AEM8423 Project: Final Report
Robust Filtering for Systems with Parameter Uncertainties

Siew, Peng Mun
siewx007@umn.edu

***Abstract*—Kalman filter is one of the most widely used filter and it can produce optimal state estimates when the system model and noise statistics are well known. However, the performance of Kalman filter degrades as uncertainties in the model increases and as the system's noise statistics deviates from that of a Gaussian distribution. In this project, performance of the traditional discrete time Kalman filter under system model parameter uncertainties is simulated and compared to that of a robust filter designed using a a semi-definite programming approach. From the simulations, it was found that the Kalman filter performs slightly better than the robust filter when the system model is perfectly known, however performance of the Kalman filter degrades at a more rapid rate than the robust filter as the system model uncertainties increases. In this case, the robust filter performs better.**

I. Introduction

Kalman filter has been the go to filter to be used in state estimation and signal filtering, ever since it was first introduced in the 1960s, and this was mainly driven by the success of Kalman filter in aerospace industries during that period. However, Kalman filter does not perform optimally under all scenarios.

The Kalman filter is an optimal filter only when under the following conditions:

1) The mean and correlation of the process noise, w_k and measurement noise, v_k at each time step k are well known, and the noises are Gaussian.
2) A good knowledge of the covariance matrix for process noise, Q_k and measurement noise, R_k is available.
3) The system model matrices F_k and H_k are exactly known

It is the minimum variance estimator when the noise is Guassian, and it is a linear minimum variance estimator when the noise is not Gaussian.

As the Kalman filter is not robust to plant uncertainties and noise uncertainties, various research has been conducted to come up with robust state estimation methods to better estimate the states of the system under model uncertainties. One such approach is the H$_\infty$ filter.

In contrast to the Kalman filter that minimizes the expected value of the variance of the estimation error, the robust filter (H$_\infty$ filter) minimizes the worst-case estimation error and does not make any assumptions on the process and measurement noise. The H$_\infty$ filter is also called the minimax filter, as it minimizes the bound on the variance of the estimation error under the worst-case scenario.

In this project, a Kalman filter, a steady state H$_\infty$ filter and a robust mixed Kalman/H$_\infty$ filter is used to estimate the states of a 1-dimensional moving cart subjected to a sinusoidal acceleration input. Performances of the three filters at different level of model uncertainties are examined and compared.

II. Problem Formulation

A. Continuous Linear Time Invariant System

Consider the following time invariant, continuous time, uncertain linear state-space model:

$$\dot{X} = FX + Bu + Gw \tag{1}$$

$$y = HX + Jv \tag{2}$$

where F, B, G, H, J are system matrices that defines the dynamic system, and x is the state of the system at time t, w and v are uncorrelated Gaussian process noise and Gaussian measurement noise respectively.

For this project, we will be looking at state estimation for a simple 1D cart displacement problem, where the cart starts from an initial condition with $x = 0$ and $\dot{x} = 1$, and we are interested in estimating the speed, $\dot{x}$ and displacement, x of the cart when subjected to a sinusoidal force. For the nominal system, it is assumed that the cart rolling resistance and air drag are proportional to the car's speed and the system can then be simplified to a simple mass and damper system (with a damping coefficient of b) represented by the following state space equations, with u as the external force input.

$$X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{3}$$

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ 0 & -b/m \end{bmatrix} X + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} w \tag{4}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} X + v \tag{5}$$

B. Discrete Linear Time Invariant System

The continuous linear time invariant system presented in the previous subsection is then discretized into a discrete linear time invariant system to better emulate the performance of the various filter in real life applications. The system is discretized using a time period, T of 0.1s.

$$F_{discrete} = \exp^{FT} \tag{6}$$

$$B_{discrete} = \frac{1}{b}\begin{bmatrix}[T - \frac{m}{b} + \frac{m}{b}\exp^{-bT/m}] \\ 1 - \exp^{-bT/m}\end{bmatrix} \quad (7)$$

$$G_{discrete} = \frac{1}{b}\begin{bmatrix}[T - \frac{m}{b} + \frac{m}{b}\exp^{-bT/m}] \\ 1 - \exp^{-bT/m}\end{bmatrix} \quad (8)$$

