SCIENCE CHINA
Information Sciences

• **Supplementary File** •

Nonlinear Fusion Estimation for False Data Injection Attack Signals in Cyber-Physical Systems

Yawen TAN[1,2], Pindi WENG[1,2], Bo CHEN[1,2*] & Li YU[1,2]

[1]*Department of Automation, Zhejiang University of Technology, Hangzhou* 310023*, China;*
[2]*Institute of Cyberspace Security, Zhejiang University of Technology, HangZhou* 310023*, China*

Appendix A Theorem 1 and selection criteria

Before deriving the main results, let us define

$$\begin{cases} F_i^{(j)}(k) = \left.\dfrac{\partial f(x(k))}{\partial x(k)}\right|_{x(k)=\hat{x}_i^{(j)}(k)} \\ H_i^{(j)}(k) = \left.\dfrac{\partial h_i(x(k))}{\partial x(k)}\right|_{x(k)=x_{p,i}^{(j)}(k)} \end{cases} \tag{A1}$$

where $x_{p,i}^{(j)}(k)$ will be determined in Theorem 1.

Theorem 1. Given $\hat{x}_i^{(j)}(k-1)$, $\Upsilon_i^{(j)}(k-1)$, $\hat{a}_i^{(j)}(k-1)$, $S_i^{(j)}(k-1)$, $0<\lambda_i\leqslant 1$, the intermediate variables $\bar{a}_i^{(j)}(k)$, $\bar{x}_i^{(j)}(k)$ of LAE particle and LSE particle can be obtained by the following formulas:

$$\begin{cases} \bar{a}_i^{(j)}(k) = \hat{a}_i^{(j)}(k-1) + \Gamma_i^{(j)}(k)\varepsilon_i^{(j)}(k) \\ \bar{x}_i^{(j)}(k) = x_{p,i}^{(j)}(k) + [K_i^{(j)}(k) + \Upsilon_i^{(j)}(k)\Gamma_i^{(j)}(k)]\varepsilon_i^{(j)}(k) \end{cases} \tag{A2}$$

where

$$\begin{cases} x_{p,i}^{(j)}(k) = f(\hat{x}_i^{(j)}(k-1)) + B(k)[u(k) + \hat{a}_i^{(j)}(k-1)] \\ \varepsilon_i^{(j)}(k) = y_i(k) - h(x_{p,i}^{(j)}(k)) \end{cases} \tag{A3}$$

$$\begin{cases} P_{p,i}^{(j)}(k) = Q \\ \Sigma_i^{(j)}(k) = H_i^{(j)}(k)P_{p,i}^{(j)}(k)H_i^{(j)}(k)^T + R_i \\ K_i^{(j)}(k) = P_{p,i}^{(j)}(k)H_i^{(j)}(k)^T\Sigma_i^{(j)}(k)^{-1} \\ P_i^{(j)}(k) = P_{p,i}^{(j)}(k) - K_i^{(j)}(k)\Sigma_i^{(j)}(k)(K_i^{(j)}(k))^T \end{cases} \tag{A4}$$

$$\begin{cases} \Upsilon_i^{(j)}(k) = [I_n - K_i^{(j)}(k)H_i^{(j)}(k)]F_i^{(j)}(k-1)\Upsilon_i^{(j)}(k-1) \\ \qquad\qquad + [I_n - K_i^{(j)}(k)H_i^{(j)}(k)]B(k) \\ \Omega_i^{(j)}(k) = H_i^{(j)}(k)F_i^{(j)}(k-1)\Upsilon_i^{(j)}(k-1) + H_i^{(j)}(k)B(k) \\ \Lambda_i^{(j)}(k) = [\lambda_i\Sigma_i^{(j)}(k) + \Omega_i^{(j)}(k)S_i^{(j)}(k-1)\Omega_i^{(j)}(k)^T]^{-1} \\ \Gamma_i^{(j)}(k) = S_i^{(j)}(k-1)\Omega_i^{(j)}(k)^T\Lambda_i^{(j)}(k) \\ S_i^{(j)}(k) = \dfrac{1}{\lambda_i}S_i^{(j)}(k-1) - \dfrac{1}{\lambda_i}\Gamma_i^{(j)}(k)\Omega_i^{(j)}(k)S_i^{(j)}(k-1) \end{cases} \tag{A5}$$

