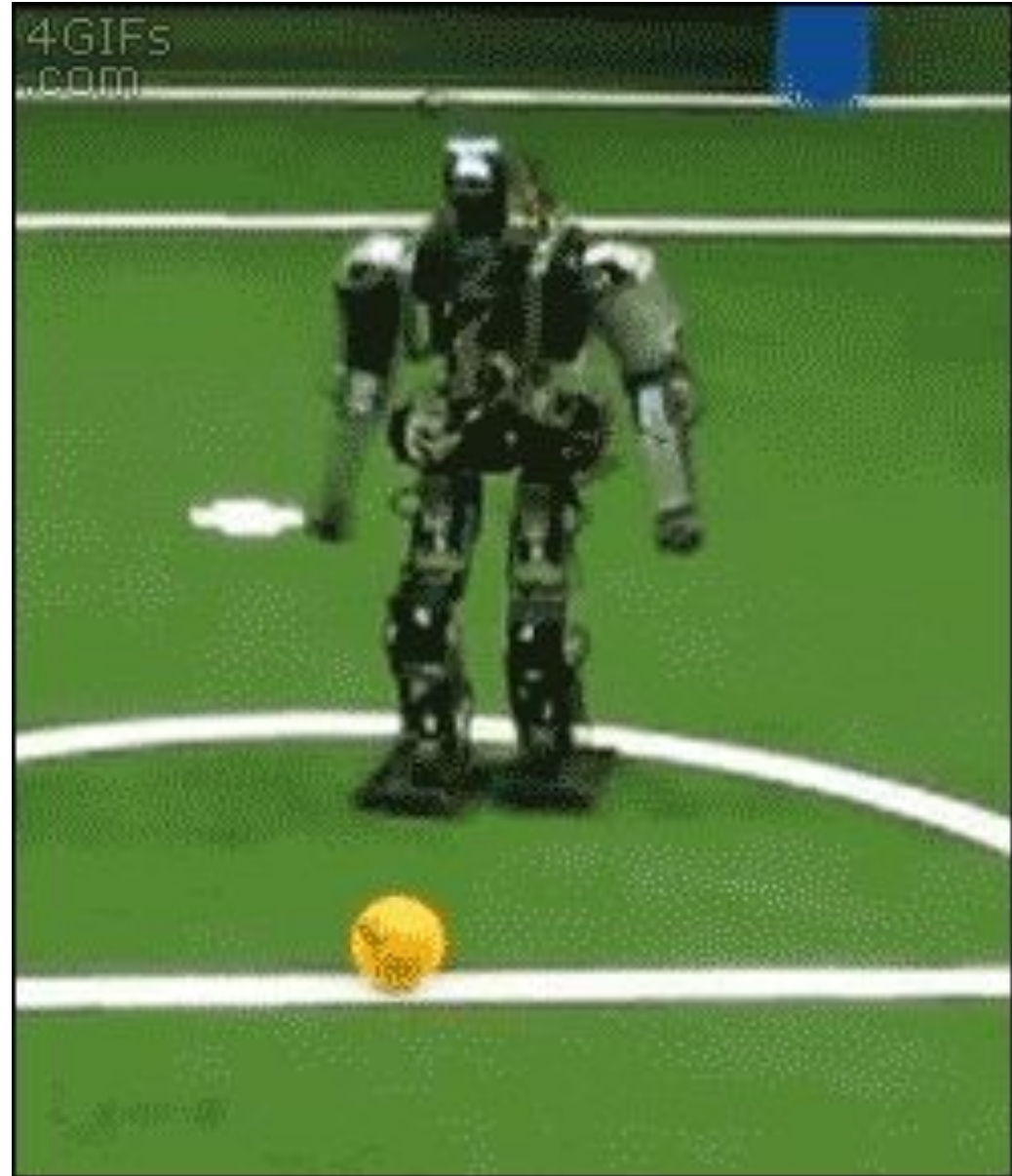


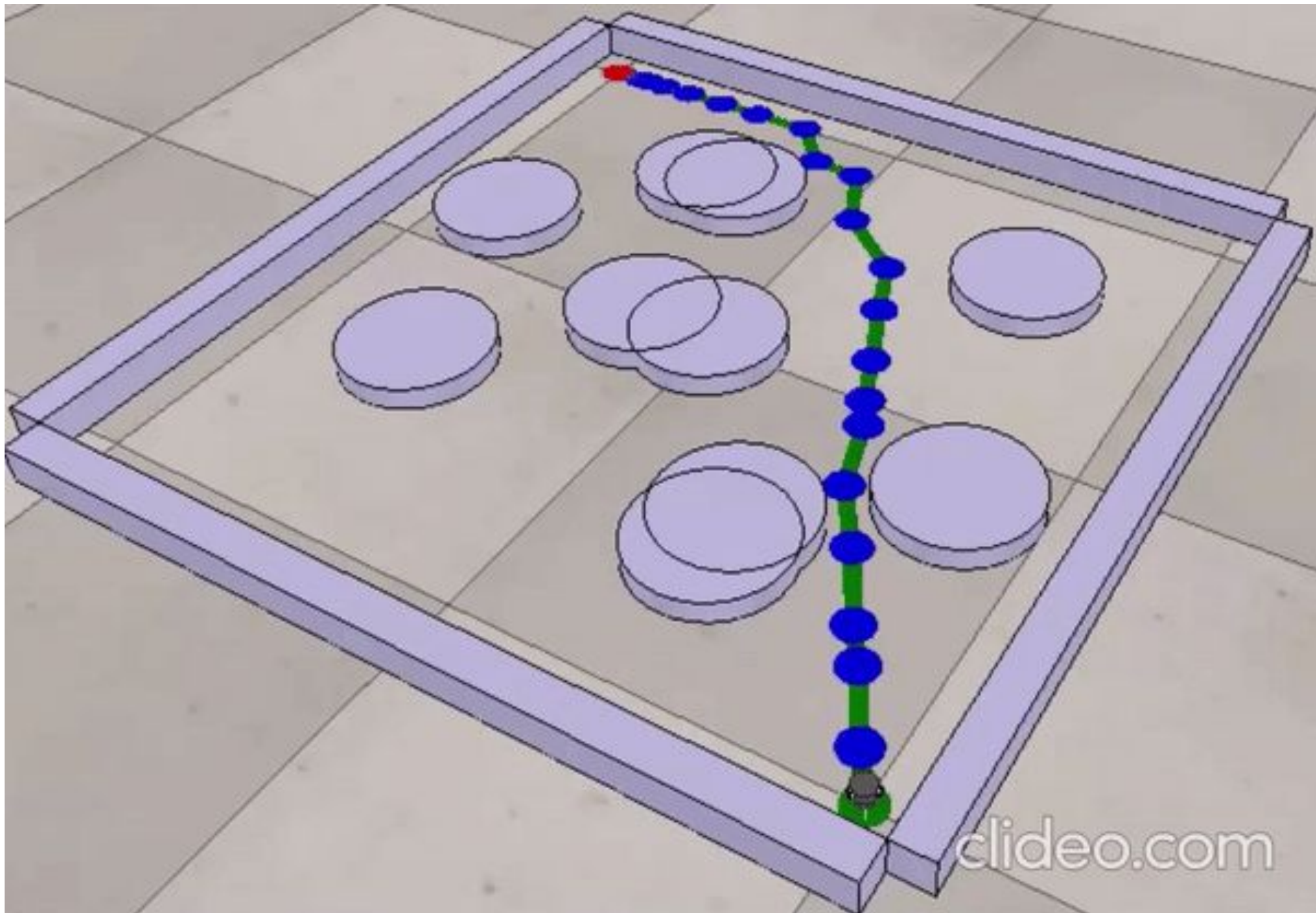
Introduction to Robotics

CSE 461

Class Topic: Robot Navigation

Riad Ahmed
Lecturer
Brac University