C. Simulation Parameters

For the simulation, it is assumed that we have a perfect knowledge of the system model and noise distribution, however the mass can change based on the type and amount of load and we have no knowledge about the load. Thus, the mass of our system can be modeled as an additive uncertain parameter.

$$m_{filter} = (1 + \delta_m)m_{true} \quad (9)$$

The m_{true} is kept at 10kg, while the $m_{estimated}$ is varied from 10kg to 4000kg.

The damping coefficient, b is set to 40Ns/m. The process noise is modeled as a zero mean Gaussian distribution with standard deviation of 1, while the measurement noise is modeled as a zero mean Gaussian distribution with a standard deviation of 0.004.

The control input is modeled as a sinusoidal force with magnitude of 1N and a period of 10s.

The model is simulated using a time step of 0.1s.

III. Filter

A. Kalman Filter Formulation

A standard discrete time Kalman Filter is designed using the following formulation.

$$P_k^- = F_{k-1}P_{k-1}^+F_{k-1}^T + Q_{k-1} \quad (10)$$

$$K_k = P_k^- H_k^T(H_k P_k^- H_k^T + R_k)^{-1} \quad (11)$$

$$\hat{x}_k^- = F_{k-1}\hat{x}_{k-1}^+ + G_{k-1}u_{k-1} \quad (12)$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k(y_k - H_k\hat{x}_k^-) \quad (13)$$

$$P_k^+ = (I - K_kH_k)P_k^-(I - K_kH_k)^T + K_kR_kK_k^T \quad (14)$$

The system's gain and state covariance are updated at each time step.

B. H_∞ Filter Formulation

Under the H_∞ filter formulation, no assumptions are made on the noise terms and they can be random with unknown statistics. The goal is to estimate a linear combination of the state, z_k, given by the following equation.

$$z_k = L_kx_k \quad (15)$$

As we are interested in estimating all of the system's states, therefore $L_k = I$. Defining the estimate of z_k as $\hat{z}_k$, the estimate of x_0 as $\hat{x}_0$ and using measurements up to time step N-1, the cost function for the H_∞ filter can be formulated using a game theory approach as follows:

$$J_1 = \frac{\sum_{k=0}^{N-1}\|z_k - \hat{z}_k\|^2_{S_k}}{\|x_0 - \hat{x}_0\|^2_{P_0^{-1}} + \sum_{k=0}^{N-1}(\|w_k\|^2_{Q_k^{-1}} + (\|v_k\|^2_{R_k^{-1}})} \quad (16)$$

We are interested in finding an estimate $\hat{z}_k$ that minimizes J_1, while our adversary is trying to maximize the J_1 by selecting the right combination of x_0, w_k and v_k. In the Kalman filter formulation, there is no adversary and the noise are assumed to be indifference. In the design of a Kalman filter, we leverages the knowledge of these indifference noises to arrive at a statistically optimal state estimation.

In contrast to the kalman filter, the state covariance matrix, P_0, process noise covariance matrix, Q_k, measurement noise covariance matrix, R_k, and the innovation covariance matrix, S_k are arbitrary chosen depending on the specific problem, to be symmetric positive definite matrices. The relative weightage of diagonal terms dictate either how accurate the state estimates need to be or how significant the noise disturbance are.

As the direct minimization of the cost function is not easy, a performance bound is used to upper bound the cost function and instead of searching for $\hat{z}_k$ that produces the minimal cost function, we search for the $\hat{z}_k$ that can satisfy the following bound.

$$J_1 \le \frac{1}{\theta} \quad (17)$$

where θ is the user defined performance bound.

