MGE-MSR-01 – Sensors and State Estimation
WS 2020/2021

Exercise 2: Kalman filtering
M.Sc. Tomislav Medic

1 Learning objective

Theory of discrete and extended Kalman filtering; setup of system model and measurement model; application to real data for a simplified 2D example; filter tuning; comparison of the filter results with the results from Strapdown algorithm and GNSS

2 Description of the exercise

This exercise is about the determination of the trajectory of a moving platform by using a Kalman filter. For this purpose a kinematic multi sensor system (MSS) is used, which is equipped with three fiber-optic gyroscopes and three servo accelerometers. Additionally, the MSS contains an accurate RTK-GNSS. The system is adapted to a trolley. However, it is also possible to adapt the system to a car or another moving vehicle. During the exercise real measurements are recorded at the Campus Poppelsdorf with the system. The goal is to determine the trajectory of the trolley via Kalman filtering in 2D by using a simplified motion model (constant accelerations, constant angular rates, motion only possible along the x-axis of the body frame) as well as the observations of the IMU (i. e. accelerations and angular rates) and the GNSS receiver (2D positions). The filtered trajectory should then be compared with the results of the Strapdown algorithm (see Exercise 1) and the RTK-GNSS positions of the MSS.

- **Execution:** group work with 2-4 students
- **Certification:** oral presentation (with slides)
- **Deadline:** submission of the presentation via eCampus until Tuesday, 12/01/2020
- **Examination:** Wednesday, 13/01/2020, 08:30, via Zoom

3 Prior knowledge

- MGE-MSR-01 – Sensors and State Estimation (Lectures: Sensors, Inertial Navigation and Kalman Filtering)

4 References

- Groves, P. D. (2013). *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*, 2nd edition. Artech House, Boston, London, ISBN: 978-1-60807-005-3.
- Groves, P. D. (2015). Navigation Using Inertial Sensors. University College London, United Kingdom, IEEE A&E Systems Magazine, February 2015, Part II of II.
- Titterton, D. H., Weston, J. L. (2004): *Strapdown Inertial Navigation Technology*, 2nd edition. The Institute of Electrical Engineers, ISBN: 978-0-86341-358-2.
- Teunissen, P. J. G. & Montenbruck, O. (Eds.) (2017). *Springer Handbook of Global Navigation Satellite Systems*. Springer Verlag, Berlin, ISBN: 978-3-319-42926-7.

5 Description of the tasks

1. **Explain the idea and principles of Kalman filtering** in your own words. For this purpose, create a flow diagram that allows you to represent the Kalman filter as a recursive algorithm with all functional and stochastic relationships and formulas. Please address the following questions:

   a. What are the requirements of the basic Kalman filter regarding system model, measurement model and noise? What requirements can be neglected in an extended Kalman filter?

   b. Which functional and stochastic variables have to be predefined before starting the algorithm and which variables are created and updated in the algorithm?

   c. What are the two major steps of a basic Kalman filter? Name all important variables and their meaning. What influence do the predefined stochastic settings, i. e. system noise and measurement noise, have on the results?

   d. Why is the Kalman filter a real time algorithm?

Exercise 2: Kalman filtering
M.Sc. Tomislav Medic

2. **Explain choice of the system and measurement model**:
    a. Explain the simplified 2D system model for a moving platform as shown in Example 2 in the lectures. How can this model be integrated into the prediction step of the Kalman filter?
    b. What are the consequences of this simplified model that does not coincide with the reality?
    c. Explain how the measurements of the MSS (2D positions, accelerations and angular rates) can be integrated into the update step of the Kalman filter.

General task: Determine the trajectory of the trolley at the Campus Poppelsdorf on the basis of the self-implemented 2D extended Kalman filter (EKF) algorithm according to Example 2 from the lectures (lecture Kalman Filter 3). Use the simplified 2D system model for the prediction of the state vector. Subsequently, update the state vector by using the observations of the MSS, i. e. *2D positions*, accelerations and angular rates. For the trajectory computation use the same IMAR.mat data file provided for Exercise 1.

3. **Analyzing filter tuning:** Perform a filter tuning, i. e. setting of a proper system noise and measurement noise parameters in the corresponding covariance matrices. Additionally vary the uncertainty of the initial values. Analyze the differences in 2D trajectory when you over-/under- estimate system noise and when you over-/under- estimate measurement noise parameters. Compare the estimated trajectories with the RTK-GPS position provided by MSS. Present the final result of your filter tuning (the 2D trajectory you subjectively consider to be the best one vs. RTK-GPS). Present the noise parameter values you selected for this trajectory estimation. Explain why you selected these parameter values and why you believe this trajectory estimation is the best one. Additionally, visualize a few other trajectory estimations/realizations (at least 3 additional) with different noise parameters you consider to be inferior to the best solution (indicate the selected parameter values). Explain why you believe these solutions are suboptimal. (End of the task 3).
4. **Analyzing 2D trajectory (2D position), attitude (yaw/heading) and velocity:** Compare the results of your own 2D EKF implementation with **A)** the results of the Strapdown algorithm from Exercise 1 (you will need to transform the trajectory to UTM coordinates) and **B)** with the RTK-GPS position provided by MSS. All trajectories must be compared in 1 figure/plot. What is the maximal difference between trajectories and where does it occur? Calculate the differences between both estimated trajectories (Strapdown and EKF) and the RTK-GPS positions in different time points (2D distances in meters). Visualize differences against time and analyze them. Calculate the average difference per time (e.g. cm/s or m/min). Analyze the yaw or heading direction. Compare the results of the Strapdown and EKF. Visualize differences against time and analyze them. Do you observe notable differences and where? Analyze the velocity over time for your own EKF and Strapdown implementations. Visualize the differences against time and analyze them. Based on all these analyses answer: How does the trajectory estimation benefit from the fusion of relative and absolute observations? (End of the task 4).

Remarks

- It is strongly recommended to utilize Python or Matlab/Octave for the implementations and calculations (only in this case we can offer help)
- The Kalman filter algorithm must be done by using own source code. The use of existing source code and algorithms is not permitted
- You are encouraged to use and modify visualization parts of the provided “shell” python scripts for Exercise 1 and 2
- Design your filter in such a way that it can handle missing GNSS positions
- All plots/figures needs to have readable axis labels and units in presentations and you must be able to discuss the results you are presenting