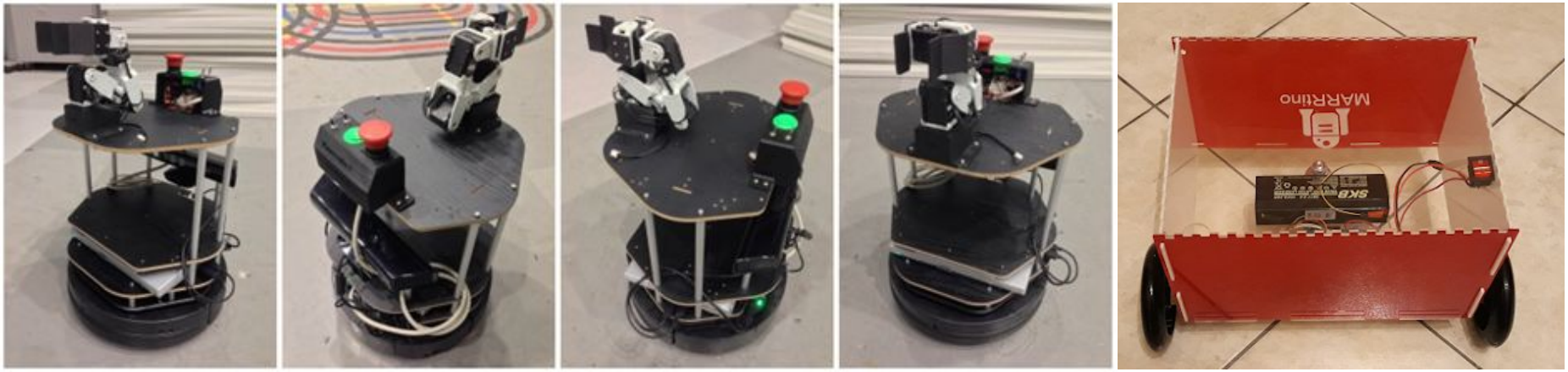


RoboCup@Home Education
ONLINE CHALLENGE 2020
Online Classroom Open Platform

05 Robot Navigation (2/2) : Autonomous Navigation (Simulation)

Jeffrey Tan, Jupiter Robot | 2020.05.14

RoboCup@Home EDUCATION

MathWorks®

JUPITER ROBOT

Online Challenge 2020: Online Classroom OP

05 Robot Navigation (1/2) : SLAM Map Building (Simulation)

Speakers: Jeffrey Tan, Jupiter Robot

Time: **May 14, 2020 (Thu) 10:00 - 11:00 am (GMT+8)**

05 Robot Navigation (2/2) : Autonomous Navigation (Simulation)

Speakers: Jeffrey Tan, Jupiter Robot

Time: **May 14, 2020 (Thu) 11:00 - 12:00 noon (GMT+8)**

Zoom: https://cernet.zoom.com.cn/j/63946172707 | PW: robocup

Facebook Live: https://www.facebook.com/robocupathomeedu/live/

Web:
https://www.robocupathomeedu.org/challenges/robocuphome-education-online-challenge-2020

Online Classroom:
https://www.robocupathomeedu.org/learn/online-classroom/online-challenge-2020

