Project Report EEC264

Toshinori Kitamura

March 16, 2019

I. Introduction

This report explains about a paper "Optimal Bayesian Kalman Filtering with Prior Update"[1]. The optimal bayesian Kalman Filter(OBKF) is an advanced version of the intrinsically bayesian robust Kalman filter(IBRKF)[2], which is a Kalman filter exploiting the prior knowledge for the model.

The Kalman filter(classic KF)[3] is a widely used technique to estimate the state vector of a linear dynamics system from its previous estimation and the measurement. Although it provides the best estimation in some condition, it has a problem. That is, the classic KF is highly sensitive to the noise covariance of the target linear dynamic system[4]. To manage this problem, mainly two robust approaches, bayesian approach and non-bayesian approach, have been studied.

The adaptive Kalman filter[5][6] is a non-bayesian approach to achieve the robustness towards the uncertain system. It achieves the robustness by estimating the covariance matrixes during the state estimation. Although it doesn't require any prior knowledge, it needs a lot of observation data to tune the parameter. For a certain problem like gene regulatory network, the cost to obtain the data is expensive, so algorithms which doesn't require a lot of data are preferred.

On the other hand, because of its prior knowledge, the bayesian approach doesn't require so many data comared with non-bayesian approaches. The bayesian approach Kalman filter, IBRKF, is robust in the sense of that it minimizes the average MSE relative to the prior distribution.

The IBRKF achieves the robustness in bayesian sense, but it doesn't utilizes the any information obtained from the observation. The OBKF exploits the both prior knowledge and the observation data, and it achieves the optimal estimation relative to the posterior distribution.

The rest of this paper is organized as follows. Section II explains the overview of the classic Kalman filter and its problem. In section III, a bayesian approach, IBR Kalman filter is explained. Section IV describes the OBKF, and section V shows the simulated results and the performance of the OBKF. Finally, section VI summerizes the OBKF.

II. Kalman filter

The classic KF is the optimal linear estimator for linear dynamic systems. A linear dynamic system is written as the following equations.

$$\mathbf{x}_{k+1} = \boldsymbol{\Phi}_k \mathbf{x}_k + \boldsymbol{\Gamma}_k \mathbf{u}_k \tag{1}$$
$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \tag{2}$$

Where,

- $\mathbf{x}_k$: *state vector*
- $\mathbf{y}_k$: *observation vector*
- $\mathbf{u}_k$, $\mathbf{v}_k$: *zero-mean gaussian noise vector*
- $\boldsymbol{\Phi}_k,\ \boldsymbol{\Gamma}_k,\ \boldsymbol{H}_k$: *transition matrix*

And the noise statistics are,

$$\mathbb{E}[\mathbf{u}_k \mathbf{u}_l^T] = \mathbf{Q}\delta_{kl},\ \mathbb{E}[\mathbf{v}_k \mathbf{v}_l^T] = \mathbf{R}\delta_{kl},\ \forall k, l = 0, 1, ... \tag{3}$$
$$\mathbb{E}[\mathbf{v}_k \mathbf{x}_l^T] = \mathbf{0}, \quad \mathbb{E}[\mathbf{u}_k \mathbf{v}_l^T] = \mathbf{0}\ \forall k, l = 0, 1, ... \tag{4}$$
$$\mathbb{E}[\mathbf{u}_k \mathbf{y}_l^T] = \mathbf{0},\ \ 0 \le l \le k, \tag{5}$$

For such linear dynamic systems, the classic KF provides the linear estimation $\hat{\boldsymbol{x}}_k$ minimizing the mean squared error based on the observations $\boldsymbol{y}_l$, $l \le k-1$. The estimated state vector is given as,

$$\hat{\boldsymbol{x}}_k = \arg\min_{\hat{\boldsymbol{x}}_k} \mathbb{E}[|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k|^2] \tag{6}$$

Let the mean of the estimated $\hat{\boldsymbol{x}}_k$ be itself $\hat{\boldsymbol{x}}_k$ and the covariance of it be as $\mathbf{P^x}_k$. Then, the classic KF algorithm is obtained by the Algorithm 1.

