Project: Multi-sensor distributed data-gathering using Kalman filtering

Seongjun Moon

***Abstract*— Feedback data-gathering by applying Kalman filter on multisensor.**

I. Introduction

As the central processor outperforms the sensors, the communication delay of the sensors significantly exceeds that of the central processor, as does the transmission rate of power per packet. Feedback communication surpasses non-feedback communication in terms of power efficiency, accuracy, and speed.

In this study, we employ random-walk and random linear state-space equations as the foundational environment for each experimental scenario. To address potential errors, we incorporate zero-mean white Gaussian noise. Subsequently, we implement a fusion rule that utilizes weighted sum aggregation. We then apply a multi-sensor Kalman filtering approach to the state-space equations. We evaluate performance and assess the degree of delay in each experimental setting.

II. Data-Gathering Model of Random-walk environment

1. Environment Setting.

we consider a random walk function denoted as $d(t)$ We model this function using a Random-Walk equation, which is characterized by an equal probability distribution within the interval [2.0, 4.0] with a probability of p/2, and within the interval [-2.0, -4.0] with the same probability of p/2.

$$x(t+1) = x(t) + d(t), t \in Z \geq 0 \tag{1}$$

Our experimental setup involves the presence of M sensors, each denoted by$\in$ {1, ..., M} For each sensor j, we formulate an observation equation, accounting for the influence of zero-mean white Gaussian noise. Following is observation equation with zero-mean white gaussian noise of Sensor number j.

$$y_j(t) = H_j(t)(x(t) + w_j(t)), t \in Z \geq 0 \tag{2}$$

H_j represents the observation matrix that conveys the sensing dimensions of sensor j, while w_j signifies random Gaussian noise. The observation vector for sensor j is denoted as y_j .

We use triggering equation with some threshold ϵ. We denote $\hat{X}$ as the optimized estimation generated by the central processor.

$$|\hat{x}_j(t) - y_j(t)| \geq \epsilon, t \in Z \geq 0 \tag{3}$$

Within our experimental framework, we have triggering delays, denoted as $\Delta_u(t)$ for triggering rule. And we have broadcasting delay referred to as $\Delta_d(t)$ for broadcasting.

2.Implement.

Environment 2x2: sensor number 2, dimension 2.

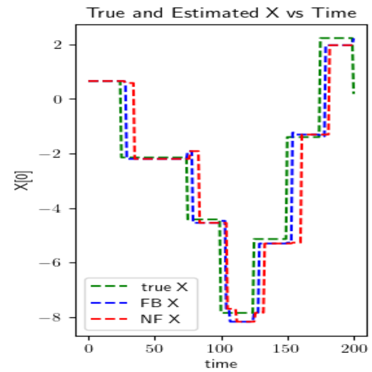

Fig1. value of first dimension of X when ϵ is 1.0 with communication delay.

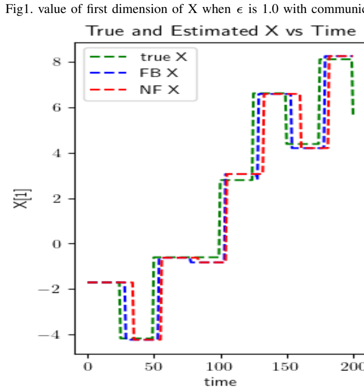

Fig2. value of second dimension of X when ϵ is 1.0 with communication delay.

3.Measuring Performance.

we use MSE(mean-sqaure error) eqaution for measuring accuracy.

$$MSE = \sum(\hat{x} - x)^2 \tag{4}$$

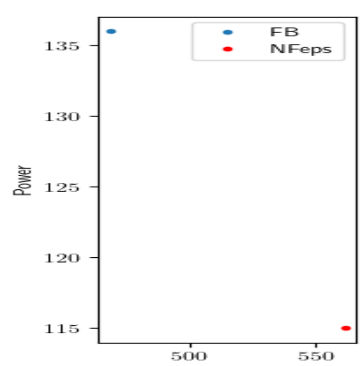

Fig3. MSE of 2x2 model

In the experiment, the measured power for the uplink rate was found to be 5, while for the downlink rate it was 3. Consequently, the FB demonstrated a higher power level compared to the NF. Additionally, the mean squared error

(MSE) of the FB was smaller than that of the NF, indicating that the FB more quickly aligns with the true value. This suggests that FB is more effective in capturing the accurate dynamics of the system under study.

