Delft University of Technology

Intelligent vehicles
ME41105

Assignment 2: Bayesian filtering and state estimation

Authors:
Michael Schoustra(4308611)

Kevin Meyer(4947843)

December 1, 2018

1 Kalman filters and multi-object tracking

Question 1.1. - LDS Model Formulation

$$\mathbf{x}_t = \begin{bmatrix} x_t \\ y_t \\ \dot{x}_t \\ \dot{y}_t \end{bmatrix} = \begin{bmatrix} x_{t-1} + \dot{x}_{t-1}\Delta t + \epsilon_t^x \\ y_{t-1} + \dot{y}_{t-1}\Delta t + \epsilon_t^y \\ \dot{x}_{t-1} + \epsilon_t^{\dot{x}} \\ \dot{y}_{t-1} + \epsilon_t^{\dot{y}} \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \dot{x}_{t-1} \\ \dot{y}_{t-1} \end{bmatrix}}_{\mathbf{x}_{t-1}} + \underbrace{\begin{bmatrix} \epsilon_t^x \\ \epsilon_t^y \\ \epsilon_t^{\dot{x}} \\ \epsilon_t^{\dot{y}} \end{bmatrix}}_{\boldsymbol{\epsilon}_t}, \tag{1a}$$

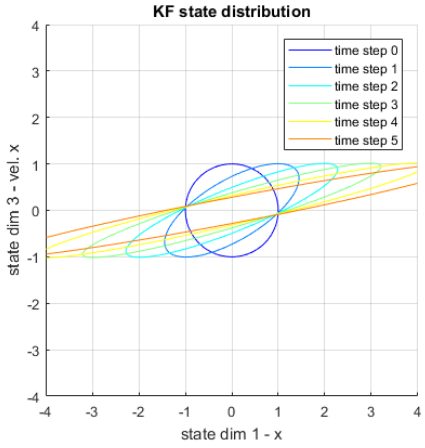
$$\mathbf{z}_t = \begin{bmatrix} z_t^x \\ z_t^y \end{bmatrix} = \begin{bmatrix} x_t + \eta_t^x \\ y_t + \eta_t^y \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{H} \mathbf{x}_t + \underbrace{\begin{bmatrix} \eta_t^x \\ \eta_t^y \end{bmatrix}}_{\boldsymbol{\eta}_t}, \tag{1b}$$

$$\boldsymbol{\epsilon}_t \sim \mathcal{N}\left(\mathbf{0}, \underbrace{\begin{bmatrix} \sigma_{pos}^2 & 0 & 0 & 0 \\ 0 & \sigma_{pos}^2 & 0 & 0 \\ 0 & 0 & \sigma_{vel}^2 & 0 \\ 0 & 0 & 0 & \sigma_{vel}^2 \end{bmatrix}}_{\boldsymbol{\Sigma}_\mathbf{x}}\right), \quad \boldsymbol{\eta}_t \sim \mathcal{N}\left(\mathbf{0}, \underbrace{\begin{bmatrix} \sigma_{obs}^2 & 0 \\ 0 & \sigma_{obs}^2 \end{bmatrix}}_{\boldsymbol{\Sigma}_\mathbf{z}}\right) \tag{1c}$$

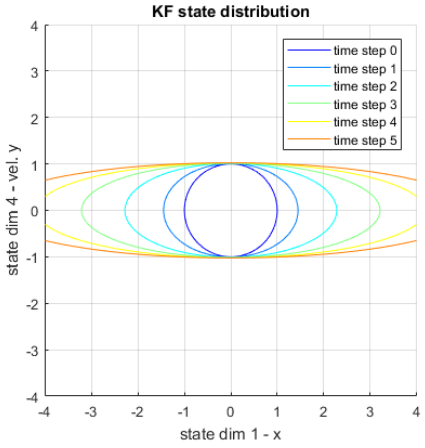
Question 1.2. - Plotting states and correlation

In figure 1a the x position is plotted on the first axis and the x velocity is plotted on the second axis. These two are seen to be positively correlated, as the Gaussian distribution is diagonally skewed positively. One can interpret this skew by for example considering a fixed positive x velocity, and observing that the distribution of x position tends to larger values as time progresses. Thus, a positive velocity leads to an increased position over time, as one would expect. Therefore the x position and x velocity are **positively correlated**.