Proof. The derivation process is divided into two main parts: system state estimation and attack signals estimation. For the system state estimation, similar to (13) in [1], the posterior probability can be expressed as

$$p(x(k)|y_i(k)) = \frac{\sum_{i=1}^{M} p(x(k)|x_i^{(j)}(k-1), y_i(k))p(y_i(k)|x_i^{(j)}(k-1))}{\sum_{i=1}^{M} p(y_i(k)|x_i^{(j)}(k-1))} \tag{A6}$$

Then, $p(x(k), y_i(k)|x_i^{(j)}(k-1))$ approximates the following Gaussian distribution

$$N\left(\begin{pmatrix} x_{p,i}^{(j)}(k) \\ \bar{y}_i(k) \end{pmatrix}, \begin{pmatrix} P_{p,i}^{(j)}(k) & P_{p,i}^{(j)}(k)H_i^{(j)}(k) \\ (P_{p,i}^{(j)}(k)H_i^{(j)}(k))^T & \Sigma_i^{(j)}(k) \end{pmatrix}\right) \tag{A7}$$

Through certain algebraic operations, we can get Kalman gain $K_i^{(j)}(k)$ and covariance $P_i^{(j)}(k)$ as shown in (A4).

For the attack signals estimation, (A5) is similar to (5e)-(5i) in [2], hence the derivation process is omitted here.

* Corresponding author (email: bchen@zjut.edu.cn)

Remark A1. Notice that the design of intermediate variables is similar to the adaptive Kalman filter in [2]. However, the method proposed by [2] was designed for linear cases, while the proposed method in this letter can be used for nonlinear systems by combining multi-distribution particle filter.

Based on Theorem 1, the weight of j-th particle is given by [1]

$$w_i^{(j)}(k) = \frac{p(y_i(k)|h(x_{p,i}^{(j)}(k)), \Sigma_i^{(j)}(k))}{\sum_{m=1}^{M} p(y_i(k)|h(x_{p,i}^{(m)}(k)), \Sigma_i^{(m)}(k))} \tag{A8}$$

where $p(y_i(k)|h(x_{p,i}^{(j)}(k)), \Sigma_i^{(j)}(k))$ is the probability which reflects the accuracy of the prior particle $x_{p,i}^{(j)}(k)$ by the measurement $y_i(k)$, and the denominator represents the normalization operation to ensure that $\sum_{j=0}^{M} w_i^{(j)}(k) = 1$. Subsequently, the selection of l is described as follows:

1. Generate a random number Θ sampled from $[0, 1]$ under uniform distribution.
2. Let $l = \min\left\{1 \leqslant l_0 \leqslant M | \sum_{m=1}^{l_0} w_i^{(m)}(k) \geqslant \Theta\right\}$.
3. Obtain $\hat{a}_i^{(j)}(k)$ and $\hat{x}_i^{(j)}(k)$ by (5) in letter and let $\Upsilon_i^{(j)}(k) = \Upsilon_i^{(l)}(k)$, $S_i^{(j)}(k) = S_i^{(l)}(k), j = j + 1$.
4. Repeat this procedure until $j = M$.

Remark A2. As stated in [3], in order to solve the degeneracy problem where the weight of some particles becomes small, the above-mentioned process is performed based on the proportion of the weight of each particle. This ensures that the weight of each particle is equal.

Appendix B Simulation Result

Consider a mobile robot tracking system based on wireless lidar sensors with known position coordinates. The motion function of the mobile robot is modeled by [4]

$$x(k) = f(x(k-1), u(k)) + w(k-1) \tag{B1}$$

where

$$f(x(k-1), u(k)) = \begin{bmatrix} s_x(k-1) + T_0 u_p(k) \cos\theta(k-1) \\ s_y(k-1) + T_0 u_p(k) \sin\theta(k-1) \\ \theta(k-1) + T_0 u_r(k) \end{bmatrix} \tag{B2}$$

Here, $x(k) \triangleq \operatorname{col}\{s_x(k), s_y(k), \theta(k)\}$ represent the system state, where $s_x(k)$ and $s_y(k)$ denote the position of the target, while $\theta(k)$ denotes the angular orientation. T_0 is the sampling period. $u(k) \triangleq \operatorname{col}\{u_p(k), u_r(k)\}$ are control signals, which can control the linear velocity and angular velocity of the mobile robot.

