After Final 14.11.23

@SE461

"Lecture-8"

Robot Navigation

→ the act, activity or process to find the way

→ It is a the ability to determine a location and plan a path to some goal.

Robots need the ability to navigate,

- Learn the Env → 'Model' → (my goal) → (my location)
- Estimate where it is in the Env → 'Localize'
- Move to desired locations in the Env.

→ Environment:

Problem?

→ Env are know vs unknown

→ Env are static vs dynamic

→ Env are structured vs unstructured

→ (Indoor vs Outdoor)

We have to design the Navigation System

Robots Navigation:

- Path Planning → how to reach my goal from current location
- Localization → find my current location
- Mapping → Env এর Map (কোথায় কি আছে)
- Exploration → visit the Env and do the map of unvisit places.

→ Path Planning:

path will be optimal according to our requirement.

3 things need to consider:

- Env
- Success metrics
- Robot capability

Algo-1: "Visual Homing" → (Purely Reactive Nav)

→ if we just have only one info then we will use this Algo.

- → If we see the goal.
- → If we don't see the goal

afte selecting this algo, it will work like,

→ if we see goal then move straight to it.

→ if we don't see goal then move randomly to reach the goal.

Algo-2: "Bug-based Path Planning"

→ if we know two info. that ~~the~~ [where is my goal] and ~~the~~ [it's distance] from me.

→ we will move straight when we see the goal. If we face a obstacles then we will move by the side of the obstacles until it ~~finishes~~ finishes. Then again we will move straight to our goal.

~~Here~~

→ As we know the distance, our robot will move to the shortest distance direction. But we don't know the env fully. So, that it is not optimal.

~~Algo 2~~

If we have the map of the Env then we can select the ~~bet~~ best path.

Map Representation:

→ we will use weighted Graph for that

→ free path এরকম weight is distance/time

→ obstacle path এরকম weight is infinite.

~~Algo 2 will use~~

Now, the Algo-3

Algo-3: "Metric/Global Path Planning"

→ when we have the full knowladge

→ use Shortest Path Grap Algo

Summary:

- Visual Homing (local sensing and feedback control)
- Bug-based Path Planning (local without map)
- Metric (A*) Path Planning (global with map)

→ Localization:

current location – where am I?

do I have a map?

↳ yes → a global position in the world
↳ no → position in ref to other obj.

We will learn 2 types of localizations;

(1) Dead-Reckoning (Motion)

→ initial position থেকে track রাখবো and next position খুঁজে বের করবো।

→ prev position is my ref position.

→ here we do not have any map

→ calculate position in each step.

→ for example: Inertial Navigation System (INS)

(2) Land Mark (Sensing) Based

→ Env এর know landmark use করে location বের করবে।

→ landmark এর position fixed থাকে own position calculate করবে।

→ For example: visual landmark, radio tower, GPS.

(3) State Estimation

• uncertainty in motion and sensing

Combine Motion and Sensing

~~Dead-Reckoning~~

(dead-reckoning+uncertinity) + (landmark+ uncertinity)

→ Kalmal Filters
→ Particle Filters