

# EXAMINATION IN TSRT14 SENSOR FUSION

ROOM: Asgård, Egypten, Hunn, Olympen, Roxen

TIME: 2023-06-01 at 8:00–12:00

EDUCODE: TSRT14 Sensor Fusion

MODULE: DAT1

DEPARTMENT: ISY

NUMBER OF EXERCISES: 4

RESPONSIBLE TEACHER:

Gustaf Hendeby, tel. 013-28 58 15, [gustaf.hendeby@liu.se](mailto:gustaf.hendeby@liu.se)

VISITS: cirka 09:00, 10:00, 11:00

COURSE ADMINISTRATOR:

Ninna Stensgård, 013-28 22 25, [ninna.stensgard@liu.se](mailto:ninna.stensgard@liu.se)

APPROVED TOOLS: 1. Book: *F. Gustafsson*, “Statistical Sensor Fusion”, any edition.

PROVIDED MATERIAL:

1. Lecture slides; available from `/courses/TSRT14/`
2. Signal and Systems toolbox manual; available from `/courses/TSRT14/`
3. Current up to date errata for the textbook; available from `/courses/TSRT14/`

MATLAB FILES: The files that are needed for the exam are available at `/courses/TSRT14/`.

SOLUTIONS: Available at the course homepage after the exam.

The exam can be inspected and checked out 2023-06-22 at 12.30–13.00 in Gustaf Hendeby’s office, room 2A:503, B-house, entrance 27, A corridor to the right.

PRELIMINARY GRADE LIMITS:

grade 3	15 points
grade 4	23 points
grade 5	30 points

NB! Solutions should include code and plots and clear cross references between these. Mark all print-outs with your AID-number, date, course code, and exam code.

Good luck!

### STARTING MATLAB (Linux)

Type `matlab &` in a terminal.

### PRINTING (Linux):

Printouts of regular files can be sent to a specific printer using the command

```
lp -d printername file.pdf
```

in a terminal. (Exchange `printername` for the actual printer name.) When selecting `File/Print` for a Simulink diagram, select the target printer by adding

```
-Pprintername
```

in the `Device option` box.

### ADDING YOUR AID ETC TO PRINTOUTS:

Text can be added in Matlab plots with the commands `title` and `gtext`, and for scope plots in Simulink by right clicking and then change the `Axes properties`. In Simulink diagrams it is possible to double click any empty area and then simply add text by typing it.

### FURTHER GUIDELINES:

- Make sure to read all exercises and prioritize before getting started. The level of difficulty is not necessarily increasing.
- Make sure to motivate every step of your solution carefully!
- Comment nontrivial steps in the code; including model choices and tuning.
- Put code for each exercise on a separate printout and keep all related paper (hand written material, code, and plots) together when you hand in your solution.

1. The following questions all require relatively short answers, a few sentences or short calculations should be enough. (Note, an incorrect statement will result in 0 p on that subexercise.)
  - (a) Based on your experiences from lab 2 “Orientation estimation using smartphone sensors”, shortly describe how you calibrated the accelerometer, gyroscope, and magnetometer. (2p)
  - (b) Figure 1 contains the cost functions for position estimation using four different sensor setups. Pair the following sensor networks with the matching representative cost function:
    - (i) Two *time difference of arrival* (TDOA) sensors
    - (ii) Two *time of arrival* (TOA) sensors
    - (iii) A range-bearing radar
    - (iv) A *direction of arrival* (DOA) sensor

