2. laboratory exercise

Vehicle location estimation using the global positioning system

Name and surname: Luka Vukelic **JMBAG:** 0036530728

Uvodne napomene

- **Exercise goal**
  Kalman filter implementation for estimating the state of a linear dynamic stochastic system.
- **Preparation**
  This exercise is done in Matlab. We apply the linear Kalman filter for estimating 2D vehicle location via global positioning system measurements. With the exercise you get m-functions including the system model which you need to fill in with the requested implementation of the Kalman filter. Study the lecture sections related to linear Kalman filter properties and implementation.
- **Figures and Equations**
  Record the figures as images and attach them in the report at the designated boxes. Write the equations by using the tool of your choice and attach them to the report also as images at the designated boxes. (For Adobe Reader: Tools->Comment & Markup->Attach a File as a Comment).
- **Useful Matlab functions:**
  `help, diag, trace, cond, chol, std, mean`

Rad na vježbi

Zadatak 1: Vehicle location estimation using the global positioning system

Knowing the exact vehicle location is one of the major prerequisites for operation of autonomous vehicles. One of the sensors that can be used to solve this issue is the global positioning system (GPS), which can be used to calculate the location of the vehicle anywhere on Earth. GPS measurements are usually received in the ecliptic coordinate system consisting of latitude and longitude; however, for this laboratory exercise these measurements were transformed into the Euclidean coordinate system using the Mercator projection. Even though for autonomous driving we also need to know the vehicle orientation, in this laboratory exercise we will focus on estimating just the location, i.e., (x, y) vehicle coordinates. Measurements from commercial GPS sensors are often noisy, and sometimes measurements can disappear (in a tunnel) or yield outlier measurements (*street canyon*). Given that, we need to use stochastic estimation methods in order to determine the vehicle location in challenging conditions.

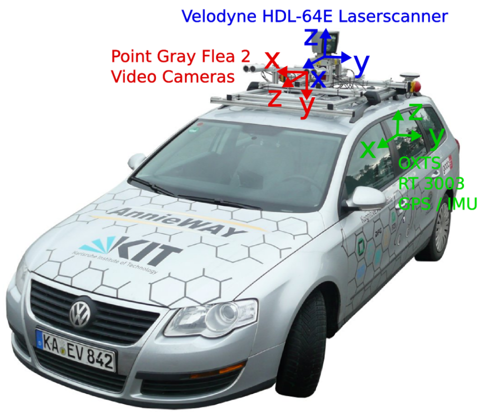
In the laboratory exercise we will use measurements obtained by the vehicle shown in Fig. 1 that was used to record the KITTI dataset developed for benchmarking autonomous vehicle localization algorithms. Since the vehicle was also equipped with a differential GPS setup, which enables high accuracy measurements, we will use these data to evaluate the estimation accuracy (the `x_gt` variable).

Vehicle state $\boldsymbol{x}_k$ at a discrete time instant k is given by the following vector

$$\boldsymbol{x}_k = [x_k \; y_k \; \dot{x}_k \; \dot{y}_k]^{\mathrm{T}}, \tag{1}$$

where x_k, y_k is vehicle location at time k, while $\dot{x}_k, \dot{y}_k$ are respective velocities.

Discrete system model, i.e., vehicle motion model, is described by the following stochastic constant velocity



Slika 1: Vehicle used for recording the KITTI dataset developed to benchmark autonomous vehicle localization algorithms

model:

$$\begin{aligned} x_k &= x_{k-1} + T\dot{x}_{k-1} + w_{x,k-1} \\ y_k &= y_{k-1} + T\dot{y}_{k-1} + w_{y,k-1} \\ \dot{x}_k &= \dot{x}_{k-1} + w_{\dot{x},k-1} \\ \dot{y}_k &= \dot{y}_{k-1} + w_{\dot{y},k-1}, \end{aligned} \tag{2}$$

where w_{k-1} represents process noise, while T is sensor sampling rate of 10 Hz. Process noise is modeled as a discrete Gauss white noise with zero mean value and Q covariance matrix.

Discrete measurement model $\boldsymbol{y}_k$ is described by the following linear stochastic equation:

$$\boldsymbol{y}_k = \begin{bmatrix} x_k + v_{x,k} \\ y_k + v_{y,k} \end{bmatrix},$$

where v_k represents measurement noise. Measurement noise is modeled as a discrete Ga with zero mean value and R covariance matrix.

a) Write the discrete state space model using state variables $\boldsymbol{x}_1 = x_k$, $\boldsymbol{x}_2 = y_k$, $\boldsymbol{x}_3 = \dot{x}_k$, matrices A, B, L, H, M.

