MOBILE ROBOTS 6 - Localisation

Prof. Francesco Mondada
Laila El-Hamamsy

EPFL

2024-2025

Bibliography and Sources

Elements of Robotics (ch.5, ch.8 and 9), M. Ben-Ari, F. Mondada, Springer, 2017.

Springer Handbook of Robotics (ch. 45, 46 and 60), B. Siciliano, and O. Khatib (Eds.), 2nd edition, Springer, 2016.

Probabilistic Robotics by Thrun, Burgard, and Fox, MIT Press, 2005

Mobile Robots - EPFL - J.-C. Zufferey, Felix Schill, 2013.

Mobile Robot Localisation

The localization problem is the problem of **estimating a robot's pose** relative to a map of its environment.

Unfortunately, the pose can usually not be sensed directly, i.e., most robots do not possess a noise-free sensor for measuring pose (even GPS is not noise-free and does not directly provide orientation).

Usually, a single sensor measurement is insufficient to determine the pose

-> data from various sources need to be combined (sensor fusion),

-> the robot often has to integrate over time to determine its pose.

Bayesian filtering (probabilistic algorithms) is widely used to recursively merge motion commands and sensor data into an updated pose.

A Taxonomy Of Localisation Problems

Local vs. Global Localisation

- ***Position tracking*** addresses the problem of accommodating the local uncertainty of a robot whose initial pose is known.
- ***Global localization*** is the more general problem of localizing a robot from scratch.
- ***Kidnapping*** is a localization problem in which a well-localized robot is “secretly” deported somewhere else without being told.

Static versus dynamic environments

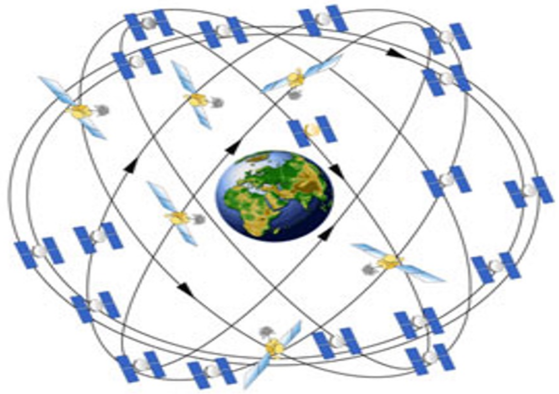
The difficulty of the localization problem is clearly dependent of the degree to which the environment changes over time.

Passive versus active approaches

- Passive localization approaches are filters: they process data acquired by the robot but do not control the robot.
- Active techniques control the robot during localization with the purpose of minimizing the robot’s uncertainty (directed exploration).

Table of Contents

Global Positioning System (GPS) - Facts



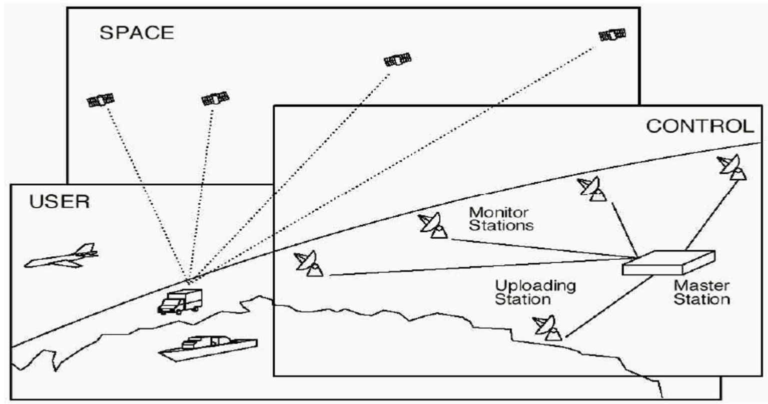
www.gps.gov

- Developed for military use by the US (NAVSTAR)
- Became then accessible for commercial applications
- 24 satellites (including three spares) orbiting the Earth every 12 hours at a height of 20’190 km
- 4 satellites are located in each of six planes inclined by 55° with respect to the plane of the Earth’s equator
- Location of any GPS receiver is determined through a time of flight measurement

Most GPS receivers have a position accuracy within 20 m (typ. 10 m) in the horizontal plane and 45 m in the vertical plane, depending on the number of satellites within line of sight and multipath issues.

The update rate is typically between 1 and 10 Hz only.

The European system, Galileo, has similar features in its Galileo Open Service mode. Better precision is available for commercial and governamental purposes.



