EL2320 Lab1

Adel Abdelsamed, 20000321-2412

4 November 2024

1 Part 1

Linear Kalman Filter:
1. The state x_t encompasses all variables that are necessary to describe the environment. Control u_t are variables that produce a change in the state x_t. They are part of the dynamic model governing the relation between x_t and x_{t-1}, since the next state depends also on the control as in $x_t = g(u_t, x_{t-1})$. Lastly, measurements z_t are a result of a sensing process, where sensors are used to gather information about the state. An example is a robot, where the state describe the robot's pose. They are modeled depending only on the state at time t x_t through $z_t = h(x_t)$. Hence, they give indirect information on the state. The difference is that the control is used in the prediction step, whereas the measurement is used in the update step. In the case of a 2d-localization, the state would be (x, y, θ). The control in this case might be the velocity v of the robot's wheels and the heading angle ω. The measurements can include range and bearing to a specfic landmark.

2. Mathematically, the update step in a Kalman filter can not result in an increase in uncertainty. From the information filter we know that the covariance of the belief after the update as

$$\Sigma_t = (\bar{\Sigma}_t^{-1} + C_t^T Q_t^{-1} C_t)^{-1}, \quad (1)$$

where the term $C_t^T Q_t^{-1} C_t$ is positive semi-definite and hence the uncertainty cannot increase after the update.

3. During the update step, the covariance matrix of the measurement model (Q-matrix) decides the relative weighting between measurements and belief, assuming that we set the R-matrix as fixed. The influence is explained more in Question 4.

4. A larger covariance for the measurement model (Q-matrix) (while the R-matrix remains the same) results in a smaller Kalman Gain, which will lead to the predicted mean $\bar{\mu}_t$ dominating the updated mean μ_t and the error between the predicted measurement $C_t\mu_t$ and the actual measurement z_t contributing minimally to the updated mean. Furthermore, the updated covariance matrix Σ_t will decrease minimally and will be close to the predicted covariance matrix $\bar{\Sigma}_t$. Hence, in short the update step will trust the prediction step more than the actual measurement.

5. Measurements have an increased effect on the updated state estimate if the Kalman Gain K_t is increased. A smaller covariance matrix of the measurement model (Q-matrix) would increase the Kalman Gain (assuming the R-matrix remains the same) and would lead to the desired effect. The deciding factor is the relative weighting between the R and Q covariance matrix.

6. During the prediction step, the belief uncertainty typically increases in most real systems, but it depends on the state transition matrix A_t. Since the state transition is stochastic $p(x_t|u_t, x_{t-1})$, it is expected that the belief uncertainty grows with the uncertainty in the state transition. This can be shown by looking at the equation of the covariance in the prediction step

$$\bar{\Sigma}_t = A_t \Sigma_{t-1} A_t^T + R_t, \tag{2}$$

where the uncertainty of the belief after the prediction is the covariance in the previous time step t-1 pre- and postmultiplied with A_t and the process uncertainty covariance R_t is added. Hence, in general the uncertainty can theoretically go down during the prediction step but inmost real systems it increases , since A_t is often about I and R is signifcantly larger.

7. Since the Kalman Filter is not an approximation and leads to a closed-form solution, we can say it is optimal.

8. The parameters of the belief when using a Kalman Filter are the first two moments of the belief distribution; the mean vector μ_t and the covariance matrix Σ_t. Since the belief in a Kalman Filter is always a Gaussian, the distribution is fully characterized by the parameters: μ_t, Σ_t.

Extended Kalman Filter:

9. The extended Kalman Filter relaxes the assumption of a linear motion model and a a linear measurement model to general nonlinear systems

$$\begin{aligned} x_t &= g(u_t, x_{t-1}) + \epsilon_t \\ z_t &= h(x_t) + \delta_t, \end{aligned} \tag{3}$$

that are approximated using Taylor Series at μ_{t-1} and applies Kalman Filter to the linearized system. The result is a linearized system and such that the resulting distributions are still closed-form Gaussians and the Kalman Filter computations remain unchanged apart from three differences. The state and measurement predictions are replaced by the nonlinear systems shown above. Moreover, EKFs use the Jacobians G_t, H_t of the nonlinear systems evaluated at μ_{t-1} instead of A_t and C_t.

10. No, the EKF can also converge to a wrong solution or diverge completely. This can arise if the nonlinearity is strong or the uncertainty is big such that the Gaussians are distorted by the non-linear transform. Another reason might be that the assumption that the measurement noise and/or the process noise are not normally distributed. Furthermore, modeling errors and numerical problems could cause the EKF to diverge.

