

## MTRN4010.2021 – Project #3

### Applying EKF for map-based robot localization

Project 3 involves applying the EKF, for solving a case of robot localization problem, based on maps. We intend to use it with the real data which was previously used in Project 2. For project 3, we try to solve it incrementally, so that we are able to secure that its many components are working well, before we try the full solution under real data conditions. The first part of the project asks the student to adapt the EKF to solve the map-based localization, in a purely synthetic simulation context. The subsequent project parts aim to use the EKF for processing real data, still in an off-line fashion, but in a loop, processing measurements sequentially as it would occur in a real time implementation.

#### Part A)

Modify the provided example “DemoEKF\_2021.m” for

a) Processing bearing (angle) observations (in addition to the range ones which are already implemented in the example.)

b) Using a proper Q matrix, based on the assumed noise in the inputs of the process model.

Note: for solving item (a) you need to also modify certain parts of the simulator components, for simulating the bearing measurements; you should see, as an example, the source code for implementing the simulated ranges.

**Part B)** You may use parts of the solution developed for Part A, for implementing the EKF-based solution for processing the data previously used in Project2. You will adapt your solution for Project 2, adding the EKF component. The obtained estimates should be more accurate than those obtained in project 2, which were based on simple dead-reckoning.

Assume the following realistic conditions:

Noise in angular rate measurements: standard deviation = 1.7 degrees/second.

Noise in speed sensor: standard deviation = 0.3m/s.

Noise in range measurements: standard deviation = 0.35m.

Noise in bearing measurements: standard deviation = 2.5 degrees

In addition, you need to consider that the LIDAR sensor is located at the front of the platform (as you did in Project 2)

Note: For solving part B, you must remove the bias which is present in the angular rate measurements (as you have done in Project 2)

#### Showing your results for parts A and B

For Part A, you will plot the result at the end of the process. The style of the plots may be as the ones produced by the provided example program (“DemoEKF\_2021.m”).

However, if you have clearly solved part B, you may skip the demonstration of this part.

For Part B, you will include dynamic plots, as you have already done in Project 2. You may (this is not mandatory) simultaneously run the pure dead-reckoning solution, for comparing the performances of both approaches. The lecturer will show his solution, in class, in which both estimation processes are performed simultaneously, and their estimates compared.

For verifying that your pose estimates are consistent with the real pose, we will infer it by inspecting the expected global position of the detected OOI's. Those will appear close enough to the map's landmarks. This visualization approach is the same you had implemented for solving Project 2.

An additional verification will be based on the LiDAR scans, which, when are shown in the global coordinate frame, will appear as almost static surfaces (a minor drift will be acceptable).

**Part C)** You are required to estimate the gyroscope's bias, simultaneously with the estimation of the UGV's pose. In this case, you will not preprocess the angular rate measurements, for removing the bias. The bias removal must be performed by the augmented estimator, in "real-time".

For verifying that your solution is properly estimating the actual bias (in addition to the UGV's pose being estimated) , we will compare your bias estimates with the one which have been obtained by the simple off-line approach (used in Part B and in Project 2). You are required to show, dynamically, the currently estimated bias (by printing its value, or by using text in the figures, or by plotting curves; any approach will be accepted, provided it shows the currently estimated values frequently enough, e.g.@1HZ ). Again, the estimation of the UGV's pose will be verified in the same way we have done for Project 2 and for Project 3/part B.

Note: the approach applied for estimating the bias will be explained on week 7.

### **Relevance of project parts**

Part A: 20% (of the whole project's mark)

Part B: 40%

Part C: 40% (35% for the working solution, 5% for the associated report)

### **Submission details**

You will submit your program and the associated report for part C, uploading your files via Moodle. The deadline is Week 9, Friday 16/April,10:00PM.

The specifications, about how the files are to be submitted, will be given via Moodle, during week 8.

Demonstration: During your lab session, on week 10.

(Optional to students: students who submitted their solutions during week 8, may give their demonstrations during week 9)

Late submissions: We will apply the penalties according to the course outline.

Questions: Ask the lecturer, via Moodle or via email ([j.guivant@unsw.edu.au](mailto:j.guivant@unsw.edu.au))