When the control signals of the mobile robot is tampered by FDI attacks, (B1) can be rewritten as

$$x(k) = f(x(k-1), u(k)) + B(k)a(k) + w(k-1) \tag{B3}$$

where $a(k) \triangleq \operatorname{col}\{a_p(k), a_r(k)\}$, which tamper linear velocity and angular velocity of the mobile robot, are the FDI attack signals and

$$B(k) = \begin{bmatrix} T_0 \cos\theta(k-1) & 0 \\ T_0 \sin\theta(k-1) & 0 \\ 0 & T_0 \end{bmatrix} \tag{B4}$$

Note that $\theta(k-1)$ in $B(k)$ is unknown. One solution is to replace $\theta(k-1)$ with the local estimated value $\hat{\theta}_i(k-1)$. Another alternative method is to estimate the impact of the attack signals on the state $a_e(k) \triangleq \operatorname{col}\{a_x(k), a_y(k), a_\theta(k)\} = B(k)a(k)$. However, this method will change the dimension of attack signals, which increases the difficulty of estimation, and the estimated value of $\theta(k)$ is still needed to be used when reconstructing the attack signals. Based on the above analysis, the former is adopted in this letter.

In the simulation, six sensors are employed to monitor the mobile robot, and the sensor measurements are modeled by

$$y_i(k) = h_i(x(k)) + v_i(k) \tag{B5}$$

where

$$h_i(x(k)) = \begin{bmatrix} \sqrt{[s_{x_i} - s_x(k)]^2 + [s_{y_i} - s_y(k)]^2} \\ \theta(k) - \arctan\left(\frac{s_{y_i} - s_y(k)}{s_{x_i} - s_x(k)}\right) \end{bmatrix} \tag{B6}$$

In (B6), s_{x_i}, s_{y_i} are the position of the i-th sensor, which are given as follows:

$$\begin{cases} (s_{x_1}, s_{y_1}) = (-120 \text{ cm}, 0 \text{ cm}) \\ (s_{x_2}, s_{y_2}) = (-60 \text{ cm}, -60 \text{ cm}) \\ (s_{x_3}, s_{y_3}) = (60 \text{ cm}, -60 \text{ cm}) \\ (s_{x_4}, s_{y_4}) = (-60 \text{ cm}, 60 \text{ cm}) \\ (s_{x_5}, s_{y_5}) = (60 \text{ cm}, 60 \text{ cm}) \\ (s_{x_6}, s_{y_6}) = (120 \text{ cm}, 0 \text{ cm}) \end{cases} \tag{B7}$$

Meanwhile, the sampling period is taken as $T_0 = 1$s, initial state $x(0) = \operatorname{col}\{0 \text{ cm}, 0 \text{ cm}, 0 \text{ deg}\}$, the covariance matrices of process noise and measurement noise are taken as $Q = \operatorname{diag}\left\{1 \text{ cm}^2/\text{s}^2, 1 \text{ deg}^2/\text{s}^2\right\}$, $R_i = \operatorname{diag}\left\{1 \text{ cm}^2, 1 \text{ deg}^2\right\}$, $i = 1, 2, ..., 6$.

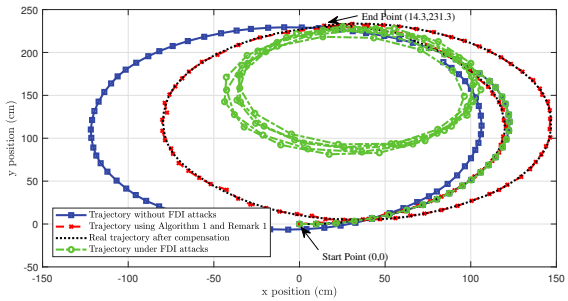

Figure B1 Trajectories of the mobile robot. Blue line: the trajectory without FDI attacks. Green line: the trajectory under FDI attacks. Black line: the real trajectory with compensation. Red line: the estimated trajectory obtained by Algorithm 1 with compensation.

