A practical note on evaluating Kalman filter performance optimality and degradation

Dah-Jing Jwo *, Ta-Shun Cho

Department of Communications and Guidance Engineering, National Taiwan Ocean University, 2 Peining Road, Keelung 202-24, Taiwan, ROC

Abstract

This paper presents useful remarks to the readers on the Kalman filter (KF) performance optimality, degradation, and some innovation related parameters. Guidelines for efficient approach for evaluation of the KF performance optimality and sensitivity analysis are presented. Performance degradation due to uncertainty in process and measurement noise statistics is discussed. Consistency check between the filter-calculated covariances versus actual mean square errors are provided, which can be used not only as a verification procedure for the filtering correctness, but also as a approach for making trade-off in designing a suitable Kalman filter. In addition to numerical algorithms, useful Matlab programs are accompanied where necessary to the readers for getting better insight in practical implementation. Exploration of the behaviour of some innovation based parameters useful in adaptive filter and system integrity designs, including covariance of innovation sequence, degree of mismatch (DOM), and degree of divergence (DOD), etc., is also involved.

© 2007 Elsevier Inc. All rights reserved.

Keywords: Kalman filter; Optimality; Consistency; Sensitivity; Adaptive; Integrity

1. Introduction

The Kalman filter (KF) [1–3] is commonly used to estimate the system state variables and suppress the measurement noise. It has been applied in the areas as diverse as aerospace, marine navigation, radar target tracking, control systems, manufacturing, and many others. Studying the operation of the Kalman filter leads to an appreciation of the inter-disciplinary nature of system engineering. The Kalman filter not only works well in practice, but also it is theoretically attractive because it has been shown that it is the filter that minimizes the variance of the estimation mean square error (MSE).

In Kalman filter designs, the divergence due to modeling errors is critical. The implementation of Kalman filter requires that the complete a priori statistical knowledge of the process noise and measurement noise are available. Poor knowledge of the noise statistics may seriously degrade the Kalman filter performance, and even provoke the filter divergence. If the theoretical behaviour of a filter and its actual behaviour do not agree,
divergence problems will occur. That is, if the Kalman filter is provided with information that the process behaves a certain way, whereas, in fact, it behaves a different way, the filter will continually intend to fit an incorrect process signal. When the measurement situation does not provide sufficient information to estimate all the system state variables, then the estimation error covariance matrix becomes unrealistically small and the filter disregards the measurement. When apparent divergence occurs, the actual estimate error covariance remains bounded, but it approaches a larger bound than the predicted error covariance; when true divergence occurs, the actual estimation covariance eventually becomes infinite.

In various circumstances where there are uncertainties in the system model and noise description, and the assumptions on the statistics of disturbances are violated due to the fact that in a number of practical situations, the availability of a precisely known model is unrealistic since in the modelling step, some phenomena are disregarded. The suboptimal configuration is typically based on a simplified error state dynamic/measurement model. One way to take them into account is to consider a nominal model affected by uncertainty. Covariance analysis is a common tool to provide numerical histories depicting the accuracy of a given configuration in terms of the covariance of its associated error state vector. Therefore, the analysis can be used to evaluate the performance of the suboptimal filter that operates in a real world environment, and can be utilized as a basic design tool during the synthesis and test of the suboptimal configuration.

To fulfil the requirement of achieving the filter optimality, an adaptive Kalman filter (AKF) [4–7] can be utilized as the noise-adaptive filter for tuning the noise covariance matrices. Adaptive filters are based on dynamically adjusting the parameters of the supposedly optimum filter based on the estimates of the unknown parameters. Adaptive Kalman filters can be based on an on-line estimation of motion as well as the signal and noise statistics available data. Many efforts have been made to improve the estimation of the covariance matrices based on the innovation-based estimation approach. The two major approaches that have been proposed for AKF are multiple model adaptive estimate (MMAE) and innovation adaptive estimation (IAE). The innovation sequences have been utilized by the correlation and covariance-matching techniques to estimate the noise covariances. The basic idea behind the covariance-matching approach is to make the actual value of the covariance of the residual consistent with its theoretical value. The IAE approach coupled with fuzzy logic techniques with membership functions designed using heuristic method has been very popular to adjust the noise statistics [8–10].

Any practical data fusion system is susceptible to faults that may lead to violations of estimate consistency. Robustness to such faults is necessary to maintain the integrity of the information in the system. Statistical decision tools for detecting abrupt changes in the properties of stochastic signals and dynamical systems have numerous applications, from the on-line fault detection in complex technical systems to detection of signals with unknown arrival time in radar and sonar signal processing. The main goal of the sensor fault detection is to detect the system degradation when it leads to an unacceptable growth of the output errors [11,12].

Several practical issues regarding the Kalman filter performance optimality and degradation will be presented for conveying some important phenomena to the readers. Content of discussion covers performance degradation due to uncertainty in process and measurement noise statistics, consistency check between the filter-calculated covariances versus actual mean square errors, and behaviour for some innovation related parameters. The remarks presented in this paper are beneficial to the Kalman filter designers, which can be employed as guidelines for developing a suitable filter configuration, and can provide useful information in the adaptive Kalman filter and system integrity design.

This paper is organized as follows. In Section 2, practical notes on Kalman filter and arbitrary gain suboptimal filter are pointed out. In Section 3, Kalman filter solution and optimality evaluation is presented. Performance degradation due to uncertainty in process noise is discussed in Section 4. In Section 5, performance degradation due to uncertainty in measurement noise is presented. The conclusion is given in Section 6.

2. Kalman filter and arbitrary gain suboptimal filter

2.1. Continuous Kalman filter

Consider a dynamical system whose state is described by a linear, vector differential equation. The process model and measurement model are represented as
Process model: \( \dot{x} = Fx + Gu \)

Measurement model: \( z = Hx + v \)

where the vectors \( u(t) \) and \( v(t) \) are both white noise sequences with zero means and mutually independent:

\[
E[u(t)u^T(\tau)] = Q_\delta(t-\tau),
\]

\[
E[G(t)u(t)(G(t)u(\tau))^T] = GQG^T_\delta(t-\tau),
\]

\[
E[v(t)v^T(\tau)] = R_\delta(t-\tau),
\]

\[
E[u(t)v^T(\tau)] = 0,
\]

where \( \delta(t-\tau) \) is the Dirac delta function, \( E[\cdot] \) represents expectation, and superscript “\( T \)” denotes matrix transpose.

