Simultaneous Localizaiton and Mapping (SLAM)

AMR Final Presentation

Ahmed M Ali

November 2022

Outline

1. Introduction
2. Algorithm details
3. Least Square SLAM
4. Remarks
5. Case study

Introduction

Introduction

- SLAM aims to build the map while localizing the robot within it.
- It is an independent discipline not a single algorithm.
- It needs computational power (to some extent) and uses multiple sensors (internal and external).
- It depends mainly on **range measurement** and **data extraction**

Introduction

- SLAM aims to build the map while localizing the robot within it.
- It is an independent discipline not a single algorithm.
- It needs computational power (to some extent) and uses multiple sensors (internal and external).
- It depends mainly on **range measurement** and **data extraction**

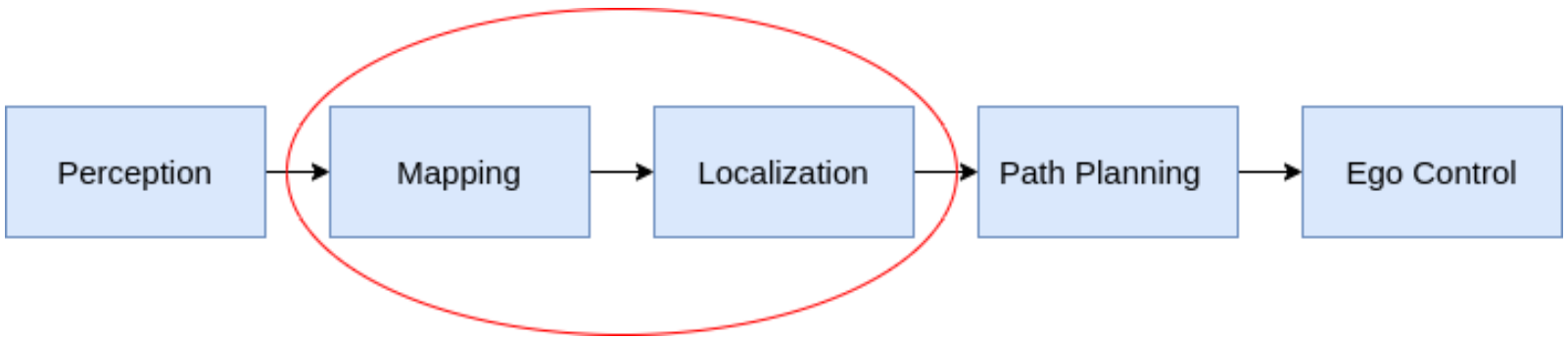


Figure 1: Navigation building blocks

Introduction

The following table gives a brief comparison between SLAM and other algorithms in AMR course.

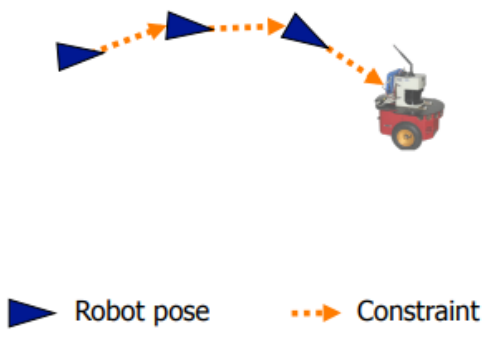
	Localization	Unknown initial pose	Mapping
Bayes filter	✓	X	X
Particle filter	✓	✓	X
SLAM	✓	✓	✓

Algorithm Details

Algorithm Details

How does it work?

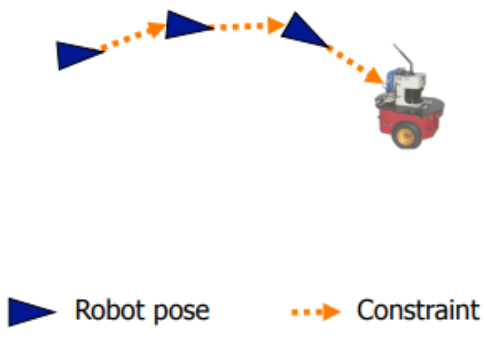
- Simply, robot moves and each new pose will be considered as a node.
- Each two nodes are connected by an edge or constraint (could be odometry data or an observation)
- runs optimization problem that minimizes the errors due to uncertainty and noise.
- Virtual Measurement

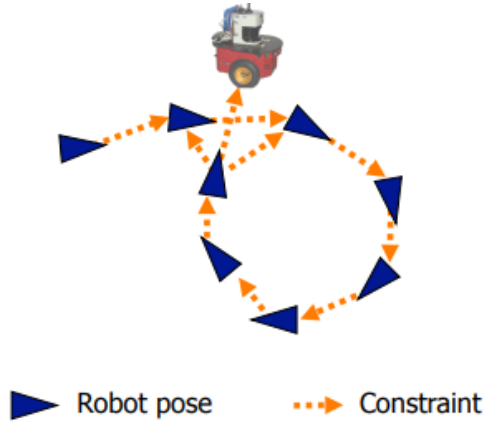


Algorithm Details

How does it work?