GPS Technical Challenges

Real time update of the exact location of the satellites:

- monitoring the satellites from a number of widely distributed ground stations
- master station analyses all the measurements and transmits the actual position to each of the satellites

Time synchronization:

- ultra-high precision time synchronization is extremely important
- roughly 0.3m/ns => position accuracy proportional to precision of time measurement
- atomic clocks on each satellite

Precise measurement of the time of flight

- the receiver correlates a pseudocode with the same code coming from the satellite
- the delay time for best correlation represents the time of flight
- quartz clock on the GPS receivers are not very precise
- the range measurement with four satellites allows to identify the three values (x, y, z) for the position and the clock correction ΔT

Interferences with other signals, reflections

Already one of the key sensors for outdoor mobile robotics but not applicable for indoor robots

Beacon Based Positioning

Beacon-based navigation has been used since the humans started to travel

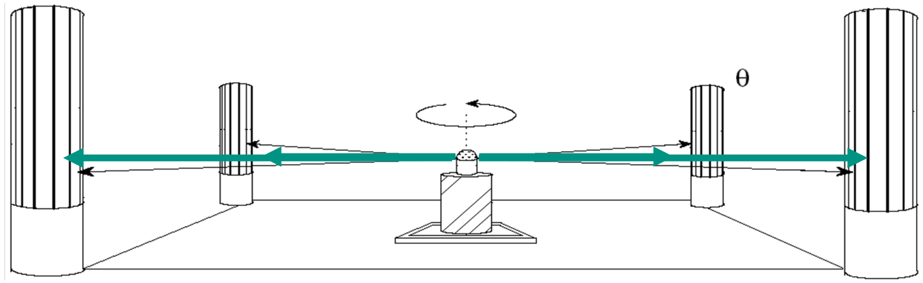
- Natural beacons (landmarks) like stars, mountains or the sun
- Artificial beacons like lighthouses

Beacons = signaling devices of precisely known position.

Major drawback with the use of beacons indoors:

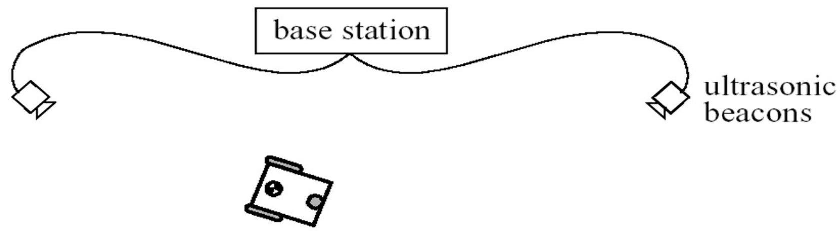
- Beacons require changes in the environment -> costly
- Limit flexibility and adaptability to changing environments

Passive Beacons : optical / retroreflective of known pos.



Distance and heading of at least two beacons are measured (e.g. using a scanning laser range finder)

