MCE 412 - Autonomous Robotics

Asst. Prof. Pınar Oğuz Ekim

Office: A 305

Webpage: http://people.ieu.edu.tr/tr/pinaroguzekim

e-mail : pinar.ekim@ieu.edu.tr

Goal of this course

- Provide an overview of problems and approaches in mobile robotics
- Probabilistic reasoning: Dealing with noisy data
- Hands-on experience with Python

Content of the Course

- Week 0: Introduction *+ Sheet 1 (Python Setup)*
- Week 1: Linear Algrebra Review *+ Sheet 2 (Linear Algebra practice in Python)*
- Week 2: Wheeled Locomotion *+ Sheet 3 (Locomotion-Differential Drive Kinematics in Python)*
- Week 3: Sensors
- Week 4: Probabilities and Bayes Review *+ Sheet 4 (Bayes Rule)*
- Week 5: Probabilistic Motion Models *+ Sheet 5 (Motion Models in Python)*
- Week 6: Probabilistic Sensor Models *+ Sheet 6 (Sensor Models in Python)*
- Week 7: Discrete Filters
- Week 8: The Particle Filter *+ Sheet 7 (Discrete Filter, Particle Filter Implementation in Python)*
- Week 9: The Kalman Filter
- Week 10: The Extended Kalman Filter *+ Sheet 8 (Extended Kalman Filter Implementation in Python)*
- Week 11: Mapping with Known Poses *+ Sheet 9 (Mapping with Known Poses in Python)*
- Week 12: Simultaneous Localization and Mapping (SLAM)
- Week 13: Working on a Project
- Week 14: Working on a Project

Reference Books

- Probabilistic Robotics, Sebastian Thrun, Wolfram Burgard and Dieter Fox
- Introduction to Autonomous Mobile Robots, Roland Siegwart and Illah R. Nourbakhsh, 2004
- Computational Principles of Mobile Robotics, Gregory Dudek and Michael Jenkin
- Handbook of Robotics, Bruno Siciliano and Oussama Khatib
- Matrix Cookbook, http://matrixcookbook.com
- Python 3, Fırat Özgül
- Hands-On Python: A Tutorial Introduction for Beginners, Andrew N. Harrington
- Introduction to Probability, Dimitri P. Bertsekas and John N. Tsitsiklis

INTRODUCTION

- Mobile Robotics is a young but a large and multidisciplinary field
- It lies at the intersection of artificial intelligence, computational vision, and robotics.
- This course will provide a foundation - a formal introduction to the field, (the rapid progress in all of mobile robotics' sub-disciplines)
- It will be an introduction to all aspects of mobile robotics, including software and hardware design considerations, related technologies, and algorithmic techniques.
- From mechanism to perception to localization and navigation, this course focuses on the techniques and technologies that enable robust *mobility*.

The study of mobile robots is an intrinsically interdisciplinary research area that involves:

- **Mechanical engineering**: vehicle design and in particular locomotive mechanisms.
- **Computer science**: representations and sensing and planning algorithms.
- **Electrical engineering**: system integration, sensors, and communications.
- **Cognitive pyschology, perception and neuroscience**: insights into how biological organisms solve similar problems.
- **Mechatronics**: the combination of mechanical engineering with computer science, computer engineering, and/or electrical engineering.

- Robot arms (manipulators):
  - Great speed and accuracy to perform repetitive tasks
  - Lack of mobility