** Privacy reminder: Video will be recorded and published online.

Robot Navigation

SLAM Map Building

- Mobile Robot
- Robot Odometry and Localization

Autonomous Navigation

- Robot Motion Planning
- Autonomous Navigation

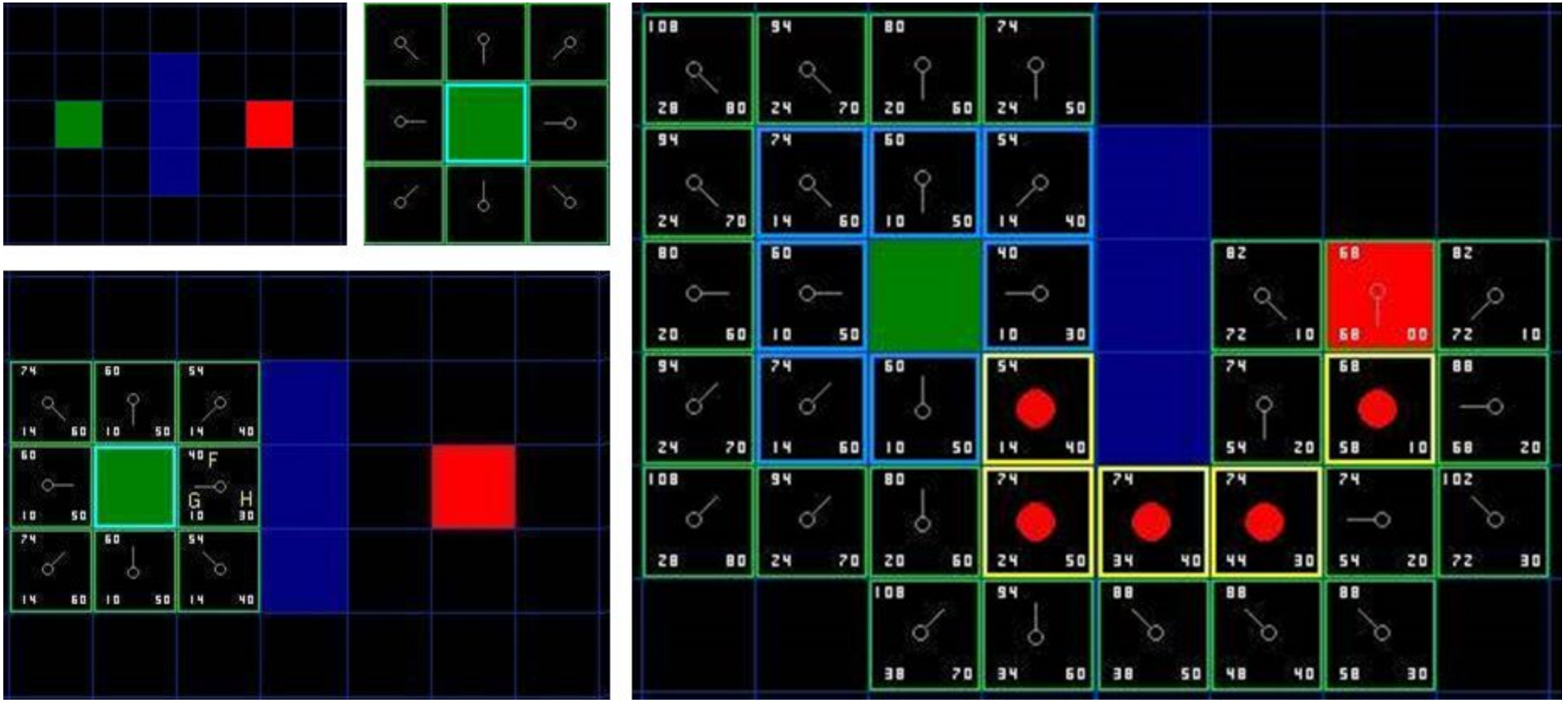
Robot Motion Planning
Autonomous Motion Generation

A * algorithm

[http://www.policyalmanac.org/games/aStarTutorial.htm]

