GraphSLAM with ICP

David FILLIAT – ENSTA Paris

9 septembre 2024

1 Introduction

In this practical work, we will work on two SLAM approaches based on the laser scan correlation method named Iterated Closest Point (ICP) [1]. For this, we will use the python code available on the course Moodle. The provided code makes it possible to read laser scan datasets associated with the odometry of the robot during the acquisition, register the scans using ICP and map the environment with two different approaches : a simple incremental SLAM approach and a simplified GraphSLAM algorithm.

Upload your report as a pdf file that includes your answers to the questions and the code you wrote on the Moodle (Paste the code in your report, don't send the python scripts).

2 Provided code

The provided code requires the installation of the `numpy` [1] and `matplotlib` [2] python packages. It contains several python scripts :

– `readDatasets.py` contains a set of functions to read and process the datasets provided in the `dataset` directory. Mainly the `read_u2is` and `read_fr079` functions read and return data as a list of dictionaries representing laser scans. The elements of each scan are :

- `ranges` : a numpy array of distances read by the laser scanner
- `angles` : a numpy array of angles corresponding to the distances above, it is different for each laser scanner
- `pose` : the absolute pose of the laser scanner computed from the robot odometry when the scan was recorded
- `x` et `y` : absolute position of each point of the laser scan computed from `pose`, `ranges` et `angles`

– `icp.py` contains a very basic implementation of the ICP method.

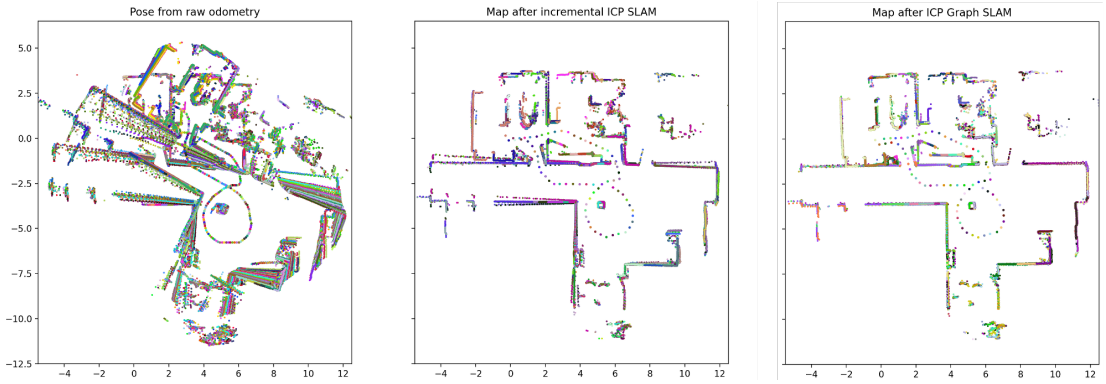

FIGURE 1 – Sample of the `icpIncrementalSLAM` and `icpGraphSLAM` results.

– `icpIncrementalSLAM.py` performs an incremental SLAM (Fig. 1, center).

– `icpGraphSLAM.py` performs a Graph SLAM approach (Fig. 1, right).

1. https://numpy.org/
2. https://matplotlib.org/

3 Incremental SLAM

In this part, you will use the `icpIncrementalSLAM.py` script (use only the U2IS dataset).

Question 1 : By looking at the code and at the course, explain in details what is the algorithm implemented in this script. In particular explain what is the scan used as a reference to compute a new scan position. Highlight the difference with the ICP localisation proposed in the first ICP practical work.

Question 2 : When a loop is closed in the environment, explain what is corrected or not in the localisation and in the map.

Question 3 : What are the role of the `STEP` and `DIST_THRESHOLD_ADD` parameters ? Explain what happens for each of them when they are decreased or increased. What eventually happens when they are too large ? Suggest a reasonable compromise for their values.

4 GraphSLAM

In this part, you will use the `icpGraphSLAM.py` script (use only the U2IS dataset).

Question 4 : By looking at the code and at the course, explain in details what is the algorithm implemented in this script. In particular explain what is the scan used as a reference to compute a new scan position. Highlight the difference with the incremental SLAM proposed in the first part.

Question 5 : When a loop is closed in the environment, explain what is corrected or not in the localisation and in the map.

Question 6 : What are the role of the `DIST_THRESHOLD_ADD` and `DIST_THRESHOLD_MATCH` parameters ? Explain what happens for each of them when they are decreased or increased. What happens when `DIST_THRESHOLD_ADD` > `DIST_THRESHOLD_MATCH` ? Suggest a reasonable compromise.

5 Influence of the dataset

Question 7 : Try the two approaches on the `FR79` dataset (You can use **MAXSCAN = 550** to analyse the behavior while reducing experimentation time). Explain what are the key differences that make this environment harder to map correctly. Is it the kind of environment, the kind of trajectory, the kind of laser sensor ? Propose the best parameter values you can find for this dataset.

Références

[1] Yang Chen and Gerard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3) :145 – 155, 1992. Range Image Understanding.