Algorithm 1 Classic Kalman Filter

Input: $\hat{\mathbf{x}}_k$, ${\mathbf{P}^{\mathbf{x}}}_k$, $\mathbf{y}_k$

1: $\tilde{\mathbf{z}}_k = \mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k$

2: $\mathbf{K}_k = {\mathbf{P}^{\mathbf{x}}}_k \mathbf{H}_k^T (\mathbf{H}_k {\mathbf{P}^{\mathbf{x}}}_k \mathbf{H}_k^T + \mathbf{R})^{-1}$

3: $\hat{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \hat{\mathbf{x}}_k + \mathbf{\Phi}_k \mathbf{K}_k \tilde{\mathbf{z}}_k$

4: ${\mathbf{P}^{\mathbf{x}}}_{k+1} = \mathbf{\Phi}_k (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) {\mathbf{P}^{\mathbf{x}}}_k \mathbf{\Phi}_k^T + \mathbf{\Gamma}_k \mathbf{Q} \mathbf{\Gamma}_k^T$

Output: $\hat{\mathbf{x}}_{k+1}$, ${\mathbf{P}^{\mathbf{x}}}_{k+1}$

This algorithm provides the best estimation which minimizes the cost function (6) and provides the optimal estimation. However, the algorithm estimates poorly if the covariance matrix **Q** or **R** are very different from the exact value. This drawback is explained in [4].

III. IBR Kalman Filter

The IBRKF is a robust Kalman filter exploiting the prior knowledge. To begin with, suppose the noise covariance are expressed by unknown parameter vector $\boldsymbol{\theta} = [\theta_1, \theta_2]$. It is governed by the prior distribution $\pi(\boldsymbol{\theta})$, and the noise statistics is written as

$$\mathbb{E}[\mathbf{u}_k^{\theta_1} (\mathbf{u}_l^{\theta_1})^T] = \mathbf{Q}^{\theta_1} \delta_{kl} \tag{7}$$

$$\mathbb{E}[\mathbf{v}_k^{\theta_2} (\mathbf{v}_l^{\theta_2})^T] = \mathbf{R}^{\theta_2} \delta_{kl} \tag{8}$$

The IBRKF is robust in the sense of that it minimizes the cost function (6) relative to the prior distribution. That means, the IBRKF produces the $\hat{\boldsymbol{x}}_k$ given as

$$\hat{\boldsymbol{x}}_k = \arg\min_{\hat{\boldsymbol{x}}_k} \mathbb{E}_{\boldsymbol{\theta}}[\mathbb{E}[|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k|^2]] \tag{9}$$

The algorithm which provides (9) is similar to the one of classic KF. It's obtained by the Algorithm 2.

Algorithm 2 IBR Kalman Filter

Input: $\hat{\mathbf{x}}_k^{\boldsymbol{\theta}}$, $\mathbb{E}_{\boldsymbol{\theta}}[\mathbf{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}]$, $\mathbf{y}_k^{\boldsymbol{\theta}}$

1: $\tilde{\mathbf{z}}_k^{\boldsymbol{\theta}} = \mathbf{y}_k^{\boldsymbol{\theta}} - \mathbf{H}_k \hat{\mathbf{x}}_k^{\boldsymbol{\theta}}$

2: $\mathbf{K}_k^{\Theta} = \mathbb{E}_{\boldsymbol{\theta}}[\mathbf{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}] \mathbf{H}_k^T \mathbb{E}_{\boldsymbol{\theta}}^{-1}[\mathbf{H}_k \mathbf{P}_k^{\boldsymbol{x},\boldsymbol{\theta}} \mathbf{H}_k^T + \mathbf{R}^{\boldsymbol{\theta}_2}]$

