bmatrix} r_t^1 \\ r_t^2 \\ \vdots \\ r_t^n \end{bmatrix} = \begin{bmatrix} y_t^1 \\ y_t^2 \\ \vdots \\ y_t^n \end{bmatrix} - \begin{bmatrix} C^1 \\ C^2 \\ \vdots \\ C^n \end{bmatrix} \widehat{x}_{t+1|t}$$

- limitation
  - hard to distinguish between large errors and stealthy attacks
  - the time lag between attack and detection has an impact on state estimate

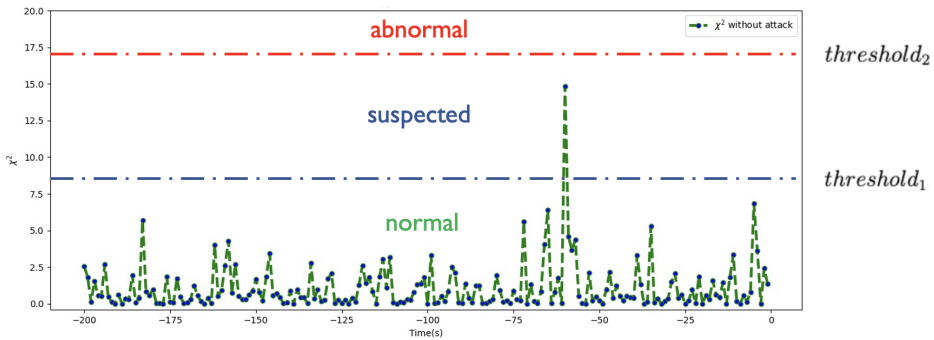
# Our motivation

$$\chi^2 = \|r_t\|^2$$

- The value of $\chi^2$ is positively correlated with the probability of being attacked, and we compare it to a fixed threshold.

| | $\chi^2 \leq threshold_1$ | $threshold_1 < \chi^2 \leq threshold_2$ | $threshold_2 < \chi^2$ |
|---|---|---|---|
| system error | ✓ | ✓ | |
| sensor attack | | ✓ | ✓ |

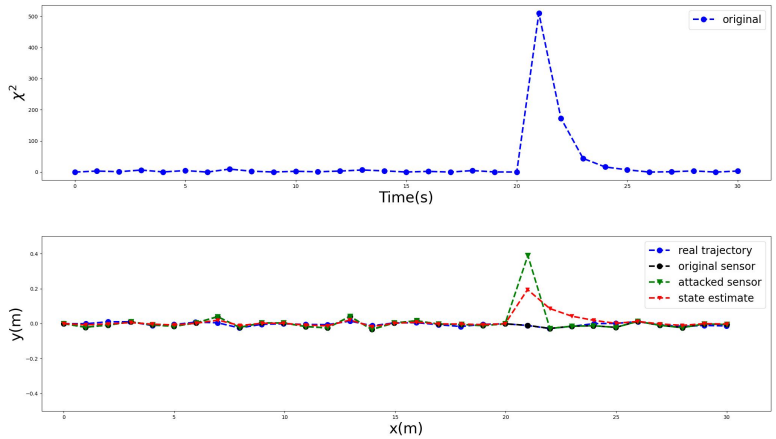
- classify sensor status into three categories
  - **normal**: trust this sensor and judge the data as normal
  - **suspected**: compensate the data from other functioning sensors
  - **abnormal**: ignore this data



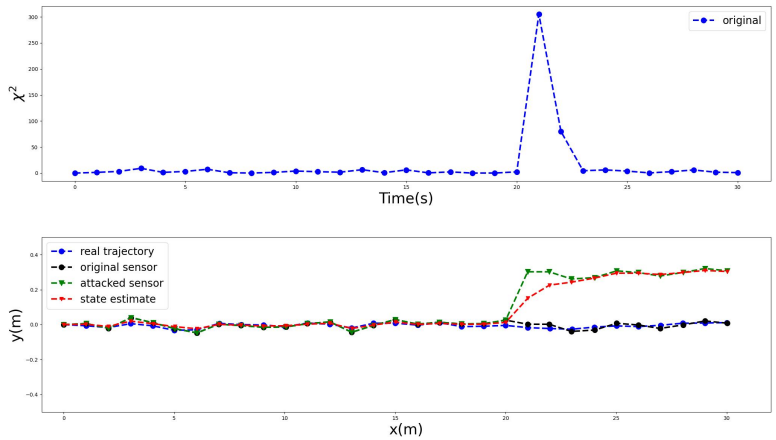
# Our motivation

How to distinguish between these two cases? Compensation!