Active Ultrasonic Beacons

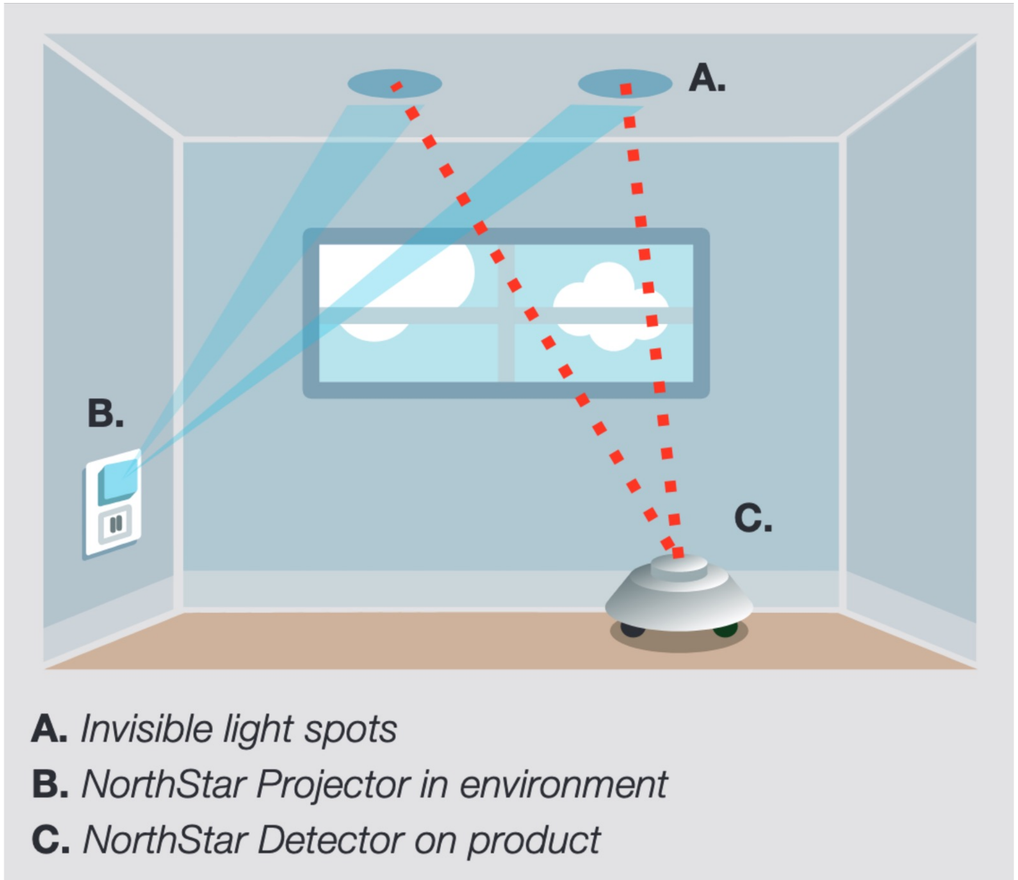


Robots must know emitter locations & deduce their pos based on TOF. Accuracy depends on: 1) Precise location of the beams 2) Time synchronization (e.g. RF/IR) 3) Measurement of TOF

Beacon Based Positioning for home use

Example of system: NorthStar (Evolution Robotics / iRobot)

- IR beacons toward the ceiling: invisible
- One projector: minimally invasive
- Very specific signal: easy to decode



Example of navigation system for home

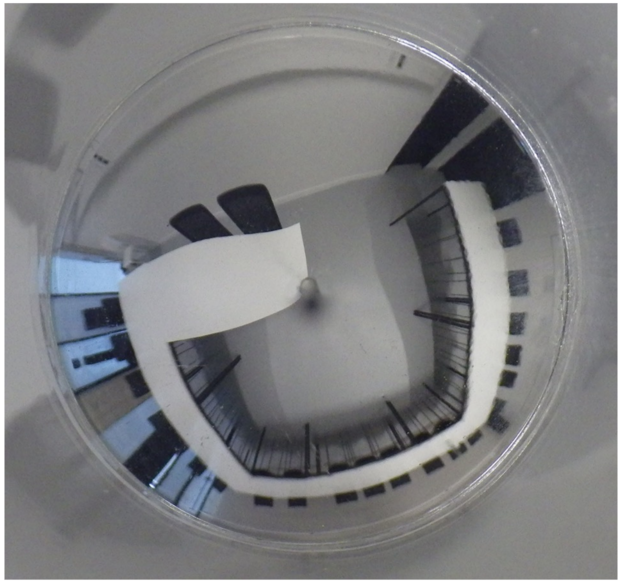
Example of complete system: iRobot Braava using NorthStar

Extremely few systems using this technology: cheaper to localize by vision

Vision-based navigation systems

Several approaches: looking to the ceiling or all around (360 vision)