3: $\hat{\mathbf{x}}_{k+1}^{\boldsymbol{\theta}} = \Phi_k \hat{\mathbf{x}}_k^{\boldsymbol{\theta}} + \Phi_k \mathbf{K}_k^{\boldsymbol{\Theta}} \tilde{\mathbf{z}}_k^{\boldsymbol{\theta}}$

4: $\mathbb{E}_{\boldsymbol{\theta}}[\mathbf{P}_{k+1}^{\boldsymbol{x},\boldsymbol{\theta}}] = \Phi_k (\mathbf{I} - \mathbf{K}_k^{\Theta} \mathbf{H}_k) \mathbb{E}_{\boldsymbol{\theta}}[\mathbf{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}] \Phi_k^T + \Gamma_k \mathbb{E}_{\theta_1}[\mathbf{Q}_k^{\theta_1}] \Gamma_k^T$

Output: $\hat{\mathbf{x}}_{k+1}^{\boldsymbol{\theta}}$, $\mathbb{E}_{\boldsymbol{\theta}}[\mathbf{P}_{k+1}^{\boldsymbol{x},\boldsymbol{\theta}}]$

It's worth nothing that the IBR Kalman Filter is obtained just replacing **Q** and **R** in Classic Kalman Filter with $\mathbb{E}_{\theta_1}[\mathbf{R}^{\theta_1}]$ and $\mathbb{E}_{\theta_2}[\mathbf{Q}^{\theta_2}]$ respectively. Since the covariance matrixes are same during the algorithm, the computational cost is exactly same as the classic KF.

IV. Optimal Bayesian Kalman Filter

Although IBRKF achieves the robustness relative to its prior knowledge, it doesn't exploit the whole information provided by the observation. The OBKF utilizes both the prior and posterior knowledge to achieve a better estimation. Compared with the IBRKF, it minimizes the cost relative to the posterior distribution, i.e.,

$$\hat{\boldsymbol{x}}_k = \arg\min_{\hat{\boldsymbol{x}}_k} \mathbb{E}_{\boldsymbol{\theta}}[\mathbb{E}[|\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k|^2]|\mathcal{Y}_k] \tag{10}$$

where $\mathcal{Y}_k$ is $\mathcal{Y}_k = \{\boldsymbol{y}_0, \boldsymbol{y}_1, ..., \boldsymbol{y}_k\}$

Same as the IBRKF, the algorithm providing the $\hat{\boldsymbol{x}}_k$ satisfying (10) is obtained just by replacing **Q** and **R** in classic KF with $\mathbb{E}_{\theta_1}[\mathbf{R}^{\theta_1}|\mathcal{Y}_k]$ and $\mathbb{E}_{\theta_2}[\mathbf{Q}^{\theta_2}|\mathcal{Y}_k]$ respectively, shown as Algorithm 3.

Algorithm 3 OBKF

Input: $\hat{\mathbf{x}}_k^{\theta}$, $\mathbb{E}_{\theta}[\boldsymbol{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}|\mathcal{Y}_{k-1}]$, $\mathcal{Y}_k$

1: $\tilde{\mathbf{z}}_k^{\boldsymbol{\theta}} = \mathbf{y}_k^{\boldsymbol{\theta}} - \mathbf{H}_k \hat{\mathbf{x}}_k^{\boldsymbol{\theta}}$

2: $\mathbf{K}_k^{\boldsymbol{\Theta}^*} = \mathbb{E}_{\theta}[\boldsymbol{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}|\mathcal{Y}_{k-1}] \mathbf{H}_k^T \mathbb{E}_{\theta}^{-1}[\mathbf{H}_k \boldsymbol{P}_k^{\boldsymbol{x},\boldsymbol{\theta}} \mathbf{H}_k^T + \mathbf{R}^{`}{}_{2}|\mathcal{Y}_{k-1}]$