**Normal exception:** occasional offset




**Malicious attack:** constant bias



# Problem formulation

How to compensate the suspected sensor data?

$$\begin{bmatrix} r_t^1 \\ r_t^2 \\ \vdots \\ r_t^n \end{bmatrix} = \begin{bmatrix} C^1 \\ C^2 \\ \vdots \\ C^n \end{bmatrix} (x_t - \widehat{x}_{t|t-1}) + \begin{bmatrix} v_t^1 \\ v_t^2 \\ \vdots \\ v_t^n \end{bmatrix} + \begin{bmatrix} \vdots \\ \xi_t^i \\ \vdots \\ \vdots \end{bmatrix}$$

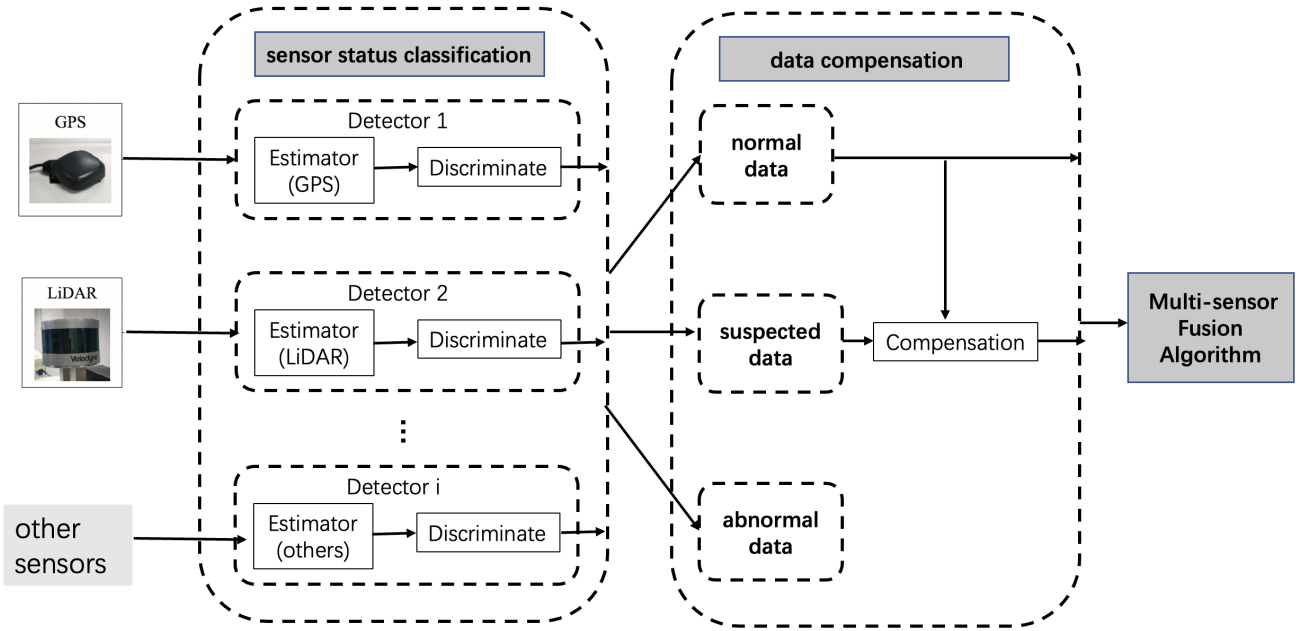
## Formulation

$$\begin{aligned} \min_{\Delta x_t, \xi_t^1, \cdots, \xi_t^n} \quad & J = \alpha \left\| \Delta x_t \right\| + \sum \left\| \xi_t^i \right\| \\ \text{s.t.} \quad & \left\| r_t^i - C^i \Delta x_t - \frac{1+sgn(\left\| r_t^i \right\| - \tau^i)}{2} \xi_t^i \right\| < \tau^i, i = 1, ..., N \end{aligned} \tag{1}$$

## Interpretation

- Calculating compensation is equivalent to estimating the attack signal
- $\{C^1, C^2, \cdots, C^n\}$ contains similar information, so as the $\{r_t^1, r_t^2, \cdots, r_t^n\}$
- when $\left\| r_t^i \right\| < \tau^i$, $\frac{1+sgn(\left\| r_t^i \right\| - \tau^i)}{2} = 0$, this sensor will be regarded as normal
- when $\left\| r_t^i \right\| > \tau^i$, $\frac{1+sgn(\left\| r_t^i \right\| - \tau^i)}{2} = 1$, this sensor will get a compensation