The state estimate equation of the continuous Kalman filter equations is represented as

\[
\hat{x} = Fx + K(z - Hx).
\]

The propagation of the error for a continuous Kalman filter can be described by the Riccati equation:

\[
\dot{P} = FP + PF^T - PH^TR^{-1}HP + GQG^T,
\]

and the continuous filter gain is obtained through the calculation

\[
K = PH^TR^{-1}.
\]

The discrete filter gain and continuous filter gain are related by

\[
K = \frac{K_k}{\Delta t},
\]

where \( \Delta t = t_{k+1} - t_k \) represents the sampling period. When the system reaches steady-state, \( \dot{P} = 0 \), Eq. (7) becomes an algebraic Riccati equation (ARE), which can be solved for the steady-state minimum covariance matrix.

2.2. Discrete time Kalman filter

Expressing Eqs. (1) and (2) in discrete-time equivalent form via discretization of a continuous time system leads to

\[
x(t_{k+1}) = \Phi(t_{k+1}, t_k)x(t_k) + \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)G(\tau)u(\tau) d\tau,
\]

for which either two of the following abbreviated notations can be used:

\[
x_{k+1} = \Phi x_k + w_k,
\]

\[
x_{k+1} = \Phi x_k + \Gamma w_k,
\]

and

\[
z_k = H x_k + v_k.
\]

In the above equation, the state vector \( x_k \in \mathbb{R}^n \), process noise vector \( w_k \in \mathbb{R}^n \), measurement vector \( z_k \in \mathbb{R}^m \), and measurement noise vector \( v_k \in \mathbb{R}^m \). Eq. (11b) can be treated as a special case of (11a) for \( \Gamma_k = I \). In Eqs. (11) and (12), both the vectors \( w_k \) and \( v_k \) are zero mean Gaussian white sequences having zero cross-correlation with each other:

\[
E[w_k w^T] = Q_k \delta_k,
\]

or alternatively

\[
E[(\Gamma_k w_k)(\Gamma'_k w'_k)^T] = \Gamma_k Q_k \Gamma'_k \delta_k,
\]
If Eq. (11b) is used, Eq. (20) should be used to replace Eq. (17a):

\[ E[w_i^T] = R_k \delta_{ik}, \quad (14) \]
\[ E[w_k^T] = 0 \] for all \( i \) and \( k \),

where \( Q_k \) is the process noise covariance matrix, \( R_k \) is the measurement noise covariance matrix, and \( \Phi_k \) is the state transition matrix. The symbol \( \delta_{ik} \) stands for the Kronecker delta function:

\[ \delta_{ik} = \begin{cases} 1, & i = k, \\ 0, & i \neq k. \end{cases} \]

The discrete-time Kalman filter loop algorithm iteratively applies two stages of computations [2]:

Stage 1: prediction steps/time update equations

\[
\begin{align*}
\hat{x}_{k+1} &= \Phi_k \hat{x}_k, \\
P_{k+1}^{-} &= \Phi_k P_k \Phi_k^T + Q_k.
\end{align*}
\]  (16)

Stage 2: correction steps/measurement update equations

\[
\begin{align*}
K_k &= P_k H_k^T[H_k P_k H_k^T + R_k]^{-1}, \\
\hat{x}_k &= \hat{x}_k^- + K_k [z_k - H_k \hat{x}_k^-], \\
P_k &= [I - K_k H_k] P_k^-.
\end{align*}
\]  (18)

If Eq. (11b) is used, Eq. (20) should be used to replace Eq. (17a):

\[ P_{k+1}^- = \Phi_k \Phi_k^T + K_k Q_k K_k^T. \]  (17b)

Eqs. (16)–(20) constitute the Kalman filter for the model of Eqs. (11) and (12). Eqs. (16) and (17) are the time update equations of the algorithm from \( k \) to step \( k + 1 \), and Eqs. (18)–(20) are the measurement update equations. These equations incorporate a measurement value into a priori estimation to obtain an improved \textit{a posteriori} estimation. In the above equations, \( P_k \) is the error covariance matrix defined by \( E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] \), in which \( \hat{x}_k \) is an estimation of the system state vector \( x_k \), and the weighting matrix \( K_k \) is called the Kalman gain matrix. The Kalman filter algorithm starts with an initial condition value, \( \hat{x}_0^- \) and \( P_0^- \). When new measurement \( z_k \) becomes available with the progression of time, the estimation of states and the corresponding error covariance would follow recursively ad infinity.

### 2.3. Arbitrary gain filter

The error covariance relationships for a discrete filter with the same structure as the Kalman filter, but with an arbitrary gain matrix are written as

\[ P_k = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T \]  (21)

with

\[ P_{k+1}^- = \Phi_k \Phi_k^T + Q_k \]  (17a)

(or, if Eq. (17b) is used, \( P_{k+1}^- = \Phi_k \Phi_k^T + K_k Q_k K_k^T \)). The error covariance \( P_k \) described in a single differential Eq. is obtained by substituting Eq. (17a) into Eq. (21):

\[ P_k = (\Phi_k - K_k H_k \Phi_k) P_k^- (\Phi_k - K_k H_k \Phi_k)^T + (I - K_k H_k) Q_k (I - K_k H_k)^T + K_k R_k K_k^T, \]  (22)

which has an equivalent form in the CKF:

\[ \dot{P} = (F - KH) P + P (F - KH)^T + GQG^T + K K^T. \]  (23)

In view of CKF, Eq. (23) defines the error covariance for the filter with a general filter gain matrix \( K \), which can be solved for the covariance of an arbitrary gain model. The sensitivity analysis can be conveniently implemented by using such representation. Taking the partial derivative of \( P_k \) with respect to \( K \), using \( \frac{\partial \ln(P_k)}{\partial K} = 0 \) for a minimum leads to the same result as Eq. (7).
For the standard Kalman filter, the filter gain $K_k$ minimizes the following performance index:

$$J = \text{tr}(E[(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T]) = \text{tr}(\mathbf{P}),$$  \hspace{1cm} (24)

where $\text{tr}(\cdot)$ denotes the trace of a matrix. If there is no uncertainty in the process and measurement noise covariances the performance index $J$ attains the global minimum using the standard Kalman filter. However, in the case that there are uncertainties in $Q_k$ or $R_k$, $J$ would not be able to attain the minimum.