11. Yes, we can inject noise into the system thereby increasing the model uncertainty. This can be done by choosing the relative weighting between the process and measurement noise covariance matrices. Hence, the matrices Q and R can be used as tuning knobs in order to get a stable EKF. However, if the divergence is due to the data association process, we can change the matching thresholds.

Localization:

12. Since the orientation is not known and not measured and we only get a distance measurement, the posterior distribution will be a ring with the landmark as the center point. Since the

measurement is inflicted by Gaussian noise, the line of the circular posterior distribution is the mean of the location and for each point on a circle there is an ellipsoid around the mean capturing the second moment. Lastly, since we do no measure the bearing, the posterior of the orientation is uniformly distributed between $[-\pi,\pi]$.

13. Nothing will change except that now the heading is not uniform but is correlated.

14. The posterior distribution will form an arc or c-shape.

15. Since the orientation was unknown and we traveled a distance without a measurement, the posterior is now as explained in the previous question. The problem we face now when measuring the range and bearing is that data association becomes much more harder. We might associate the measurement with the wrong landmark and perform the update. In this case, the EKF might not recover from this wrong update and hence will diverge or become inconsistent. Furthermore, the linearization will produce an update 'direction' that is a straight line and hence the estimate can not move along the c-shape, which will lead to the update diverging2

2 Part 2

Question 1: Since the dimension of the state is n = $\dim(x_k) = 2$, the process noise is of dimension $\dim(\epsilon_k) = \text{n} = 2$. Furthermore, the dimension of the measurement noise is the same as the dimension of z_k, $\dim(\delta_k) = \dim(z_k) = 1$. Since white Gaussian have zero mean, it is only required to define the covariance of ϵ_k and variance and δ_k, Σ_{ϵ_k} $\sigma^2_{\delta_k}$ respectively.

Question 2: Table 1 shows the role of each variable used in WarmUp/kf car.

Table 1: Roles of the variable used in WarmUp/kf car.

Parameter	Simulated state vector
x	Estimated state vector
xhat	Mean of the estimated state vector.
P	Covariance matrix of the estimated state vector.
G	Models how the process noise influences the state. In this case $G = I_2$, for the consistency of dimensions.
D	Models how the measurement noise influences the expected measurement. In this case $D = 1$, for the consistency of dimensions.
Q	Process noise covariance matrix; used as tuning parameter for the filter.
R	Measurement noise covarinace matrix; used as a tuning parameter for the filter.
wStdP	Standard deviation of the postion noise, used only for simulation.
wStdV	Standard deviation of the speed noise, used only for simulation.
vStd	Standard deviation of the measurement noise, used only for simulation.
u	Control input for simulation and estimation models
PP	Matrix containing the covariance matrix of the estimated state at each time step reshaped into its columns.

Question 3:

The relative weighting of the measurement noise covariance matrix and the process noise covariance matrix is the deciding factor. If the measurement noise covariance matrix R is increased, the resulting Kalman Gain is increased and hence more weight is given to the predicted state estimate and a faster convergence rate is expected. If the process noise covariance matrix Q is increased, the resulting Kalman Gain is decreased and hence more weight is given to the measurement and a slower convergence rate is expected. If both the modeled measurement and process noise covariance matrix are increased, nothing should change in the Kalman Gain, since the relative weighting remains unchanged, but the estimated covariance matrix should be larger. The above is confirmed by simulation.

In Figure 1 the effect of increased Q and R one at a time is illustrated. For an increased process noise covariance matrix Q (second row), the Kalman Gain is indeed decreased compared to the default case, which results in slower convergence. The reliance os more on the predicted estimate and hence the trajectories are more smooth compared to the default case. For an increased measurement noise error covariance matrix R (third row), the Kalman gain is increased and the convergence rate is faster compared to the default case.