Samsung: top looking camera



dyson: 360 camera

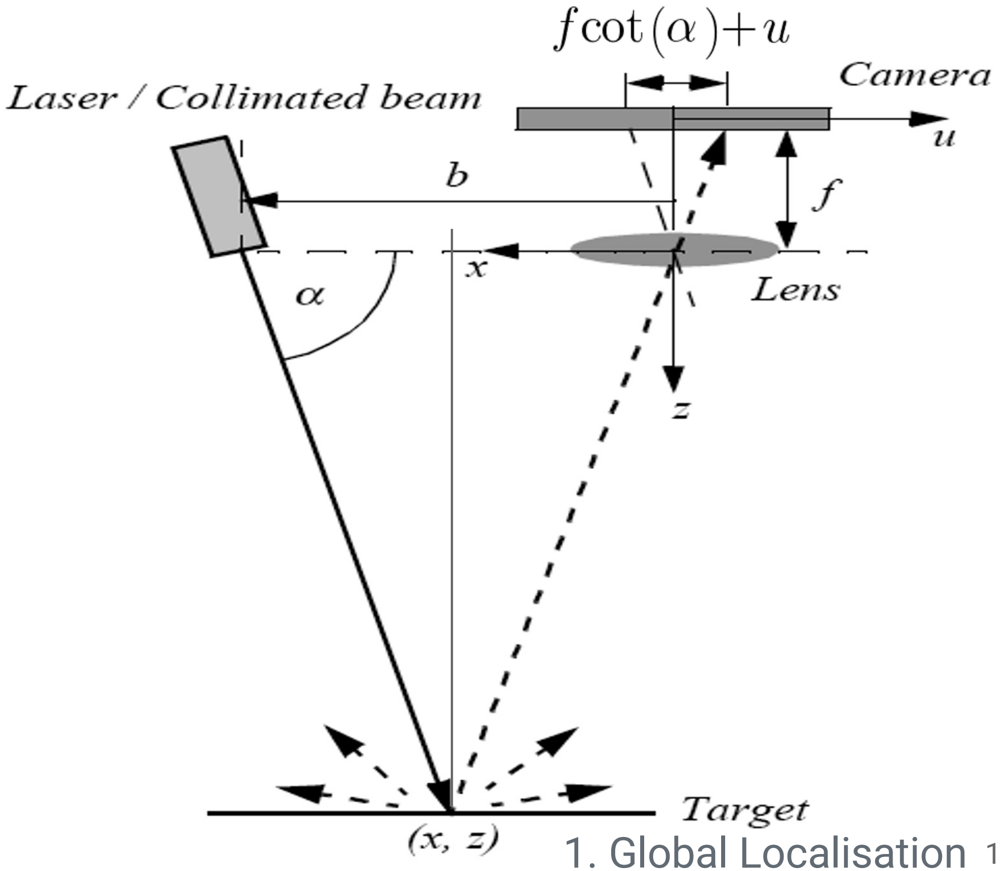
Triangulation - Laser / Collimated Beam

Independent from object detected

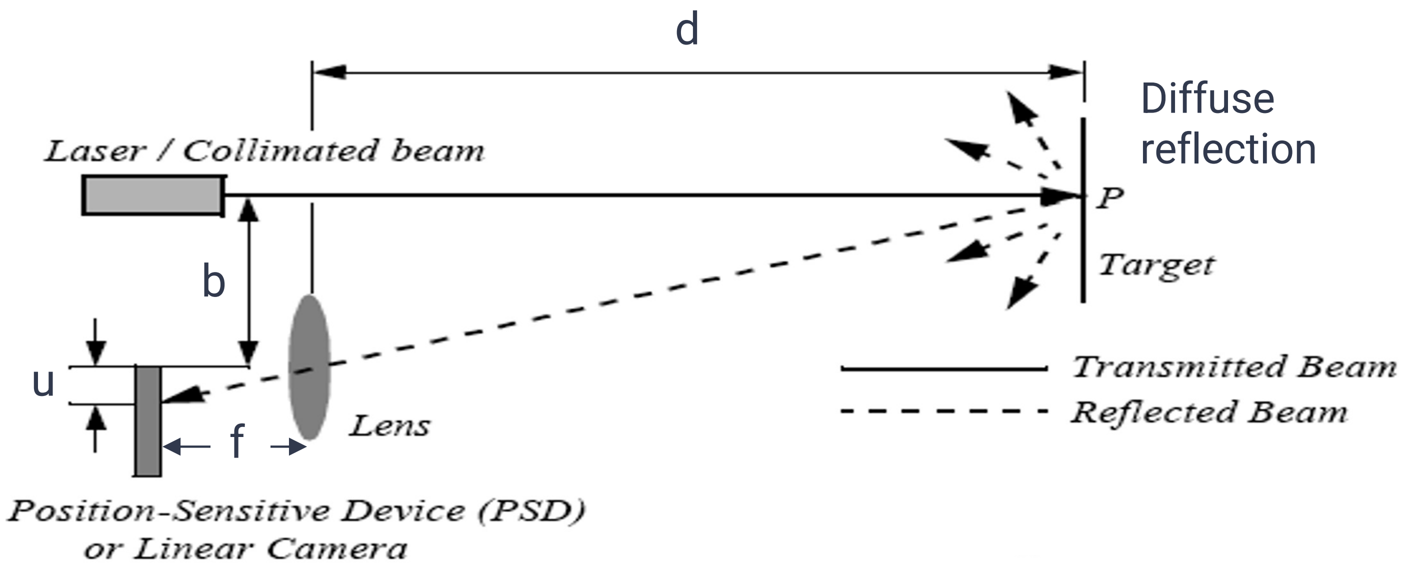
One dimensional schematic of the principle.

From the figure, geometry of similar triangles shows that:

$$z = \frac{f b}{2 u} = \frac{f b}{f \cot(\alpha) + u}$$



Triangulation - PSD or Linear Camera



$$d = f\,\frac{b}{u}$$

- Both collimated LED or lasers are used to emit a visible or infrared (IR) beam.
- Position-Sensing Detectors (PSD) are the simplest and fastest way to measure the position of the centroid of a light spot. PSDs are based on a lateral-effect diode that converts the position of incident radiation into signal currents that are directly proportional to the light's position on the active area of the detector.
- Another approach consists of using a linear camera.