3. Kalman filter solution and optimality evaluation

For the sake of better illustration, one dimension target tracking problem is employed in this and also the following sections where applicable. When modelling for a vehicle kinematics, the positions at $t_k$ and $t_{k+1}$ may be related by

$$p_{k+1} = p_k + \ddot{p}_k \Delta t + \frac{\dddot{p}_k \Delta t^2}{2!} + \frac{\dddot{p}_k \Delta t^3}{3!} + \cdots,$$

where the number of dots indicates the order of differentiation. The model is approximated by a truncated series, either using a constant velocity model:

$$p_{k+1} = p_k + \dot{p}_k \Delta t,$$

(25)

or a constant acceleration model:

$$p_{t+1} = p_t + \dot{p}_t \Delta t + \frac{\ddot{p}_t \Delta t^2}{2},$$

(26)

where $p_t$ and $\dot{p}_t$ of Eqs. (25) and (26) are considered constant for the time interval $\Delta t$, and $\ddot{p}_k$ of Eq. (26) is considered constant for the time interval $\Delta t$. The validity of these assumptions is dependent on the length of $\Delta t$ and the degree of the linearity of the vehicle’s motion. The differential equation of the observed system can be represented either of the following forms:

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} u(t),$$

(27)

or

$$\frac{d}{dt} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t),$$

(28)

where the state variables are the position $x_1$ and the velocity $x_2$. The model governed by Eq. (1) leads to

$$\mathbf{F} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}; \quad \mathbf{H} = \begin{bmatrix} 1 & 0 \end{bmatrix},$$

and these signals satisfy the following:

$$E[u(t)u^T(\tau)] = q\delta(t - \tau); \quad E[v(t)v^T(\tau)] = r\delta(t - \tau); \quad E[u(t)v^T(\tau)] = 0.$$  \hspace{1cm} (29)

3.1. Kalman filter solution for time-varying noise statistics

The general discrete time recursive Kalman filter (DTKF) with very small discretization size ($\Delta t \to 0$) and the continuous Kalman filter (CKF) approaches both have the same (statistical) steady-state solution if the signal and noise are stationary. Therefore, the correctness of solutions obtained by the DTKF can be checked with those by the CKF. The main reason for choosing the CKF as a comparison basis is due to the fact that CKF solutions can be obtained without complicated computation and with better confidence on the correctness of solutions. In the following discussion, the results based on the DTKF for $\Delta t \to 0$ and $k \to \infty$ will be
compared to the steady-state continuous Kalman filter (SSCKF). The SSCKF uses gains derived from the steady-state covariance and provides sub-optimal solutions. Expanding Eq. (7) leads to

\[
\begin{align*}
\dot{P}_{11} &= 2P_{12} - \frac{1}{r}P_{21}^2, \\
\dot{P}_{12} &= P_{22} - \frac{1}{r}P_{11}P_{12}, \\
\dot{P}_{22} &= q - \frac{1}{r}P_{12}^2.
\end{align*}
\]

The time varying error covariance and Kalman gain matrix can be obtained using the numerical integration such as the Euler or the Runge–Kutta integrator. When the system reaches the steady-state, we have

\[
P_\infty = \begin{bmatrix} \sqrt{2} q^{1/4} r^{3/4} & (qr)^{1/2} \\ (qr)^{1/2} & \sqrt{2} q^{3/4} r^{1/4} \end{bmatrix}, \quad K_\infty = \begin{bmatrix} \sqrt{2} (q)^{1/4} \\ (q)^{1/2} \end{bmatrix}.
\]

For the example of measurement noise statistic \( r = 1 \) and time-varying process noise statistics \( q = 1 \to 4 \to 9 \to 1 \), the steady-state solutions for the four time intervals are as follows

\[
P_\infty = \begin{bmatrix} \sqrt{2} & 1 \\ 1 & \sqrt{2} \end{bmatrix} \to \begin{bmatrix} 2 & 2 \\ 2 & 4 \end{bmatrix} \to \begin{bmatrix} 2.4495 & 3 \\ 3 & 7.3485 \end{bmatrix} \to \begin{bmatrix} \sqrt{2} & 1 \\ 1 & \sqrt{2} \end{bmatrix},
\]

\[
K_\infty = \begin{bmatrix} \sqrt{2} \\ 1 \end{bmatrix} \to \begin{bmatrix} 2 \\ 2 \end{bmatrix} \to \begin{bmatrix} 2.4495 \\ 3 \end{bmatrix} \to \begin{bmatrix} \sqrt{2} \end{bmatrix}.
\]

The ‘arrows (→)’ is employed for indicating the time-varying trajectory of variables \((q, P_\infty, K_\infty)\). Fig. 1 shows the gain trajectories for the DTKF, for the cases of sampling period \( \Delta t = 0.001 \), in the case of perfect adaptation of noise statistics. These results are very consistent to those obtained by the CKF. Fig. 2 provides the associated trajectories of covariance propagation for the discrete time Kalman filter \( (\Delta t = 0.1) \) as compared to the continuous time Kalman filter.