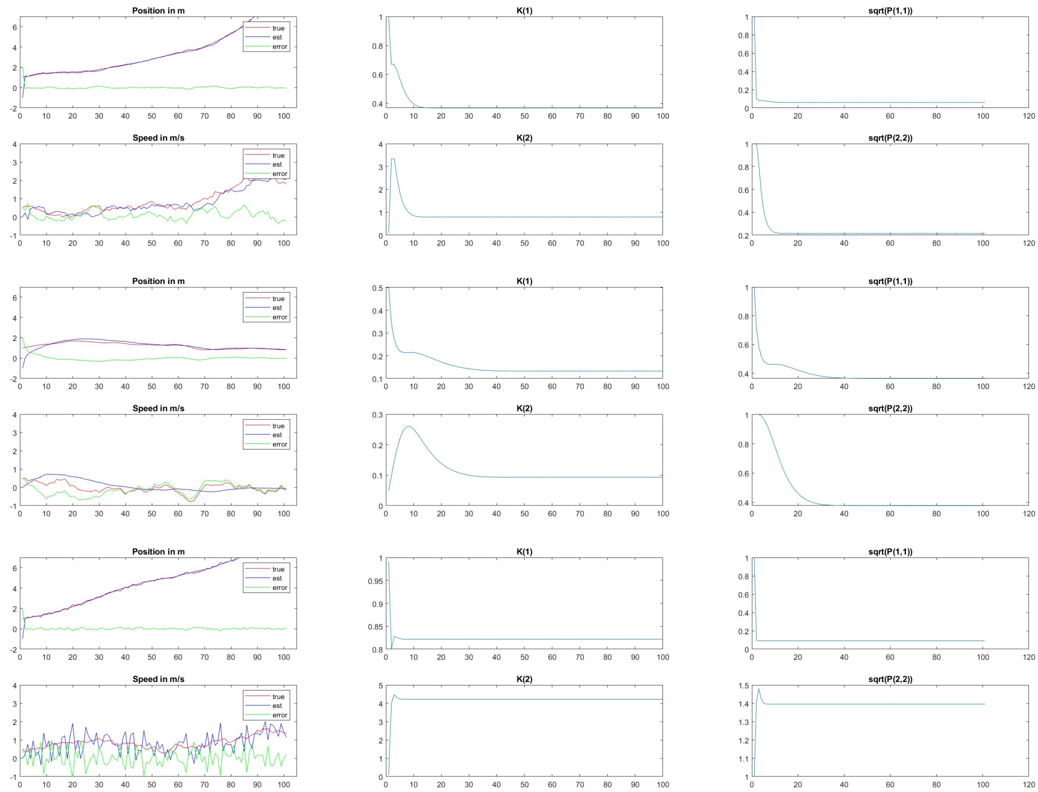

Figure 1: Effect of increasing the process (Q) and measurement noise (R) error covariance matrix one at a time. The second row represents the default value of R and Q. The second row represents the case with 100Q and the last row the case with 100R. The first column represents the state and velocity estimates and the true values. The second column represents the Kalman Gain, while the last column represents the standard deviation of the estimated state.

In Figure 2 the effect of increased Q and R (by the same factor) simultaneously is illustrated.

The Kalman Gain stays approximately the same and hence convergence is not affected. The steady state values for the estimated state covariance matrix increases/decreases when Q and R are increased/decreased. This is expected since we introduce/reduced the uncertainty when increasing/decreasing Q and R.

