Signal Processing, Homework Assignment 2

- Kalman filtering

Elvira Boman
Uppsala University, 2016

1. Introduction

This report describes the implementation of a Kalman filter to estimate position and velocity of an unknown unit using two different methods for combining sensor measurements.

2. System

A state space linear system is described by

$$x_{k+1} = A x_k + w_k = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k + w_k \tag{1}$$

with noise w_k being i.i.d. Gaussian with mean zero. The covariance matrix is

$$Q = \begin{bmatrix} (T^4/4) & (T^3/2) \\ (T^3/2) & T^2 \end{bmatrix} q \tag{2}$$

(1) can be used to model a unit which movements are approximated with constant velocity and constant acceleration between two sampling instants. The first component of the vector x is the position and the second one is the velocity.

$$y_{1,k} = [1 \; 0] x_k + v_{1,k} \tag{3}$$

$$y_{2,k} = [1 \; 0] x_k + v_{2,k} \tag{4}$$

The measurement noise v is assumed to be i.i.d Gaussian and independent with mean zero and variances σ_1^2 and σ_2^2.

The system parameters are given as

$T = 0.1$
$q = 0.2$
$\sigma_1^2 = 0.2$
$\sigma_2^2 = 0.3$

3. The Kalman filter

A Kalman filter uses a series of measurements from which estimates of unknowns are made. It is done recursively, with only the estimated state from the previous measurement and the current measurement are needed to make an estimation for the current state. The Kalman filter can be understood as two phases: “predict” and “update”. In the predict phase the previous state estimate is used to predict the current state. This is known as *a priori* state estimation. In the update phase, the a priori estimate is updated with current measurements. This is known as *a posteriori* state estimation.

Predict	
A priori state estimation	$\hat{\mathbf{x}}_{k\|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1\|k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}$
A priori estimate covariance	$\mathbf{P}_{k\|k-1} = \mathbf{F}_k \mathbf{P}_{k-1\|k-1} \mathbf{F}_k^{\mathrm{T}} + \mathbf{Q}_k$
Update	
Measurement residual	$\tilde{\mathbf{y}}_k = \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k\|k-1}$
Residual covariance	$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k\|k-1} \mathbf{H}_k^T + \mathbf{R}_k$

Kalman gain	$\mathbf{K}_k = \mathbf{P}_{k\|k-1}\mathbf{H}_k^T\mathbf{S}_k^{-1}$
A posteriori state estimate	$\hat{\mathbf{x}}_{k\|k} = \hat{\mathbf{x}}_{k\|k-1} + \mathbf{K}_k\tilde{\mathbf{y}}_k$
A posteriori estimate covariance	$\mathbf{P}_{k\|k} = (I - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k\|k-1}$

Table 1. Pseudo code for Kalman filtering.

An example of implemented Kalman filtering can be seen in table 1. Kalman filtering is commonly used for guidance, navigation and control of vehicles.

4. Method

1. *Method 1*

The two measurements $y_{1,k}$ and $y_{2,k}$ are combined into one vector

$$y_k = \begin{bmatrix} y_{1,k} \\ y_{2,k} \end{bmatrix} \quad (5)$$

which is seen as a single measurement y_k. A Kalman filter is then used to estimate x_k from y_k.

2. *Method 2*

The two measurements $y_{1,k}$ and $y_{2,k}$ are linearly combined into y_k

$$y_k = y_{1,k} + y_{2,k} \quad (6)$$

A Kalman filter is then used to estimate x_k from y_k.