- Simply, robot moves and each new pose will be considered as a node.
- Each two nodes are connected by an edge or constraint (could be odometry data or an observation)
- runs optimization problem that minimizes the errors due to uncertainty and noise.
- Virtual Measurement



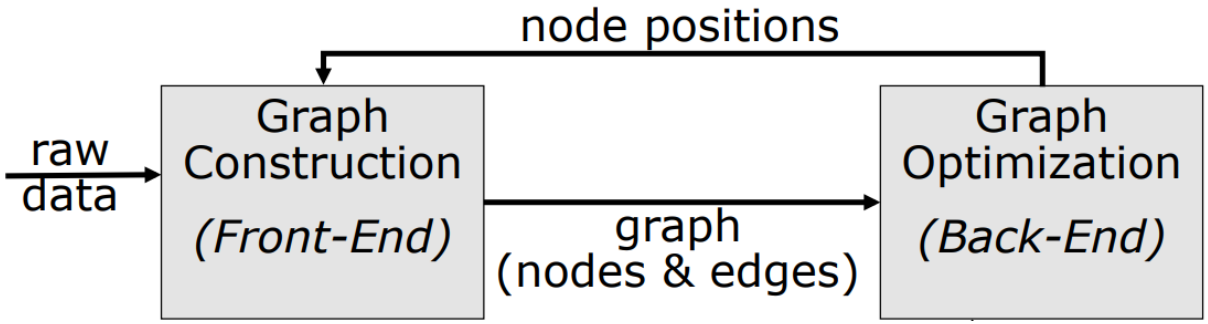


Least-Square SLAM

LS SLAM

SLAM can be composed into two levels:

- Front End
- Back end



LS SLAM

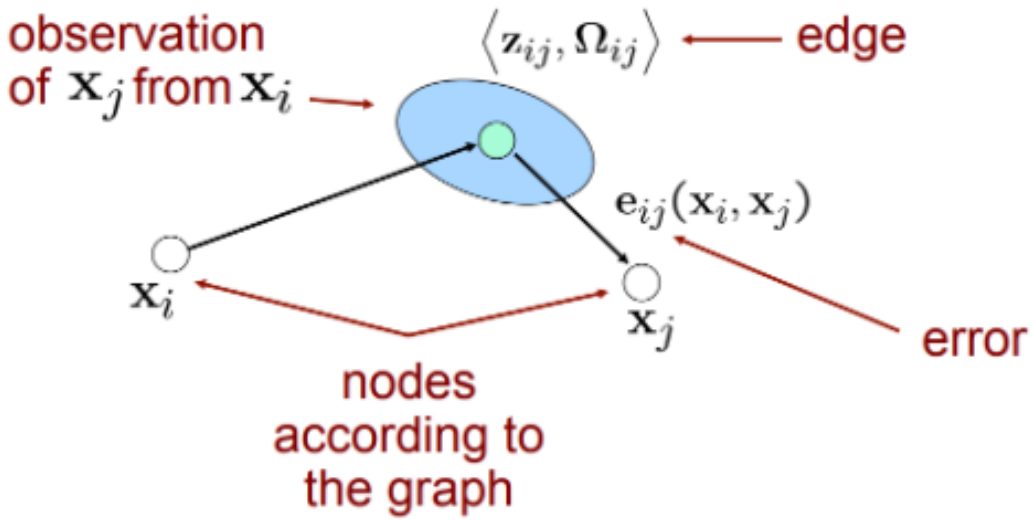
In the back end, LS can be used as optimization problem to minimize the error.

$$x^* = \underset{x}{\operatorname{argmin}} \sum_k e_k^T(x)\Omega_k e_k(x)$$

LS SLAM

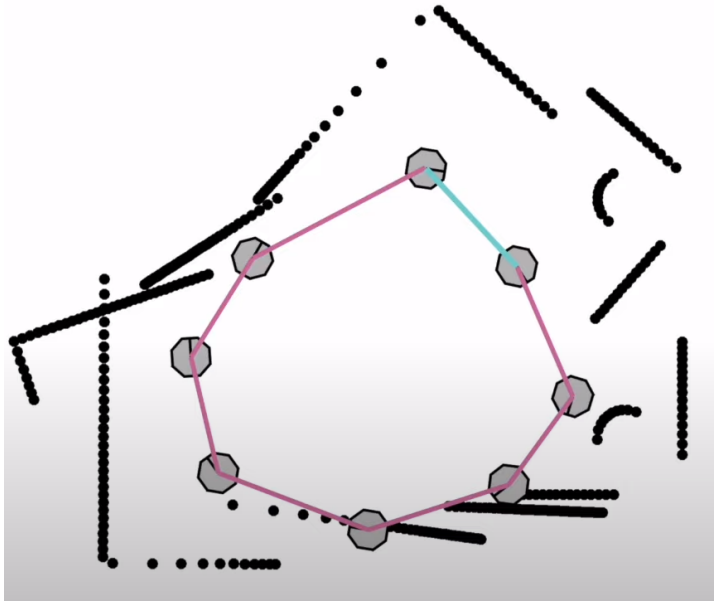
In the back end, LS can be used as optimization problem to minimize the error.