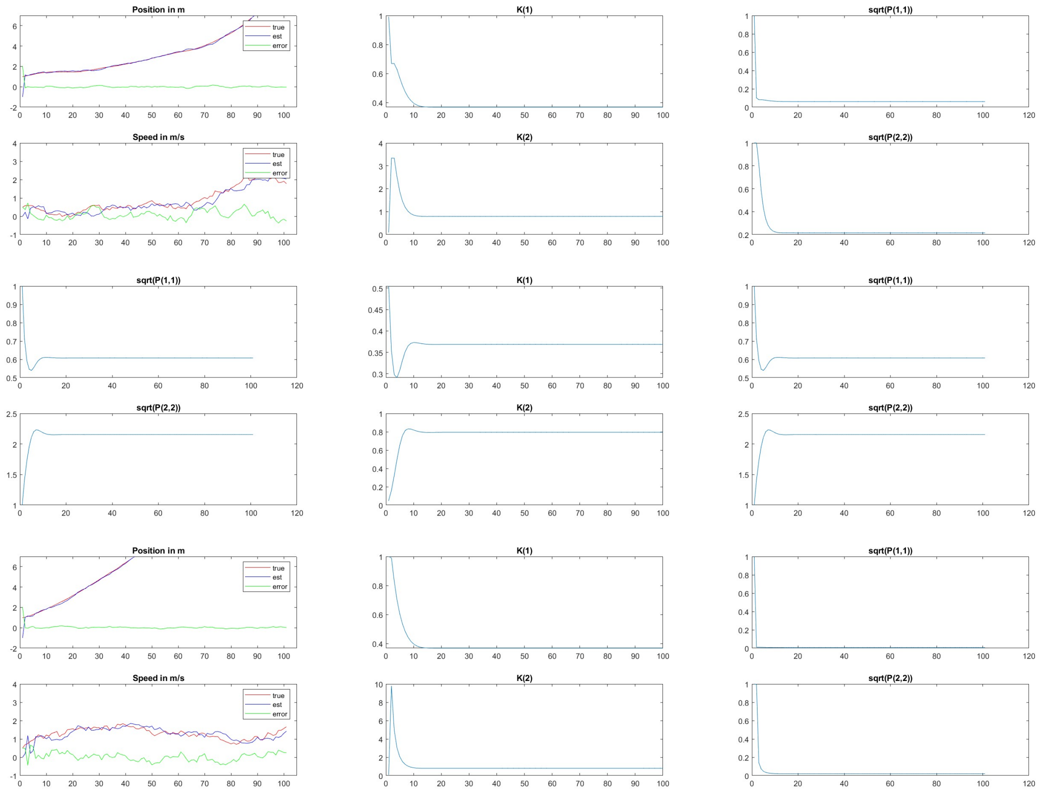

Figure 2: Effect of increasing the process (Q) and measurement noise (R) error covariance matrix at the same time. The second row represents the default value of R and Q. The second row represents the case with 100Q and 100R and the last row the case with 0.01Q and 0.01R. The first column represents the state and velocity estimates and the true values. The second column represents the Kalman Gain, while the last column represents the standard deviation of the estimated state.

Question 4: Decreasing the covariance of the initial state estimate makes convergence slower, while increasing the covariance will lead to faster convergence after a large initial error and will make the effect of the initial covariance negligible. If the error between the norm of the initial estimate xhat and the true state is increased, the estimated state converges slower to the true state. If the norm is decreased, convergence is fast.

Question 5: The upper equation in (2) is responsible for both the prediction and update step simultaneously. The following is a break up of the prediction and update step in (2) and the relation to the equations in (3).

The inner integral in the first equation corresponds to the belief $\overline{bel}(x_t)$ after the prediction step as shown in (3). The update step corresponds to the multiplication of $\overline{bel}(x_t)$ by the measurement model $p(z_t|x_t, M)$ and a constant normalization parameter η. This corresponds to the first equation in (3).

Question 6: The assumption that measurements are independent is valid based on our model. Since the measurement model depends only on the state and the landmark at each time and the measurement noise is assumed white Gaussian (hence no correlation between the variables and only diagonal entries in the covariance matrix).

Question 7: Since $\delta_m = X_2^2(\lambda_M)$ is the cumulative distribution function of the χ_2^2-distribution, the parameter δ_M must lie within the bounded region [0, 1]. If δ_m is chosen high, this means that only measurements with high Mahalonobis distances are disregarded as outliers. Hence, we accept many measurements as valid and rarely reject any measurements as outliers. If we choose a δ_M small, this means that threshold for outliers is reduced and hence many measurements will be disregarded as outliers. For the case of reliable measurements we should probably increase the threshold for outliers since we can be sure that the measurements made are actually a feature in the map. Assuming a significance level of $\alpha = 0.95$ we can choose $\lambda_M = 5.99$. In the case of unreliable measurments we should choose a lower threshold. For example, given a significance level of $\alpha = 0.5$ we choose $\lambda_M = 1.386$.

Question 8: In the sequential update approach, initial noisy measurements will introduce bias into the estimated state, which might lead to slower convergence or divergence if the bias is large enough. This is because the noise propagates through the system until the next measurements. Lastly, sequential update approach will make the filter more sensitive to outliers causing shift to the mean and possibly culminating in inconsistency.

Question 9:
In Algorithm 4, the expected measurement (from our measurement model) $\hat{z}t, j$, the Jacobian of the measurement model H_{t_j} and the measurement uncertainty $S_{t,j}$ are computed in the nested loop. Hence, if we assume N observations in z_t and M landmarks, the above variables are computed $N \cdot M$ times. Since the above variables do not depend on the observations, they should only be computed once for each landmark (M times). Hence, they can be pulled out of the observations loop and performed only M times. The pre-computed values can then be used inside the nested loop for the computation of $\nu_t^{i,j}$, $D_t^{i,j}$ and $\phi_t^{i,j}$.

The reduction of redundant computations increases the more observations there are.

Question 10: The dimensions are shown in Table 2, where n is the number of inliers. As the number of observations (hence n) grow, the batch update will become more computationally expensive.

Table 2: Comparison of dimensions between sequential and batch update in the observations

	Sequential Update	Batch Update
$\dim(\bar{\nu}_t)$	1 x 2	1 x 2n
$\dim(\bar{H}_t)$	2 x 3	2 x 3n

3 Simulation

Dataset 1:

The mean absolute values of the simulation using the first dataset are shown in Figure 3. The mean absolute error in the x- and y-direction are less than 0.01 m. Furthermore, the mean absolute error in the angle corresponds to $\frac{0.15^\circ}{180^\circ}\pi \approx 0.002618$ *rad*, which is less than the required 0.01 rad. Furthermore, the evolution of the estimated covariance matrix is shown in Figure 4. The observed peaks at approx. 50 s, 60 s, 110 s correspond to the case where the robot is at a corner and due to symmetries in the map the uncertainty is expected to grow.