5. Results

Both methods share the matrixes A and Q, which are given by (1) and (2) and the respectively to be

$$A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$$

$$Q = \begin{bmatrix} 5e\text{-}6 & (1e\text{-}4) \\ 1e\text{-}4 & 2e\text{-}2 \end{bmatrix}$$

1. *Method 1*

The matrix C can be found by combining the vectors (3) and (4) to a new vector which makes the space state model

$$y_k = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} x_k + \begin{bmatrix} v_{1,k} \\ v_{2,k} \end{bmatrix}$$

This gives us that the matrix C is

$$C = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}$$

Knowing this, the covariance matrix R can be written as

$$R = \begin{bmatrix} \sigma_2^1 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.3 \end{bmatrix}$$

Fig. 1 shows the actual position and velocity and the estimations of position and velocity done with Kalman filtering using method 1, for a 100 samples.

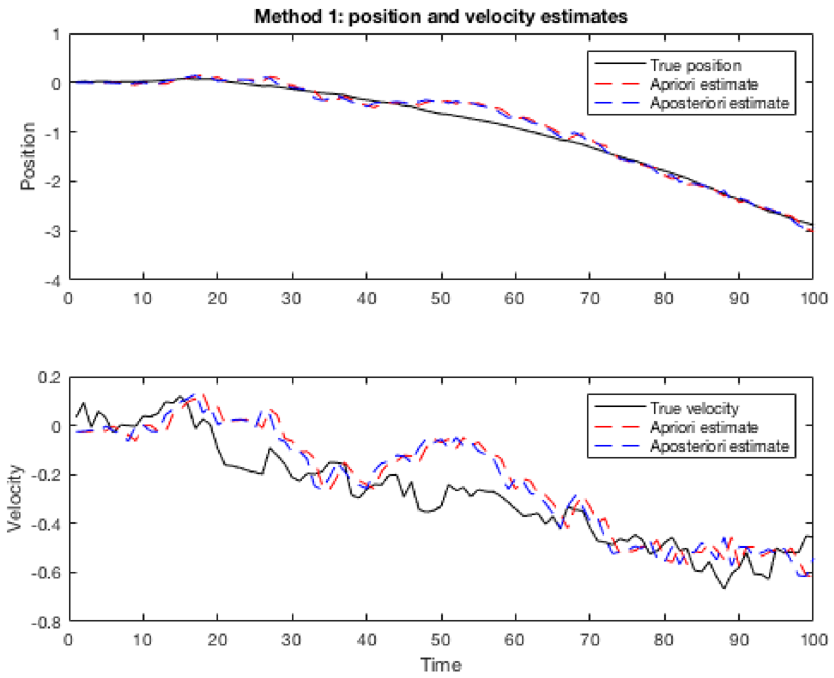

Figure 1. Actual position and velocity, and position and velocity estimates made with a Kalman filter using method 1, 100 samples.

Fig. 2 shows the trace of the error covariance matrices for a priori and a posteriori measurements using method 1.

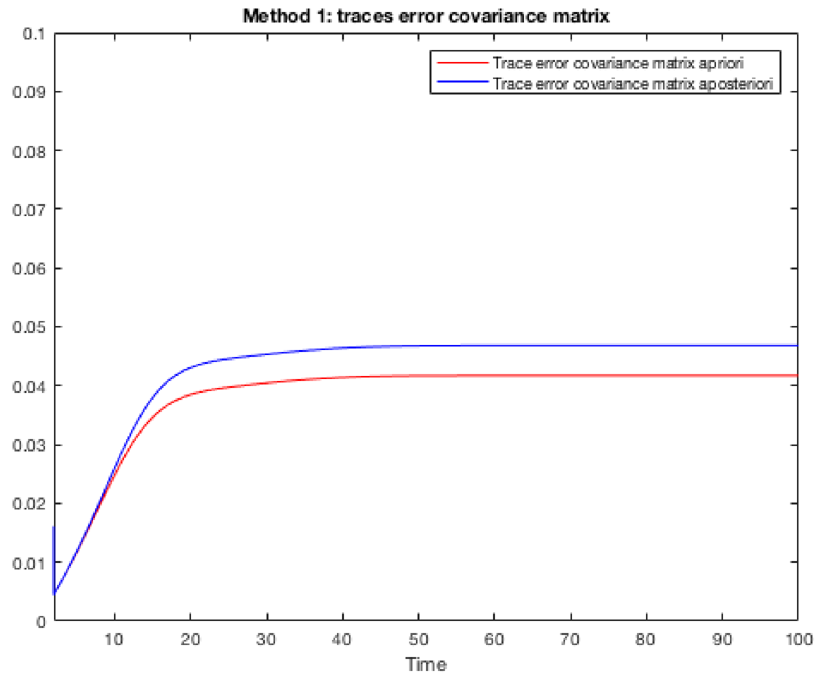

Figure 2. Trace of the error covariance matrices using method 1.