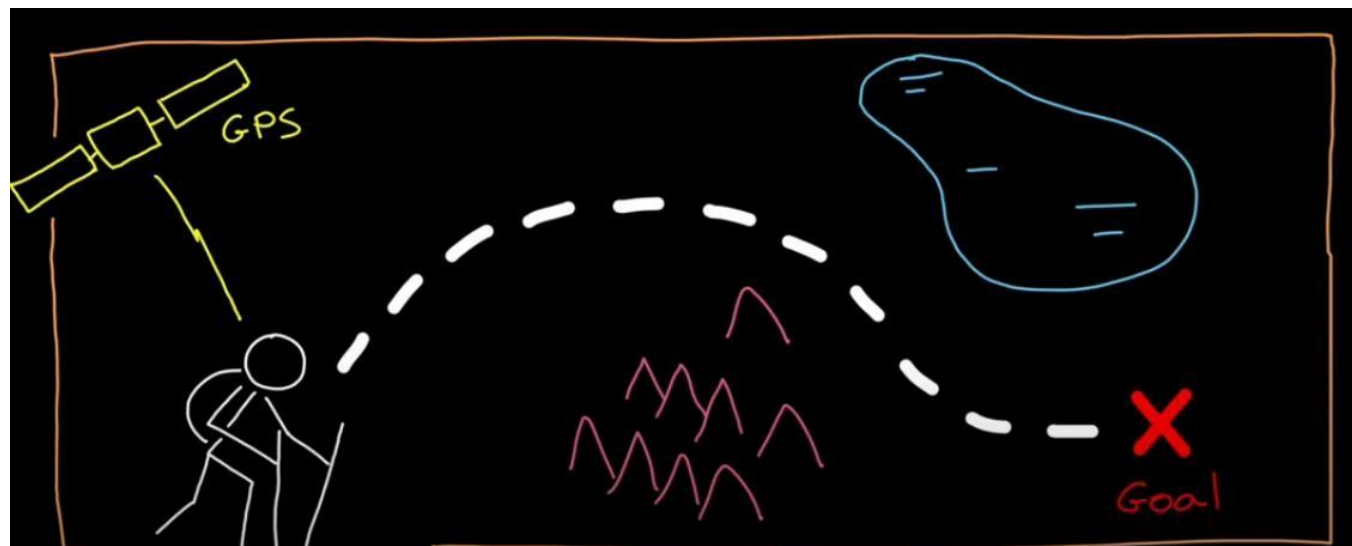




clideo.com

Navigation

- The act, activity, or process of finding the way to get to a place
- Navigation is the ability to determine your location and plan a path to some goal



Robot Navigation

For autonomous behavior, robots need the ability to navigate:

- Learn the environment->“Model”
- Estimate where it is in the environment->“Localize”
- Move to desired locations in the environment