3: $\hat{\mathbf{x}}_{k+1}^{\theta} = \mathbf{\Phi}_k \hat{\mathbf{x}}_k^{\theta} + \mathbf{\Phi}_k \mathbf{K}_k^{\boldsymbol{\Theta}} \tilde{\mathbf{z}}_k^{\theta}$

4: $\mathbb{E}_{\theta}[\boldsymbol{P}_{k+1}^{\boldsymbol{x},\boldsymbol{\theta}}|\mathcal{Y}_k] = \mathbf{\Phi}_k (\mathbf{I} - \mathbf{K}_k^{\boldsymbol{\Theta}^*} \mathbf{H}_k) \mathbb{E}_{\theta}[\boldsymbol{P}_k^{\boldsymbol{x},\boldsymbol{\theta}}|\mathcal{Y}_k] \mathbf{\Phi}_k^T + \mathbf{\Gamma}_k \mathbb{E}_{\theta_1}[\boldsymbol{Q}^{\boldsymbol{\theta}_1}|\mathcal{Y}_k] \mathbf{\Gamma}_k^T$

Output: $\hat{\mathbf{x}}_{k+1}^{\theta}$, $\mathbb{E}_{\theta}[\boldsymbol{P}_{k+1}^{\boldsymbol{x},\boldsymbol{\theta}}|\mathcal{Y}_k]$

Now the only remaining problem is how to find the posterior expectations: $\mathbb{E}_{\theta_1}[\boldsymbol{Q}^{\boldsymbol{\theta}_1}|\mathcal{Y}_k]$ and $\mathbb{E}_{\theta_2}[\boldsymbol{R}^{\boldsymbol{\theta}_2}|\mathcal{Y}_k]$. To calculate them, the posterior distribution $\pi(\boldsymbol{\theta}|\mathcal{Y}_k)$ is necessary. Since there is no closed-form solution for $\pi(\boldsymbol{\theta}|\mathcal{Y}_k)$, the author of this paper employs MCMC to approximate $\mathbb{E}_{\theta_1}[\boldsymbol{Q}^{\boldsymbol{\theta}_1}|\mathcal{Y}_k]$ and $\mathbb{E}_{\theta_2}[\boldsymbol{R}^{\boldsymbol{\theta}_2}|\mathcal{Y}_k]$. The posterior distribution is proportinal to the product of its likelihood and prior distribution, i.e., $\pi(\boldsymbol{\theta}|\mathcal{Y}_k) \propto f(\mathcal{Y}_k|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$. Then, once the likelihood function $f(\mathcal{Y}_k|\boldsymbol{\theta})$ is obtained, we can run the MCMC. From the Markov assumption of the system, it can be assumed that $\boldsymbol{y}_k$ depends only on $\boldsymbol{x}_k$ and $\boldsymbol{\theta}$. Then, the likelihood function is calculated as,

$$f(\mathcal{Y}_k|\boldsymbol{\theta}) = \int \cdots \int f(\mathcal{Y}_k, \mathcal{X}_k|\boldsymbol{\theta}) dx_0 ... dx_k = \int \cdots \int \prod_{i=0}^{k} f(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\theta}) \prod_{i=1}^{k} f(\boldsymbol{x}_i|\boldsymbol{x}_{i-1}, \boldsymbol{\theta}) f(\boldsymbol{x}_0) dx_0 ... dx_k \tag{11}$$

Since this equation (11) is a factorization of a grobal function, it can be expressed as a factor-graph, and we can use sum-product algorithm[7] to compute $f(\mathcal{Y}_k|\boldsymbol{\theta})$. Finally, the likelihood function is obtained by the Algorithm 4.