The a priori error covariance matrix converges to 0.047 when the samples k goes to infinity, as shown in fig. 2. As can be seen in fig. 1 (100 samples) and fig. 3 (1000 samples), the a priori estimate of the position x_k^1 gets more accurate the more samples that are taken.

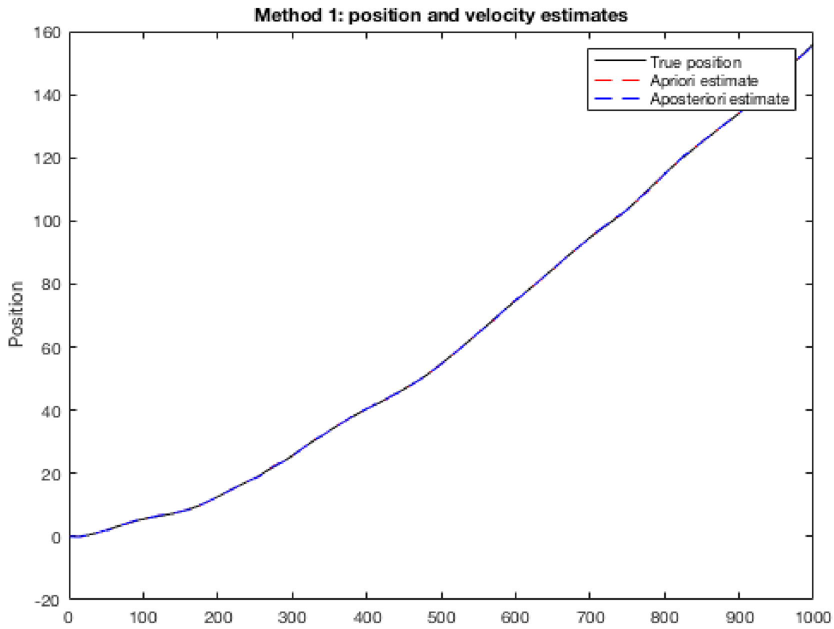

Figure 3. Actual position and position estimates made with a Kalman filter using method 1, 1000 samples.

2. *Method 2*

The matrix C can be found by linearly combining the vectors (3) and (4)

$$y_k = [1\,0]\,x_k + v_{1,k} + [1\,0]\,x_k + v_{2,k} = [2\,0]\,x_k + v_k$$

This gives us that the matrix C is

$$C = [2\,0]$$

Knowing this, the covariance matrix R can be written as

$$R = \sigma_2^1 + \sigma_2^2 = 0.5$$

Fig. 4 shows the actual position and velocity and the estimations of position and velocity done with Kalman filtering using method 2, for a 100 samples.