Navigation Example



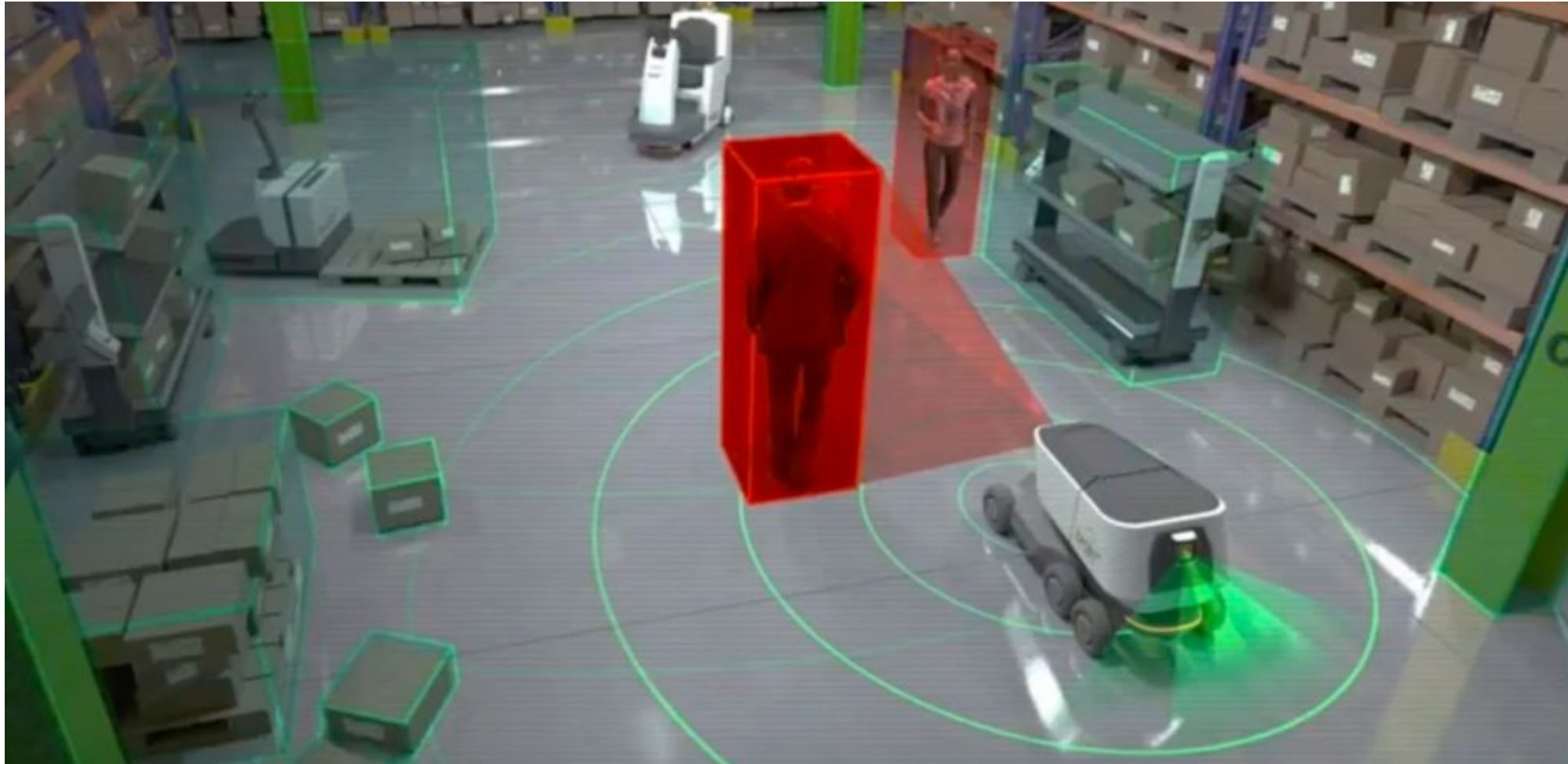
Scenarios

- Hospital Helper
(e.g. Diligent, Tugs)
- Office security or mail-delivery
(e.g. Cobalt, Savioke)
- Tour Guide robot in a museum (Minerva)
- Autonomous Car with GPS and Nav system

Biological analogies:

Humans, bees and ants, migrating birds, herds

Environment



Navigation Problem

Problem Characteristics

- Environments are Known versus Unknown
- Environments are Static versus Dynamic
- Environments are Structured versus Unstructured (Indoors versus Outdoors?)

We have to design the navigation system

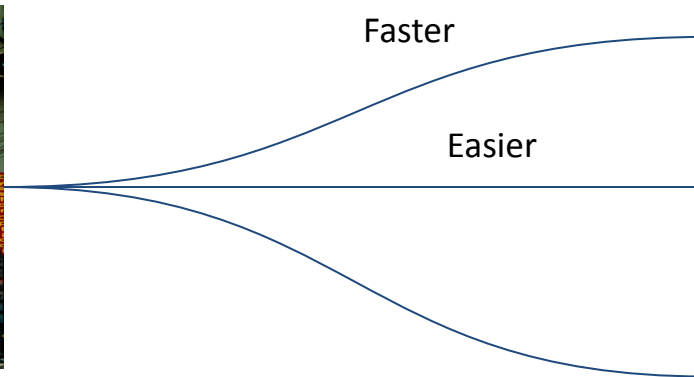
Robots Navigating

- **Path Planning**: How I get to my Goal?
- **Localization**: Where am I?
- **Mapping**: Where have I been?
- **Exploration**: Where haven't I been?

What is Path Planning?



Banani



Badda

What is Path Planning?