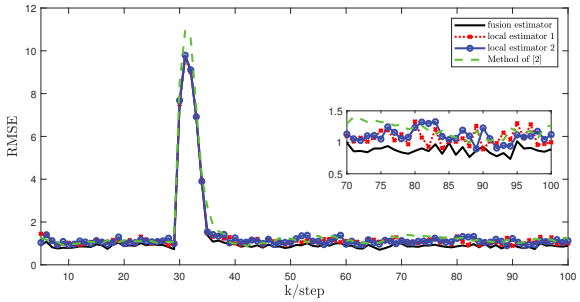

Figure B2 The performance comparison of the local estimators, the fusion estimator and method of [2].

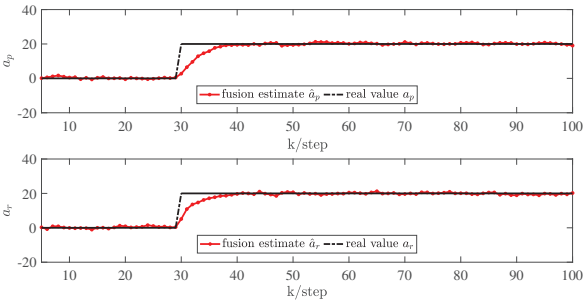

Figure B3 The real values and estimated values of attack signals.

Moreover, sensors $1, 2, 3$ and $4, 5, 6$ are utilized to obtain the local estimates, respectively. In this case, the root mean squared error (RMSE) function is defined as the estimation error indicator function:

$$\mathrm{RMSE}(k) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}[(s_{x,m}(k)-\hat{s}_{x,m}(k))^2+(s_{y,m}(k)-\hat{s}_{y,m}(k))^2]} \tag{B8}$$

where $s_{x,m}(k)$, $s_{y,m}(k)$ and $\hat{s}_{x,m}(k)$, $\hat{s}_{y,m}(k)$ represent the real position and estimated position of the m-th Monte Carlo run, and M is the number of Monte Carlo run.

Let the forgetting factor be $\lambda_1 = \lambda_2 = 0.7$, and the attack signal $a(k)$ is taken as

$$\begin{cases} a_p(k) = 20 \text{ cm/s}, \ a_r(k) = 20 \text{ deg/s}, & k \geqslant 30 \\ a_p(k) = 0 \text{ cm/s}, \quad a_r(k) = 0 \text{ deg/s}, & k < 30 \end{cases} \tag{B9}$$

By implementing Algorithms 1, the simulation results are shown by Fig. B1-Fig. B3. Fig. B1 shows the trajectories of the mobile robot in different situtations. Especially, the start and end points of the real trajectory with compensation are given. From Fig. B1, when the control signals of mobile robot are tampered by FDI attacks, the proposed nonlinear fusion estimator can follow the trajectory of the mobile robot well. Fig. B2 shows the RMSE of different estimators by Monte Carlo method with an average of 100 runs. In Fig. B2, the RMSE of the fusion estimator is smaller than the RMSE of the local estimators, which further verifies the effectiveness of the proposed fusion method. It can be seen from Fig. B2 that compared with the estimator proposed in [2], the proposed fusion method has better performance, which verifies the advantage of the proposed method. The real values of the

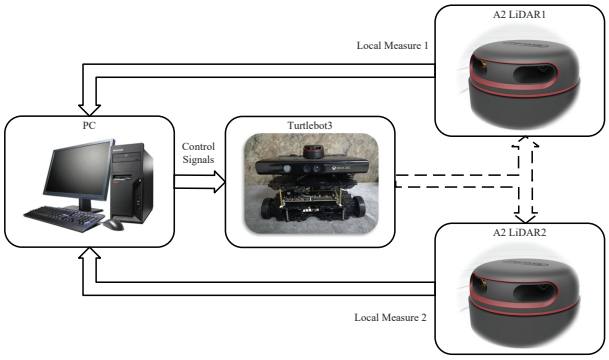

Figure C1 The illustration of mobile robot experiment platform.

attack signals and its FAE are shown in Fig. B3. One can see that $\hat{a}(k)$ obtained by the Algorithm 1 can converge to the real value of $a(k)$. On the other hand, it can be seen from Fig. B1 that the mobile robot under FDI attacks deviates from the original trajectory (the blue line) completely. By implementing the compensation method in remark 1, it is observed that the compensation method can prevent the attack from further degrading the system performance.