Figure 1b plots the distributions of x position to y velocity on the first and second axes respectively. The distributions do not exhibit any diagonal skew, and using the same method of fixing a positive velocity, now in the y direction, one can see that this does not affect the x position positively or negatively. Therefore the x position and y velocity are **not correlated**.



(a) x position and x velocity

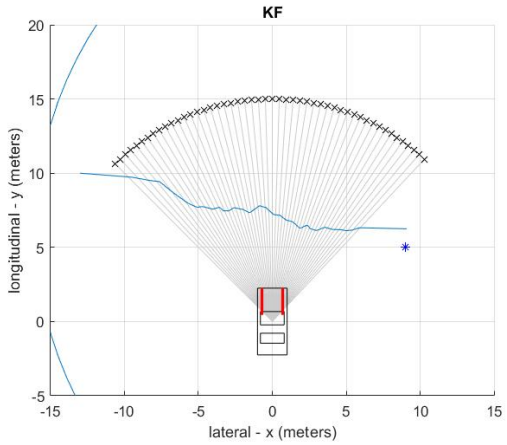


(b) x position and y velocity

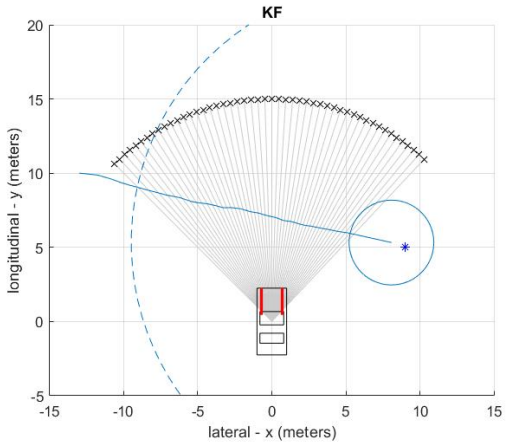
Figure 1: Plotting two states time sequences against each other

Question 1.3 - Track smoothness and noise variation

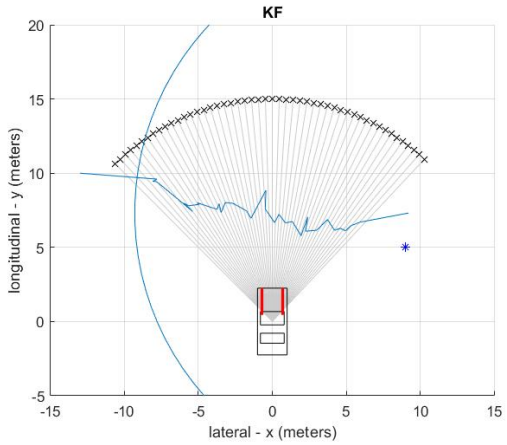
Four experiments concerning process and observation noise were run, where a high noise variance of 100 was compared with a low variance of 0.001. The resulting tracks are shown in figure 2.



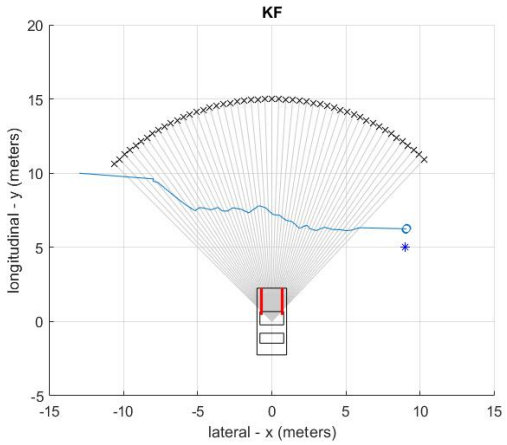
(a) High process and observation noise



(b) Low process and high observation noise



(c) High process and low observation noise



(d) Low process and low observation noise

Figure 2: Influence of process and observation noise covariance on the smoothness of trajectory

It was observed that a high observation noise coupled with low process noise resulted in a smoother track. For the pedestrian stopping scenario, it was observed that this noise combination would result in the predicted track overshooting the actual movement, as shown in figure 3. This can be attributed to a low confidence in the measurements, which show that the pedestrian has stopped, and a high confidence in the model, which would suggest continuous movement.