$$x^* = \underset{x}{\operatorname{argmin}} \sum_k e_k^T(x)\Omega_k e_k(x)$$



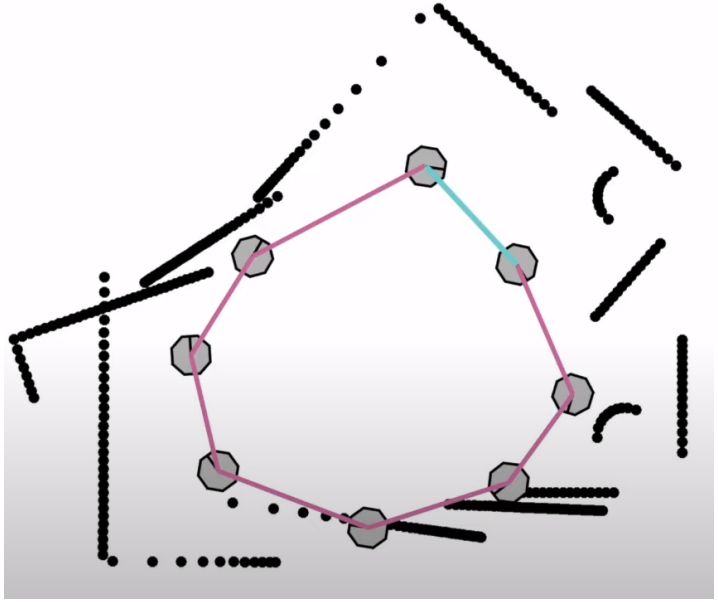
LS SLAM

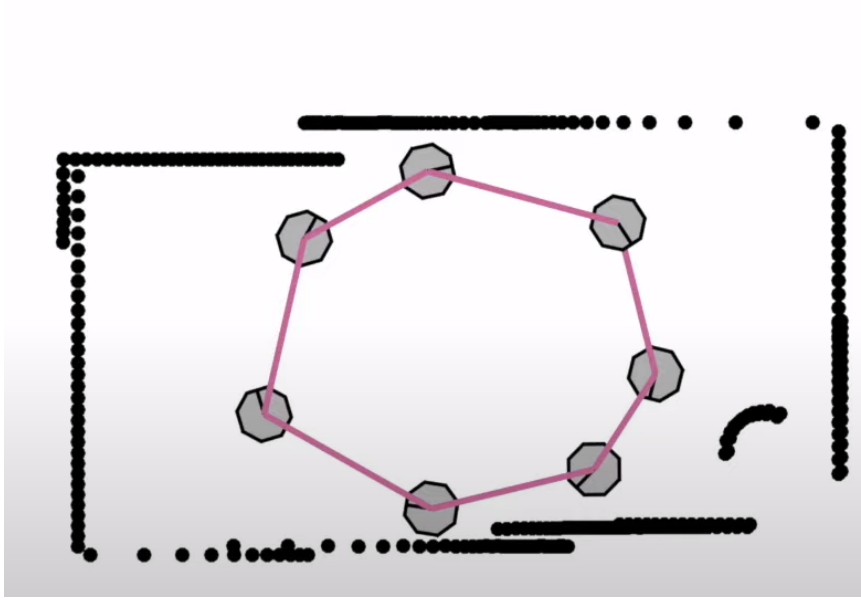
- In general, we want to adjust the poses till the error becomes the least.



LS SLAM

- In general, we want to adjust the poses till the error becomes the least.





Remarks

- **Accumulation of error** due to sequential nature of the algorithm.
  Solution: Draw a pose graph and minimize the error using optimization problem and landmarks.
- **Discontinuous localization** may result from SLAM.
  Solution: add motion model with sensor fusion such as kalman filter. Also, use feature extraction algorithms (BoF or BoVW) for obtaining landmarks
- **High computation power** is needed in SLAM.
  Solution: we make parallel processing and give lower priority to pose graph optimization, making it in fixed interval not every time.

Case Study

I used SLAM in my bachelor graduation project. The video can found be in this drive link.

bibliography

B Douglas

Understanding SLAM Using Pose Graph Optimization

https://www.youtube.com/watch?v=saVZtgPyyJQ&t.

C Stachniss

Least Squares SLAM

https://www.youtube.com/watch?v=VRGOLRGwAjg.

Thank you