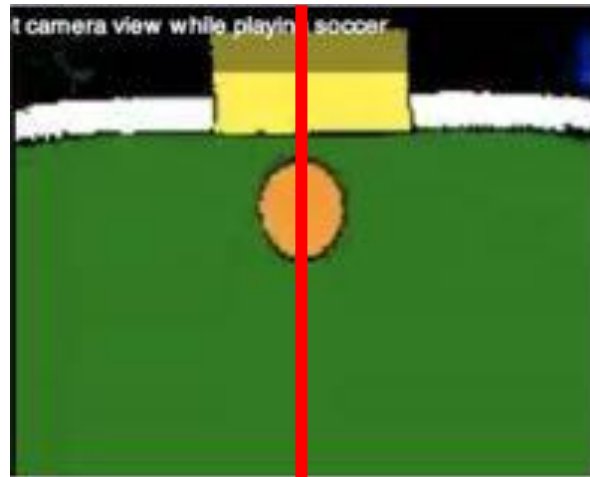
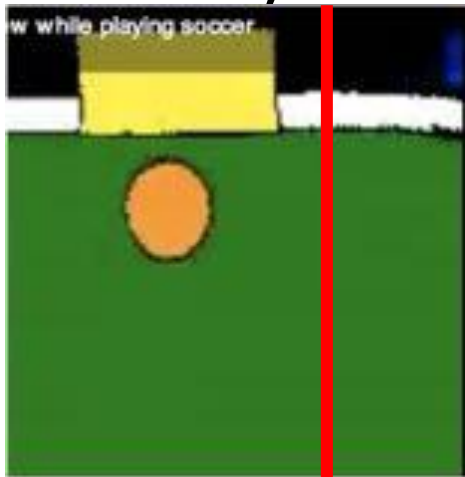
- Simple Question: **How do I get to my Goal?**
- Not a simple answer!
 - Can you see your goal?
 - Do you have a map?
 - Are obstacles unknown or dynamic?
 - **Does it matter how fast you get there?**
 - **Does it matter how smooth the path is ?**
 - **How much compute power do you have?**
- Path Planning is best thought of as a Collection of Algorithms

We know our criterias now what?

Basics: Visual Homing

Purely Reactive Navigation

- Measure Visual (x,y) Position of Goal
- Move to bring goal to Visual Center
- Proportional Control (if you see the goal), Random walk (if you don't)



If Map is given ?

Map Representation: Feature based



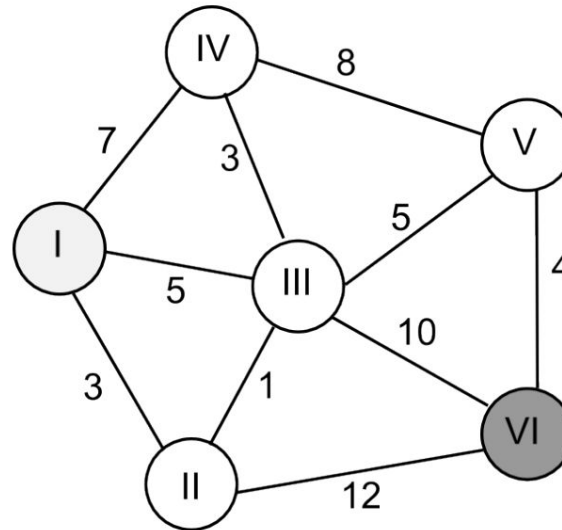
Path Finding Algorithms

- All Map Representations are a weighted “graph”

- Nice part is that you only need to do this once

- Algorithm: Compute shortest paths in the graph

- Path is represented by a series of waypoints



Metric/Global Path Planning

□ What if the Robot has Full Knowledge

- A map of the environment and robot + goal's locations
- Goal: Find a “optimal” path (typically distance)

□ Two Components

- Map Representation (“graph”)
- Path Finding Algorithms:
 - Shortest-Path Graph Algorithms (Breadth-First-Search, A* Algorithm)



Types of Path Planning Approaches

- Basics
 - Visual homing (Purely local sensing and feedback control)
- Bug-based Path Planning (mostly-local without a map)
- Metric (A^*) Path Planning (global with a map)

Robots Navigating

- Path Planning: How to I get to my Goal?
- Localization: Where am I?
- Mapping: Where have I been?
- Exploration: Where haven't I been?