As our system is time invariant, we can solve for a steady state solution to the H_∞ filtering problem. The initial states and control inputs are assumed to be perfectly known to simplify the problem. The performance bound of the cost functions can then be simplified to the following equation.

$$J_1 = \lim_{N\to\infty}\frac{\sum_{k=0}^{N-1}\|z_k - \hat{z}_k\|^2_{S}}{\sum_{k=0}^{N-1}(\|w_k\|^2_{Q^{-1}} + (\|v_k\|^2_{R^{-1}})} \le \frac{1}{\theta} \quad (18)$$

where Q, R and S are symmetric positive definite matrices that are arbitrary chosen depending on the problem. The steady state filter equations are as followed.

$$\bar{S} = L^TSL \quad (19)$$

$$K = P[I - \theta\bar{S}P + H^TR^{-1}HP]^{-1}H^TR^{-1} \quad (20)$$

$$\hat{x}_{k+1} = F\hat{x}_k + FK(y_k - H\hat{x}_k) \quad (21)$$

$$P = FP[I - \theta\bar{S}P + H^TR^{-1}HP]^{-1}F^T + Q \quad (22)$$

The subsequent equation need to hold for the above estimator to be a solution to the H_∞ filtering problem.

$$P^{-1} - \theta\bar{S} + H^TR^{-1}H > 0 \quad (23)$$

C. Formulating Steady State H_∞ Filter as a SDP Problem

The H_∞ filtering problem can be formulated as a Semi-Definite Programming (SDP) problem by changing the constraints (Equation 22 and Equation 23) into a Linear Matrix Inequalities (LMI) and searching for a solution that minimizes the upper bound on the cost function, $\frac{1}{\theta}$.

From equation 22 above, we can apply a matrix inversion lemma to obtain the following equation

$$
\begin{aligned}
P &= F\{P - P[(-\theta\bar{S} + H^T R^{-1} H)^{-1} + P]^{-1} P\} F^T + Q \\
&= FPF^T - FP[(-\theta\bar{S} + H^T R^{-1} H)^{-1} + P]^{-1} PF^T + Q
\end{aligned} \tag{24}
$$

Using the Schur Complement Lemma and setting it to be greater than zero, we arrive at the following equation.

$$
\begin{bmatrix} FPF^T + Q - P & FP \\ PF^T & (-\theta\bar{S} + H^T R^{-1} H)^{-1} + P \end{bmatrix} > 0 \tag{25}
$$

Using another matrix inversion lemma on the 2,2 element of the matrix, followed by a Schur Complement Lemma, we arrive at the following equation.

$$
\begin{bmatrix} FPF^T + Q - P & FP & 0 \\ PF^T & P - (\theta\bar{S})^{-1} & (\theta\bar{S})^{-1} H^T \\ 0 & H(\theta\bar{S})^{-1} & R - H(\theta\bar{S})^{-1} H^T \end{bmatrix} > 0 \tag{26}
$$

Notice that in our case, the L is an identity matrix and S is a scalar value, therefore

$$
\begin{aligned}
\theta\bar{S} &= \theta L^T S L \\
&= \theta S I
\end{aligned} \tag{27}
$$

After using a change of variable (Let $\gamma I = \theta S I = (\theta\bar{S})^{-1}$), we finally arrive at an LMI of the following form.

$$
\begin{bmatrix} FPF^T + Q - P & FP & 0 \\ PF^T & P - \gamma I & \gamma H^T \\ 0 & H\gamma & R - H\gamma H^T \end{bmatrix} > 0 \tag{28}
$$

From equation 23 above, we can apply a matrix inversion lemma to obtain the following equation.

$$
P - \gamma I - PH^T (R + HPH^T)^{-1} HP > 0 \tag{29}
$$

Using the Schur Complement Lemma, we would arrive at the following LMI.

$$
\begin{bmatrix} P - \gamma I & PH^T \\ HP & R + HPH^T \end{bmatrix} > 0 \tag{30}
$$

The semi-definite programming problem can be summarized as follows
minimize γ
subjected to

$$
P > 0 \tag{31}
$$

$$
\begin{bmatrix} FPF^T + Q - P & FP & 0 \\ PF^T & P - \gamma I & \gamma H^T \\ 0 & H\gamma & R - H\gamma H^T \end{bmatrix} > 0 \tag{32}
$$

$$
\begin{bmatrix} P - \gamma I & PH^T \\ HP & R + HPH^T \end{bmatrix} > 0 \tag{33}
$$

D. Robust Mixed Kalman/H_∞ Filtering

Based on [2], a mixed Kalman/H_∞ Filter that is robust to model uncertainties can be designed using the following formulations.