Algorithm 4 Factor-Graph-Based Likelihood Function Calculation

Input: $\boldsymbol{\theta}$, $\mathcal{Y}_k$

1: $\boldsymbol{M}_0 \leftarrow \mathbb{E}[\boldsymbol{x}_0]$
2: $S_0 \leftarrow 1$
3: $\boldsymbol{\Sigma}_0 \leftarrow \mathrm{cov}[\boldsymbol{x}_0]$
4: $i \leftarrow 0$
5: **while** $i \leq k-1$ **do**
6: $\quad \boldsymbol{W}_i \leftarrow \boldsymbol{H}_i^T (\boldsymbol{R}^{\theta_2})^{-1} \boldsymbol{y}_i + \boldsymbol{\Sigma}_i^{-1} \boldsymbol{M}_i$
7: $\quad \boldsymbol{\Lambda}_i^{-1} \leftarrow \boldsymbol{\Phi}_i^T (\tilde{\boldsymbol{Q}}_i^{\theta_1})^{-1} \boldsymbol{\Phi}_i + \boldsymbol{H}_i^T (\boldsymbol{R}^{\theta_2})^{-1} \boldsymbol{H}_i + \boldsymbol{\Sigma}_i^{-1}$
8: $\quad \boldsymbol{\Sigma}_{i+1}^{-1} \leftarrow (\tilde{\boldsymbol{Q}}_i^{\theta_1})^{-1} - (\tilde{\boldsymbol{Q}}_i^{\theta_1})^{-1} \boldsymbol{\Phi}_i \boldsymbol{\Lambda}_i \boldsymbol{\Phi}_i^T (\tilde{\boldsymbol{Q}}_i^{\theta_1})^{-1}$
9: $\quad \boldsymbol{M}_{i+1} \leftarrow \boldsymbol{\Sigma}_{i+1} (\tilde{\boldsymbol{Q}}_i^{\theta_1})^{-1} \boldsymbol{\Phi}_i \boldsymbol{\Lambda}_i (\boldsymbol{H}_i^T (\boldsymbol{R}^{\theta_2})^{-1} \boldsymbol{y}_i + \boldsymbol{\Sigma}_i^{-1} \boldsymbol{M}_i)$
10: $\quad S_{i+1} \leftarrow S_i \sqrt{\frac{|\boldsymbol{\Lambda}_i||\boldsymbol{\Sigma}_{i+1}|}{|\tilde{\boldsymbol{Q}}_i^{\theta_1}||\boldsymbol{\Sigma}_i|}} \mathcal{N}(\boldsymbol{y}_i; \boldsymbol{0}_{m\times 1}, \boldsymbol{R}^{\theta_2}) \times \exp(\frac{\boldsymbol{M}_{i+1}^T \boldsymbol{\Sigma}_{i+1}^{-1} \boldsymbol{M}_{i+1} + \boldsymbol{W}_i^T \boldsymbol{\Lambda}_i \boldsymbol{W}_i - \boldsymbol{M}_i^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{M}_i}{2})$
11: $\quad i \leftarrow i+1$
12: **end while**
13: $\boldsymbol{\Delta}_k^{-1} \leftarrow \boldsymbol{H}_k^T (\boldsymbol{R}^{\theta_2})^{-1} \boldsymbol{H}_k + \boldsymbol{\Sigma}_k^{-1}$
14: $\boldsymbol{G}_k \leftarrow \boldsymbol{\Delta}_k (\boldsymbol{H}_k^T (\boldsymbol{R}^{\theta_2})^{-1} \boldsymbol{y}_k + \boldsymbol{\Sigma}_k^{-1} \boldsymbol{M}_k)$
15: $f(\mathcal{Y}_k|\theta) \leftarrow S_k \sqrt{\frac{|\boldsymbol{\Delta}_k|}{|\boldsymbol{\Sigma}_k|}} \mathcal{N}(\boldsymbol{y}_k; \boldsymbol{0}_{m\times 1}, \boldsymbol{R}^{\theta_2}) \times \exp(\frac{\boldsymbol{G}_k^T \boldsymbol{\Delta}_k^{-1} \boldsymbol{G}_k - \boldsymbol{M}_k^T \boldsymbol{\Sigma}_k^{-1} \boldsymbol{M}_k}{2})$

Output: $f(\mathcal{Y}_k|\theta)$

Although the OBKF provides the best estimation relative to the posterior distribution, it has some drawbacks. Due to its MCMC and the factor-graph algorithm, the OBKF is computationally expensive. The computational cost of the factor-graph algorithm is proportional to k due to its while loop, and the computational cost of MCMC is depends on its number of sampling.