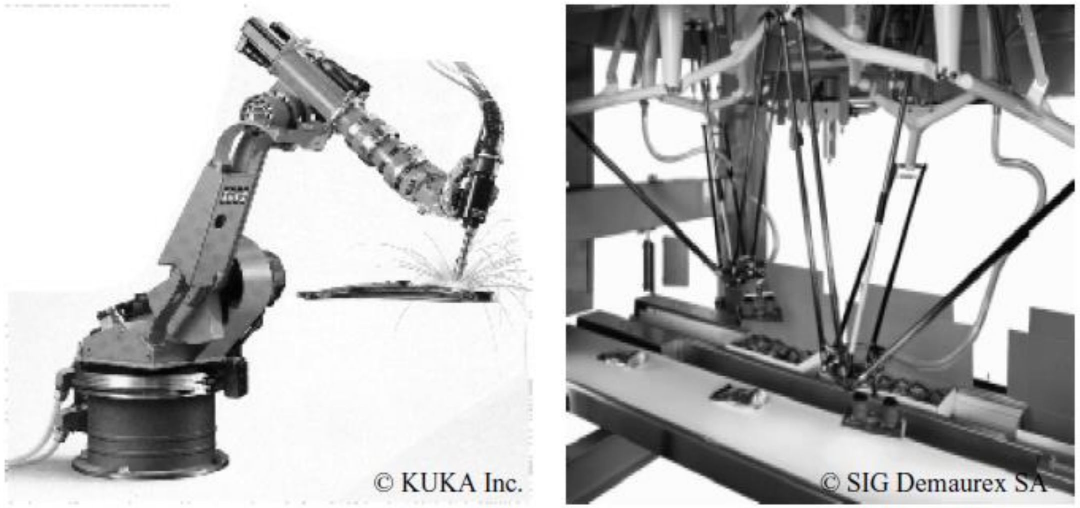


Figure 1.1
Picture of auto assembly plant-spot welding robot of KUKA and a parallel robot Delta of SIG Demaurex SA (invented at EPFL [140]) during packaging of chocolates.

What is an autonomous robot?

- The tasks of moving through space, sensing about space, and reasoning about space are fundamental problems within the study of mobile robotics.
- The words **autonomous**, as in autonomous system, and **automaton** have their roots in the Greek for *self-willed*.
- The term **robot** was introduced by Karel Capek and it has a meaning *labour* and robotnik has a meaning *workman*.
- Robots that are manufactured following the same general structure as humans are known as **antropomorphic robots** or **humanoid robots**.

- A mobile robot would be able to travel through the manufacturing plant, flexibly applying its talents wherever it is most effective.
- The focus of this course: how can a mobile robot move unsupervised through real-world environment to fulfill its tasks?
  - How should a mobile robot move? (LOCOMOTION)

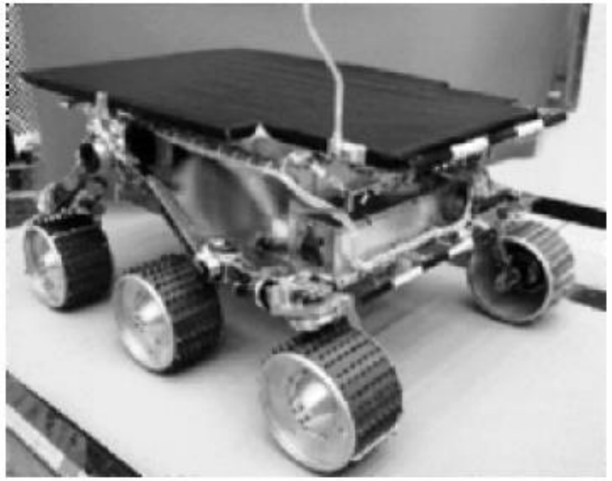


Figure 1.2
The mobile robot Sojourner was used during the Pathfinder mission to explore Mars in summer 1997. It was almost completely teleoperated from Earth. However, some on-board sensors allowed for obstacle detection. (http://ranier.oact.hq.nasa.gov/telerobotics_page/telerobotics.shtm).
© NASA/JPL

Application Areas of Autonomous Robots

- In dangerous and inhospitable environments, teleoperated systems have gained popularity

Figure 1.3
Plustech developed the first application-driven walking robot. It is designed to move wood out of the forest. The leg coordination is automated, but navigation is still done by the human operator on the robot. (http://www.plustech.fi). © Plustech.

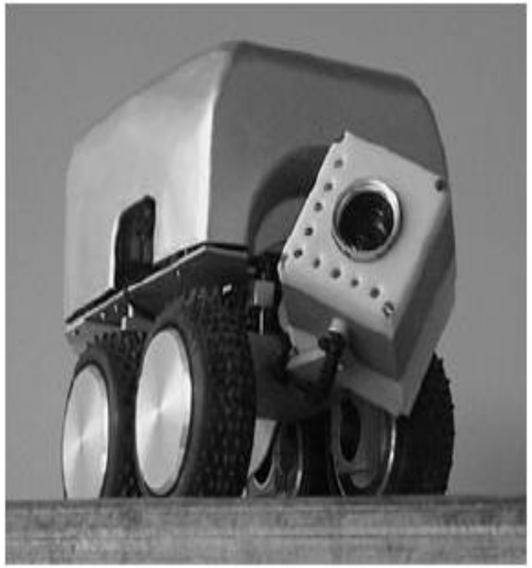


Figure 1.4
Airduct inspection robot featuring a pan-tilt camera with zoom and sensors for automatic inclination control, wall following, and intersection detection (http://asl.epfl.ch). © Sedirep / EPFL.