### 3.2. Evaluation of filtering optimality

A sensitivity analysis illustrates the importance of \( \tilde{Q} \) and \( \tilde{R} \). Gain deviation from the optimal point leads to the increase of variance/covariance. No matter the gain is decreased or increased, the performance index will
be increased in both cases. The schematic illustration for performing the filter optimality evaluation is given in Fig. 3.
Error covariance for the filter with the arbitrary gain model, can be obtained using Eq. (23):

\[
\begin{align*}
P_{11} &= -2K_1P_{11} + 2P_{12} + K_1^2r, \\
P_{12} &= -2P_{21} = -K_1P_{11} - K_1P_{12} + P_{22} + K_2K_3r, \\
P_{22} &= -2K_2P_{12} + K_2^2r + q.
\end{align*}
\] (31)

The covariances are influenced by (1) the actual noise spectral amplitudes, or more specifically, power spectral densities (PSD’s) in the external environment; (2) the gain matrix (which can be any gain matrix and not necessarily the Kalman gain matrix). If the gain matrix is the Kalman gain matrix, the above equations leads to Eq. (30) and the optimal result with minimum variance will be obtained. By setting \( \mathbf{P} = \mathbf{0} \), the solutions related to arbitrary gains are found to be

\[
\begin{bmatrix}
P_{11\infty} & P_{12\infty} \\
P_{21\infty} & P_{22\infty}
\end{bmatrix}
= \frac{1}{2K_1K_2^\infty}
\begin{bmatrix}
q + K_1^2K_2\infty r + K_2\infty^2 r & (q + K_2\infty^2 r)K_1\infty \\
(q + K_2\infty^2 r)K_1\infty & K_1^2q + K_2\infty q + K_2^3 r
\end{bmatrix}.
\] (32)

If the arbitrary gain matrix \( \mathbf{K} \) of a filter has been designed for particular values of \( \mathbf{Q} \) and \( \mathbf{R} \), the steady-state error covariance will vary linearly with the actual process noise spectral density or measurement error spectral density. If the actual noise variances are assumed fixed and the design values of \( \mathbf{Q} \) and \( \mathbf{R} \) are varied, significantly different curves result. Any deviation of the design variances, and consequently \( \mathbf{K} \), from the correct values will cause an increase in the filter error variance (a consequence of the optimality of the filter). Further information on sensitivity analysis can be referred to Gelb [1].

The derivation of the optimal filter gain in Eq. (18) assumes that the mathematical description of the system given by Eqs. (11)–(14) is exact, i.e., the system matrices and the noise statistics used in the Kalman filter model match those of the system. It also assumes that the chosen filter state fully describes the system state. In practical applications, exact knowledge about the system is not usually available, and therefore the model assumed by the filter designer is actually different from the truth. An example on sensitivity is given in Fig. 4, for which \( r = 1 \); truth \( q = 9 \); \( q \) in filter model ranges from 1 to 25. Either decreasing of increasing the gains will both lead to increase of performance index \( J \). The result also reflects the information that decreasing the \( q \) value is more sensitive than increasing it. This phenomenon is straightforward since it reflects the fact that when maneuvering occurs, insufficient large amplitude \( q \) will tend to diverge.

Due to the fact that the filter gain is computed based on the filter-calculated error covariances, it follows that the consistency check is important for filter optimality evaluation. Sensitivity analysis illustrates the importance of \( Q_k \) and \( R_k \). For a linear, time-invariant system, it is easy to perform a parameter sensitivity analysis. Keeping all other conditions constant, the filter is implemented using a range of design \( Q_k \) values (and \( R_k \) constant), and then implemented using a range of design \( R_k \) values (and \( Q_k \) constant).

Fig. 5 presents the flow chart for predicting the actual covariance. The actual MSE basically results from the hybrid information of incorrect gain information and truth noise statistics. \( Q_{k_f}, R_{k_f} \) represent the statistic of the dynamic process and measurement models, respectively, of the Kalman filter, yielding the predicted covariance (\( \mathbf{P}_{\text{predicted}} \); \( Q_{\text{truth}}, R_{\text{truth}} \) represent the actual statistics of the dynamic process and measurement in real world, leading to the actual covariance (\( \mathbf{P}_{\text{actual}} \)). When the noise matrices are correct, i.e. if \( Q_{k_f} = Q_{\text{truth}} \) and \( R_{k_f} = R_{\text{truth}} \), then \( \mathbf{P}_{\text{predicted}} = \mathbf{P}_{\text{actual}} \), while if there is incorrectness in the statistic, i.e. if \( Q_{k_f} \neq Q_{\text{truth}} \) and/or \( R_{k_f} \neq R_{\text{truth}} \), then \( \mathbf{P}_{\text{predicted}} \neq \mathbf{P}_{\text{actual}} \).
4. Performance degradation due to uncertainty in process noise statistic

The Kalman filter requires that all the dynamics and measurement models are exactly known, in addition to the noise processes are zero mean white noise. The process noise covariance $Q_k$ is usually a measure of the uncertainty in the state dynamics. If the theoretical behaviour of a filter and its actual behaviour do not match, divergence problems will occur. In the apparent divergence, the actual estimate error covariance remains bounded, but it approaches a larger bound than the predicted error covariance. In the true divergence, the actual estimation covariance eventually becomes infinite. A filter is called consistent if its state estimation errors satisfy:

\[ E[x_k - \hat{x}_k] = 0, \]
\[ E[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T] = P_k. \]  

Condition in Eq. (33) is the unbiasedness requirement for the estimates (i.e., zero-mean estimation error), while Eq. (34) is the covariance matching requirement, i.e., that the actual mean square error (left-hand side)
matches the filter-calculated covariance (right-hand side). The adaptive filtering approach requires that the parameter of interest be measured. The benefit of adaptive filtering is that it is able to tune the parameters so as to keeps the covariance consistent with the real performance.

4.1. Filter-calculated covariance: continuous time Kalman filter

In the present and next subsection, influence of covariance due to incorrect noise information, either process or measurement, is presented. The total time designed for each simulation scenario is 40 s, which is separated into four equal time intervals, each containing 10 s. Three scenarios are designed for illustration, including Scenario (a)-filter model exactly matches the real world. Scenario (b)-filter model does not match the real world (too small in two of the four time intervals). Scenario (c)-filter model does not match the real world (too large in two of the time intervals). Table 1 shows various spectral amplitudes q’s in each of the four intervals. Listing 1 provides the Matlab program for predicting actual covariance when incorrect Qk information is utilized for the continuous time Kalman filter (for fixed measurement noise spectral amplitude r = 1). Calculation of gains was performed using two approaches: analytical and numerical. The variables P11_inf, P12_inf, and P22_inf stand for the actual steady state covariance elements evaluated by analytical solutions given by Eq. (32). Illustration of covariance degradation for the three simulation scenarios is shown in Fig. 6. The steady-state covariances based on Eq. (32) for the continuous time Kalman filtering at all of the time intervals are summarized in Table 2.