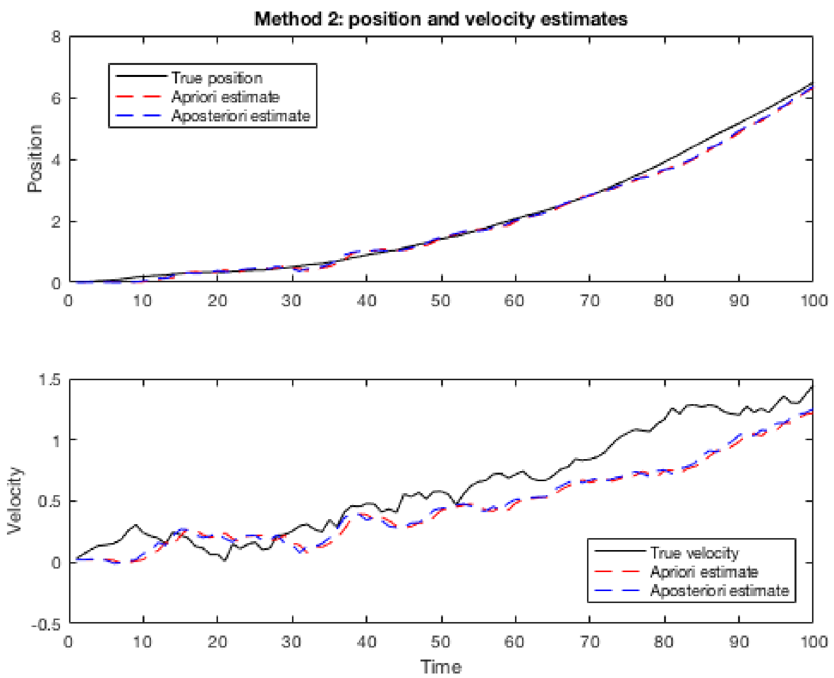

Figure 4. Actual position and velocity, and position and velocity estimates made with a Kalman filter. using method 2, 100 samples

Fig. 5 shows the trace of the error covariance matrices for a priori and a posteriori measurements using method 2.

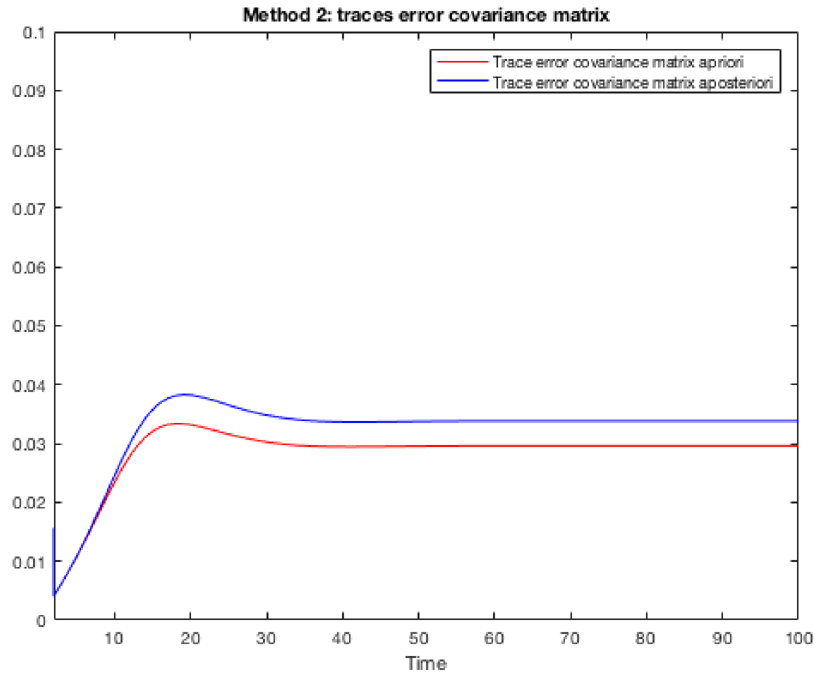

Figure 5. Trace of the error covariance matrices using method 2.

The a priori error covariance matrix converges to 0.034 when the samples k goes to infinity. The a priori estimate of x_k:s position component gets more accurate the more samples that are taken for method 2 just as for method 1, as can be seen in fig. 4 (100 samples) and fig. 6 (1000 samples).