Figure 1.5
Picture of Pioneer, a robot designed to explore the Sarcophagus at Chernobyl. © Wide World Photos.

Figure 1.6
Picture of recovering MBARI's ALTEX AUV (autonomous underwater vehicle) onto the Icebreaker Healy following a dive beneath the Arctic ice. Todd Walsh © 2001 MBARI.

- Other commercial robots operate not where humans cannot go but share space with humans in human environments.
- They are valuable due to their autonomy. Their ability to maintain a sense of position and to navigate without human intervention is paramount.

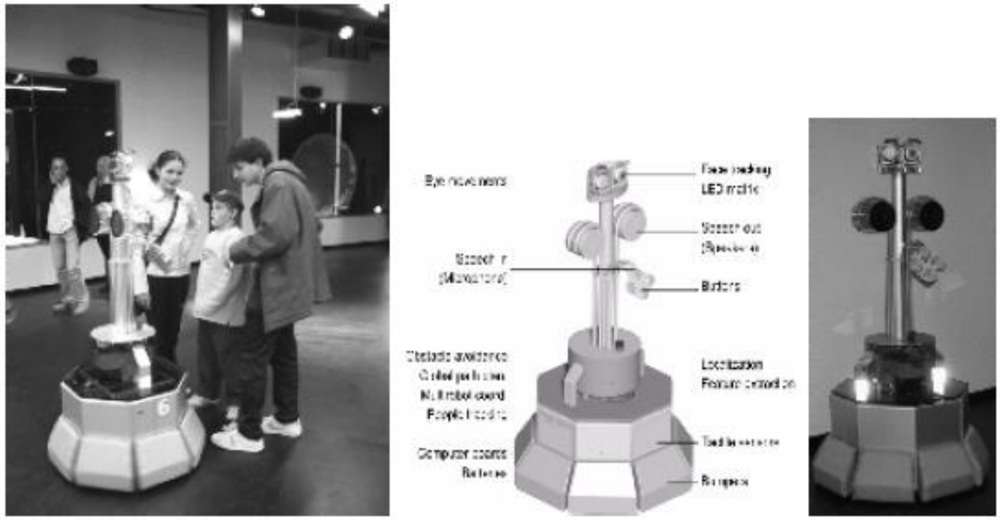


Figure 1.7
Tour-guide robots are able to interact and present exhibitions in an educational way [48, 118, 132, 143,]. Ten Roboxes have operated during 5 months at the Swiss exhibition EXPO.02, meeting hundreds of thousands of visitors. They were developed by EPFL [132] (http://robotics.epfl.ch) and commercialized by BlueBotics (http://www.bluebotics.ch).

Figure 1.8
Newest generation of the autonomous guided vehicle (AGV) of SWISSLOG used to transport motor blocks from one assembly station to another. It is guided by an electrical wire installed in the floor. There are thousands of AGVs transporting products in industry, warehouses, and even hospitals. © Swisslog.

Figure 1.9
HELPMATE is a mobile robot used in hospitals for transportation tasks. It has various on-board sensors for autonomous navigation in the corridors. The main sensor for localization is a camera looking to the ceiling. It can detect the lamps on the ceiling as references, or landmarks (http://www.pyxis.com). © Pyxis Corp.