start→n *n→goal*

Path with shortest cost, $f(n)=g(n)+h(n)$

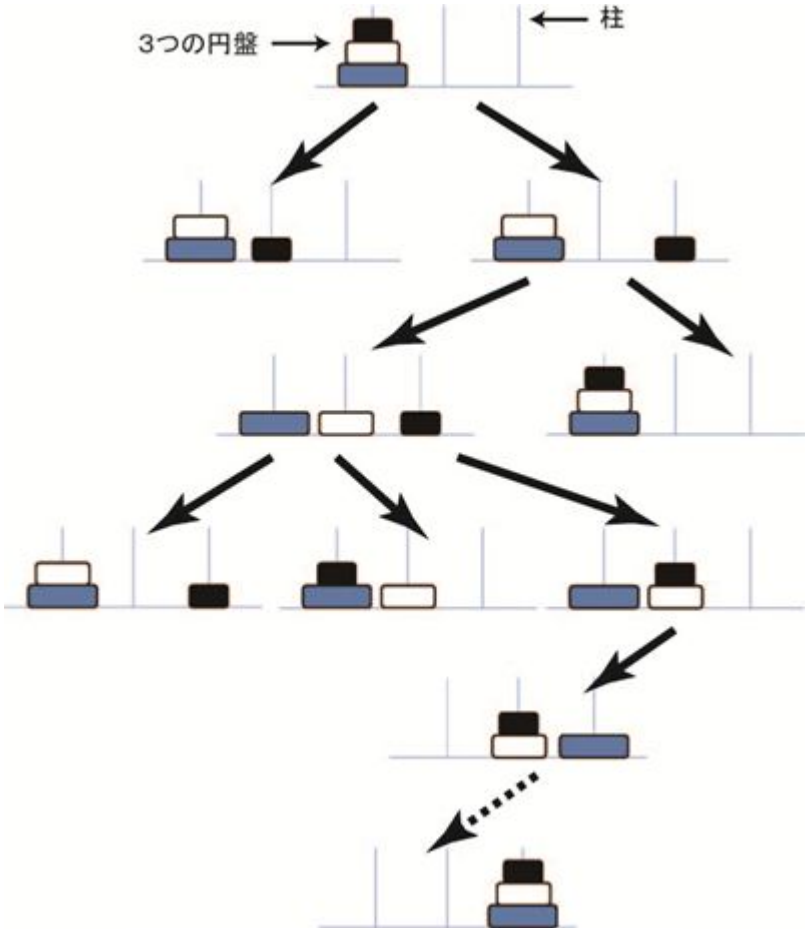


Robot Motion Planning
Autonomous Motion Generation

State Space & Search Problem

- Search Tree
- Each state, Node
- Connection between states, Link
- Start State
- Goal State

Demo of Hanoi Tower

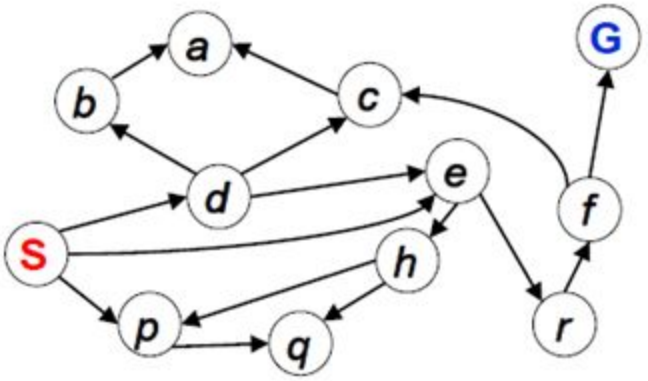


State space in Hanoi Tower

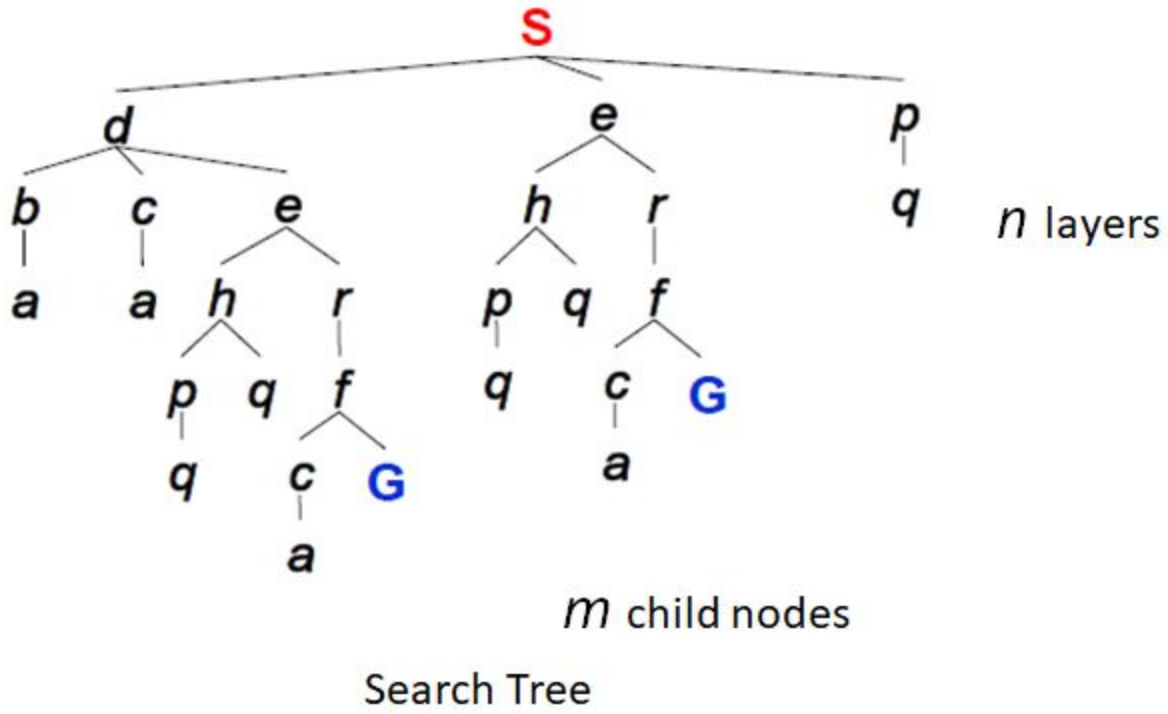
Robot Motion Planning
Autonomous Motion Generation