Triangulation - application example

Roborock

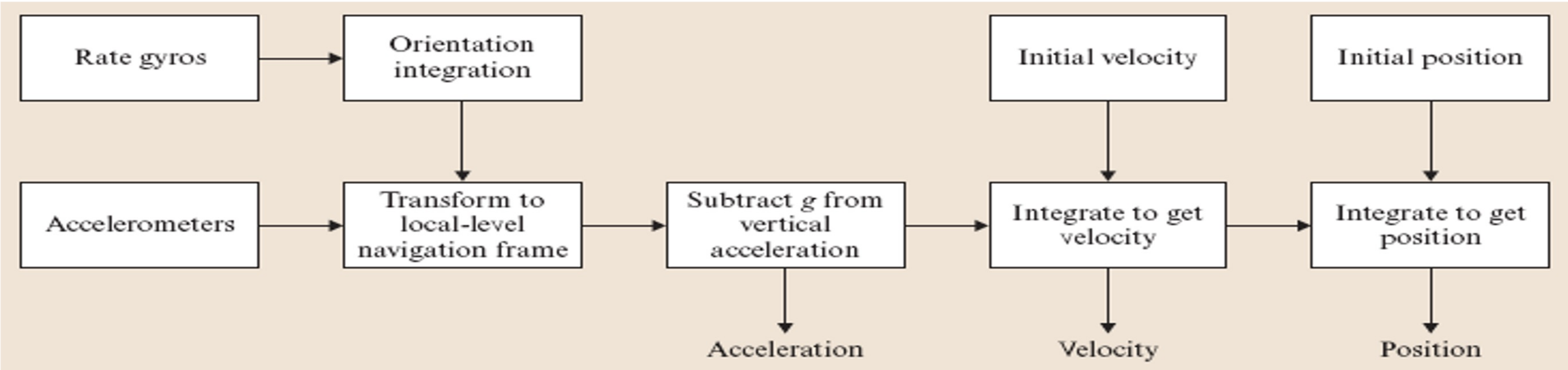
- Laser sensor
- Triangulation
- Local map

Table of Contents

Inertial Measurement Unit (IMU)

AN IMU is a device that include gyroscopes (measuring angular speed) and accelerometers (measuring linear acceleration). Based on IMU one can maintain a 6 DOF estimate of the vehicle pose: position (x,y,z) and orientation (roll, pitch, yaw)

This requires integrating information from the sensors in real time.



IMU/INS are extremely sensitive to measurement errors => an external source of information such as GPS and/or magnetometer is often required to correct this (cf. lesson on uncertainties for more information on sensor fusion).
List of IMU/INS products: http://damien.douxchamps.net/research/imu/

Wheel / Motor Incremental Encoders

Measure position or speed of the wheels or steering.

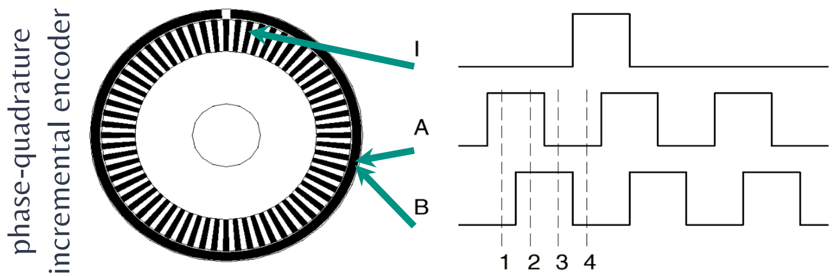
- Measurement types: resistive, inductive, magnetic, optical, etc.
- Optical method is cheap, robust and provides simple digital output.
- Many systems based on magnets and hall sensors (no need of light protection)

Integrate wheel movements to get an estimate of the position -> odometry

Encoders are proprioceptive sensors thus the position estimation in relation to a fixed reference frame is only valuable for short movements.

Incremental Encoders (e.g. Optical)

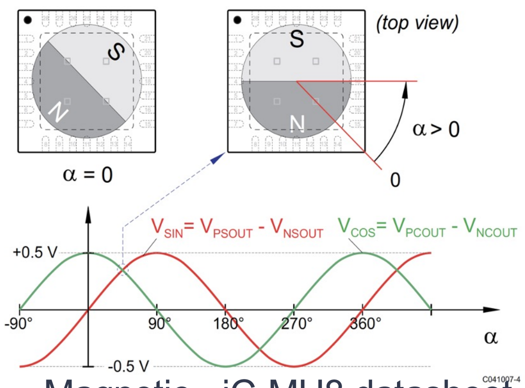
Two signals (90 degrees shift) for direction detection



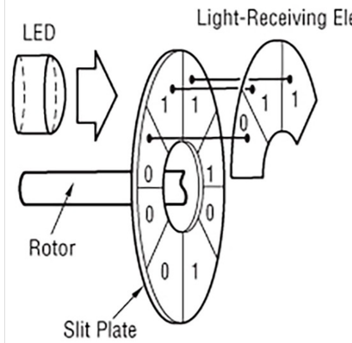
State	Ch A	Ch B
S_1	High	Low
S_2	High	High
S_3	Low	High
S_4	Low	Low

Typical resolutions: 64 - 2048 increments per revolution.