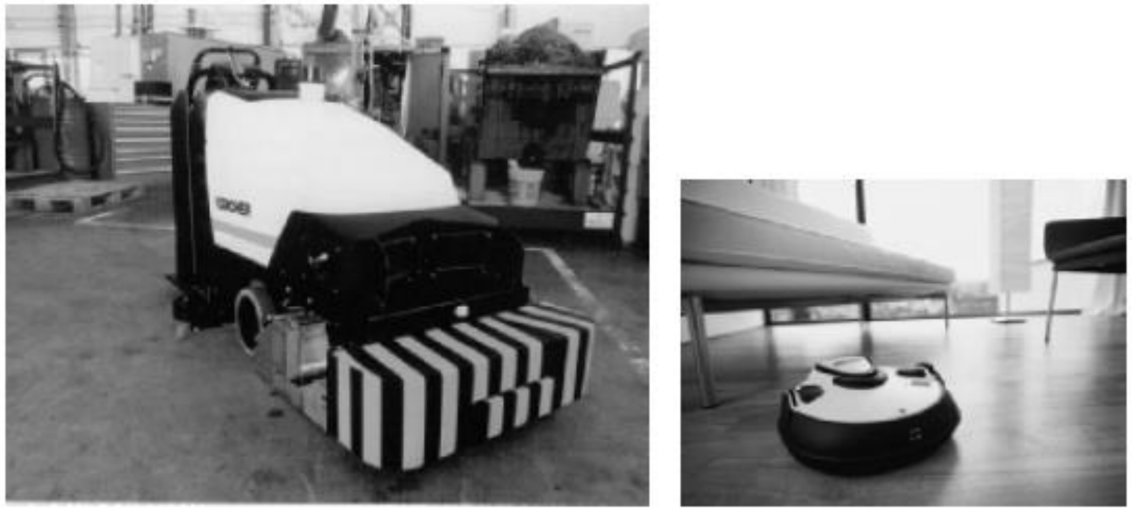


Figure 1.10
BR 700 industrial cleaning robot (left) and the RoboCleaner RC 3000 consumer robot developed and sold by Alfred Kärcher GmbH & Co., Germany. The navigation system of BR 700 is based on a very sophisticated sonar system and a gyro. The RoboCleaner RC 3000 covers badly soiled areas with a special driving strategy until it is really clean. Optical sensors measure the degree of pollution of the aspirated air (http://www.karcher.de). © Alfred Kärcher GmbH & Co.

- Research into high level questions of cognition, localization, and navigation can be performed using standard research robot platforms that are tuned to the laboratory environment.

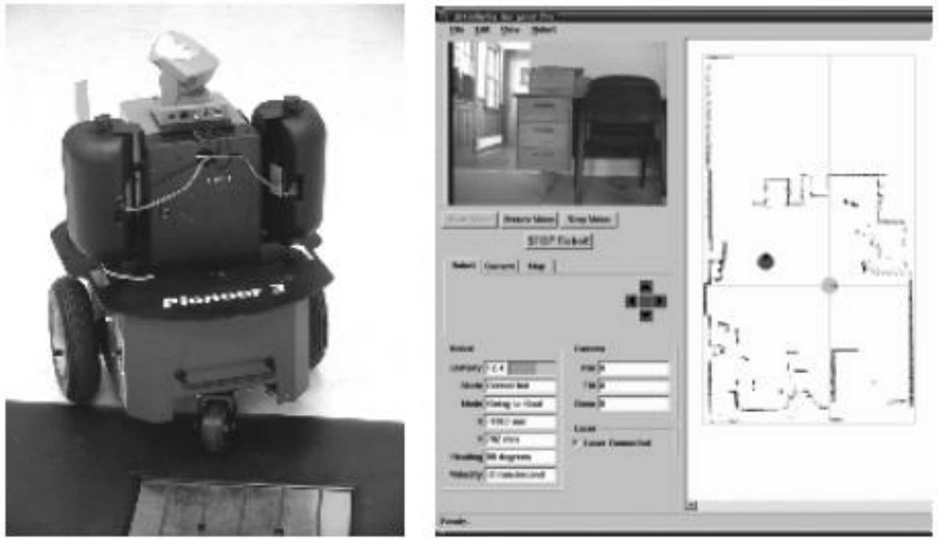


Figure 1.11
PIONEER is a modular mobile robot offering various options like a gripper or an on-board camera. It is equipped with a sophisticated navigation library developed at SRI, Stanford, CA (Reprinted with permission from ActivMedia Robotics, http://www.MobileRobots.com).

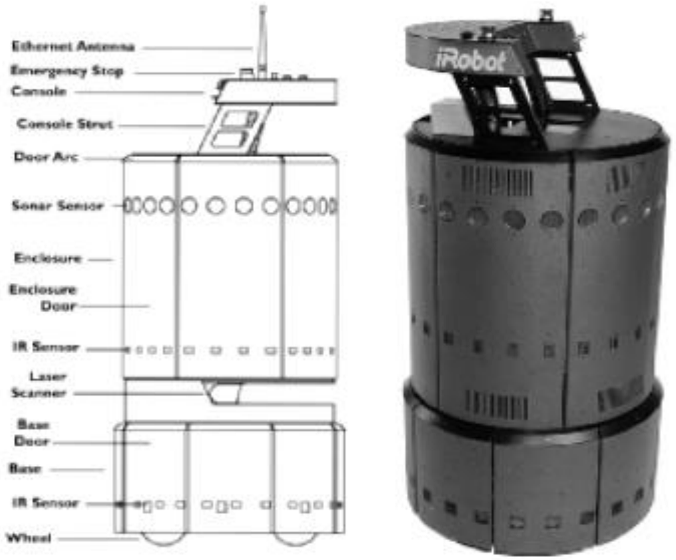


Figure 1.12
B21 of iRobot is a sophisticated mobile robot with up to three Intel Pentium processors on board. It has a large variety of sensors for high-performance navigation tasks (http://www.irobot.com/rwi/).
© iRobot Inc.