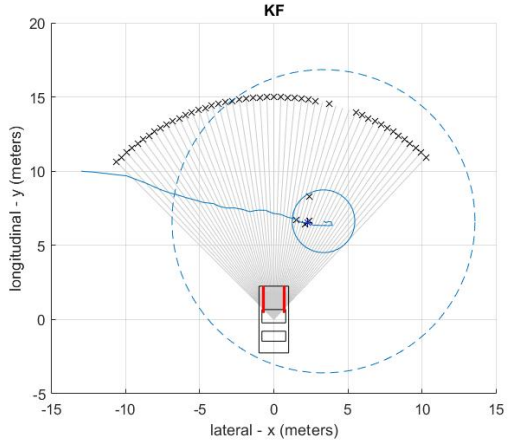
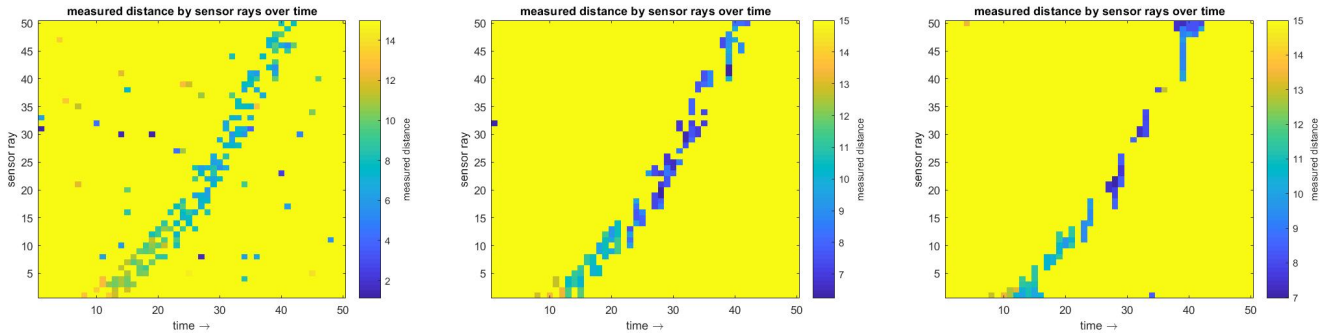


Figure 3: Pedestrian stopping while crossing

Question 1.4 - Median filter orders

The matrix visualizations are given in figure 4. It can be observed that the higher the order of the spatial filtering, the higher the amount of outliers which is filtered out. For the 7th order filter, this also results in some of the actual measurement data being lost. Therefore, a 3rd order is the best choice in this case.



(a) 1st order (b) 3rd order (c) 7th order

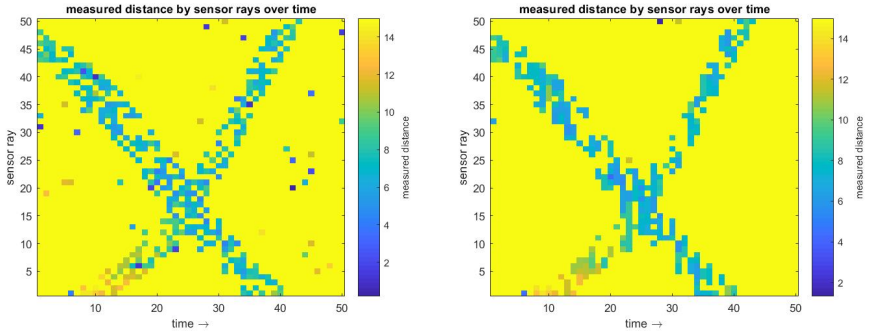
Figure 4: Influence of different order spatial filters on measurement matrix

Question 1.5 - Gating threshold selection

If the gating threshold is set too large, measurements found further away from the actual pedestrian will be counted as valid measurements thus it will constantly locate the pedestrian in different locations. Setting the gating threshold too small resulted in no measurements being assigned to the pedestrian. This could be due to the known initial position of the pedestrian being out of range of the area where measurements could be made, and therefore any measurement would be further away than the gating threshold would allow. When determining the ideal gating threshold, one would choose the smallest threshold that could still assign measurements to the pedestrian. Using this principle, a gating threshold of **3** was chosen.

Question 1.6 - Settings and tuning

The 3rd order median filter was used to preprocess the data. A 5th order filter was also experimented with, as one could argue that the benefits of more aggressive filtering to remove outliers would outweigh the loss of data. However, the results were not significantly different from the 3rd order filter, so the original 3rd order filter was kept. The visualization of the data is shown in figure 5.