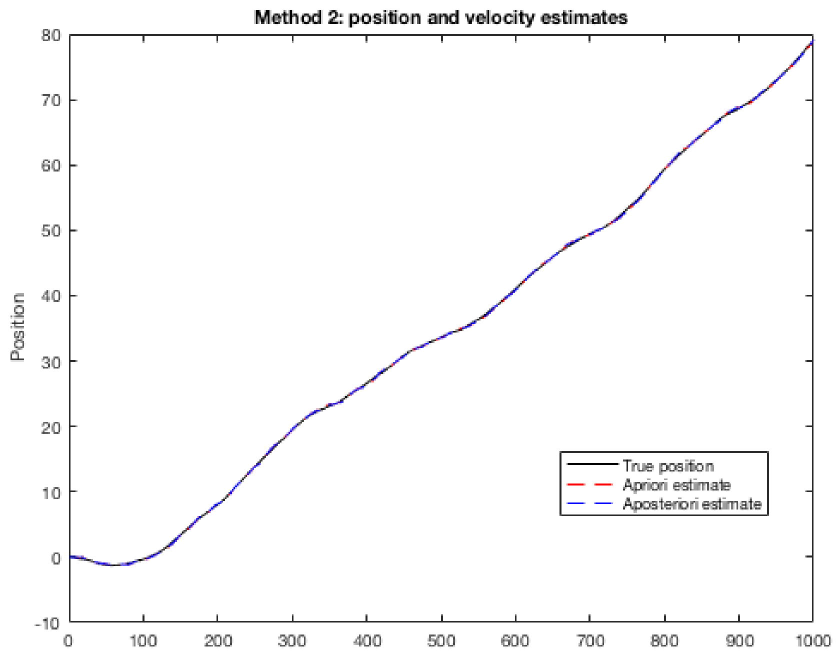

Figure 6. Actual position and position estimates made with a Kalman filter using method 2, 1000 samples.

3. *Comparison method 1 and 2 and simulations with added noise*

Fig. 7 shows a comparison between a priori error matrices for method 1 and 2. As can be seen, the curves starts off similar, but method 2 then converges to a lower value than method 1.

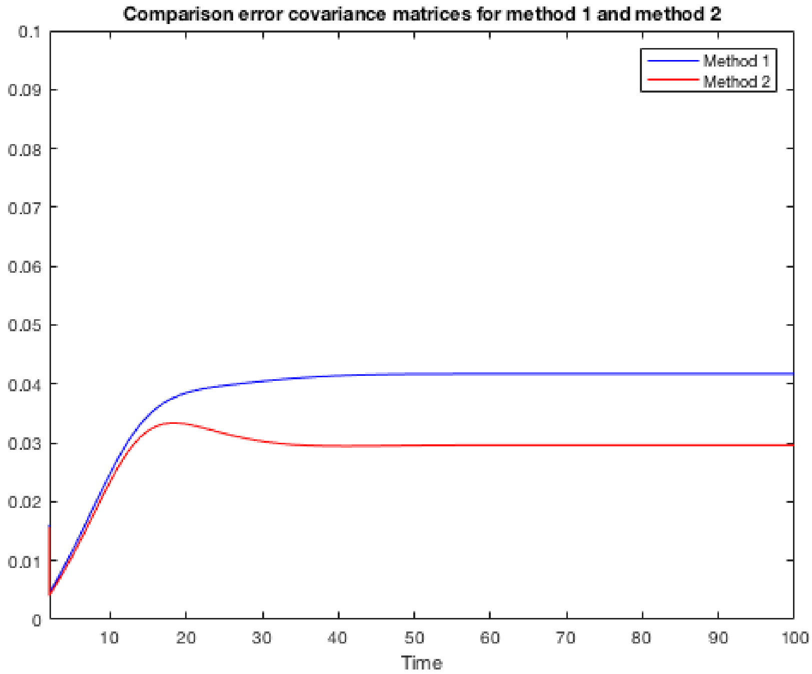

Figure 7. Comparisons trace of a priori error matrices for method 1 and 2

If the sensor measurements are not perfectly received, but instead corrupted by noise n, (5) for method 1 have to be modified to

$$y_k = \begin{bmatrix} y_{1,k} \\ y_{2,k} \end{bmatrix} + \begin{bmatrix} n_{1,k} \\ n_{2,k} \end{bmatrix}$$

where n_k is i.i.d Gaussian with zero means and variance $\sigma_n^2 = 0.25$. A Kalman filter is used to estimate x_k from y_k.

