Alternative Part C – Parameter identification via optimization

Introduction

Part C (alternative). You are required to propose, and implement, an off-line approach, based on optimization, for estimating an unknown system's parameter: the gyroscope bias, "b". You can use the optimizer you prefer, from the suite of optimizers offered by Matlab. The approach must be able to work with any of the provided datasets, individually.

In your solution, you will use the information provided by sensors' measurements (speed sensor, and gyroscope) and by part of the position **ground truth** (GT) provided in the dataset). You can use ONLY the positions (x,y) of the GT (the GT also contains the vehicle actual heading, however you assume that that component is not available, and that only x(t) and y(t) are available from the GT.)

Hints about how to solve this problem: You need to define an adequate cost function, ***C(b)***, to be minimized by the optimizer for inferring the parameter "b" (gyroscope bias).

A possible basic cost function may be implemented as follows:

1) For a given parameter "b", perform predictions of the vehicle's pose, for a horizon of time from t0=0 to t=30 seconds. For that, you will use the provided initial pose, and measurements of speed sensor and of the gyroscope (as you did in Project 1.Part A)
2) Define a cost based on the discrepancy of that trajectory (x(t),y(t)) and the actual one known from the provided Ground Truth (GT), at the times at which the GT was available (Lidar scans events).

Based on your cost function, the optimizer will be able to infer the actual gyroscope bias.

To verify if your implementation performed well, you can do it in different ways:

Way 1: Based on the obtained parameter "b", perform a prediction of the vehicle path, for a horizon of 50 seconds. Plot it, and plot (in the same figure) the GT. In addition, you can predict the vehicle pose, assuming "b=0" (no bias). You should see that the path predicted based on your inferred bias is close to that of the GT, while the one based on "b=0" will diverge from it (except if that bias, in that dataset was b=0, but that is not the case for the datasets we have tried.)

Way # 2: compare your inferred bias with the one that you obtained in Project2.PartB, for the same dataset (if you solved that part of Project 2) (a discrepancy of 0.1 degrees/second may be usual).

Alternative Part C does not require any modules or code from parts B,C,D of Project 1 or from Project 2 parts A and B.

For solving it, you need to apply concepts about defining a cost function, implementing it, and using that cost function with an optimizer. You may use the optimizer “fminsearch”. An example using that optimizer was given for solving a tutorial problem in week 9. Alternatively, you may choose to use any of the optimizers offered in Matlab.

Alternative Part C is an option for completely replacing the originally released Part C of the project. Solving one option or the other is indistinct, in terms of completing part C, and achieving the specified marks for part C.

Using ground truth (GT)

The GT is the actual pose of the platform. GT was provided for validating your results in Project 1. You can also use the GT in Project 2, for comparing with the estimates obtained in parts A and B.

In Alternative part C, you will use the GT for implementing the cost function.

The provided GT corresponds to the vehicle poses at the sample times of the LiDAR events.

So, **data.verify.poseL(:,k)** was the actual vehicle’s pose at time t_k, being t_k the sample time of LiDAR scan #k. That means that in the loop of events, when an event corresponds to a LiDAR scan event, associated to LiDAR scan #k, then the GT pose at that time is available in **data.verify.poseL(:, k)**

You can always know the GT pose, at the time of a LiDAR scan (even if you do not process the LiDAR scan)

If you have questions about this Part of Project 2 you may ask the lecturer j.guivant@unsw.edu.au).