Localization

□ Simple Question: *Where am I?*

□ Not a simple answer:

□ Do you have a map?

Yes => a global position in the world

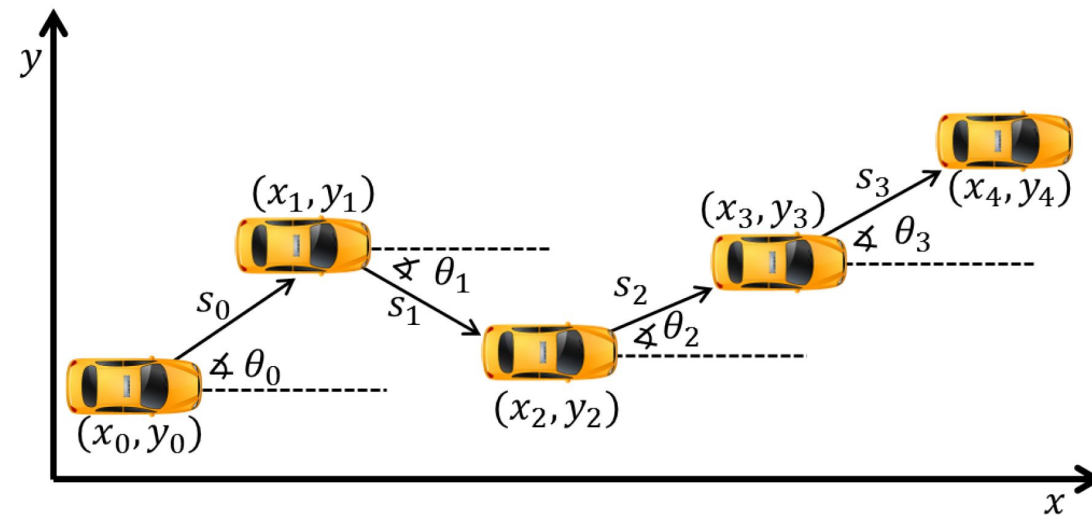
No => position in reference to other objects? Or your own past?

□ What can you sense?

□ Localization is a “collection of algorithms”

Dead-Reckoning(Motion)

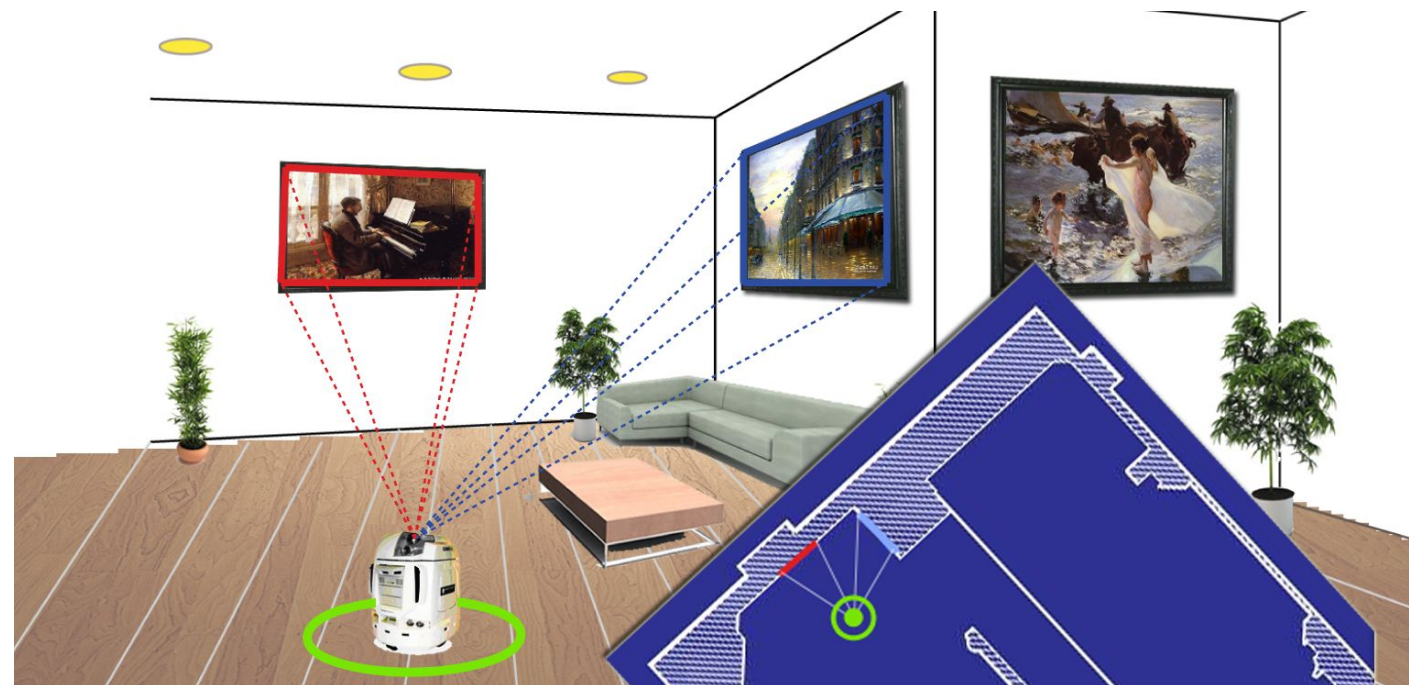
- Keep track of initial position and **the series of movements/actions** that you made.
- **Method: Take a “step”, compute new position.**
- Also called odometry or path integration Example:
Inertial navigation systems (INS)