Consider a scalar sequence $\alpha_k > 0$ and a small scalar $\epsilon > 0$. Let,

$$
R_{11k} = Q_k + \alpha_k M_{1k} M_{1k}^T \tag{34}
$$

$$
R_{12k} = \alpha_k M_{1k} M_{2k}^T \tag{35}
$$

$$
R_{22k} = R_k + \alpha_k M_{2k} M_{2k}^T \tag{36}
$$

Initialize P_k and $\tilde{P}_k$ as S.
Solve for P_{k+1} and $\tilde{P}_{k+1}$ using the following equation.

$$
\begin{aligned}
P_{k+1} =& F_{1k} T_k F_{1k}^T + R_{11k} + R_{11k} R_{2k} R_{11k}^T - \\
& [F_{1k} T_k H_{1k}^T + R_{12k} + R_{11k} R_{2k} R_{12k}] R_k^{-1} [\dots]^T + \epsilon I
\end{aligned} \tag{37}
$$

$$
\begin{aligned}
\tilde{P_{k+1}} =& F_k \tilde{P}_k F_k^T + F_k \tilde{P}_k N_k^T (\alpha_k I - N_k \tilde{P}_k N_k^T)^{-1} N_k \tilde{P}_k F_k^T + \\
& R_{11k} + \epsilon I
\end{aligned} \tag{38}
$$

where,

$$
R_{1k} = (\tilde{P}_k^{-1} - N_k^T N_k / \alpha_k)^{-1} F_k^T \tag{39}
$$

$$
R_{2k} = R_{1k}^{-1} (\tilde{P}_k^{-1} - N_k^T N_k / \alpha_k)^{-1} R_{1k}^{-T} \tag{40}
$$

$$
F_{1k} = F_k + R_{11k} R_{1k}^{-1} \tag{41}
$$

$$
H_{1k} = H_k + R_{12k}^T R_{1k}^{-1} \tag{42}
$$

$$
T_k = (P_k^{-1} - \theta^2 I)^{-1} \tag{43}
$$

the gain and estimated state matrix at each time step are given by the following equations

$$
K_k = [F_{1k} T_k H_{1k}^T + R_{12k} + R_{11k} R_{2k} R_{12k}] \tilde{R}_k^{-1} \tag{44}
$$

$$
\tilde{R}_k = H_{1k} T_k H_{1k}^T + R_{22k} + R_{12k}^T R_{2k} R_{12k} \tag{45}
$$

$$
\hat{F}_k = F_{1k} - K_k H_{1k} \tag{46}
$$

If the following two constraints hold then the estimator is a solution to the robust mixed Kalman/H_∞ filtering problem.

$$
\frac{1}{\theta^2} I > P_k \tag{47}
$$

$$
\alpha_k I > N_k \tilde{P}_k N_k^T \tag{48}
$$

ϵ is chosen to be a very small positive number, while α_k and θ are variables that we are trying to minimize. A small α_k would results in a smaller P_k, making the bound on the RMS estimation error to be smaller, whereas a small θ would results in a lower bound on the state estimation error.

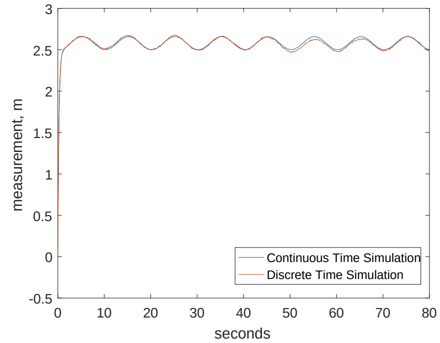

Fig. 1. Error in Displacement Estimation when the System Model is Perfectly Known.