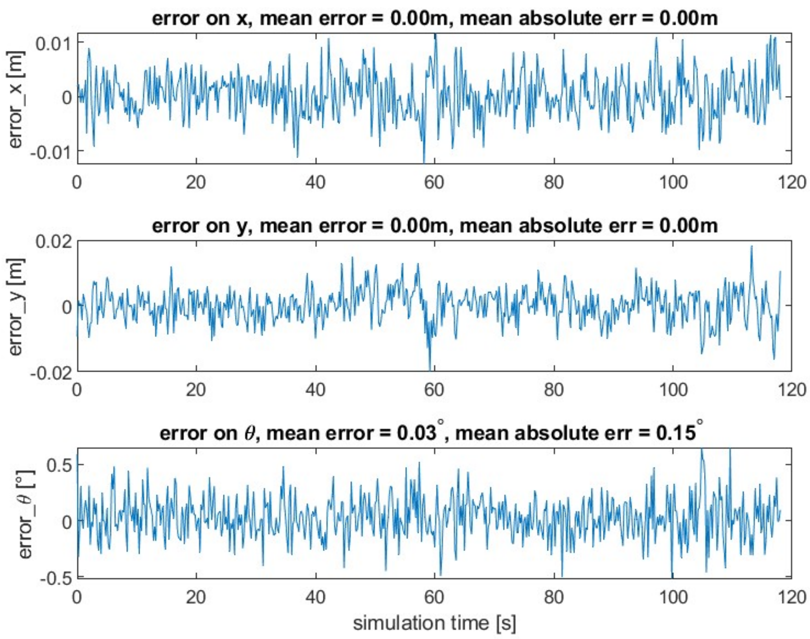

Figure 3: Evolution of the error for dataset 1.

Dataset 2:
The process noise covariance matrix was tuned to the following values

$$R = \begin{pmatrix} 0.01^2 & 0 & 0 \\ 0 & 0.01^2 & 0 \\ 0 & 0 & \left(\frac{\pi}{180}\right)^2 \end{pmatrix}. \tag{4}$$

The threshold of the outlier detection was selected to 12 % ($\delta_M = 88\%$) to dismiss the bad measurements and thereby prevent an inconsistent estimate. The mean absolute values of the simulation using the second dataset are shown in Figure 5. The mean absolute values in x- and y-directions are 0.05 m and hence lower than the required 0.06 m. The mean absolute value of the angle corresponds to $\frac{3.01^\circ}{180^\circ}\pi \approx 0.0525$ *rad* and thereby also lower than the required 0.06 *rad*. The evolution of the covariance matrix is shown in Figure 6.

Dataset 3:
The measurement noise covariance matrix is selected as

$$Q = \begin{pmatrix} 0.1^2 & 0 \\ 0 & 0.1^2 \end{pmatrix}, \tag{5}$$

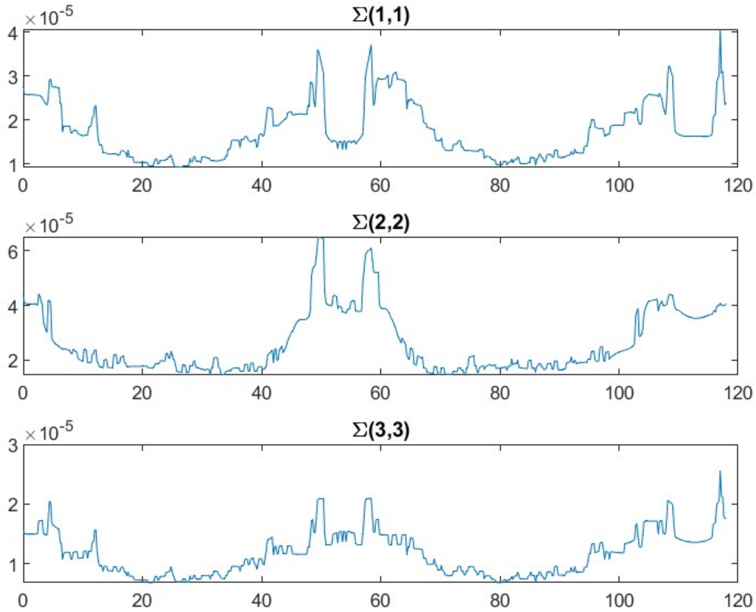

Figure 4: Evolution of the estimated covariance matrix for dataset 1.

and the process noise covariance matrix was chosen to

$$R = \begin{pmatrix} 1^2 & 0 & 0 \\ 0 & 1^2 & 0 \\ 0 & 0 & 1^2 \end{pmatrix} \tag{6}$$

.