LandMark(Sensing) Based

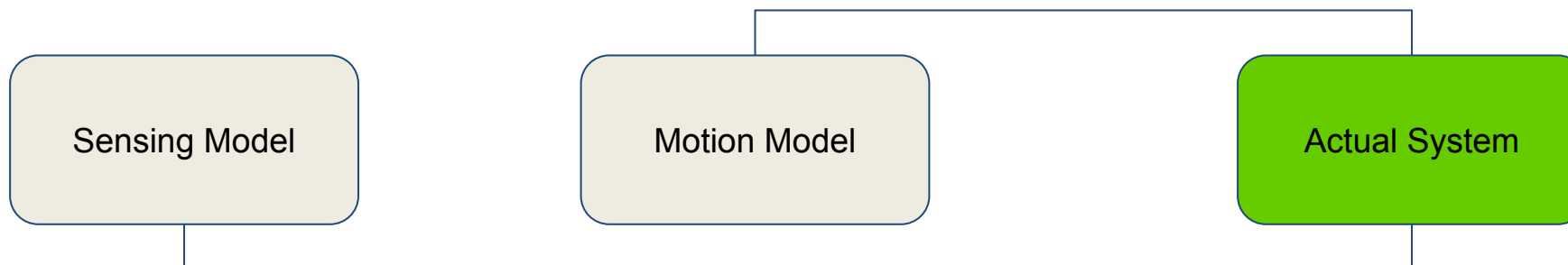
- **How it works**

- *Opposite of dead-reckoning!*
- Use measurements to external landmarks of known position
- Examples: visual landmarks, radio towers, GPS!



State Estimation (uncertainty in motion & sensing)

- **Key Idea: Combine Motion and Sensing**
 - **(Dead-reckoning + uncertainty) + (Landmarks + uncertainty)**
 - Each has error, but the error can be complementary
- **Kalman Filters**
- **Particle Filters (Monte Carlo Localization)**



Localization Techniques

□ Dead-reckoning (motion)

- Keep track of where you are without a map, by recording the series of actions that you made, using internal sensors. (also called Odometry, Path Integration)

□ Landmarks (sensing)

- Triangulate your position geometrically, by measuring distance to one or more known landmarks
E.g. Visual beacons or features, Radio/Cell towers and signal strength, GPS!

□ State Estimation (uncertainty in motion & sensing)

□ *Probabilistic Reasoning*

- **Kalman Filters** (combine both motion and sensing)
- Particle Filters (also known as Monte Carlo Localization)

Next Class

Robots Navigating

- **Path Planning**: How to I get to my Goal?
- **Localization**: Where am I?
- **Mapping**: Where have I been?
- **Exploration**: Where haven't I been?

Thank You