(a) Original data (b) 3rd order spatial filter

Figure 5: Influence of of preprocessing on multi object tracker

For the single pedestrian scenario, gating threshold was used to filter outliers in the data and assign measurements to a known amount of pedestrians. As the filtering function has been replaced with the median filter in the multi-object scenario, the gating thresholds role is to assign measurements to the correct pedestrian. For this purpose, a higher gating threshold than the one originally proposed gave better results, as a low threshold resulted in new pedestrians being erroneously assigned to data just outside the gating threshold of an actual

predicted pedestrian. Choosing too high of a threshold would however result in all measurements being assigned to one pedestrian. To this end, a gating threshold of **7** was chosen.

With the settings as described above, it was observed that in most cases only 2 tracks were created. There were cases of pedestrians being assigned outside of these 2 tracks, however no other subsequent measurements would be assigned to them as to create a track. A possible solution to this could be to increase the amount of measurements needed before assigning measurements as a new pedestrian.

Question 1.7 - jk-srand and testing

While optimizing, the jk-srand was not changed since the tracker was optimized for a certain random seed. However, to verify the proposed optimal tracker, one should use a new random seed to evaluate the performance and test if the performance is consistent for different random seeds. Verification is a very important step any software related problems to avoid overfitting.

2 Vehicle self-localization

Question 2.1. - Calculating expected state

An expression for the expected state $\mathbf{x}_2$ at time 2 is calculated in equation 2. Of these, the resulting equations for x_2 and y_2 are nonlinear with respect to θ_0 and θ_1. The equation for z_2 is linear.

$$\mathbf{x}_1 = f(\mathbf{x}_0, \mathbf{u}_0, \Delta t) = \mathbf{x}_0 + \begin{bmatrix} v_0 \Delta t \sin\theta_0 \\ v_0 \Delta t \cos\theta_0 \\ \omega_0 \Delta t \end{bmatrix}, \tag{2a}$$

$$\mathbf{x}_2 = f(\mathbf{x}_1, \mathbf{u}_1, \Delta t) = \mathbf{x}_1 + \begin{bmatrix} v_1 \Delta t \sin\theta_1 \\ v_1 \Delta t \cos\theta_1 \\ \omega_1 \Delta t \end{bmatrix} \tag{2b}$$

$$= \mathbf{x}_0 + \begin{bmatrix} v_0 \Delta t \sin\theta_0 \\ v_0 \Delta t \cos\theta_0 \\ \omega_0 \Delta t \end{bmatrix} + \begin{bmatrix} v_1 \Delta t \sin\theta_1 \\ v_1 \Delta t \cos\theta_1 \\ \omega_1 \Delta t \end{bmatrix}, \tag{2c}$$

$$\Rightarrow x_2 = x_0 + \Delta t(v_0 \sin\theta_0 + v_1 \sin\theta_1), \tag{2d}$$

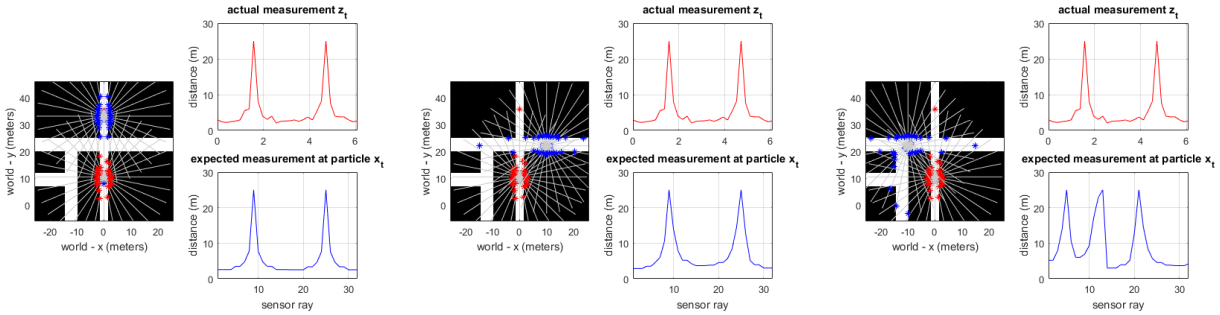
$$\Rightarrow y_2 = y_0 + \Delta t(v_0 \cos\theta_0 + v_1 \cos\theta_1), \tag{2e}$$

$$\Rightarrow z_2 = z_0 + \Delta t(\omega_0 + \omega_1). \tag{2f}$$

Question 2.2 - Particle patterns

The patterns of 3 of the 7 particles are shown in figure 6. One should note that the actual vehicle is located in the middle of a corridor, a situation that could occur multiple places within the map. Particle 4 exhibited the most similarity with the actual measurements, resulting in the maximum weight of ca. 0.91. The particle is located in the same corridor as the actual vehicle, so the detected particle corridor width matches that of the measurements. It is however not located correctly, ending up in a position exactly mirroring the actual position across the intersection. This would be impossible to tell from the measurements alone.