Listing 1. Simulation for predicting actual covariance with incorrect Qk information for the continuous time Kalman filter (with fixed measurement noise spectral amplitude)

```matlab
% (Power spectral densities for process and measurement noises)
r = 1;
q_kf = 1;
q_truth = 9;
% (Predicted covariance matrix)
P11_kf(i+1) = P11_kf(i) + dt * (2 * P12_kf(i) - 1/r * P11_kf(i) * P11_kf(i));
P12_kf(i+1) = P12_kf(i) + dt * (P22_kf(i) - 1/r * P11_kf(i) * P12_kf(i));
P22_kf(i+1) = P22_kf(i) + dt * (q_kf - 1/r * P12_kf(i) * P12_kf(i));
for i = FirstStep + FinalStep
  % (Option #1: calculating gains via analytical solutions)
  K1 = sqrt(2) * (q_kf/r) * 0.25;
  K2 = (q_kf/r) * 0.5;
  % (Option # 2: calculating gains via numerical integration for the Kalman filter formulations)
  K1 = P11_kf(i+1)/r;
  K2 = P12_kf(i+1)/r;
% (Actual covariance matrix)
P11(i+1) = P11(i) + dt * (-2*K1*P11(i) + 2*P12(i) + K1*K1*r);
P12(i+1) = P12(i) + dt * (-2*K2*P11(i) - K1*K1*P12(i) + P22(i)+K1*K2*r);
P22(i+1) = P22(i) + dt * (-2*K2*P12(i) + K2*K2*r+q_truth);
end
% (Actual steady state covariance matrix by analytical solutions)
P11_inf = 1/(2*K1*K2)*(q_truth + K1*K1*K2*r + K2*K2*r);
P12_inf = 1/(2*K1*K2)*(q_truth + K2*K2*r+K1); P22_inf = 1/(2*K1*K2)*(K1*K1*q_truth + K2*K2*r+K2*K2*r);
```
4.2. Filter-calculated covariance: discrete time Kalman filter

Fig. 7 provides the flow chart for actual covariance prediction with incorrect Qk information for the discrete time Kalman filter, which is based on the idea given as in Fig. 5. Here, Rk is used representing both Rk_truth and Rk_kf since these two are assumed the same. Practical implementation is provided in Listing 2. It should be noted that in practical implementation, the actual covariance matrix P_plus_actual may not be available due to the fact that Qk_truth may not be available.

Listing 2. Simulation for actual covariance prediction with incorrect Qk information for the discrete time Kalman filter

```
P_minus = PHI*P_plus*PHI’ + Qk_kf;
P_plus = (eye(size(K_kf*H))−K_kf*H)*P_minus;
K_kf=P_minus*H’*inv(H*P_minus*H’ + Rk);
P_minus_actual = PHI*P_plus_actual*PHI’ + Qk_truth;
P_plus_actual = I_KH*P_minus_actual*I_KH’ + K_kf*Rk*K_kf’;
```

4.3. Filter-calculated covariances versus actual mean square errors – the consistency check

To implement statistical analysis of the actual system performance in comparison with the filter-predicted statistical performance, the filter-calculated covariance (optimal when the noise statistics are exactly known) is compared to the actual mean squared error. Continued from the preceding example for illustration, the results were verified using multiple simulations runs. Figs. 8 and 9 show the 100 ensembles and the one sigma (1 − σ) confidence bound. Data were collected from the 100 simulation runs. Total time for each simulation run is set to be 40 s with Δt = 0.1 s, which leads to 401 samples per run, and totally gives 80,200 samples for the entire 200 runs. The filter-calculated covariance (or 1 − σ bound), resulting from the wrong gains, is not consistent to the actual estimation error covariance, and is unreliable to reflect the true estimation accuracy.

In subplots (a) of Figs. 8 and 9, although the predicted 1 − σ bound shows that good result has been obtained, nevertheless, the actual result do not have consistent errors with the predicted covariances. It can be considered that the filter seemed to be ‘lied’. On the other hand, in subplots (b) of Figs. 8 and 9, the predicted 1 − σ bound shows the change in covariance due to change of noise statistics and the actual mean square errors are consistent with the filter-calculated (predicted) covariance. The predicted covariances given by the filter are very likely to be incorrect by comparing (a) and (b) of Figs. 8 and 9. This can be easily justified from: (1) If the incorrect statistic is utilized, the predicted covariances (or equivalently, 1 − σ bound) show that the steady state has been reached in about 3 s, nevertheless, the actual covariances do not reflect the same phenomena. The predicted covariance for this case is not trustable. (2) If the correct statistic are utilized, the predicted covariances show that the covariances have different values in four time intervals, which do reflect the
Fig. 6. Illustration of covariance degradation for three simulation scenarios.
consistent information as provided by the estimation result. The predicted covariance seems to be more reasonable. (3) Even though the predicted covariances in (a) are smaller than those in (b) in two of the time intervals, nevertheless, the actual MSEs are, on the contrary, larger.

4.4. Detection on the change of process noise statistic

A limitation in applying Kalman filter to real-world problems is that the a priori statistics of the stochastic errors in both dynamic process and measurement models are assumed to be available, which is difficult in practical application due to the fact that the noise statistics may change with time. As a result, the set of unknown time-varying statistical parameters of noise, \( \{Q_k, R_k\} \), needs to be simultaneously estimated with the system state and the error covariance. To fulfill the requirement, an adaptive Kalman filter can be utilized as the noise-adaptive filter to estimate the noise covariance matrices and overcome the deficiency of Kalman filter. The benefit of the adaptive algorithm is that it keeps the covariance consistent with the real performance. The innovation sequences have been utilized by the correlation and covariance-matching techniques to estimate the noise covariances. The basic idea behind the covariance-matching approach is to make the actual value of the covariance of the residual consistent with its theoretical value.