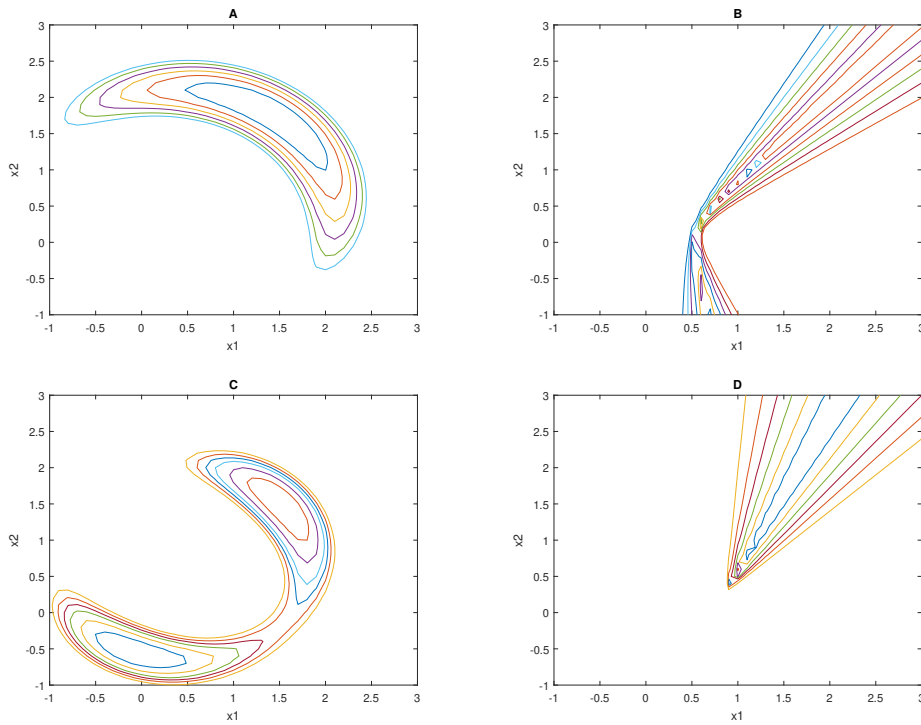


Figure 1: Cost functions for position estimation for the different sensor setups in Exercise 1b.

(2p)

(c) Assume two estimates of a parameter  $x^0$ ,

$$\begin{aligned}x_1 &= x^0 + e_0 + e_1 \\x_2 &= x^0 + e_0 + e_2,\end{aligned}$$

where  $x_1 \sim \mathcal{N}(5, 6)$ ,  $x_2 \sim \mathcal{N}(4, 7)$ , and  $e_0 \sim \mathcal{N}(0, 4)$ . Furthermore,  $e_0$ ,  $e_1$ , and  $e_2$  are all mutually independent. Given this information, provide an optimal estimate  $x = \mathcal{N}(\hat{x}, P)$  of  $x^0$ . (2p)

(d) Which of the following statements are correct?

- A. Properly tuned, the *unscented Kalman filter* (UKF) is always a better alternative than the *extended Kalman filter* (EKF).
- B. The Kalman filter is the *best linear unbiased estimator* (BLUE) for linear Gaussian problems.
- C. The Kalman filter reaches the *Cramér-Rao lower bound* (CRLB) for all linear systems.
- D. The *weighted least squares* (WLS) is an optimal estimator for linear Gaussian batch problems.
- E. The *maximum likelihood estimator* (MLE) is equivalent to the WLS, it has only been derived in a different way.

(2p)

(e) You are trying to use a SIR *particle filter* (PF) to estimate the state of the following nonlinear system

$$\begin{aligned}x_{k+1} &= \begin{pmatrix} x_{k+1}^p \\ x_{k+1}^v \end{pmatrix} = \begin{pmatrix} I & T \\ 0 & I \end{pmatrix} x_k + \begin{pmatrix} T^2 \\ T \end{pmatrix} w_k \\ y_k &= \begin{pmatrix} \|x_k^p\| \\ (x_k^p \cdot x_k^v) / \|x_k^p\| \end{pmatrix} + e_k,\end{aligned}$$

where neither the process noise nor the measurement noise is extreme. You are having problems with particle depletion, suggest how to improve the filter performance. (2p)

2. Assume an area with three distinct landmarks with the positions  $m^1$ ,  $m^2$ , and  $m^3$ . A sensor mounted on an RC car is moving around in the area, observing the relative position of the landmarks in the global Cartesian coordinate system. At each time, and for each landmark, the following measurement is obtained

$$y_k^i = m^i - x_k + e_k^i,$$

where  $x_k$  is the position of the sensor, and the measurement noise  $e_k \sim \mathcal{N}(0, R)$ . The covariance matrix  $R = 10^2 I$ .

The file `data20230601.mat` contains the variables `ex2_m`, with approximate positions of the landmarks (one landmark per column), and `ex2_y`, with measurements from the sensor collected at 1 Hz. The measurements of the landmarks are stacked, *i.e.*, the first two values in each column is  $y^1$  according to the notation above, the next two  $y^2$ , and so on. (That is, the landmark association is assumed known.)