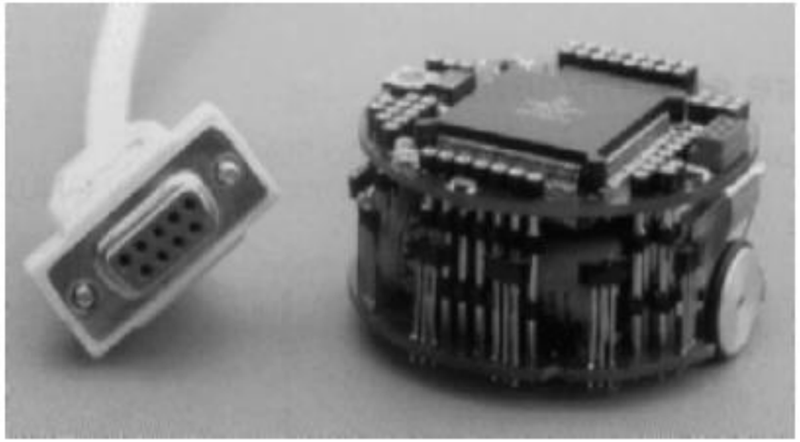


Figure 1.13
KHEPERA is a small mobile robot for research and education. It is only about 60 mm in diameter. Various additional modules such as cameras and grippers are available. More then 700 units had already been sold by the end of 1998. KHEPERA is manufactured and distributed by K-Team SA, Switzerland (http://www.k-team.com). © K-Team SA.

Figure 1.14
Alice is one of the smallest fully autonomous robots. It is approximately 2 x 2 x 2 cm, it has an autonomy of about 8 hours and uses infrared distance sensors, tactile whiskers, or even a small camera for navigation [54].

Mobile Robot Design

- A mobile robot design involves the integration of many different bodies of knowledge (interdisciplinary field)
  - To solve locomotion problems: dynamics and control theory
  - To create robust perceptual sytems: signal analysis, computer vision, sensor technologies
  - Localization and navigation : computer algorithms, information theory, artificial intelligence and probability theory