As given, the outlier detection is disabled. The evolution of the error is shown in Figure 7. The left plot exhibits the sequential mode update approach, while the right plot the batch mode update approach. The sequential mode results in an absolute error above 0.1 (m, rad), since the sequential mode is more sensitive to outliers as discussed in Question 8. The right plot shows the evolution of the mean error , where the absolute error in all dimensions satisfy the requirement of being below 0.1 (m, rad) in all dimensions. In Figure 8 the evolution of the estimated covariance matrix. The estimated covariance matrix is larger in the case of the sequential mode, since the outliers propagate through the system.

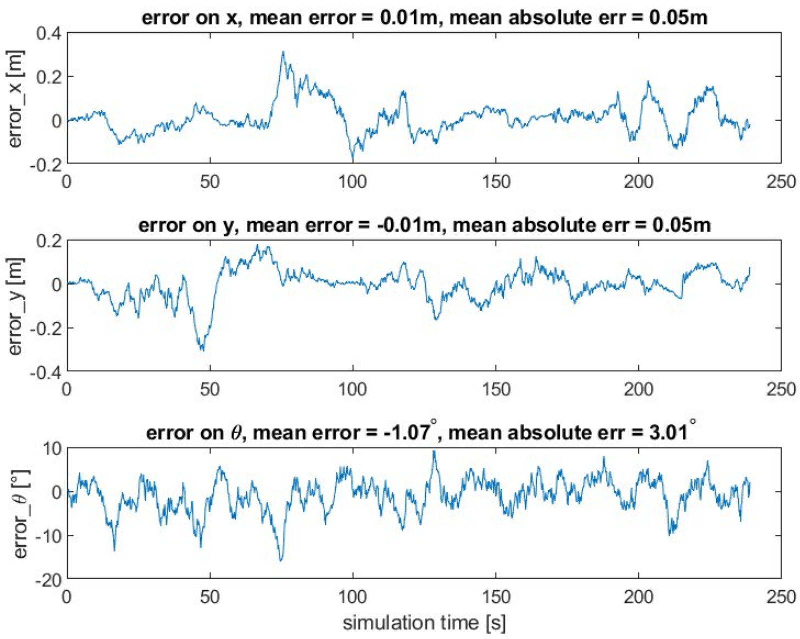

Figure 5: Evolution of the error for dataset 2.

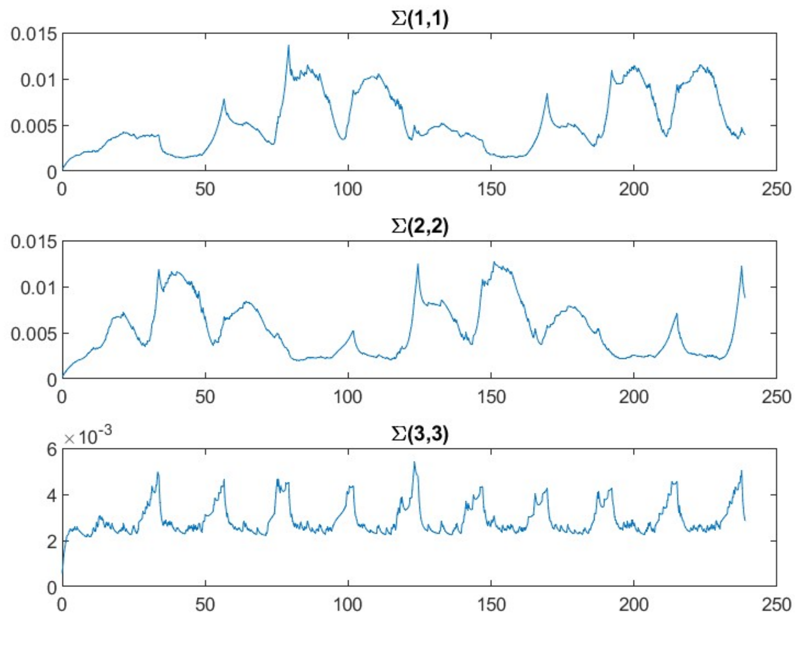

Figure 6: Evolution of the covariance matrix for dataset 2.

