Introduction to SLAM

- Known maps require only localization, Known Poses make mapping easy.
  ↳ The joint estimation is more tricky.

SLAM is the task of estimating a map of the environment and at the same time localizing your Sensor/robot in the map that you are currently building.

- **Frontend** : takes the raw sensor data and turns it into an immediate representation. Such as constrains in optimization problem or probability distributions about landmarks derived from sensor data.
- **Backend** : takes this intermediate representation and solve the underlying state estimation or optimization problem. Three different categories.
  - EKF
  - Particle filter
  - Least squares "Graph based SLAM approaches" ›› the most popular
    ↳ use a graph to represent the variables and relations between those variables. most popular : Pose graph - Factor graph

  Pose graph : every node represents the pose of the sensor at a certain point of time. edges represent spatial relations that extracted from sensor data. optimization system tries to find the best node configuration.

Bundle Adjustment : a special form of visual SLAM which has a special form of constraint that is basically the reproduction error of pixel coordinates

---

What is SLAM

localization : estimating the robot's location — Where am I
mapping : building a map — What the World looks like) coupled
SLAM : building a map and localizing the robot simultaneously.
↳ Can be a dense model, Position of distinct Points "landmarks"

localization example — given landmarks "map" help to correct our pose.

mapping example — estimate the landmarks given the robot's Poses.

SLAM example — estimate the robot's Poses and the landmarks at the same time.

- SLAM is an online variant of the bundle adjustment Problem for arbitrary sensors.
- Considered a fundamental Problem for truly autonomous robots.
- SLAM is the basis for most navigation systems.

Definition of the SLAM Problem

Given : the robot's Controls $u_{1:T} = \{u_1, u_2, \dots, u_T\}$
observations $z_{1:T} = \{z_1, z_2, \dots, z_T\}$

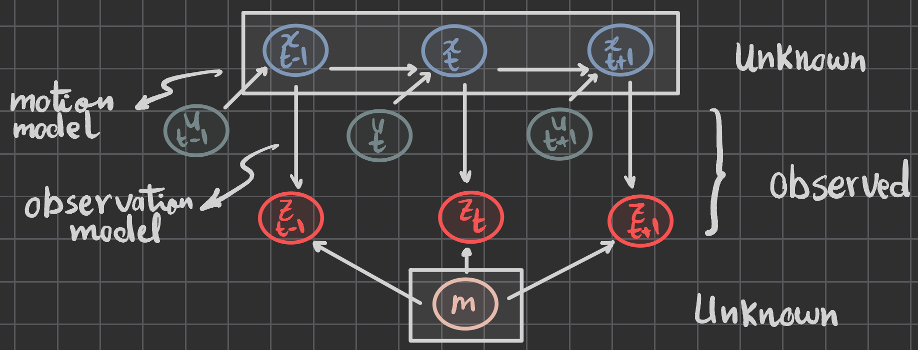
Wanted : map of the environment m
Path of the robot $x_{0:T} = \{x_0, x_1, \dots, x_T\}$

Probabilistic approach

Use the Probability theory to explicitly represent the uncertainty.

$$P(x_{0:T}, m \mid u_{1:T}, z_{1:T})$$

Graphical model



Full SLAM : estimates the entire Path.

Online SLAM : Seeks to recover only the most recent Path. $P(x_t, m \mid z_{1:t}, u_{1:t})$

↳ marginalizing out the Previous Poses.

$$P(x_t, m \mid u_{1:t}, z_{1:t}) = \int \dots \int P(x_{0:t}, m \mid u_{1:t}, z_{1:t})\, dx_{t-1} \dots dx_0$$

Why SLAM is a hard Problem

. Robot Path and map are both unknown.
↳ Map and Pose estimates Correlated.

. Known vs. Unknown Correspondence . "data association"
↳ The mapping between observations and the map is unknown
Picking wrong data associations Can have catastrophic Consequences. (divergence)

Volumetric vs. Feature-based SLAM

. Feature-based maps
. Volumetric : 3D Scans building dense models or grid maps

Three Traditional Paradigms

Kalman Filter

Particle Filter

Graph-based
↳ The most frequently used representation