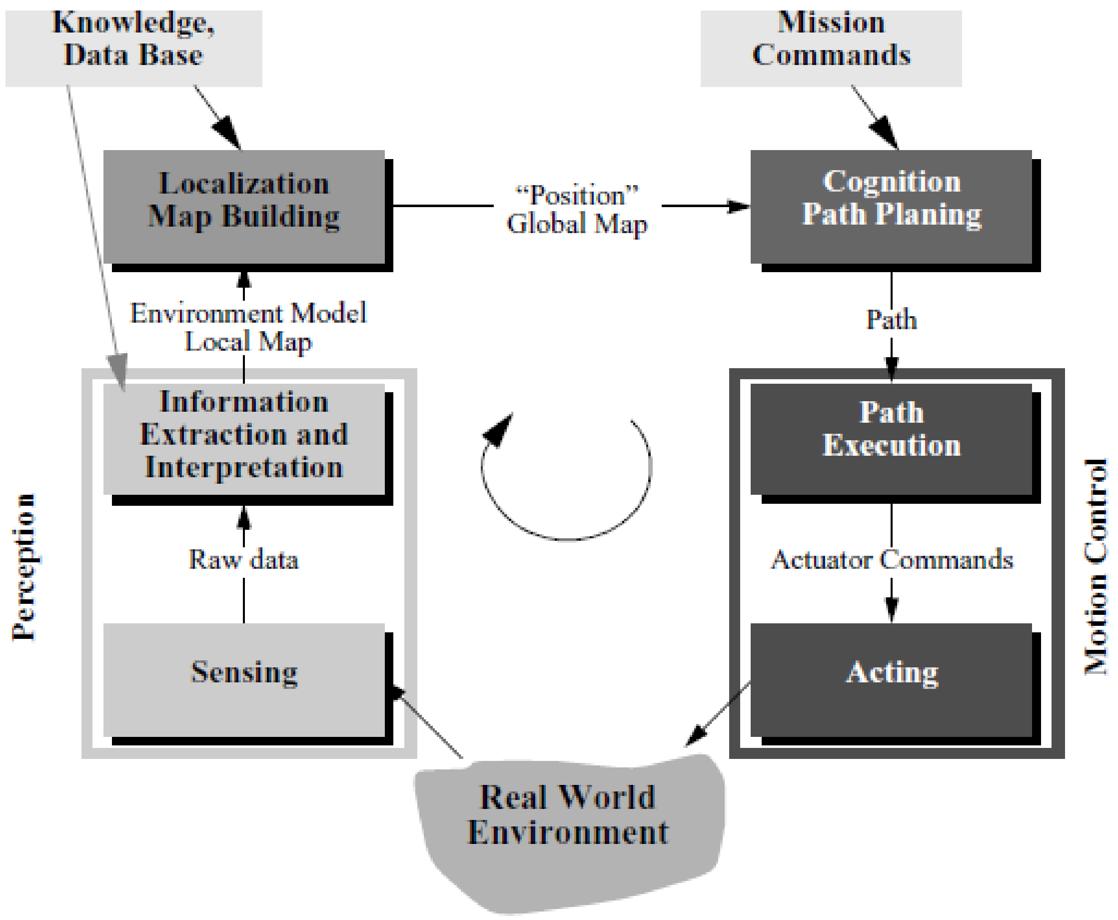


Figure 1.15
Reference control scheme for mobile robot systems used throughout this book.

Fundamental Problems

- **Path Planning :** Is it possible to get the robot from one configuration to another while remaining within C_{free}?
- **Localization :** How can the robot determine its state if it has local measurements of C_{free}?
- **Robot perception or sensing :** How can the robot determine which parts of its environment are occupied?
- **Mapping :** How can the robot determine C_{free}, assuming it always knows where it is?
- **Simultaneous localization and mapping (SLAM) :** How can the robot determine its pose and C_{free} if it knows neither?

- The point (x,y) fully describes the state (**pose or configuration**) of the point robot
- The set of valid poses of the robot are known as its **free space**, C_{free}.
- Some states are not valid; **obstacles**.

Path Planning for a Point Robot

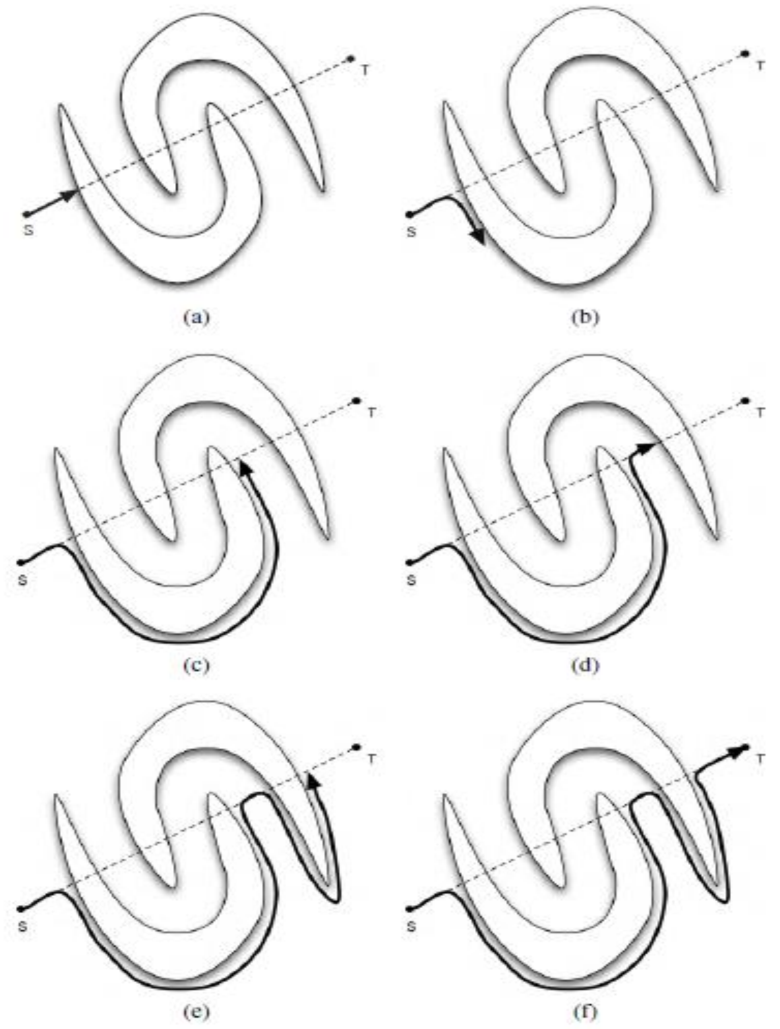


Figure 2.1. Bug algorithm example. The robot moves from S to T in the sequence (a)–(f). The dashed line shows the direct path, and the solid line shows the path followed by the robot.