State Space and Search Problem

- Search Space Graph & Search Tree



Search Space Graph



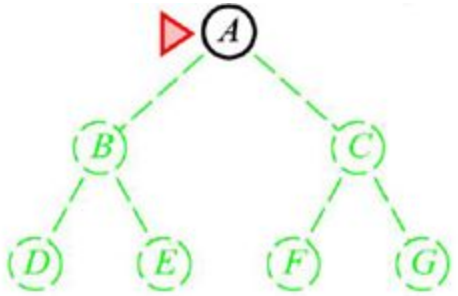
Search Tree

Robot Motion Planning
Autonomous Motion Generation

State Space and Search Problem

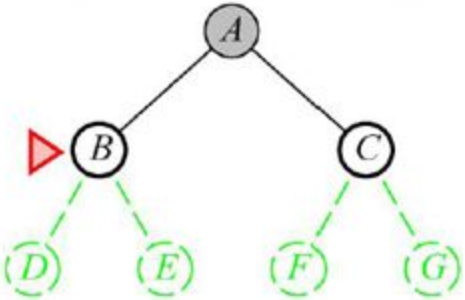
- Breadth First Search
- Depth First Search
- Uniform Cost Search
- Depth Limited Search
- Iterative Deepening Search

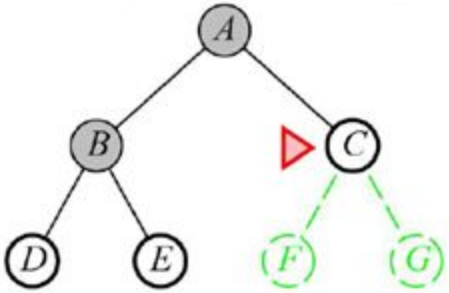
Robot Motion Planning
Autonomous Motion Generation