However, in some situation, the algorithm 4 and the MCMC can be omitted. For example, if the environment is stationary, it's not necessary to run the MCMC once the $\mathbb{E}_{\theta_1}[\boldsymbol{Q}^{\boldsymbol{\theta_1}}|\mathcal{Y}_k]$ and $\mathbb{E}_{\theta_2}[\boldsymbol{R}^{\boldsymbol{\theta_2}}|\mathcal{Y}_k]$ are obtained. Moreover, even if the covariance $\mathbb{E}_{\theta_1}[\boldsymbol{Q}^{\boldsymbol{\theta_1}}|\mathcal{Y}_k]$ and $\mathbb{E}_{\theta_2}[\boldsymbol{R}^{\boldsymbol{\theta_2}}|\mathcal{Y}_k]$ change, we can assume that the change of them are small between a short interval. In that case, it's not necessary to run the MCMC every measurement k, and the computational cost will become much small.

V. Simulation and Performance

The author compares the OBKF with the model specific Kalman filter, IBRKF, minimax Kalman filter, and MAP Kalman filter. The model specific Kalman filter is the classic KF with true unknown parameter. The minimax Kalman filter provides the best worst case performance. The MAP Kalman filter is the classic KF with the MAP estimates of the unknown parameter. To evaluate the estimation, the covariance of the estimation error is used. In the equation (12), $\mathbf{P^x}_{k+1}(\boldsymbol{\theta};\boldsymbol{\theta}')$ is the covariance of the estimation error $\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k$.

$$\mathbf{P^x}_{k+1}(\boldsymbol{\theta};\boldsymbol{\theta}') = \boldsymbol{\Phi}_k(\mathbf{I} - \mathbf{K}_k^{\theta'}\mathbf{H}_k)\mathbf{P^x}_k(\boldsymbol{\theta};\boldsymbol{\theta}')(\mathbf{I} - \mathbf{K}_k^{\theta'}\mathbf{H}_k)^T\boldsymbol{\Phi}_k^T + \boldsymbol{\Gamma}_k\mathbf{Q}^{\theta_1}\boldsymbol{\Gamma}_k^T + \boldsymbol{\Phi}_k\boldsymbol{K}_k^{\theta'}\boldsymbol{R}^{\theta_2}(\boldsymbol{K}_k^{\theta'})^T\boldsymbol{\Phi}_k^T \tag{12}$$

The trace of $\mathbf{P^x}_{k+1}(\boldsymbol{\theta};\boldsymbol{\theta}')$ is computed as the MSE of the estimation, and it is used as the metric.

To evaluate the performance of the OBKF, consider a 2-D tracking problem with an unknown parameter. In the problem, the state vector is $\boldsymbol{x}_k = [p_x\, v_x\, p_y\, v_y]^T$, where p_x, v_x, p_y, v_y are the position and velocity of the x and y dimmensions, respectively. The parameters in the (1) and (2) are given as follows.

$$\boldsymbol{\Phi}_k = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{H}_k = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad \boldsymbol{\Gamma}_k = \boldsymbol{I}$$

$$\boldsymbol{Q} = q \times \begin{bmatrix} \tau^3/3 & \tau^2/2 & 0 & 0 \\ \tau^2/2 & \tau & 0 & 0 \\ 0 & \tau & 0 & 0 \\ 0 & 0 & \tau^2/2 & \tau \end{bmatrix}, \quad \boldsymbol{R} = r \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Here, τ is the measurement interval. q and r are the covariance noise intensity.