Particle 5 exhibited the median weight of the set, ca .0.16, and is also located in the middle of a corridor. It is however located in a corridor that is slightly wider than the actual measurements would suggest, and is therefore given a lower weight. Particle 7, which exhibited the lowest weight of ca. $1.19 * 10^{-14}$, is both located in the wrong corridor as with particle 5, and is turning into another corridor. Both of these lead to measurements that mismatch greatly with the actual measurements.



(a) Particle 4 (b) Particle 5 (c) Particle 7

Figure 6: Patterns of predicted sensor measurements and actual measurements

Question 2.3 - Proof of log measurements likelihood equation

Using equations (33) and (35), denoted in the appendix as equations 8 and 9 respectively, a proof is shown in equation 3.

$$P(\mathbf{z}|\mathbf{x}^{(i)}) \overset{(8)}{=} \prod_{r=1}^{R} \mathcal{N}(z_r - \hat{z}_r^{(i)}|0, \sigma^2) \tag{3a}$$

$$\overset{(9)}{=} \prod_{r=1}^{R} \frac{1}{(2\pi)^{1/2}|\sigma^2|^{1/2}} \exp\left[-\frac{1}{2}(z_r - \hat{z}_r^{(i)} - 0)^\top (\sigma^2)^{-1}(z_r - \hat{z}_r^{(i)} - 0)\right] \tag{3b}$$

$$= \prod_{r=1}^{R} \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{1}{2\sigma^2}(z_r - \hat{z}_r^{(i)})^2\right] = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^R \exp\left[-\frac{1}{2\sigma^2}\sum_{r=1}^{R}(z_r - \hat{z}_r^{(i)})^2\right] \tag{3c}$$

$$\Rightarrow \log P(\mathbf{z}|\mathbf{x}^{(i)}) = R\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) - \frac{1}{2\sigma^2}\sum_{r=1}^{R}(z_r - \hat{z}_r^{(i)})^2, \quad R\log\left(\frac{1}{\sqrt{2\pi}\sigma}\right) = \alpha, \tag{3d}$$

$$\Rightarrow \log P(\mathbf{z}|\mathbf{x}^{(i)}) = \alpha + \sum_{r=1}^{R} -\frac{(z_r - \hat{z}_r^{(i)})^2}{2\sigma^2} \tag{3e}$$

This shows that the constant α only depends on the number of measurements R and the standard deviation in the measurements σ, which is the same for all particles.

Question 2.4 - Particle amount selection with known initial state

Increasing the amount of particles will result in a more accurate tracking of the vehicle, as the particle distribution will more adequately cover the actual state of the vehicle. However, it will also results in a higher computational load. Fewer particles take significantly less computation time, but result in a lower chance of tracking the vehicle accurately. The most optimal value found for this scenario was 20, with the experiments being repeated several times to check if this amount of particles is sufficient to give an accurate tracker. This conclusion was drawn by equating reliable tracking to the tracker constantly being able to locate the defined vehicle and follow it through the sequence.

Question 2.5 - Particle amount selection with unknown initial state

In the current situation, should all particles lose track of the vehicles position, it would be almost impossible for the vehicle position to recovered by the end of the sequence. Therefore, a large amounts of particles should be used so that from the start of the sequence some of the particles are in the correct position. Experiments were conducted with $N = [50, 100, 250, 500, 1000]$, each repeated 10 times. The results are given in the table below:

	50	100	250	500	1000
# of reliable trackings	1	2	3	3	6
# vehicle postion recovered	1	2	3	3	6

One could argue that these results show that a higher amount of particles correlates with a higher likelihood that the vehicle is tracked accurately through the simulation. However, it was observed during these experiments that an accurate tracking relied heavily on one of the initial particles coinciding with the actual initial state of the vehicle. It was found that even low values, such as $N = 20$, could result in as accurate of a tracking as the higher values, given that an initial particle coincided with the actual initial state.