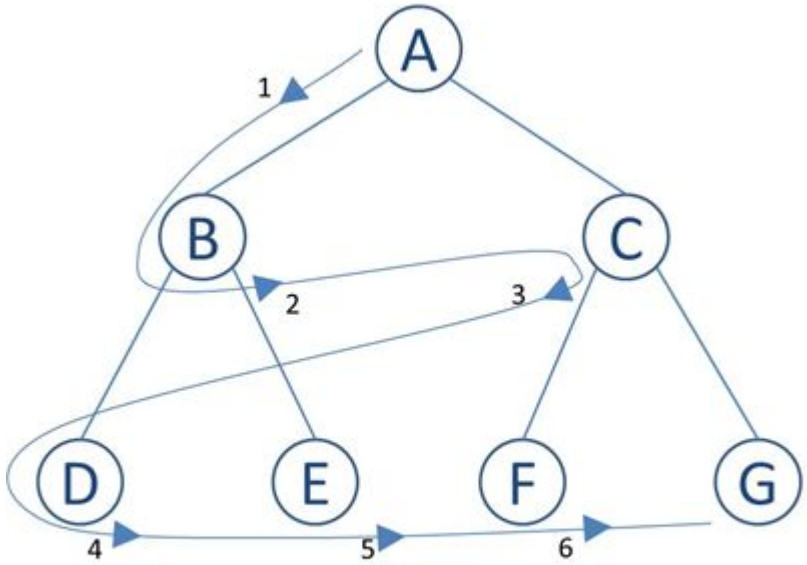


Breadth First Search

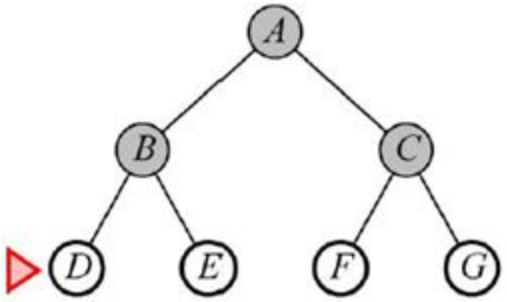
- Order of time computation: $O(m^n)$
- Order of space computation: $O(m^n)$







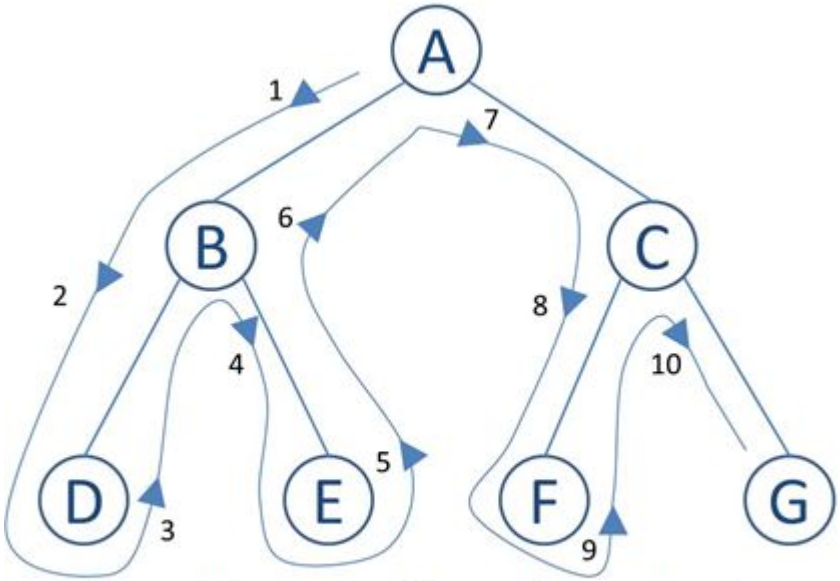
Search tree and flow of search of breadth first search approach



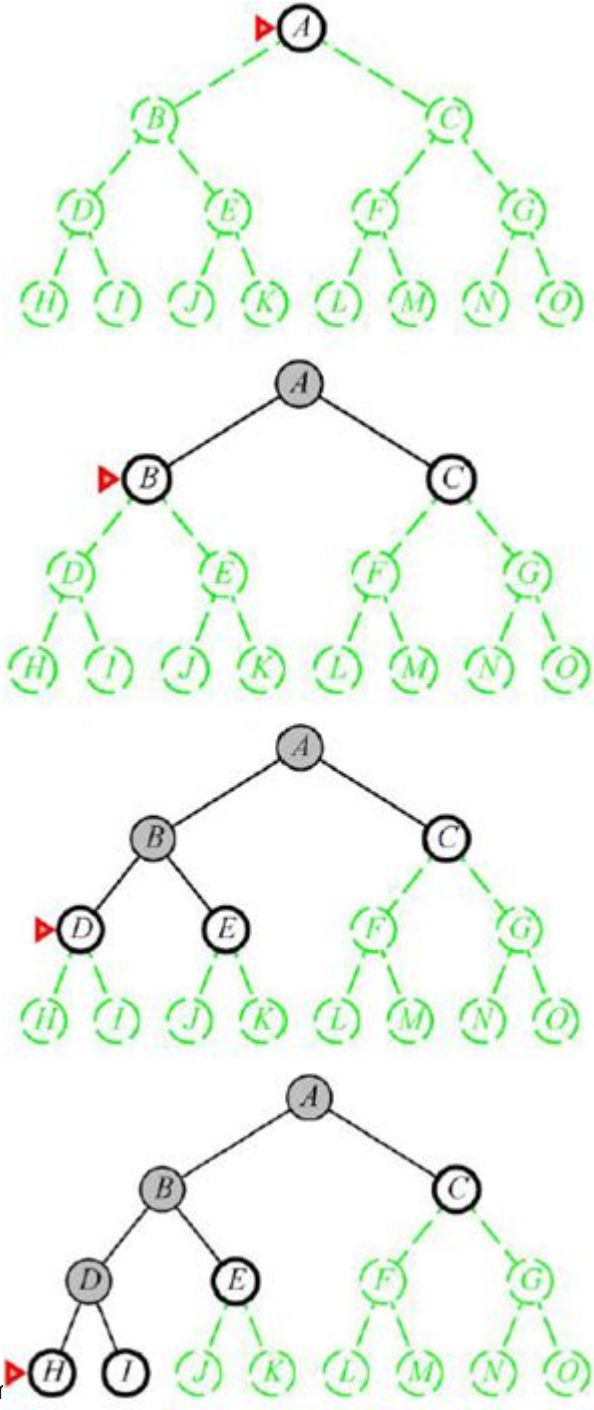
Robot Motion Planning
Autonomous Motion Generation