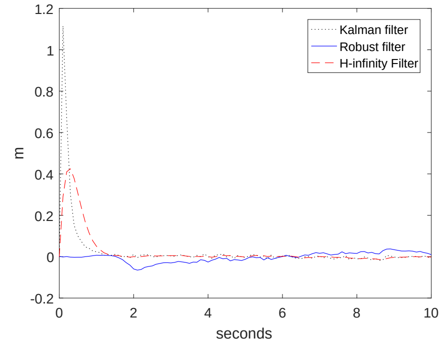

Fig. 2. Error in Displacement Estimation with $\delta_m = 0$.

IV. Results & Discussions

Figure 1 shows the comparison between displacement measurement obtained from using a continuous linear time invariant model and that obtained from the discretized linear time invariant model. Both graphs adheres well to each other, which shows that our discretized system agrees well with the continuous time system.

The three filters are first implemented on a perfectly known system with Gaussian noise distribution. Figure 2 and figure 3 shows the position and velocity estimation error respectively. We noticed that during the first few seconds of the simulation, the mixed Kalman/H$_\infty$ filter performs better than the other two filters. However, at steady state, the mixed Kalman/H$_\infty$ filter is outperformed by both the Kalman filter and Steady State H$_\infty$ filter. Initially, the Kalman filter did not perform as well as the Steady State H$_\infty$ filter, but at steady state, the Kalman filter performs slightly better than the Steady State H$_\infty$ filter. The root mean squared error (RMSE) of the three filters during steady state are as shown in table I.

TABLE I
Steady State RMSE for $\delta_m = 0$

Filter	Position RMSE	Velocity RMSE
Kalman Filter	0.008197	0.024667
Steady State H$_\infty$ Filter	0.011414	0.022407
Mixed Kalman/H$_\infty$ Filter	0.023738	0.293760

The filters are implemented next on a system with parameter uncertainty, $\delta_m = 100$. Figure 4 and figure 5 shows the position and velocity estimation error when $\delta_m = 100$ respectively. We noticed that the Kalman filter performed badly compared to both the mixed Kalman/H$_\infty$ filter and steady state H$_\infty$ filter during the beginning of the simulation as well as after the system has reached steady state. The root mean squared error (RMSE) of the three filters during steady state are as shown in table II.

The filters are also tested on a system with parameter uncertainty, $\delta_m = 400$. Figure 6 and figure 7 shows the position and velocity estimation error when $\delta_m = 400$ respectively. We noticed that the Kalman filter's performance degrades even further and performs badly compared to the other two filter. The performance of the other two filters did not degrade much and is robust to the parameter uncertainty. The root mean squared error (RMSE) of the three filters during steady state are as shown in table III.

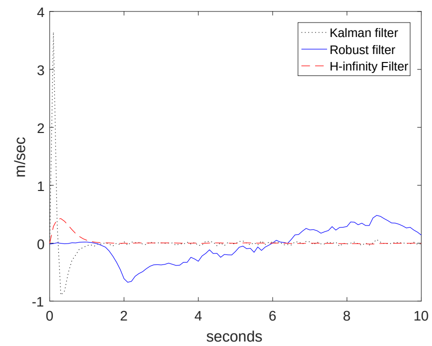

Fig. 3. Error in Velocity Estimation with $\delta_m = 0$.