Absolute Encoders (e.g. Magnetic, Optical)



Magnetic - iC-MH8 datasheet

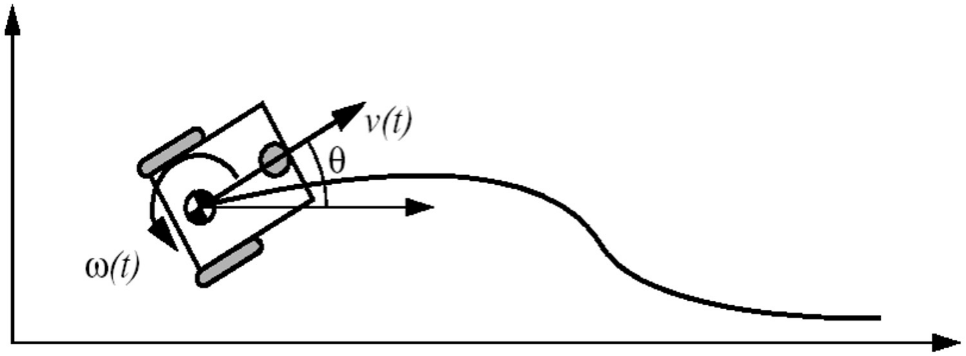


Sensor Output	Angle
011	45°
001	90°
000	135°
100	180°
010	225°
101	270°
110	315°
111	360°

Optical

Odometry of a Differential Drive Robot

$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \qquad p' = p + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{bmatrix}$$



The small displacements (translation and rotation) in the robot frame can be estimated from the encoders as follows:

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2} \qquad \Delta\theta = \frac{\Delta s_r - \Delta s_l}{b}$$

b : distance between the two wheels.

Expressed in the external fixed frame:

$$\Delta x = \Delta s \cos(\theta + \Delta\theta / 2)$$

$$\Delta y = \Delta s \sin(\theta + \Delta\theta / 2)$$

Motion Models $P(X_T \mid U_T, X_{T-1})$

exercice

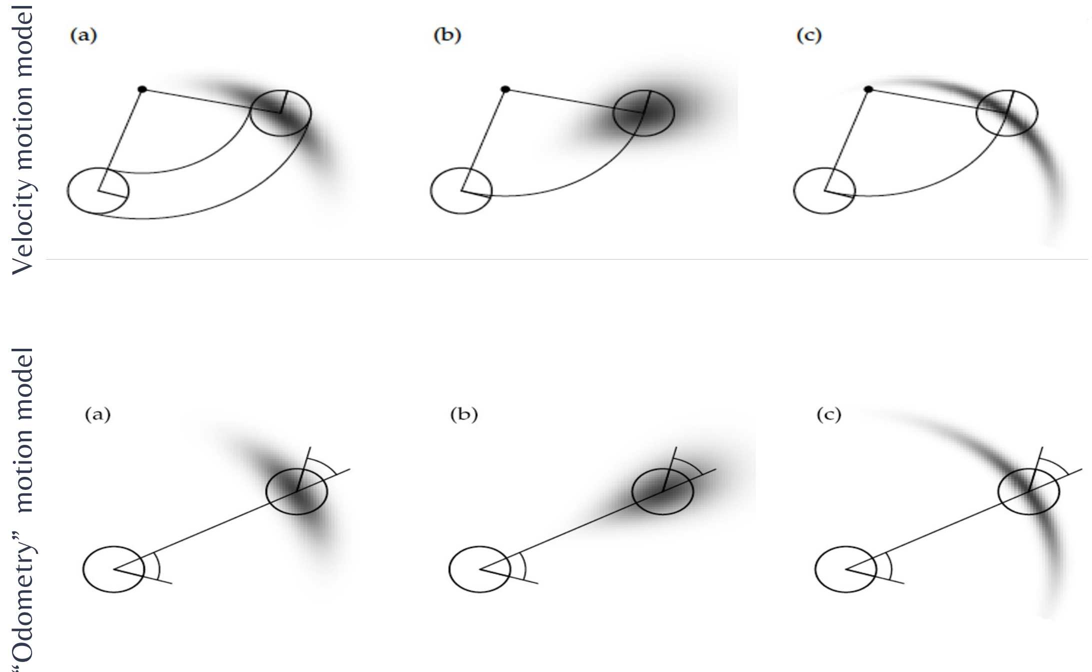
The *motion model* provides a way of estimating in a probabilistic sense the effect of motion on the state.

In mobile robotics, this is usually achieved using odometry (note that encoders are not part of the measurement z !) or low-level speed control.