Fig. 8 shows the actual position and velocity and the estimations of position and velocity for a noise corrupted signal done with Kalman filtering using method 1, for a 100 samples.

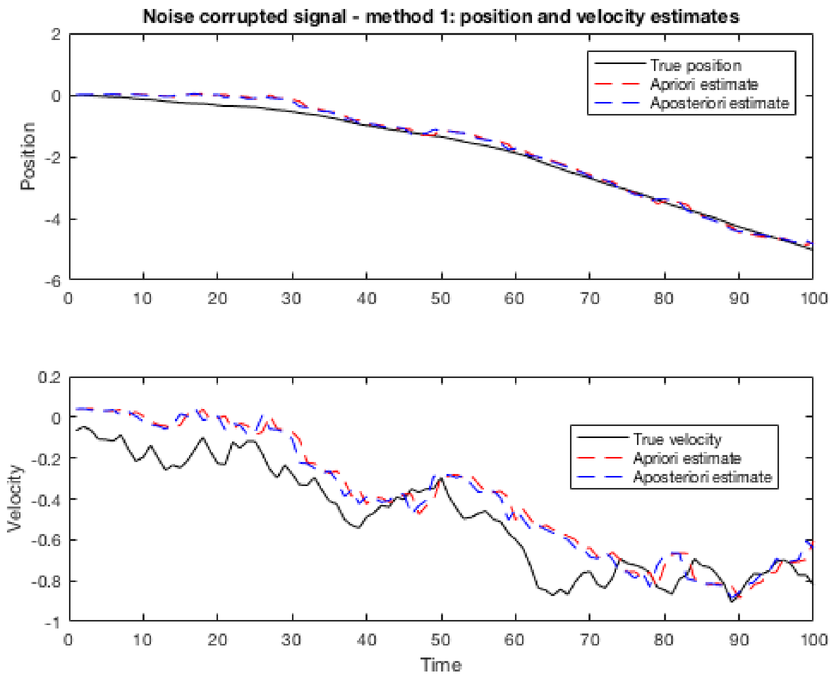

Figure 8. Noise corrupted signal: actual position and velocity, and position and velocity estimates made with a Kalman filter using method 1, 100 samples.

Fig. 9 shows the trace of the error covariance matrices for a priori and a posteriori measurements for a noise contaminated signal using method 1. Here the a priori error covariance matrix converges to 0.066 when the samples k goes to infinity – a higher value compared to the non-noise contaminated signal using the same method.