From the incoming measurement \( z_k \) and the optimal prediction \( \hat{x}_k^- \) obtained in the previous step, the innovations sequence is defined as

\[
v_k = z_k - H_k \hat{x}_k^-.
\]  

The innovation reflects the discrepancy between the predicted measurement \( H_k \hat{x}_k^- \) and the actual measurement \( z_k \). It represents the additional information available to the filter as a consequence of the new observation \( z_k \). The weighted innovation, \( K_k(z_k - H_k \hat{x}_k^-) \), acts as a correction to the predicted estimate \( \hat{x}_k^- \) to form the estimation \( \hat{x}_k \). Substituting the measurement model (11b) into (17) gives

\[
v_k = H_k(x_k - \hat{x}_k^-) + v_k,
\]  

Table 2

Steady state covariance matrices based on Eq. (32) for the continuous time Kalman filtering

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_{11} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>1.4142</td>
<td>2</td>
<td>2.4495</td>
<td>1.4142</td>
</tr>
<tr>
<td>b</td>
<td>1.4142</td>
<td>2.4749</td>
<td>4.2426</td>
<td>1.4142</td>
</tr>
<tr>
<td>c</td>
<td>1.4142</td>
<td>2.2981</td>
<td>2.6563</td>
<td>1.4142</td>
</tr>
<tr>
<td>( P_{12} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>1.0</td>
<td>2.0</td>
<td>3.0</td>
<td>1.0</td>
</tr>
<tr>
<td>b</td>
<td>1.0</td>
<td>2.5</td>
<td>5.0</td>
<td>1.0</td>
</tr>
<tr>
<td>c</td>
<td>1.0</td>
<td>2.5</td>
<td>3.4</td>
<td>1.0</td>
</tr>
<tr>
<td>( P_{22} )</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>1.4142</td>
<td>4</td>
<td>7.3485</td>
<td>1.4142</td>
</tr>
<tr>
<td>b</td>
<td>1.4142</td>
<td>4.9562</td>
<td>9.8995</td>
<td>1.4142</td>
</tr>
<tr>
<td>c</td>
<td>1.4142</td>
<td>4.9498</td>
<td>8.2219</td>
<td>1.4142</td>
</tr>
</tbody>
</table>

Fig. 7. Flow chart for actual covariance prediction due to incorrect Qk information for the discrete time Kalman filter.
which is a zero-mean Gaussian white noise sequence. An innovation of zero means that the two are in complete agreement. The mean of the corresponding error of an unbiased estimator is zero.

By taking variances on both sides, we have the theoretical covariance, the covariance matrix of the innovation sequence is given by

$$C_t = E[t_k t_k^T] = H_k P_k H_k^T + R_k; \quad (37a)$$

which can be written as

$$C_t = H_k (\Phi_k P_k \Phi_k^T + \Gamma_k Q_k \Gamma_k^T) H_k^T + R_k. \quad (37b)$$

Defining $\hat{C}_t$ as the statistical sample variance estimate of $C_t$, matrix $\hat{C}_t$ can be computed through averaging inside a moving estimation window of size $N$

$$\hat{C}_t = \frac{1}{N} \sum_{j=j_0}^{k} t_j t_j^T, \quad (38)$$

where $N$ is the number of samples (usually called the window size); $j_0 = k - N + 1$ is the first sample inside the estimation window. The window size $N$ is chosen empirically (a good size for the moving window may vary

---

**Fig. 8.** The filter-calculated $1 - \sigma$ bound is (a) incorrect/inconsistent; (b) correct/consistent to the actual estimation position MSE. (a) Inconsistency between predicted $1 - \sigma$ bound and actual position errors. (b) Consistency between predicted $1 - \sigma$ bound and actual position errors as compared to the result in (a).
from 10 to 30) to give some statistical smoothing. To detect the discrepancy between $\tilde{C}_{uk}$ and $C_{uk}$, we define the degree of mismatch (DOM)

$$\text{DOM} = C_{uk} - \tilde{C}_{uk}.$$  \hspace{1cm} (39)

Kalman filtering with motion detection is important in target tracking applications. The innovation information at the present epoch can be employed for timely reflect the change in vehicle dynamic. Selecting the degree of divergence (DOD) as the trace of innovation covariance matrix at present epoch (i.e., the window size is one), we have:

$$\zeta = v_k^T v_k = \text{tr}(v_k v_k^T).$$  \hspace{1cm} (40)

This parameter can be utilized for detection of divergence/outliers or adaptation for adaptive filtering. Alternatively, the parameters for identifying the degree of change in vehicle dynamics can be determined based on the idea of Eqs. (37) and (40). If the discrepancy for the trace of innovation covariance matrix between the present (actual) and theoretical value is used, the DOD parameter can be of the form:

$$\eta = v_k^T v_k - \text{tr}(C_{uk}).$$  \hspace{1cm} (41)
The other DOD parameter commonly used as a simple test statistic for an occurrence of failure detection is based on the normalized innovation squared, defined as the ratio given by:

$$\mu = \frac{v_k^T v_k}{\text{tr}(C_{v_k})} = v_k^T C_{v_k}^{-1} v_k.$$  \hspace{1cm} (42)

For each of the proposed approach, only one scalar value needs to be determined, and therefore the fuzzy rules can be simplified resulting in the decrease of computational efficiency. Scenario (d): filter model matches the real world. Scenario (e): filter model does not match the real world (too small in one of the time intervals). Scenario (f): filter model does not match the real world (too large in one of the time intervals).

The logic of adaptation algorithm using covariance-matching technique is described as follows. When the actual covariance value $\hat{C}_{v_k}$ is observed, if its value is within the range predicted by theory $C_{v_k}$ and the difference is very near to zero, this indicates that both covariances match almost perfectly. If the actual covariance is greater than its theoretical value, the value of the process noise should be decreased; if the actual covariance is less than its theoretical value, the value of the process noise should be increased. The fuzzy logic [8–10] is popular mainly due to its simplicity even though some other approaches such as neural network and genetic algorithm may also be applicable. When the fuzzy logic approach based on rules of the kind:

IF (antecedent) THEN (consequent)

the following rules can be utilized to implement the idea of covariance matching:

A. $\hat{C}_{v_k}$ is employed

1. IF $\langle \hat{C}_{v_k} \approx 0 \rangle$ THEN $\langle Q_k \rangle$ is unchanged (This indicates that $\hat{C}_{v_k}$ is near to zero, the process noise statistic should be remained.)