# Algorithm



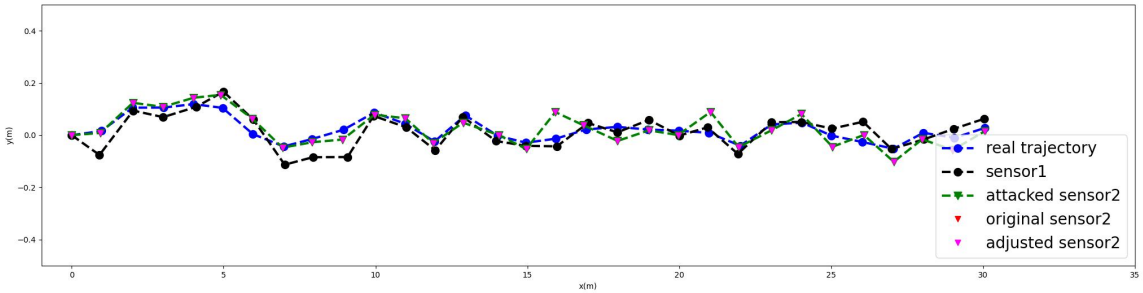

**Algorithm 1** proposed compensation algorithm

**Input:** $x_{t+1|t}, P_{t+1|t}, \{y_t^i\}, \{C^i\}, \{Q^i\}$
**Output:** $x_{t+1|t+1}, P_{t+1|t+1}$
**Step1:**
**for** $i \leftarrow 1$ **to** $n$ **do**
    the residue $r_t^i = y_t^i - C^i x_{t+1|t}$
    calculate the residue's distribution $\sum_{r_t^i} = Q^i + C^i P_{t+1|t} C^{iT}$
    get the $\chi^2$ value and classify the sensor data by comparing to the threshold
**end for**
**Step2:**
    solve the optimization problem and calculate the compensation $\{\xi_t^j\}$
**Step3:**
    run multi-sensor fusion algorithm with sensor data $\{y_t^i\}$ and $\{y_t^j + \xi_t^j\}$
    update the state estimate and return $x_{t+1|t+1}, P_{t+1|t+1}$