Appendix C Experiment Platform

We consider a mobile robot tracking experimental platform whose structure is shown in Fig. C1, where the experiment platform is composed of PC, turtlebot3 and A2 LiDARs. The PC is the monitoring center of the experiment platform and is responsible for receiving local measurement information from A2 LiDARs, executing the proposed algorithm and sending control signals. The turtlebot3 is a small (32cm*27cm*19cm), programmable mobile robot based on Robot Operating System. It can exchange data with PC via wireless network. The A2 LiDAR is a 360° 2D laser scanner developed by SLAMTEC. The digital signal-processing module in A2 LiDAR can process the sampled data and output the distance and angle between the object and the LiDAR to the PC via the network. In the experiment, PC, Turtlebot3 and A2 LiDAR are connected in the same local area network through a routor, and then configure the appropriate IP address to ensure that the PC can communicate with Turtlebot3 and A2 LiDAR.

Experiment and simulation have the same motion model shown in (B2). Different from the simulation, two A2 LiDARs are used as sensors with known coordinates in experiment part, which cause two most critical differences: one is the sensor measurement data, the other is sampling frequency. Specially, for sensor measurement data, A2 LiDAR can only obtain high-precision contour information of the mobile robot, which is different from directly measuring the center position in simulation. Hence, the center position of the mobile robot need to be extracted from the contour information. Notice that the contour of the mobile robot is a circle with known radius, Gauss-Newton method is used for minimize the deviation (C1) to obtain the center position.

$$\arg\min_{x_c,y_c} F(x_c,y_c)=\frac{1}{2}\sum_{i=1}^{N}[(x_c-x_i)^2+(y_c-y_i)^2-r_k^2]^2 \tag{C1}$$

where x_i, y_i are the measured contour data, r_k is the known radius and x_c, y_c are the center of the fitted circle.

For the latter, it will cause the PC to receive unsynchronized measurement data. According to the estimation scheme proposed by [5], the local estimate is performed when the local measurement information is obtained, and fusion estimate is performed when two local measurement information are obtained at the same time (or short time interval).

Appendix D Discussion

The main reasons for the deterioration of the estimated performance in the experiment are summarized as follows:

- It takes a certain time for the mobile robot to reach the specified control signal in the experiment. As shown in Fig. 1(c), the FAE at the beginning is negative, which is caused by the time required for the activating of the turtlebot3.
- The covariances of A2 LiDAR measurement noises are not constant, but are related to the distance between the LiDARs and the object.
- During data transmission through the network, packet delay and packet loss phenomenon may occur.

Therefore, it is necessary to propose new methods to solve these practical problems in our future works such as designing design a secure fusion estimator in the case of unknown noise and time delay/packet loss.

References

1 Murata M, Hiramatsu K. Non-gaussian filter for continuous-discrete models. IEEE Transactions on Automatic Control, 2019, 64: 5260-5264.
2 Zhang Q. Adaptive Kalman filter for actuator fault diagnosis. Automatica, 2018, 93: 333-342.
3 Bolic M, Djuric P M, Hong S. New resampling algorithms for particle filters. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.
4 Yang F, Wang Z, Lauria S, et al. Mobile robot localization using robust extended H_∞ filtering. Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 2009, 223: 1067-1080.
5 Kim S, Kim H, Yoo W, et al. Sensor fusion algorithm design in detecting vehicles using laser scanner and stereo vision. IEEE Transactions on Intelligent Transportation Systems, 2015, 17: 1072-1084.
6 Liu L, Ma L, Zhang J, et al. Distributed non-fragile set-membership filtering for nonlinear systems under fading channels and bias injection attacks. International Journal of Systems Science, 2021, 52: 1192-1205.