2. IF $\langle \hat{C}_{v_k} > 0 \rangle$ THEN $\langle Q_k \rangle$ is increased (This indicates that $\hat{C}_{v_k}$ is larger than zero, the process noise statistic is too small and should be increased.)

3. IF $\langle \hat{C}_{v_k} < 0 \rangle$ THEN $\langle Q_k \rangle$ is decreased (This indicates that $\hat{C}_{v_k}$ is less than zero, the process noise statistic is too large and should be decreased.)

B. DOM is employed

1. IF $\langle \text{DOM} \approx 0 \rangle$ THEN $\langle Q_k \rangle$ is unchanged (This indicates that $\hat{C}_{v_k}$ is about the same as $C_{v_k}$, the process noise statistic should be remained.)

2. IF $\langle \text{DOM} > 0 \rangle$ THEN $\langle Q_k \rangle$ is decreased (This indicates that $\hat{C}_{v_k}$ is less than $C_{v_k}$, the process noise statistic should be decreased.)

3. IF $\langle \text{DOM} < 0 \rangle$ THEN $\langle Q_k \rangle$ is increased (This indicates that $\hat{C}_{v_k}$ is larger than $C_{v_k}$, the process noise statistic should be increased.)

C. DOD ($\mu$) is employed.

Suppose that $\mu$ is employed as the test statistic, and $\mu_T$ represents the chosen threshold. The following fuzzy rules can be utilized:

1. IF $\langle \mu \geq \mu_T \rangle$ THEN $\langle Q_k \rangle$ is increased (There is a failure or maneuvering reported; the process noise statistic is too small and needs to be increased)

2. IF $\langle \mu < \mu_T \rangle$ THEN $\langle Q_k \rangle$ is decreased (There is no failure or non-maneuvering; the process noise statistic is too large and needs to be decreased)

Faults in dynamical systems can be detected with the aid of an innovation sequence of Kalman filter. The failure detection is based on rules of the kind:

$H_0$ System operates normally,

$H_1$ Fault occurs in the system.

Table 3

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Real world</th>
<th>Kalman filter model</th>
</tr>
</thead>
<tbody>
<tr>
<td>d</td>
<td>$q = 0.1 \rightarrow 100 \rightarrow 0.1 \rightarrow 0.1$</td>
<td>$q = 0.1 \rightarrow 100 \rightarrow 0.1 \rightarrow 0.1$</td>
</tr>
<tr>
<td>e</td>
<td>$q = 0.1 \rightarrow 100 \rightarrow 0.1 \rightarrow 0.1$</td>
<td>$q = 0.1 \rightarrow 0.1 \rightarrow 0.1 \rightarrow 0.1$</td>
</tr>
<tr>
<td>f</td>
<td>$q = 0.1 \rightarrow 0.1 \rightarrow 0.1 \rightarrow 0.1$</td>
<td>$q = 0.1 \rightarrow 100 \rightarrow 0.1 \rightarrow 0.1$</td>
</tr>
</tbody>
</table>
Fig. 10. Behaviors of the $\tilde{C}_{n}$, DOM, and DOD ($\mu$) (from top to bottom) for the cases that $q$'s of the Kalman filter model in the second time intervals is too small.
which can be represented as

$$H_0 \quad \mu < \mu_T \forall k \text{ (If } \mu < \mu_T \text{ then there is no failure)}$$

$$H_1 \quad \mu \geq \mu_T \exists k \text{ (If } \mu \geq \mu_T \text{ then there is a failure)}$$

The spectral amplitudes of the process noise ($q$'s) in the four time intervals for three simulation scenarios for detection of change in process noise statistic is given in Table 3. Fig. 10 shows the behaviors of the $\hat{C}_{12}$, DOM, and $\mu$ for the cases that $q$'s of the Kalman filter model in the second time intervals is too small ($\hat{C}_{12} > 0$, DOM < 0, and $\mu \geq \mu_T$, respectively) and need to be increased.

To account for the greater uncertainty, the covariances need to be updated, through one of the following ways:

1. $Q_k \rightarrow Q_{k-1} + \Delta Q_k$; $R_k \rightarrow R_{k-1} + \Delta R_k$.
2. $Q_k \rightarrow Q_k \sqrt{\frac{1}{C_0(k)}} + \frac{1}{R_k}$; $R_k \rightarrow R_k \sqrt{\frac{1}{C_0(k)}} + \frac{1}{R_k}$.
3. $Q_k \rightarrow a Q_k$; $R_k \rightarrow b R_k$.

If (3) is utilized as an example, the filter equations can be augmented in the following way:

$$P_{k+1} = P_k + dt \left( \frac{2 P_{12} k}{C_0} \beta^{(k+1)} \right) + \frac{1}{R_k} \beta^{(k+1)} + \frac{1}{R_k} \beta^{(k+1)}$$

$$K_k = P_k H_k^T \left( H_k P_k H_k^T + \frac{1}{R_k} \beta^{(k+1)} \right)^{-1}$$

In case that $a = b = 1$, it deteriorates to the standard Kalman filter.

5. Performance degradation due to uncertainty in measurement noise statistic

In this section, we will focus on the performance degradation due to uncertainty in measurement noise statistic.

5.1. Filter-calculated covariance: continuous time Kalman filter

Table 4 shows the time varying measurement noise statistic ($r$'s) in the four time intervals (also 10 s for each time interval) for the three simulation scenarios. Three scenarios are designed as follows: Scenario (a)-filter model exactly matches the real world. Scenario (b)-filter model does not match the real world (too small in two of the time intervals). Scenario (c)-filter model does not match the real world (too large in three of the time intervals). Listing 3 provides the Matlab program for actual covariance prediction with incorrect $R_k$ information for the continuous time Kalman filter (for fixed measurement noise statistic: $r = 1$). Illustration for performance degradation (i.e., increase of covariance) for the three simulation scenarios is provided in Fig. 11. Table 5 provides the steady state covariances based on the continuous time Kalman filtering.

```
Listing 3. Simulation for actual covariance prediction with incorrect R_k information for the continuous
time Kalman filter (with fixed process noise PSD)

q = 1;
r_kf = 1;
r_truth = 9;

% (Predicted covariance matrix)
P11_kf(i+1) = P11_kf(i)+dt*(2*P12_kf(i)*P11_kf(i))end;
P12_kf(i+1) = P12_kf(i)+dt*(P22_kf(i)*P11_kf(i)-1/r_kf*P11_kf(i)*P11_kf(i));
P22_kf(i+1) = P22_kf(i)+dt*(q-1/r_kf*P12_kf(i)*P12_kf(i));
for i = FirstStep+FinalStep
```