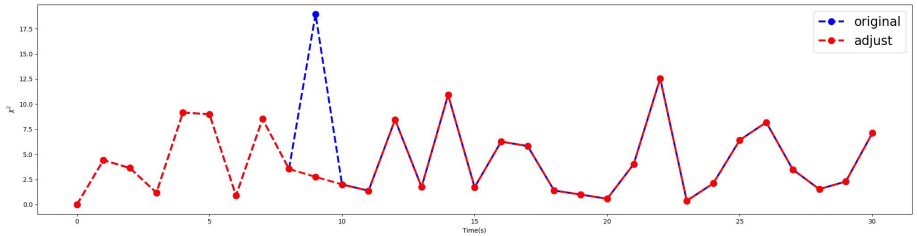
# Simulation

**Experiment 1:** without attack

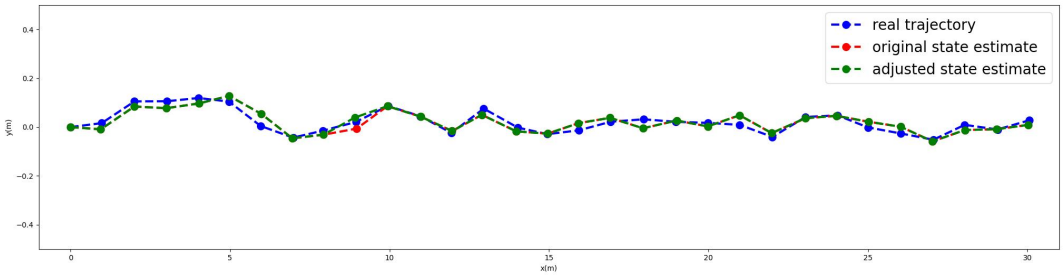
- sensor data with compensation




- adjustment of $\chi^2$ values



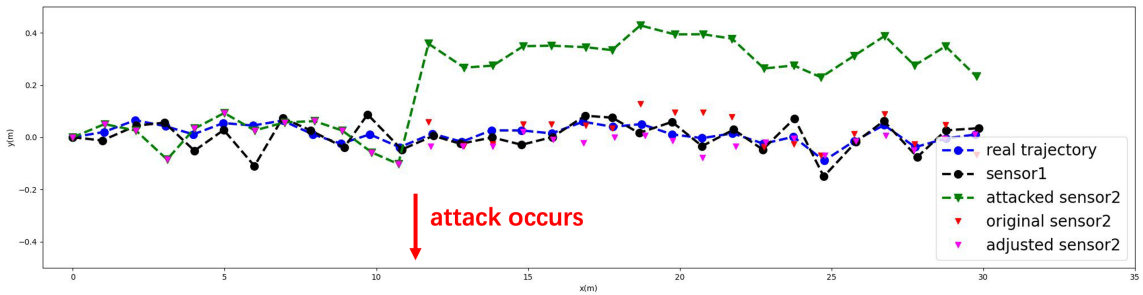

- state estimation before and after compensation



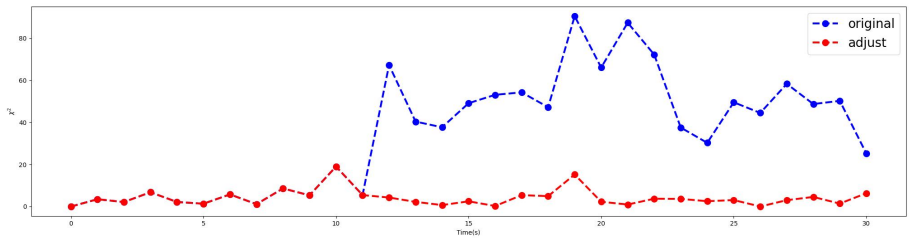
# Simulation

**Experiment 2:** under certain bias attack

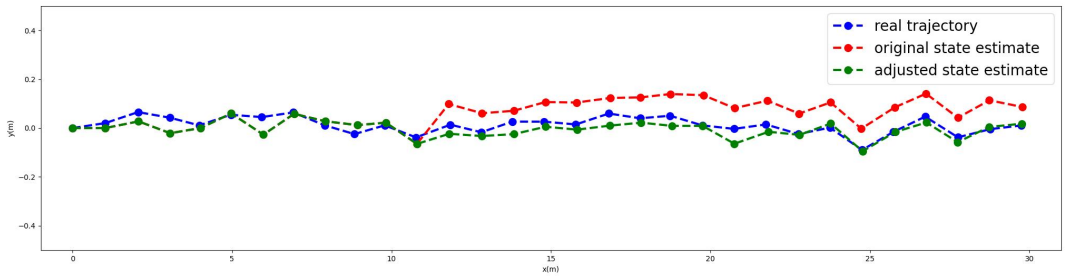
- sensor data with attack and compensation