Depth First Search

- Order of time computation: $O(m^n)$
- Order of space computation: $O(mn)$



Search tree and flow of search of depth first search approach

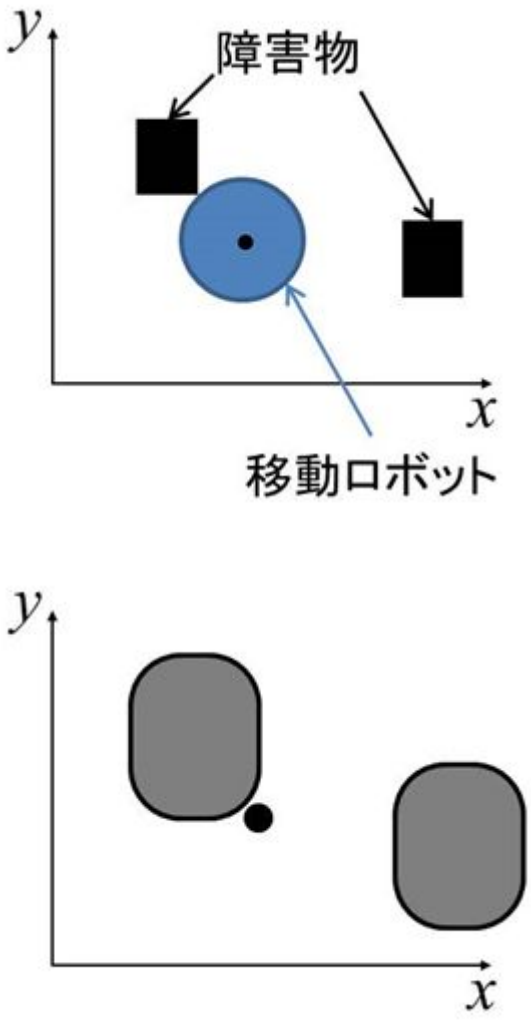


Robot Motion Planning
Autonomous Motion Generation

Motion Planning

Configuration Space, C-Space

- Instead of searching directly from visible robot position or form, C-Space is a representation of robot operational state space



Configuration space example
in mobile robot navigation

Autonomous Navigation

Adaptive Monte Carlo Localization

- https://en.wikipedia.org/wiki/Monte_Carlo_localization
- http://wiki.ros.org/amcl

B. Autonomous Navigation of a Known Map with TurtleBot
   - http://wiki.ros.org/turtlebot_navigation/Tutorials/indigo/Autonomously%20navigate%20in%20a%20known%20map

TurtleBot2 in Gazebo | Simulation

TurtleBot in Gazebo (Indigo)

- http://wiki.ros.org/turtlebot_simulator

Installation

- $ sudo apt-get install ros-kinetic-turtlebot-simulator

Bring up

- $ roslaunch turtlebot_gazebo turtlebot_world.launch

Autonomous Navigation (RViz) | Simulation

1. Launch AMCL with scanned map
   - $ roslaunch turtlebot_gazebo amcl_demo.launch map_file:=/home/<username>/my_map.yaml
2. Use RViz for navigation visualization
   - $ roslaunch turtlebot_rviz_launchers view_navigation.launch
3. Send a navigation goal with RViz
   - Click the “2D Nav Goal” button
   - Click and drag on the map for the goal location and orientation

