Statistical Signal Processing

MATLAB exercise #2

Least Squares Estimation and Signal Detection

Task 1:

One celestial body a travels along an elliptical orbit that lies on the equatorial plane of another larger body B. The position of the larger celestial body can be considered fixed with respect to a. We set the origin $(0, 0)$ of a Cartesian coordinate system in correspondence with the center of B.

The orbit of the body a is known to have a period of 6 years (72 months) but its trajectory is unknown. During the last 12 months the position of a was observed and recorded at regular time intervals, once during each month, resulting in twelve coordinate pairs $(X_1, Y_1), \ldots, (X_{12}, Y_{12})$. However, such recorded values are known to be imprecise due to imperfections in the measuring device. The orbit of a can be modeled in the following way:

$$\begin{bmatrix} X_k \\ Y_k \end{bmatrix} = \begin{bmatrix} R_1 \cos\left(\frac{2\pi}{72}k\right) + C_1 \\ R_2 \sin\left(\frac{2\pi}{72}k\right) + C_2 \end{bmatrix} + \begin{bmatrix} v_k \\ \nu_k \end{bmatrix} \tag{1}$$

where $k = 1, \ldots, 12$, and R_1, R_2, C_1, C_2 are the unknown parameters of an ellipse, representing respectively the length of the two main axes, and the $xy-$coordinates of its center. The quantities v_k, ν_k are statistically independent noise that affect the measurements.

The task is to re-express Equation (1) according to the generic linear model form and estimate numerically the orbit of a in Matlab using a LS-estimator. This must be done by inserting your code into the blank part of the script `Exercise_02.m`. The numerical values of the measurements X_k, Y_k are contained in the Matlab script. Can you predict whether a and B are going to collide? Make sure to include in your report the result figure and to explain in the report the approach you took.

Task 2:

Use the provided MATLAB Dataset data_high_snr.

This data corresponds to a real measurement of a pulsed signal (from a signal generator) done at the University of Oulu. Measurement has been carried out in passband so the signal samples are complex (inphase and quadrature) samples.

Plot the signal power [dBm] versus time [seconds] with this command

```
load data_high_snr
plot(timet,10*log10(abs(data).^2))
```

As you can see we have the case of periodic signal in noise. The samples are the complex inphase and quadrature samples that we transformed into power samples in the above MATLAB line.

- Write a program to automatically set detection threshold for power samples [dBm]. This threshold should separate the power samples into H_0: noise only and H_1: signal + noise. When setting the threshold, how did you solve the problem of having signals at unknown locations (no known noise-only samples)? Can you estimate the probability of false alarm that your detection threshold will give? Is this PFA reasonable? Please notice that using solution that will only work for this specific dataset will give less points.

- Using this threshold (if you could not solve the previous problem, then you may use here manually selected threshold like $-xx$ dBm), plot a figure showing which samples are H_1 and which samples are H_0. Does the result look reasonable?

- Our task in signal detection is to especially detect weak signals, how should you take this into account when setting the threshold?

- Using a detection threshold, automatically by MATLAB estimate the signal parameters: pulse period (after what time is the pulse again repeated?) and pulse duration (how long is each individual pulse?). Do the estimation by taking into account all the received data and not only for example the first pulse. How accurate would you say the signal generator which generated these signals is (does pulse period and duration stay constant)? *Hint: For some parts of this problem ready made functions such as findpeaks (with appropriate input parameters) could be helpful!. Please feel free to use your own solution also.*

- How would you handle the case of very weak signals (sometimes under noise), sketch an algorithm. Innovative solutions will give more points!

The MATLAB programs for above tasks must be provided in the ZIP file and they must be standalone (include also the dataset files). In the report write clear explanations and justifications for the algorithms/methods you used to solve the above tasks.

Homework delivery instructions:

Return to OPTIMA before 10.11. Upload your solutions to the return box in OPTIMA as ZIPPED file containing everything needed to run your programs. The return box in OPTIMA will open well before deadline. Use filename MATLAB2_studentnumber1_studentnumber2, where studentnumber1 and 2 are student numbers of your two team members (please form teams, do not submit alone). In addition to the well-commented program, write a short report explaining how your program works and answering the problems posted (including for example figures obtained by running the simulation program). Please note that the tasks include also the theoretical parts, which need to be addressed in the report in addition to the simulations. Include the student numbers and names of your team members (two members per team) on the top of the first page of the report. Plagiarism is strictly forbidden and will lead to immediate rejection (do not share code outside your team)! You do not need to be able to solve all the problems completely. If you can explain clearly how did you approach the problem and why you did not reach a solution, you may still get a portion of the points. Please note that since this exercise is MANDATORY, make sure to return your solutions even if you could not solve all the tasks. Notice that there is possibility of failing the MATLAB Exercise if you do not get enough points (combined from Exercise 1 and Exercise 2).

Late submissions (if accepted at all) will lead to minus points for late return.