Examples:

–Velocity motion model: the robot is controlled using two velocities v and ω and follows circular trajectories during Δt

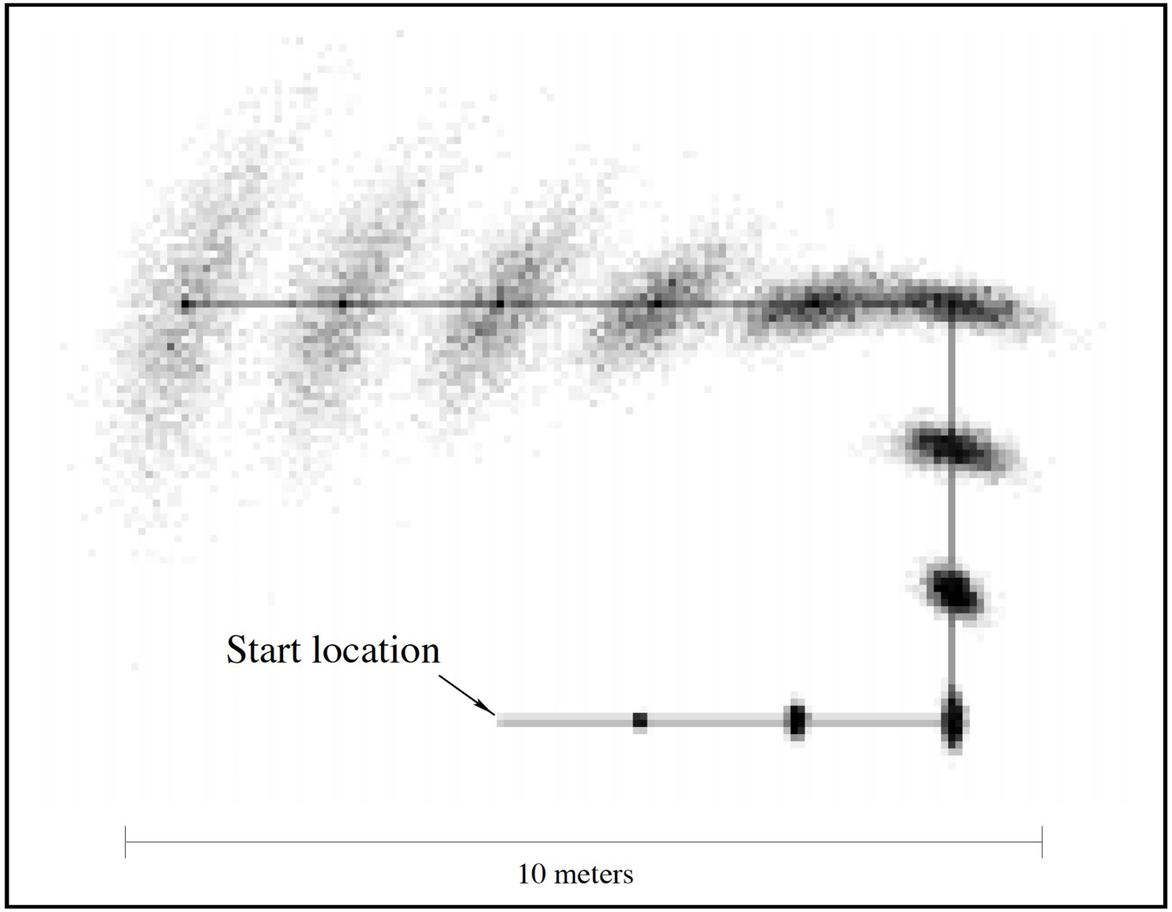
–"Odometry" motion model: u_t is given by the difference between x_t (as provided by odometry) and x_{t-1} ; u_t is transformed into a sequence of 3 steps: rotation, linear motion, rotation



Motion models with various noise parameter settings

Effect of Non Deterministic Errors

This graph shows how the odometric uncertainty grows along a straight or curved path. The error is bigger in angle than straight



Fox, D., Burgard, W., Dellaert, F., & Thrun, S. (1999). Monte carlo localization: Efficient position estimation for mobile robots. *AAAI/IAAI*, *1999*(343-349), 2-2.

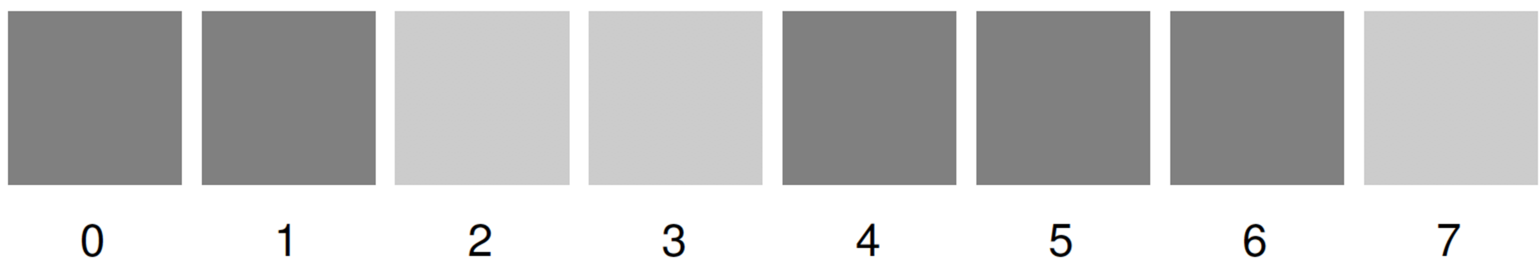
Table of Contents

Example

exercice

Robot moving in a known map, simplified here to 8 positions. The robot has the map and can sense the color of the position (for instance a ground sensor).