In the simulation, the author uses $\tau = 1$ second. The initial conditions are set as $\mathbb{E}[\boldsymbol{x}_0] = [100\,10\,30\,-10]^T$ and $\mathrm{cov}[\boldsymbol{x}_0] = \mathrm{diag}([25\,2\,25\,2])$ where $\mathrm{diag}(v)$ is a diagonal matrix whose diagonal elements are v. The parameter q is set to 2 and r is defined as a random variable, uniformly distributed over [0.25,4].

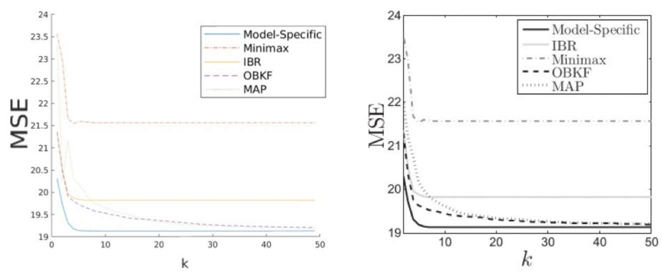

Figure 1: *MSE for $r = 1$. The left figure shows the result simulated on my laptop. Right figure is cited from the OBKF paper[1].*

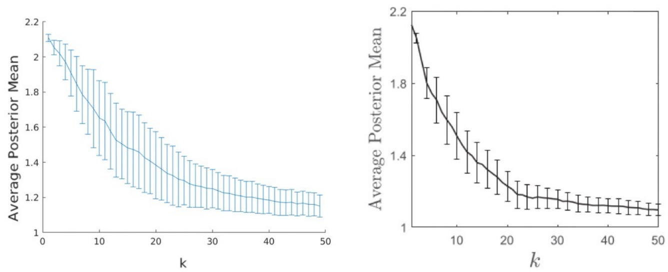

Figure 2: *Empirical average and variance of $\mathbb{E}[r|\mathcal{Y}_k]$. The left figure shows the result simulated on my laptop. Right figure is cited from the OBKF paper[1].*

From the view of the estimation accuracy, the OBKF achieves the best performance. Fig.1 compares the simulated MSE on my laptop with the figure from the paper. You can see that in the both images, the OBKF outperforms the other Kalman filters. The OBKF converges. The OBKF converges the fastest and reaches almost the model-specific KF as k increases. Fig.2 compares the estimated r simulated on my laptop and the figure from the paper. In the both image, the estimated r converges to the true value $r = 1$ as k increases. That means that the estimation is getting close to the model-specific KF as the k increases.

Compared with the non-bayesian approach, the OBKF achieves much efficient estimation in the sense of the number of data. Fig.3 compares the OBKF and the adaptive Kalman filter with unknown parameter r uniformly distributed over [0.25, 4]. The adaptive methods depend on the parameter N_s, which is the number of the observations to tune the filter. If the N_s is small, the adaptive Kalamn filter estimates poorly and can be unstable in the long run. From Fig.3, you can see that the OBKF converges much faster than the adaptive methods.

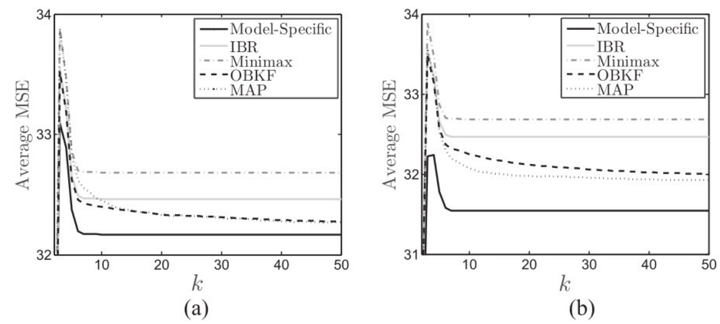

Figure 4: *Average performance with unknown r. The assumed prior distribution is uniform distribution over the interval [3, 6]. (a) True uncertainty interval [2, 7] (b) True uncertainty interval [0.5, 8.5]. Cited from [1]*