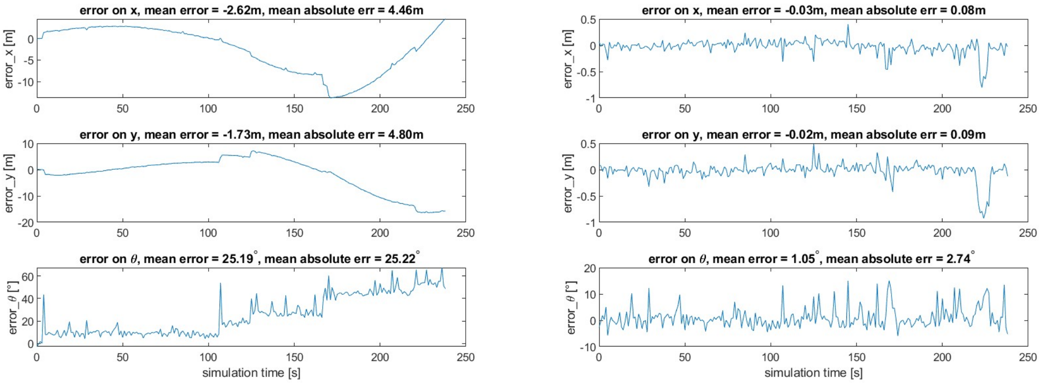

Figure 7: Evolution of the the error for dataset 3. (Left) The updates are done in sequential mode. (Right) The updates are done in batch mode.

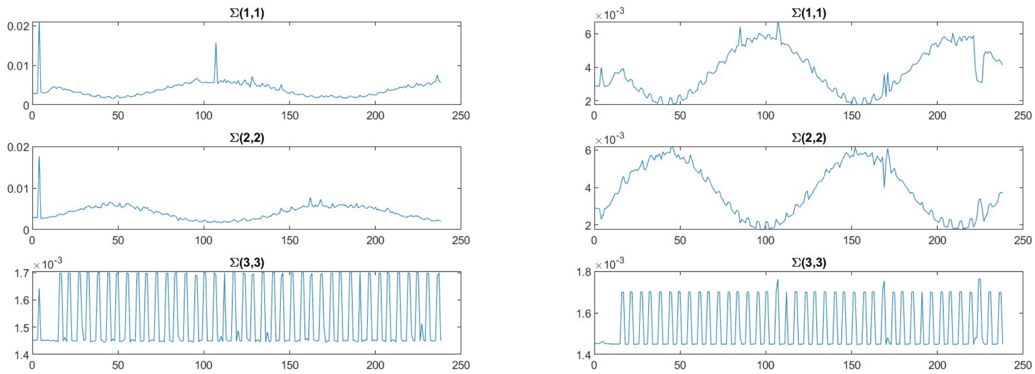

Figure 8: Evolution of the the estimated covariance matrix for dataset 3. (Left) The updates are done in sequential mode. (Right) The updates are done in batch mode.

[1] $X_0 \sim N(0,1)$ $\Sigma_{xy} = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$
$Y_0 \sim N(0,1)$

$$\begin{pmatrix} X_t \\ Y_t \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_{t-1} \\ Y_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{pmatrix} \qquad \begin{matrix} \varepsilon_{1t} \sim N(0,2) \\ \varepsilon_{2t} \sim N(0,3) \end{matrix}$$

$$\rightarrow R_t = \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$$

a) $X_1 \sim (\bar{\mu}_1, \bar{\Sigma}_1)$

$$\bar{\mu}_1 = A_1 \cdot \mu_0 + B \cdot u_1 = A \cdot \mu_0 = \begin{pmatrix} 0 & 1 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\bar{\Sigma}_1 = A_1 \Sigma_0 A_1^T + R_1$$
$$= \begin{pmatrix} 0 & 1 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 2 \\ 1 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$$
$$= \begin{pmatrix} 1 & 1 \\ 3 & 3 \end{pmatrix} \cdot \begin{pmatrix} 0 & 2 \\ 1 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 3 \\ 3 & 9 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 3 \end{pmatrix}$$
$$= \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix}$$

Posterior distribution is $X_1 \sim N(\bar{\mu}_1, \bar{\Sigma}_1)$ (Prediction step)

b) $z_1 = 1$ $\qquad z_t = \begin{pmatrix} 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} X_t \\ Y_t \end{pmatrix} + \delta_t$ $\qquad \delta_t \sim N(0,9)$

$Q = 9$

Update:

$X_1 \sim (\mu_1, \Sigma_1)$

$$\mu_1 = \bar{\mu}_1 + K_1 \cdot \eta_1 \rightarrow \quad K_1 = \bar{\Sigma}_1 C_1^T \cdot (C_1 \bar{\Sigma}_1 C_1^T + Q_1)^{-1}$$
$$= \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix} \cdot \left(\begin{pmatrix} 1 & 1 \end{pmatrix} \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 9 \right)^{-1}$$
$$= \begin{pmatrix} 6 \\ 15 \end{pmatrix} \cdot \left(\begin{pmatrix} 6 & 15 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 9 \right)^{-1}$$
$$= \begin{pmatrix} 6 \\ 15 \end{pmatrix} \cdot \frac{1}{30} = \begin{pmatrix} \frac{1}{5} \\ \frac{1}{2} \end{pmatrix} //$$

$$\mu_1 = \bar{\mu}_1 + \begin{pmatrix} 1/5 \\ 1/2 \end{pmatrix} \cdot (1 - C \cdot \bar{\mu}_1)$$
$$= \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 1/5 \\ 1/2 \end{pmatrix} \cdot (1 - 0) = \begin{pmatrix} 1/5 \\ 1/2 \end{pmatrix}$$

$$\Sigma_1 = \bar{\Sigma}_1 - K_t C_t \bar{\Sigma}_1 = \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix} - \begin{pmatrix} 1/5 \\ 1/2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix}$$

$$= \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix} - \begin{pmatrix} 1/5 \\ 1/2 \end{pmatrix} \cdot \begin{pmatrix} 6 & 15 \end{pmatrix}$$

$$= \begin{pmatrix} 3 & 3 \\ 3 & 12 \end{pmatrix} - \begin{pmatrix} \frac{6}{5} & 3 \\ 3 & \frac{15}{2} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{11}{5} & 0 \\ 0 & \frac{9}{2} \end{pmatrix}$$

Updated belief: $X_1 \sim (\mu_1, \Sigma_1)$

2) $\begin{pmatrix} X_0 \\ Y_0 \end{pmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right)$

$$\begin{pmatrix} X_t \\ Y_t \end{pmatrix} = \begin{pmatrix} Y_{t-1}^2 \\ (X_{t-1}+1) Y_{t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{pmatrix} \qquad \begin{matrix} \varepsilon_{1t} \sim N(0,2) \\ \varepsilon_{2t} \sim N(0,3) \end{matrix} \qquad R_t = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}$$

a) $G = \begin{bmatrix} 0 & 2Y \\ Y & X+1 \end{bmatrix}$ Jacobian of dynamic model

b) $X_1 \sim N(\bar{\mu}_1, \bar{\Sigma}_1)$

$$\bar{\mu}_1 = g(\bar{\mu}_{t-1}, 0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\bar{\Sigma}_1 = R_1 + G_t \Sigma_{t-1} G_t^T = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} //$$

$X_1 \sim (\bar{\mu}_1, \bar{\Sigma}_1)$ Posterior distribution

c) $z_1 = 0$ $\qquad z_t = (X_t+1)(Y_t+1) + \delta_t \qquad \delta_t \sim N(0, 10)$

$$H_t = \begin{bmatrix} Y_t+1 & X_t+1 \end{bmatrix} \qquad Q_1 = 10$$

$X_1 \sim N(\mu_1, \Sigma_1)$

$\mu_1 = \bar{\mu}_1 + K_1 \cdot \eta_1$

$$K_1 = \bar{\Sigma}_1 H_1^T \cdot (H_1 \bar{\Sigma}_1 H_1^T + Q_1)^{-1}$$

$$= \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} \cdot \left[\begin{bmatrix} 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + 10 \right]^{-1}$$

$$= \begin{bmatrix} 2 \\ 4 \end{bmatrix} \cdot \begin{bmatrix} 16 \end{bmatrix}^{-1} = \begin{bmatrix} 1/8 \\ 1/4 \end{bmatrix} //$$

$$\mu_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1/8 \\ 1/4 \end{bmatrix} \cdot (0 - 1) = \begin{bmatrix} -1/8 \\ -1/4 \end{bmatrix} //$$

$$\Sigma_1 = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} - \begin{bmatrix} 1/8 \\ 1/4 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix}$$

$$= \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} - \begin{bmatrix} 1/8 \\ 1/4 \end{bmatrix} \cdot \begin{bmatrix} 2 & 4 \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & 4 \end{bmatrix} - \begin{bmatrix} 1/4 & 1/2 \\ 1/2 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 7/4 & -1/2 \\ -1/2 & 3 \end{bmatrix} //$$

Posterior distribution: $X_1 \sim N(\mu_1, \Sigma_1)$