- The robot starts at configuration S=(a,b), and we wish to move it to configuration T=(c,d).
- **Bug Algorithms**: The **Bug2** algorithm is said to be **complete** in that it is certain to find a path to the goal if such a path exists and reports failure otherwise.

```
Visualize a direct path ST from the start S to the goal T .
while the goal T is not achieved, do:
  begin
    while the path ST to the goal is not obstructed, do
      begin
        move towards the goal along the path ST ,
        if the path is obstructed then
          begin
            mark the current location as P circumnavigate the
            object until the robot either:
            (a) hits the line ST at a point closer to T than P and
                can move towards T , in which case the robot
                follows ST ;
            (b) returns to where P in which case T is
                unreachable.
          end
      end
  end
```

Figure 2.2. The Bug2 algorithm.

Localization for a Point Robot

- When the robot moves from (x, y) to some point $(x + \Delta x, y + \Delta y)$ how do we know where the robot actually is?
- The robot's wheels will slip each time we accelerate or deaccelarate the vehicle and over these small errors will accumulate → causing any estimate we have of the robot's position to be inaccurate.
- Maintaining an estimate of the pose of the vehicle requires reference to external features or events that can be used to overcome the errors associated with movement.
- The robot requires some technique to combine the **local estimate** of the robot's position (obtained through an understanding of the motion noise process) and the **current measurements** (which are likely corrupted by some sensor noise process)

Sensing for a Point Robot

- In order to deal with the realities of navigation and localization, a robot requires some mechanism to sense its environment.
- Suppose that the point robot has a single range sensor that it can point in any direction and that the sensor returns the distance to the first object that it encounters along that direction and it returns infinity if no obstacle exists in that direction.

- Let us assumme that the robot’s global orientation, like its position, is known.
- Suppose that the robot is stationary in a static environment and fires its sensor in a number of different directions. The sensor obtains a collection of measurements $\{(\theta_i, d_i)\}$.
- Define the origin of a coordinate system at the robot’s current location, and define θ as the angle between the direction in which the robot is moving and the sensor direction.

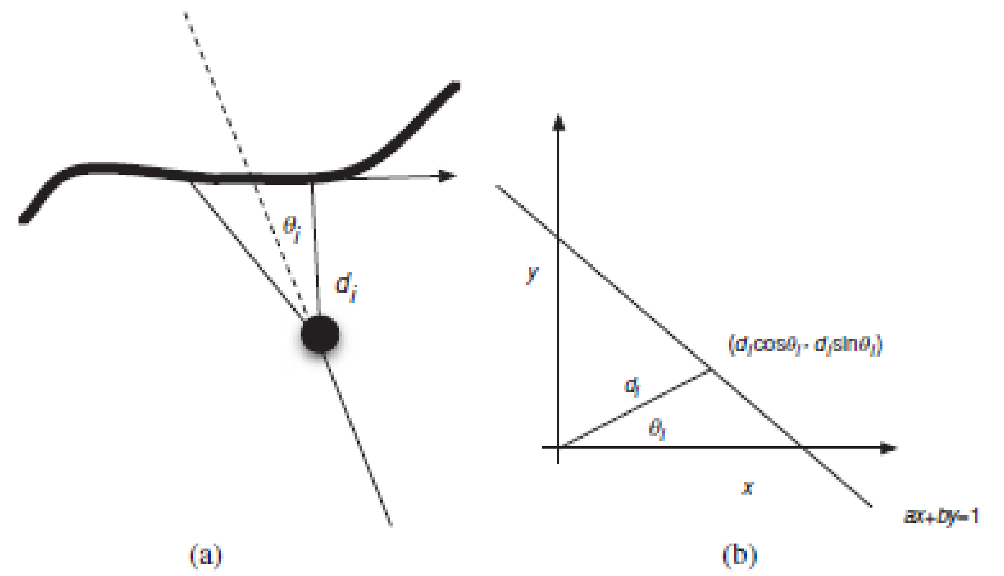


Figure 2.3. Equipping a point robot with a sensor. (a) The robot obtains range measurements d_i in different directions (given by θ_i). (b) These measurements can be integrated to estimate the orientation of the object in front of the robot.