Although the OBKF is accurate and efficient, the accuracy of the OBKF depends on the prior knowledge. If the true prior distribution is different from the prior distribution in the OBKF, the OBKF doesn't converge to the optimal performance. Fig.4 shows the MSE of the Kalman filters with incorrect prior distribution. Since the prior distribution doesn't include the true value in some simulation, there is a possibility that the OBKF doesn't find the true value.

VI. Conclusion and Future Work

The OBKF provides the optimal estimation relative to the posterior distribution. Since it utilizes the prior knowledge, it doesn't require a lot of data compared with the adaptive filter. However, the OBKF is computationally expensive. This problem can be solved if the environment is static or the change of the unknown parameter is small. Developing more efficient algorithm is a future work. Also, if the prior distribution of the OBKF is very different from the true prior distribution, the OBKF doesn't converge to the optimal estimation. This is another drawback of the

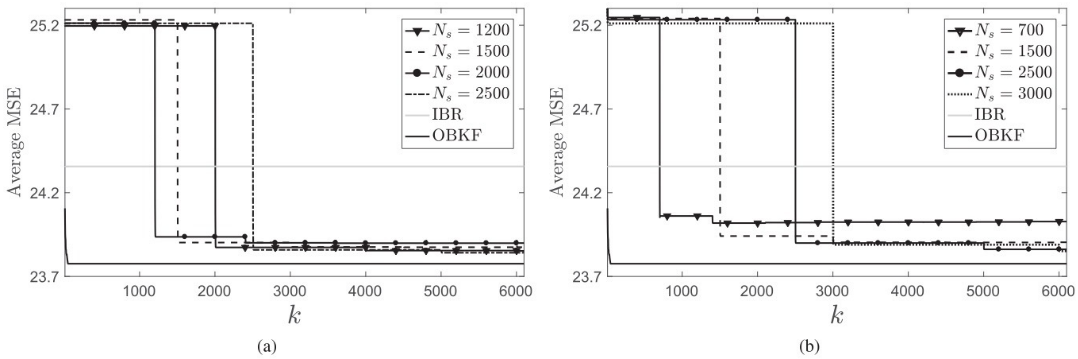

Figure 3: *Comparison with the adaptive Kalman filters. (a) Unknown r and comparison with the Myers method. (b) Unknown r and comparison with the Mehra method. Cited from [1]*

OBKF.

References

[1] Roozbeh Dehghannasiri, Mohammad Shahrokh Esfahani, Xiaoning Qian, and Edward R. Dougherty. Optimal Bayesian Kalman Filtering with Prior Update. *IEEE Transactions on Signal Processing*, 66(8):1982–1996, 2018.

[2] Roozbeh Dehghannasiri, Mohammad Shahrokh Esfahani, and Edward R. Dougherty. Intrinsically Bayesian robust kalman filter: An innovation process approach. *IEEE Transactions on Signal Processing*, 65(10):2531–2546, 2017.

[3] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35, 1960.

[4] S. Sangsuk-Iam and T.E. Bullock. Analysis of discrete-time Kalman filtering under incorrect noise covariances. *IEEE Transactions on Automatic Control*, 35(12):1304–1309, 1990.

[5] Kenneth A. Myers and Byron D. Tapley. Adaptive Sequential Estimation with Unknown Noise Statistics. *IEEE Transactions on Automatic Control*, 21(4):520–523, 1976.

[6] Raman K. Mehra. Approaches to Adaptive Filtering. *IEEE Transactions on Automatic Control*, 17(5):693–698, 1972.

[7] Frank R Kschischang, Brendan J Frey, and Hans-Andrea Loeliger. Factor Graphs and the Sum-Product Algorithm. Technical Report 2, 2001.