Autonomous Navigation (Coding) | Simulation

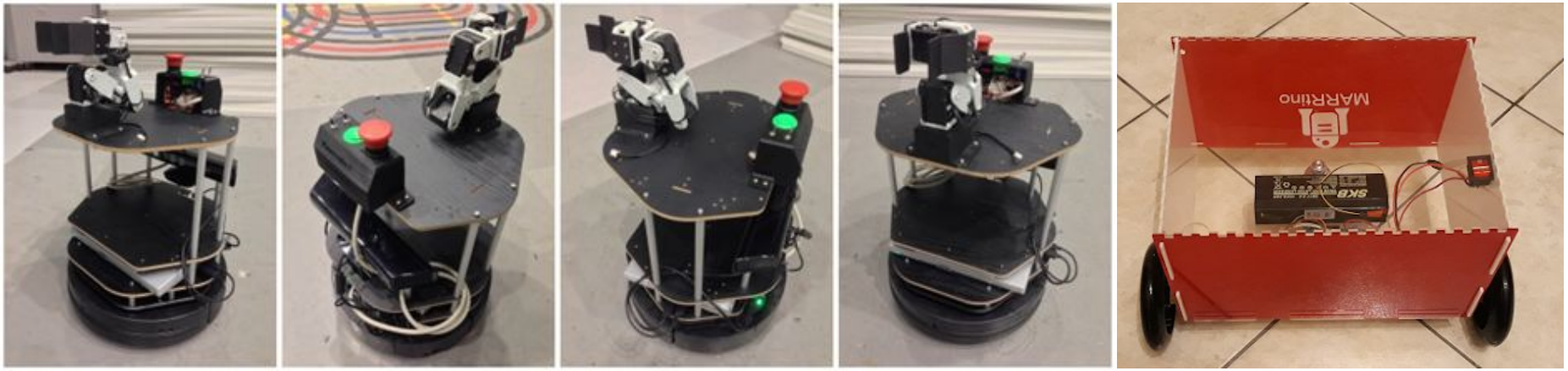
1. Launch AMCL with scanned map
   - $ roslaunch turtlebot_gazebo amcl_demo.launch map_file:=/home/<username>/my_map.yaml
2. Use RViz for navigation visualization
   - $ roslaunch turtlebot_rviz_launchers view_navigation.launch
3. Determine and update the coordinate of the goal location
   - Click the “Publish Point” button and move the cursor to a desired goal location on map (do not click on the map)
   - Record the first two numbers (x, y) on the bottom left corner of RViz window
   - Update the numbers (x, y) as “Location A” in /rc-home-edu-learn-ros/rchomeedu_navigation/scripts/navigation.py
4. Launch the navigation code
   - $ roslaunch rchomeedu_navigation navigation.launch
   - Set robot current position with “2D Pose Estimate”

Assignments

1. Combine speech interaction and robot navigation to develop an interactive robot navigation application
   a. Design and develop robot autonomous navigation application with speech interface, using ROS navigation package and Gazebo simulator.
   b. At least 3 waypoints.
2. Upload to GitHub
   a. Create own repository and upload the source code, map, system design, and navigation+speech interaction video (with terminal results) to GitHub.

Autonomous Navigation

Autonomous Vehicle [https://www.youtube.com/watch?v=tiwVMrTLUWg]



RoboCup@Home Education
ONLINE CHALLENGE 2020
Online Classroom Open Platform

Web: https://www.robocupathomeedu.org/challenges/robocuphome-education-online-challenge-2020
Online Classroom: https://www.robocupathomeedu.org/learn/online-classroom/online-challenge-2020
Online Entry Form: https://forms.gle/UBREeC1xTCVQ9wr78
Online Entry Form (backup): https://www.wjx.cn/jq/72082120.aspx
Contact: oc@robocupathomeedu.org

RoboCup@Home EDUCATION

MathWorks®

JUPITER ROBOT