- (a) Assume that the given landmark locations are exact, and estimate the trajectory of the sensor. Make sure to carefully motivate what you are doing. Plot the estimated trajectory, with 90% covariance ellipses, together with the landmark positions in a plot. (4p)
- (b) Now assume that the landmark positions are only estimates,  $m^i \sim \mathcal{N}(m^i, R^m)$ , where  $R^m = 50^2 I$ . Repeat the task in (a), with this added uncertainty of the landmarks. Illustrate the estimated trajectory, as well as location and uncertainty of the landmark positions at the end (both with 90% covariance ellipses). (4p)
- (c) Rerun your solution in (a) with the landmark positions from (b). Compare the resulting trajectory in a plot. Reflect on the result. (2p)

3. Assume  $N$  linear measurements

$$y_k = H_k x + e_k, \quad e_k \sim \mathcal{L}(\lambda),$$

where  $\mathcal{L}(\lambda)$  is the Laplace distribution, which is characterized by the pdf  $p(e|\lambda) = \frac{1}{2}\lambda e^{-\lambda|e|}$ . Furthermore, you may assume that  $x$  is observable given the available measurements and the measurement noise is independent.

- (a) Derive the *best linear unbiased estimator* (BLUE),  $x^{\text{BLUE}}$  and the corresponding covariance of the estimate, for the problem above. (3p)
- (b) Derive the *maximum likelihood estimate* (MLE) formulation of the problem above, and show that it is equivalent to minimizing the sum of the absolute value of the difference between the predicted and obtained measurement. (3p)
- (c) Compute the *Cramér-Rao lower bound* (CRLB) for the above estimation problem. Use that it can be shown that the Fisher information exists even if the derivative of  $\log(p(y|x))$  is discontinuous, as long as the computed Fisher information is positive and limited. (You do not have to show this.) (4p)

4. The examiner sometimes uses the Sensor Fusion app to track his movements. In this case as he has been walking some stairs. Your task is to track the height profile of his walk!

During the walk, the examiner held his phone horizontally in his hand, with the  $z$ -axis of the accelerometer pointing up and aligned with gravity. The following data is available in the file `data20230601.mat`:

**ex4\_acc** The accelerometer measurements from the walk;  $x$ -,  $y$ -, and  $z$ -component in the three columns, given in  $\text{m/s}^2$ .

**ex4\_prs** The barometric pressure measurements from the walk given in hPa; NaN represents no measurement available. The frequency is lower than for the accelerometer.

**ex4\_t** The time stamps for the measurements from the walk in seconds.

**ex4\_c0cm\_acc**, **ex4\_c0cm\_prs** Acceleration and pressure measurements, respectively, from a calibration experiment where the phone is placed flat on the floor, at the lowest point. Assume this as height 0 m.

**ex4\_c78cm\_acc**, **ex4\_c78cm\_prs** Another calibration experiment, this time at height 78 cm.

For the purpose of this experiment, the relation between the height and the air pressure can be approximated with

$$y_k = C_1 h_k + C_0,$$

where  $y_k$  is the measured pressure,  $h_k$  the current height, and  $C_1$  and  $C_0$  constants.

To obtain full credits, the exercise should be solved with techniques inspired by the methods discussed in the course, and all steps be well motivated.

- (a) Use the calibration datasets to determine the constants in the given height–pressure relation! Use the model to compute the height directly from the pressure measurements, and plot the result!

**Hint:** *As there are NaNs in the measurements, use a cross to mark the computed heights in the plot.* (2p)

- (b) Suggest a motion model that can be used to estimate the height of the phone and the accelerometer bias. Treat the  $z$ -accelerations as input to the model! Simulate the model and plot the result in a separate figure! Estimate the bias from the calibration datasets. (2p)

- (c) Use a filter to combine the pressure measurements with the motion model! Plot the result in the plot in (a)!

**Hint:** *The problem formulation does not fit easily in the toolbox, it is most likely easier to implement the filter yourself.*

*Use  $\sim \text{isnan}(\mathbf{y}(k))$  to test if there is a measurement available for time  $k$ .* (6p)

**Hint:** *Make sure to document how you think. Partial credits will be awarded for the correct idea even if your implementation does not work correctly.*