5.2. Filter-calculated covariance: discrete time Kalman filter

Fig. 12 shows the flow chart for actual covariance prediction with incorrect Rk information for the discrete time Kalman filter based on the flow chart for computing the actual covariance as in Fig. 5. Only Qk is used representing both Qk\_truth and Qk\_kf, since these two are assumed the same in this subsection. Detailed implementation procedure is provided in Listing 4. It should be noted that in practical implementation, the actual covariance matrix P\_plus_actual may not be available due to the fact that Rk\_truth may not be available.

<table>
<thead>
<tr>
<th>Scenarios</th>
<th>Real world</th>
<th>Kalman filter model</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>r = 1 → 4 → 9 → 1</td>
<td>r = 1 → 4 → 9 → 1</td>
</tr>
<tr>
<td>b</td>
<td>r = 1 → 9 → 1 → 1</td>
<td>r = 1 → 9 → 9 → 1</td>
</tr>
<tr>
<td>c</td>
<td>r = 1 → 9 → 9 → 1</td>
<td>r = 1 → 9 → 9 → 1</td>
</tr>
</tbody>
</table>

5.2. Filter-calculated covariance: discrete time Kalman filter

Listing 4. Simulation for actual covariance prediction with incorrect Rk information for the discrete time Kalman filter

```matlab
%(Predicted covariance matrix)
P_minus = PHI*P_plus*PHI\_0' + Qk;
P_plus = (eye(size(K_kf*H)\_0'))\_0*P_minus;
%(Calculating gains based on predicted covariance matrix)
K_kf = P_minus*H\_0'*inv(H*P_minus*H\_0' + Rk_kf);
%(Actual covariance matrix)
P_minus_actual = PHI*P_plus_actual*PHI\_0' + Qk;
I_KH = eye(size(K_kf*H))\_0' - K_kf*H;
P_plus_actual = I_KH*P_minus_actual*I_KH\_0' + K_kf*Rk_truth*K_kf\_0';
```
Fig. 11. Degradation of the covariance for the three scenarios given in Table 4.
5.3. Detection on the change of measurement noise statistic

Similar to Section 4, the following rules can be utilized to implement the idea of covariance matching:

1. IF \( \text{DOM} \approx 0 \) THEN \( (R) \) is unchanged (This indicates that \( \hat{C}_v \) is about the same as \( C_v \), the measurement noise statistic should be maintained.);
2. IF \( \text{DOM} > 0 \) THEN \( (R) \) is decreased (This indicates that \( \hat{C}_v \) is less than \( C_v \), the measurement noise statistic should be decreased.);
3. IF \( \text{DOM} < 0 \) THEN \( (R) \) is increased (This indicates that \( \hat{C}_v \) is larger than \( C_v \), the measurement noise statistic should be increased.).

Now, suppose that DOD \( (\mu) \) is employed as the test statistic, and \( \mu_T \) represents the chosen threshold, the following fuzzy rules can be utilized:

1. IF \( \mu \geq \mu_T \) too large THEN \( (R) \) is increased (the measurement noise statistic is too small and needs to be increased);
2. IF \( \mu < \mu_T \) too small THEN \( (R) \) is decreased (the measurement noise statistic is too large and needs to be decreased).

Fig. 13 shows the behaviors of DOM for the cases that spectral amplitudes of the measurement noises \( (r's) \) of the Kalman filter model do not match the real world. For subplot (a), the values utilized are too small in the second and third time intervals, and therefore negative DOM values will be obtained \( (\text{DOM} < 0) \), and the measurement noise statistics in these two intervals should be increased for performance improvement. Furthermore, for subplot (b), the values are too small in the first, second and last time intervals, and therefore positive DOM values will be obtained \( (\text{DOM} > 0) \), and the measurement noise statistics in these three intervals should be decreased. Fig. 14 shows the behaviors of \( \mu \) for the cases that \( r's \) of the Kalman filter model do not match the real world. For subplot (a), the values are too small in the second and third time intervals, and therefore, \( \mu \geq \mu_T \), the measurement noise statistics in these two intervals need to be increased. For subplot (b), the values are too small in the first, second and last time intervals, and therefore, \( \mu < \mu_T \), the measurement noise statistics in these three intervals need to be decreased.

6. Conclusions

This paper has pointed out several practical remarks on performance optimality and degradation, which are useful in designing a suitable Kalman filter. Lessons learned from this paper include: (1) additional notes on Kalman filter and arbitrary gain suboptimal filter; (2) Kalman filter solution and evaluation of estimator optimality for verification of minimum variance optimality with sensitivity analysis; (3) performance degradation due to uncertainty in process noise statistic; (4) performance degradation due to uncertainty in measurement noise statistic. Consistency check between the filter-calculated covariances versus actual mean square
errors is provided, which can be used not only as a verification procedure for the filtering correctness, but also as an approach for making trade-off in designing the suitable Kalman filter. In addition, behaviour of some innovation related parameters have been discussed, which are useful for providing the useful information in designing the noise-adaptive Kalman filter and for achieving the system integrity design.

Fig. 12. Flow chart for actual covariance prediction due to incorrect Rk information for the discrete time Kalman filter.

Fig. 13. Behaviors of DOM for the cases that r’s of the Kalman filter model are (a) too small in two time intervals; (b) too large in three time intervals, respectively.
The authors gratefully acknowledge the support of the National Science Council of the Republic of China through Grant NSC 95-2221-E-019-026.

Acknowledgement

The authors gratefully acknowledge the support of the National Science Council of the Republic of China through Grant NSC 95-2221-E-019-026.

References


Fig. 14. Behaviors of $\mu$ for the cases that $r$’s of the Kalman filter model are (a) too small in two time intervals; (b) too large in three time intervals, respectively.