- adjustment of $\chi^2$ values



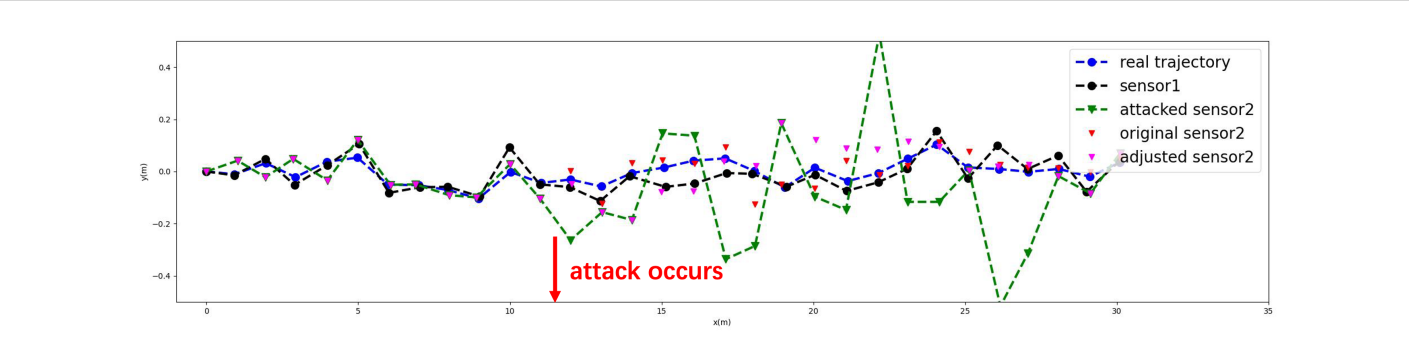

- state estimation before and after compensation



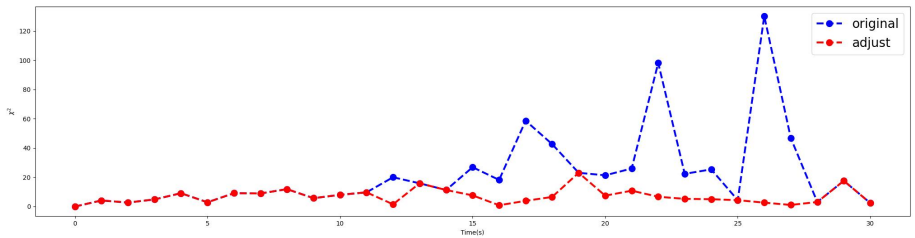
# Simulation

**Experiment 3:** under random noise attack

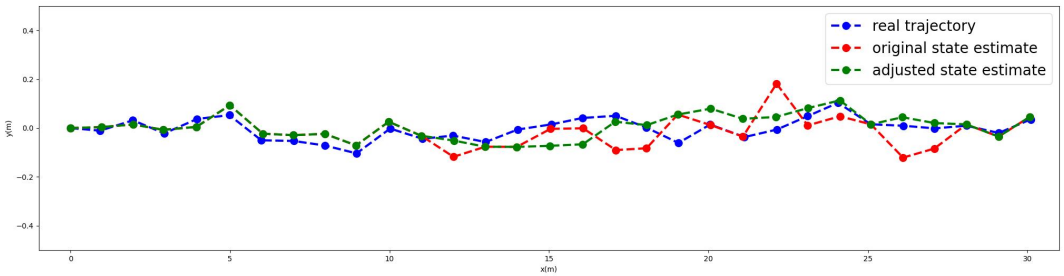
- sensor data with attack and compensation




- adjustment of $\chi^2$ values



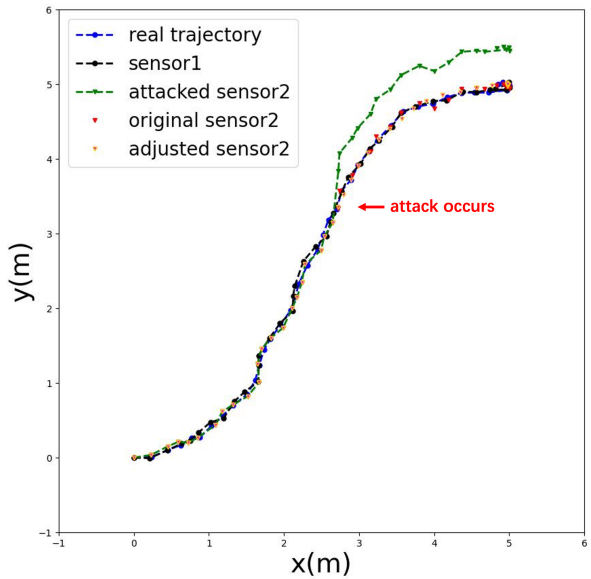

- state estimation before and after compensation

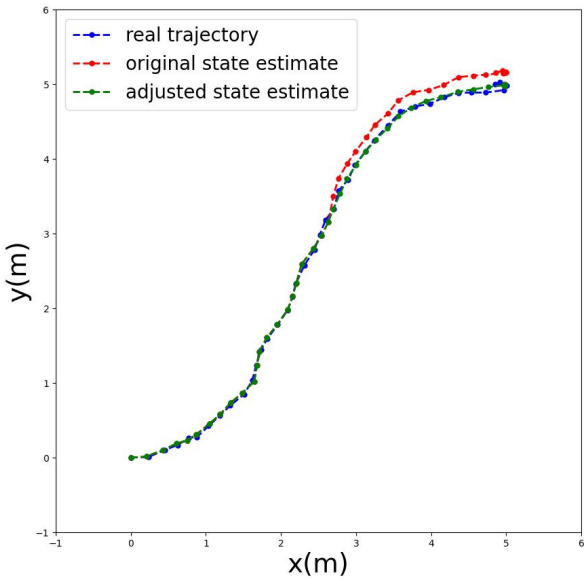


# Simulation

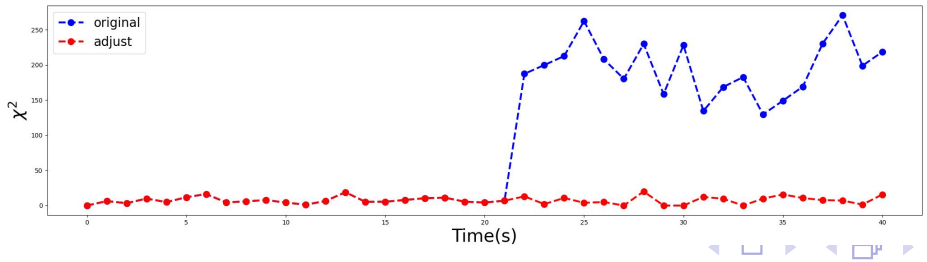
**Experiment 4:** constant bias attack in nonlinear system

- sensor data/ with attack and compensation




- adjustment of $\chi^2$ values



*Thank You*