$$\begin{bmatrix} d_1\cos(\theta_1) & d_1\sin(\theta_1) \\ d_2\cos(\theta_2) & d_2\sin(\theta_2) \\ \cdots & \cdots \\ d_n\cos(\theta_n) & d_n\sin(\theta_n) \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \cdots \\ 1 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} d_1\cos(\theta_1) & d_1\sin(\theta_1) \\ d_2\cos(\theta_2) & d_2\sin(\theta_2) \\ \cdots & \cdots \\ d_n\cos(\theta_n) & d_n\sin(\theta_n) \end{bmatrix}$$

$$X = (A'A)^{-1}A'B$$

$$\mathbf{X} = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} 1 \\ 1 \\ \cdots \\ 1 \end{bmatrix}$$

Mapping for a Point Robot

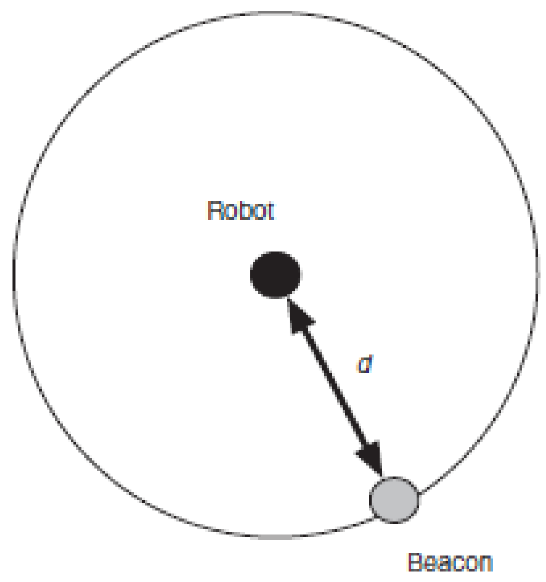
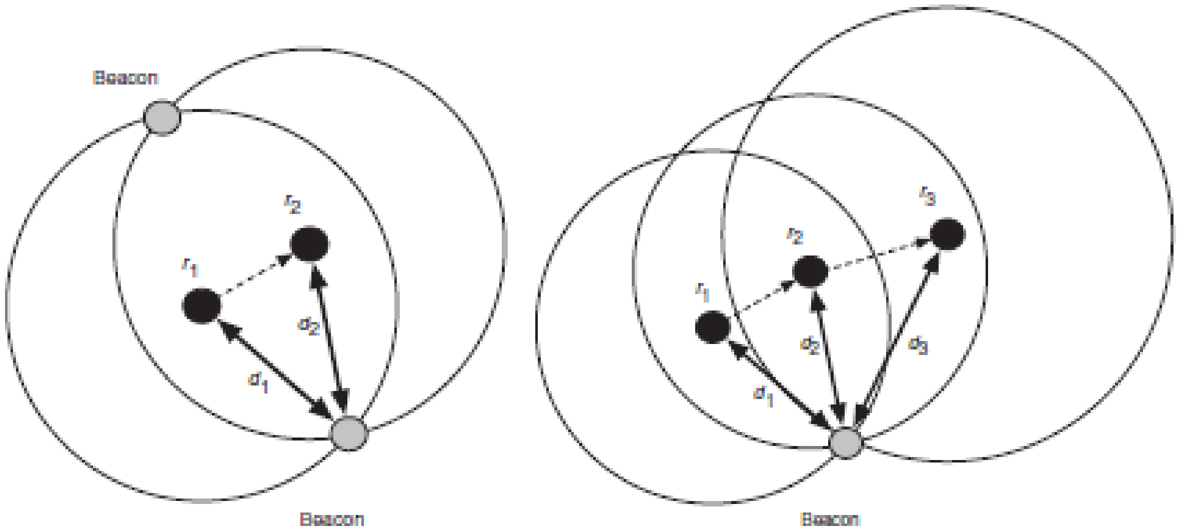


Figure 2.4. Range information. When a robot determines that it is a distance d from a beacon, that localizes the beacon to a circle with the robot at its center with radius d.



(a) After one motion

(b) After two motions

Figure 2.5. When the robot moves (indicated by the dashed line) and senses the beacon again (a), it obtains an additional set of constraints as to the location of the robot. Each set of constraints limits the possible location of the robot to a circle. The intersection of these two constraints (circles) restricts the location to one of two locations. When the robot moves a second time (b), a third constraint is introduced, and the intersection of three circles obtains a single point as the location of the beacon.

SLAM for a Point Robot

- The mapping algorithm will work for short-duration motions, it will fail for longer motion sequences as the estimate of the robot's position will become corrupted, which in turn will corrupt the robot's map of the environment.
- Readings of the sensor must be used to continually refine the map and the estimate of the robot's position.