Therefore, an increased number of particles seems to correlate with a decreased risk of not recovering the vehicles position in the situation with unknown initial state. However, this conclusion is drawn noting that the decreased risk is more likely due to the increased chance of discovering the initial state, than due to more accurate tracking after having found the initial state.

Question 2.6 - Particle amount selection with unknown initial state and re-initialization

Re-initializing a low fraction of the particles was observed to cause most of the particles to converge and stay in one cluster throughout the simulation. This would be beneficial should the vehicles position be known or discovered initially, as the filter would be better at keeping a cluster located on the vehicles position. The downside to this would appear for the cases where the correct initial position is not discovered, and the filter converges to an erroneous cluster.

This downside can be addressed by increasing the fraction of reinitialized particles. Should the initial cluster be wrong, the filter would then have a higher chance of re-initializing a particle to the correct position and forming a cluster around that particle. As one would expect, this also has the downside of increasing the chance of losing the vehicles position should too great a fraction of the particles be re-initialized to other locations.

With a low number of total particles, such as $N = 20$, one would rely on the fact that the initial state is known, or that the filter manages to discover the initial state by chance. Re-initializing particles in this situation would be to the detriment of the filters ability to accurately track the vehicles movements after converging on its state correctly.

With a higher number of total particles, such as $N = 100$, the filter has a better ability to accommodate a higher re-initialization fraction. Since only a small fraction of the particles in this situation, for example $N = 20$ to 40, are needed to accurately track a vehicle state, the remaining could be used to explore other states should the tracked cluster be incorrect. However, it was observed that a very high fraction, for example 90%, resulted in the tracking losing accuracy, as reinitialized particles appearing close to the tracked cluster would distort the predicted vehicle position distribution.

3 Mathematical proofs with Linear Algebra

3.1 Question 3.1. - Covariance update

$$\mathbf{S}_t = H\hat{\mathbf{\Sigma}}_t H^\top + \Sigma_z, \tag{4a}$$

$$K_t = \hat{\mathbf{\Sigma}}_t H^\top \boldsymbol{S}_t^{-1} = \hat{\mathbf{\Sigma}}_t H^\top (H\hat{\mathbf{\Sigma}}_t H^\top + \Sigma_z)^{-1} \tag{4b}$$

$$\overset{(7)}{=} (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} H^\top \Sigma_z^{-1}, \tag{4c}$$

$$\Rightarrow \Sigma_t = (I - KH)\hat{\mathbf{\Sigma}}_t = (I - (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} H^\top \Sigma_z^{-1} H)\hat{\mathbf{\Sigma}}_t \tag{4d}$$

$$= ((\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H - H^\top \Sigma_z^{-1} H))\hat{\mathbf{\Sigma}}_t \tag{4e}$$

$$= (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} \tag{4f}$$

3.2 Question 3.2. - Mean update A

$$(I - KH)\hat{\boldsymbol{\mu}}_t \overset{(4c)}{=} (I - (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} H^\top \Sigma_z^{-1} H)\hat{\boldsymbol{\mu}}_t \tag{5a}$$

$$= ((\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H - H^\top \Sigma_z^{-1} H))\hat{\boldsymbol{\mu}}_t \tag{5b}$$

$$= (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} \hat{\mathbf{\Sigma}}_t^{-1} \hat{\boldsymbol{\mu}}_t \tag{5c}$$

3.3 Question 3.2. - Mean update B

$$K\mathbf{z} \overset{(4c)}{=} (\hat{\mathbf{\Sigma}}_t^{-1} + H^\top \Sigma_z^{-1} H)^{-1} H^\top \Sigma_z^{-1} \mathbf{z} \tag{6a}$$

A Assignment formulas

Formula (11) from Section 3:

$$BX^\top (A + XBX^\top)^{-1} = (B^{-1} + X^\top A^{-1} X)^{-1} X^\top A^{-1}. \tag{7}$$

Formula (33) from Appendix C:

$$P(\mathbf{z}|\mathbf{x}^{(i)}) = \prod_{r=1}^{R} \mathcal{N}(z_r - \hat{z}_r^{(i)}|0, \sigma^2). \tag{8}$$

Formula (35) from Appendix E:

$$\mathcal{N}(\mathbf{x}|\mu, \Sigma) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \mu)^\top \Sigma^{-1}(\mathbf{x} - \mu)\right]. \tag{9}$$