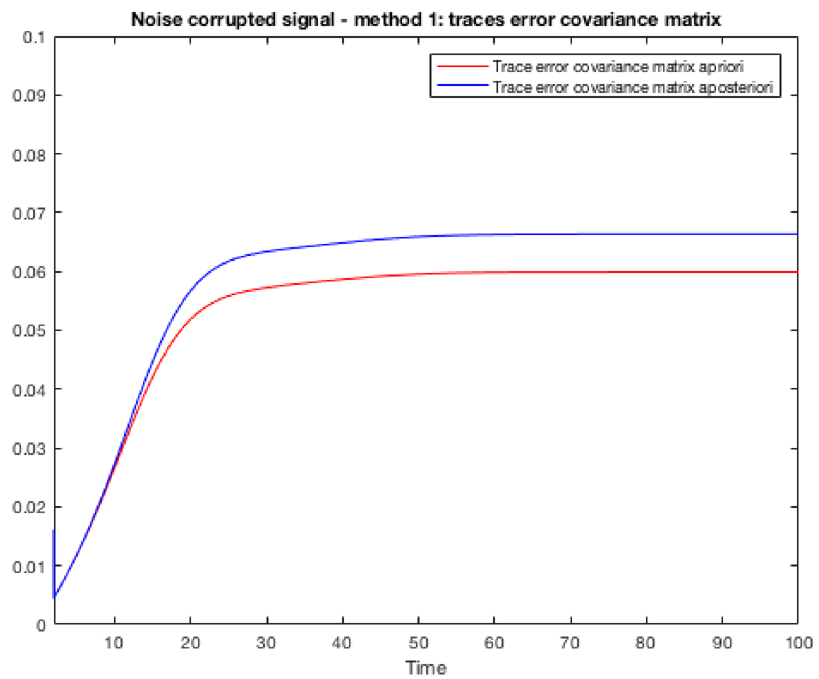

Figure 9. Noise corrupted signal: trace of the error covariance matrices using method 1.

For method 2, a noise corrupted signal demands (6) to be modified to

$$y_k = y_{1,k} + y_{2,k} + n_k$$

where n_k also is i.i.d Gaussian with zero means and variance $\sigma_n^2 = 0.25$. A Kalman filter is used to estimate x_k from y_k.

Fig. 8 shows the actual position and velocity and the estimations of position and velocity for a noise corrupted signal done with Kalman filtering using method 2, for a 100 samples.

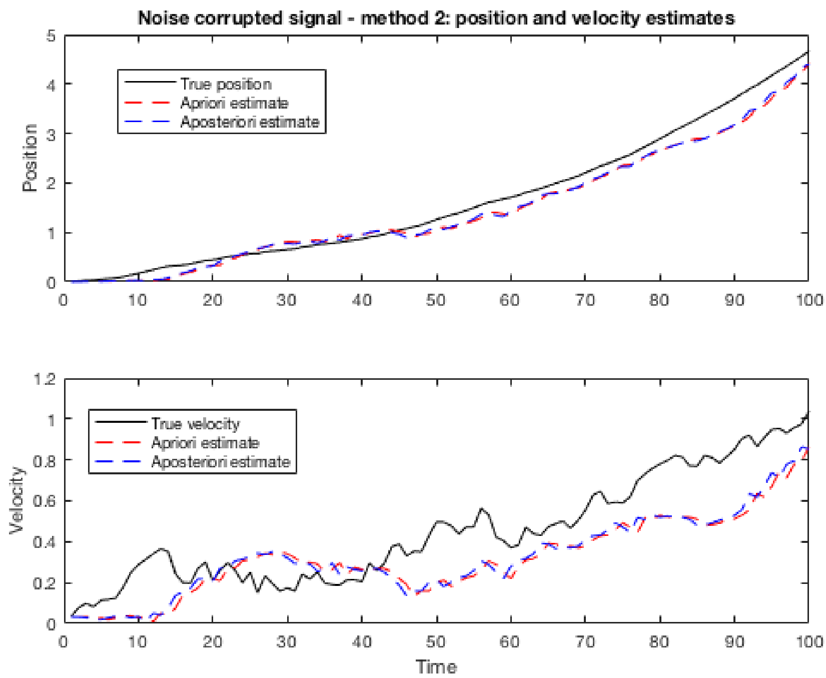

Figure 10. Noise corrupted signal: actual position and velocity, and position and velocity estimates made with a Kalman filter using method 2, 100 samples.

Fig. 11 shows the trace of the error covariance matrices for a priori and a posteriori measurements for a noise contaminated signal using method 2. Just as in the case with method 1, the a priori error covariance matrix converges to a higher value compared to the non-noise contaminated signal using the same method, in this case 0.041.