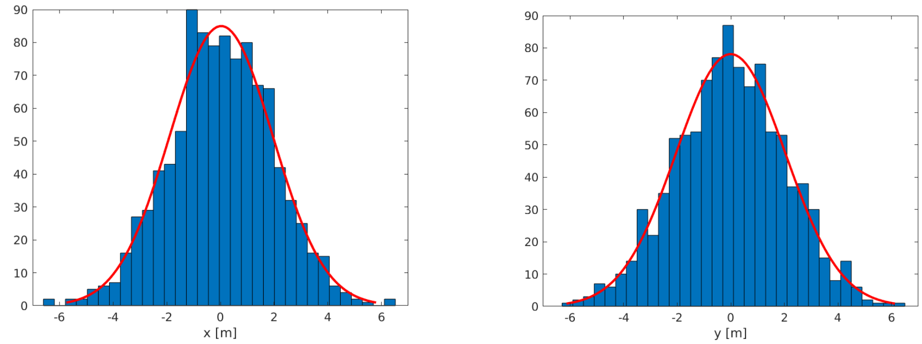
$$\boxed{\begin{aligned} \boldsymbol{x}_k &= \boldsymbol{A}\boldsymbol{x}_{k-1} + \boldsymbol{B}\cdot\boldsymbol{u} + \boldsymbol{L}\boldsymbol{w}_{k-1} \quad \boldsymbol{w}_k \approx N(0,\ Q) \\ \boldsymbol{y}_k &= \boldsymbol{H}\boldsymbol{x}_k + \boldsymbol{M}\cdot\boldsymbol{v}_k \qquad \boldsymbol{v}_k \approx N(0,\ R) \end{aligned}}$$

```
A:
    1.0000         0    0.1000         0
         0    1.0000         0    0.1000
         0         0    1.0000         0
         0         0         0    1.0000

B:
     0
     0
     0
     0

L:
     1     0     0     0
     0     1     0     0
     0     0     1     0
     0     0     0     1

H:
     1     0     0     0
     0     1     0     0

M:
     1     0
     0     1
```

b) Parametrize the process and measurement noise of the discrete linear stochastic system.

Process noise covariance matrix can be freely set as a diagonal matrix with estimated values such that they model the expected deviation of the predicted vehicle location with respect to the real vehicle location. You can assume that average velocity is 50 km/h. Elaborate your choice!

Measurement noise covariance matrix R can be estimated from the following experiment, where stationary vehicle at location $(0, 0)$ recorded GPS measurements. Histogram of measurement errors and pertaining estimated Gaussian distribution for each location coordinate are shown in Fig. 2. Assuming that errors in x and y are independent, determine the measurement noise covariance matrix R by visual inspection of the depicted histograms. Elaborate your choice!



(a) Histogram of the x coordinate measurements (b) Histogram of the y coordinate measurements

Slika 2: GPS measurement errors

For the process noise covariance, I used a value that represents 10% of the average speed. I concluded that this value would be suitable based on my personal expertise. Using this speed variance, I calculated the position variance using a basic kinematic formula. On the other hand, for the measurement noise covariance matrix, I approximately determined the sigma value by reading it from graphs.

c) Design the discrete linear Kalman filter (KF) for estimating the state of the given system. Implement the filter in the available m-function `Vozilo.m` and in the boxes below enter the equation of the filter prediction and correction.

$$Pk^- = A_{k-1} P_{k-1}^+ A_{k-1}^T + L Q_{k-1} L^T$$

$$K_k^- = P_k^- H_k^T \left(H_k P_k^- H_k^T + M R_k M^T\right)^{-1}$$

$$\hat{x}_k^- = A_{k-1} \hat{x}_{k-1}^+$$

$$\hat{x}_k^+ = \hat{x}_k^- + K_k (y_k - H_k \hat{x}_k^-)$$

$$P_k^+ = (I - K_k H_k) P_k^- (I - K_k H_k)^{\mathrm{T}} + K_k R_k K_k^{\mathrm{T}}$$

d) Simulate the KF through the whole trajectory by using the dataset `gps_easy.mat`. Initialize the KF state by using the following location components: $\hat{\boldsymbol{x}}_0^{(1:2)} = \boldsymbol{y}_0$ and $P_0^{(1:2,1:2)} = R$, while velocity components you can set as unknown via high uncertainty. Plot the estimated trajectory of the vehicle together with the ground truth and GPS measurements.

Estimated and true vehicle trajectory, sensor measurements

Compute the root-mean-squared-error of the estimation and measurements. Comment the results!

The RMSE of measurement location (GPS measurements) indicates the average deviation between the ground truth position and the GPS measurements. In your case, the RMSE values are approximately 1.57 meters in the x-direction and 1.54 meters in the y-direction. This means that, on average, the GPS measurements are within about 1.57 meters horizontally and 1.54 meters vertically from the true position.