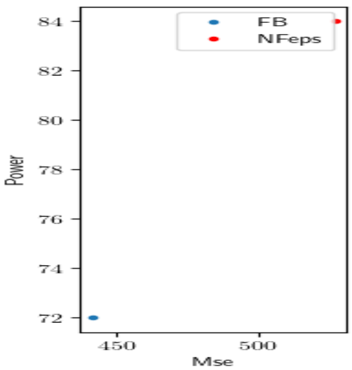

Fig4. value of MSE and Power.

we operated under the assumption that the central processor possesses a superior transmission system. To investigate the conditions under which the Feedback model (FB) exhibits a lower power value compared to the Non-feedback model (NF), we set the uplink rate at 4 and the downlink rate at 1. This setup was chosen to explore the performance of FB relative to NF under specific transmission conditions.

4.Fusion Rule.

Fusion rule of weight sum.

$$\sum \omega_j(t)\hat{x}_j(t) = \hat{x} \tag{5}$$

Weight sum matrix $\omega_j(t)$ is based on avarage.

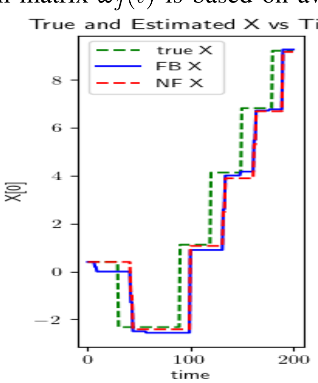

Fig5. Feedback and NoneFeedback model with weighted sum.

The measurement interval of the sensor has become narrower than the triggering interval, resulting in a graph where broadcasting becomes meaningless. So, as we adjusted the measuring interval. However, this approach is contingent on the sensors' ability to trigger values. Without sensors' triggering, the application of the weighted sum model, as dictated by our equation, is not feasible.

III. State-Space Model With Kalman Filtering

1.Environment Setting

we use State-space equation with zero-mean Gaussian white noise w(t) .

$$x(t+1) = A(t)x(t) + w(t) \tag{6}$$

$$y_j(t) = H_j(t)x(t) + v(t)(1 \geq j \geq M) \tag{7}$$

Then we can derive local optimized $\hat{x}_j[k]$ with Kalman filter(KF). Kalman filter has 2 phases estimation and prediction. P^-_{jj} is [j,j] elements of covariance matrix with prediction.

$$\begin{aligned} \hat{x}_j(t) &= (I_n - K_j(t)H_j(t))A(t-1)\hat{x}_j(t-1) + K_j(t)y_j(t) \\ K_j(t) &= P^-_{jj}(t)H^T_j(t)[H_j(t)P^-_{jj}(t)H^T_j(t) + Q_{vi}(t)]^{-1} \\ P_{jj}(t) &= [I_n - K_i(t)H_i(t)]P^-_{jj}(t) \\ P^-_{jj}(t) &= A(t-1)P_{jj}(t-1)A^T(t-1) + Q_w(t-1) \end{aligned} \tag{8}$$

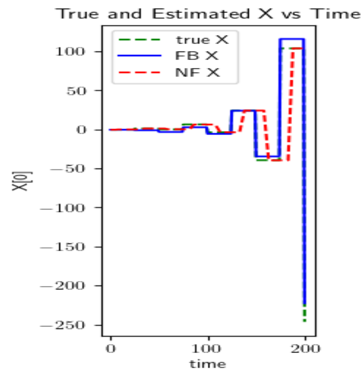

Fig3. Kalman-filtering model

We made an model using Kalman filtering over the fusion rule.

References

[1] B. Chen, W.-A. Zhang, L. Yu, G. Hu, and H. Song, "Distributed fusion estimation with communication bandwidth constraints," *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1398–1403, 2015.

[2] S. Han, "Optimizing accuracy and efficiency in distributed data gathering architectures with feedback," *Preprint*, 2022.