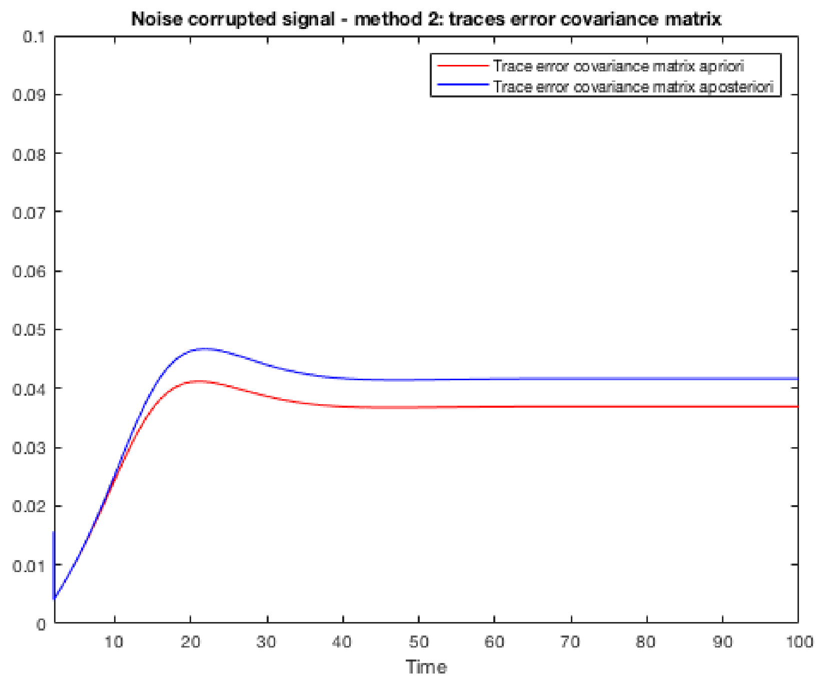

Figure 11. Noise corrupted signal: trace of the error covariance matrices using method 2.

Fig. 12 shows a comparison between a priori error matrices for noise corrupted signals using method 1 and 2.

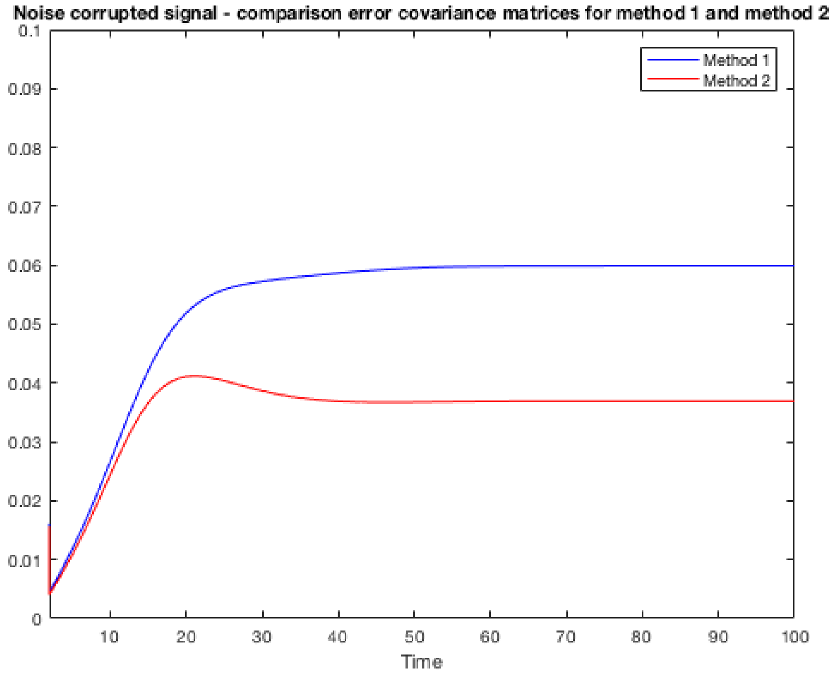

Figure 12. Comparisons trace of a priori error matrices for method 1 and 2

Fig. 12 shows a similar pattern to fig. 7 which is a comparison between the non-noise corrupt signals, only that the gap between method 1 and 2 has widened in the noise corrupt case.