TABLE II
Steady State RMSE for $\delta_m = 100$

Filter	Position RMSE	Velocity RMSE
Kalman Filter	0.064863	0.132990
Steady State H$_\infty$ Filter	0.015101	0.062055
Mixed Kalman/H$_\infty$ Filter	0.013906	0.040833

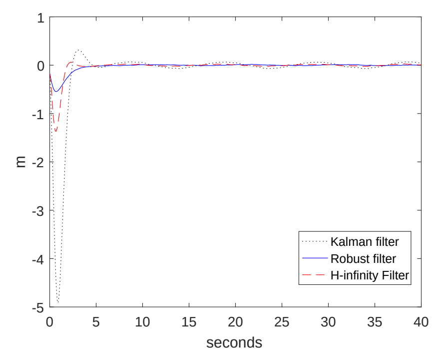

Fig. 4. Error in Displacement Estimation with $\delta_m = 100$.

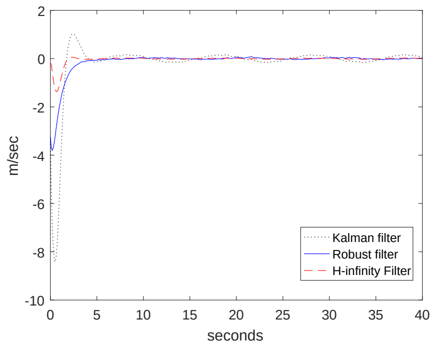

Fig. 5. Error in Velocity Estimation with $\delta_m = 100$.

TABLE III
STEADY STATE RMSE FOR $\delta_m = 400$

Filter	Position RMSE	Velocity RMSE
Kalman Filter	0.1590300	0.201870
Steady State H_∞ Filter	0.0155640	0.062965
Mixed Kalman/H_∞ Filter	0.0051373	0.030035

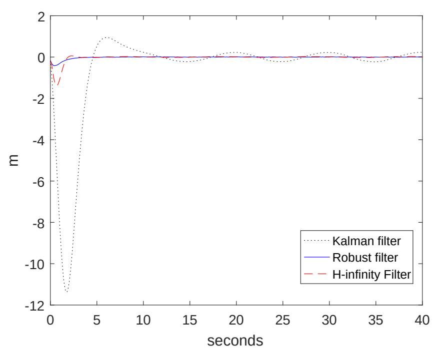

Fig. 6. Error in Displacement Estimation with $\delta_m = 400$.

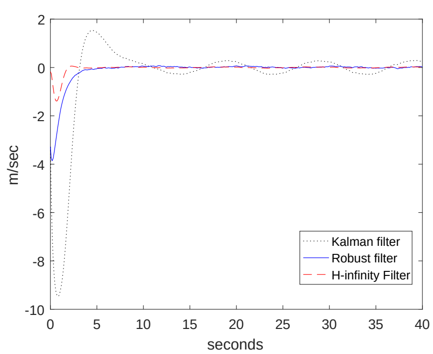

Fig. 7. Error in Velocity Estimation with $\delta_m = 400$.

V. CONCLUSION

In this project, three different filters (Kalman Filter, Steady State H_∞ Filter and Mixed Kalman/H_∞ Filter) are successfully designed and implemented on a 1-dimensional cart displacement problem, with the H_∞ Filter being formulated using a convex optimization approach. The performance of these three filters are examined when the system model is subjected to model uncertainties of different magnitudes ($\delta_m = 0, \delta_m = 100, \delta_m = 400$). The Kalman Filter performed best among the three filters when the model is perfectly known (i.e. $\delta_m = 0$). As the parameter uncertainties increases, the Kalman filter's performance degrades at a much faster rate compared to the other two filters. The Steady State H_∞ filter and mixed Kalman/H_∞ filter are robust to the model parameter uncertainties and are able to maintain consistent performance even when the parameter uncertainty is high ($\delta_m = 400$). The Mixed Kalman/H_∞ Filter have the best performance among the three filters.

REFERENCES

[1] Dan Simon, "Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches," Wiley-Interscience, 2006
[2] Y. Hung and F. Yang, "Robust H_∞ filtering with error variance constraints for discrete time-varying systems with uncertainty," Automatica, 39(7), pp. 1185-1194 (July 2003)
[3] Dan Simon, 'From Here to Infinity," Embedded System Programming, October 2001