The RMSE of estimated location (Kalman filter estimates) indicates the average deviation between the ground truth position and the estimated position by the Kalman filter. In your case

e) Plot the estimation error of the vehicle location and the trace of the filter covariance matrix P_k^+.

| Vehicle location estimation error | Trace of the filter covariance matrix P_k^+ |

f) Repeat d) and e), but with the more complex dataset `gps_hard.mat` which includes missing GPS measurements and outliers. Which changes did you have to make in KF in order to achieve accuracy close to the one from the previous case?

Incorporating more complex data, I implemented an if statement to check whether the measurements contain valid data or NaN values. For valid data, I followed the standard Kalman filter procedure. However, in the case of NaN values, I only executed the prediction step. Subsequently, I updated the state and covariance estimates (denoted as x+ and P+) with the predicted values (x- and P-) and continued the iteration. The purpose of this approach is to inform the algorithm that the measurement data is unreliable, but by utilizing the prediction based on the mathematical model, I can approximate the possible location of the vehicle. I also

Plot the estimated vehicle trajectory with the true trajectory and GPS measurements.

| Estimated and true vehicle trajectory, sensor measurements |

Plot the estimation error of the vehicle location and the trace of the filter covariance matrix P_k^+.

| Vehicle location estimation error | Trace of the filter covariance matrix P_k^+ |

Compute the root-mean-squared-error of the estimation and measurements. Comment the results!

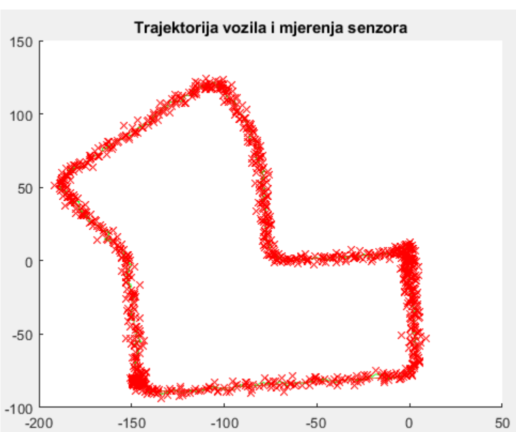
The RMSE of measurement location (GPS measurements) indicates the average deviation between the ground truth position and the GPS measurements. In your case, the RMSE values are approximately 1.57 meters in the x-direction and 1.54 meters in the y-direction. This means that, on average, the GPS measurements are within about 1.57 meters horizontally and 1.54 meters vertically from the true position.

Zaključak

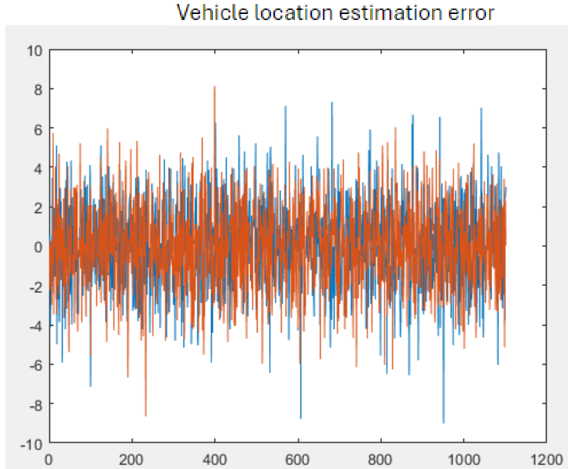
Your conclusion after the exercise which gives a concise overview and comment the obtained results.

In the realm of vehicle location estimation using GPS, the Kalman Filter (KF) is a key player for handling noisy and incomplete measurements. By using datasets like KITTI, which offer real-world measurements with high accuracy, we can see how well the KF performs in tough situations. Handling complex data scenarios with the KF gives us a chance to improve estimation algorithms. By tackling these obstacles, we can advance our understanding of autonomous vehicle localization, making transportation more efficient.

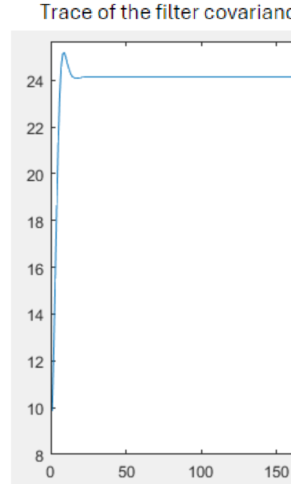
EASY DATA



Vehicle location estimation error



Trace of the filter covariance matrix P_k^+



HARD DATA