Map:

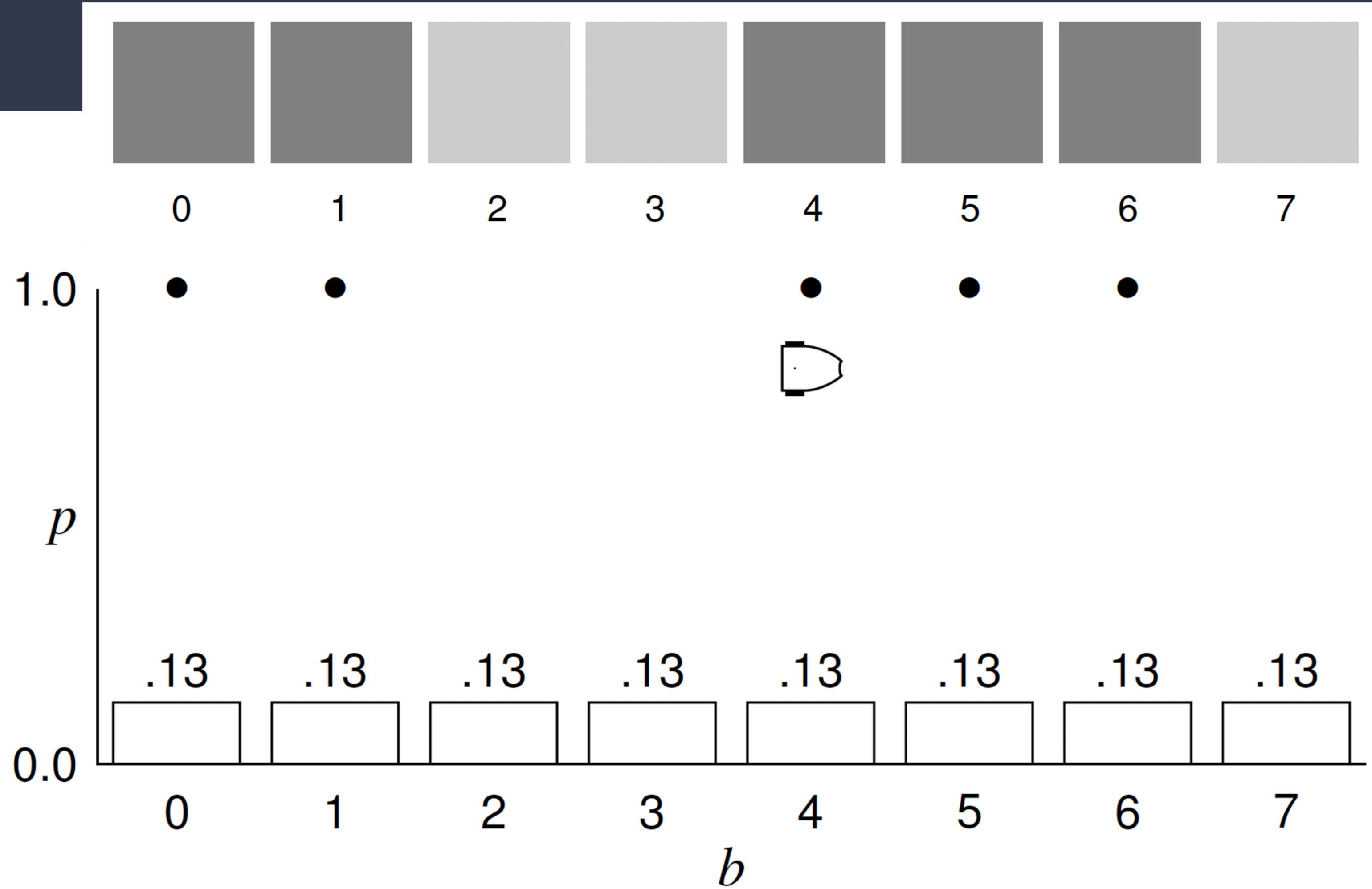


Example

Robot moving in a known map, simplified here to 8 positions. The robot has the map and can sense the color of the position (for instance a ground sensor).

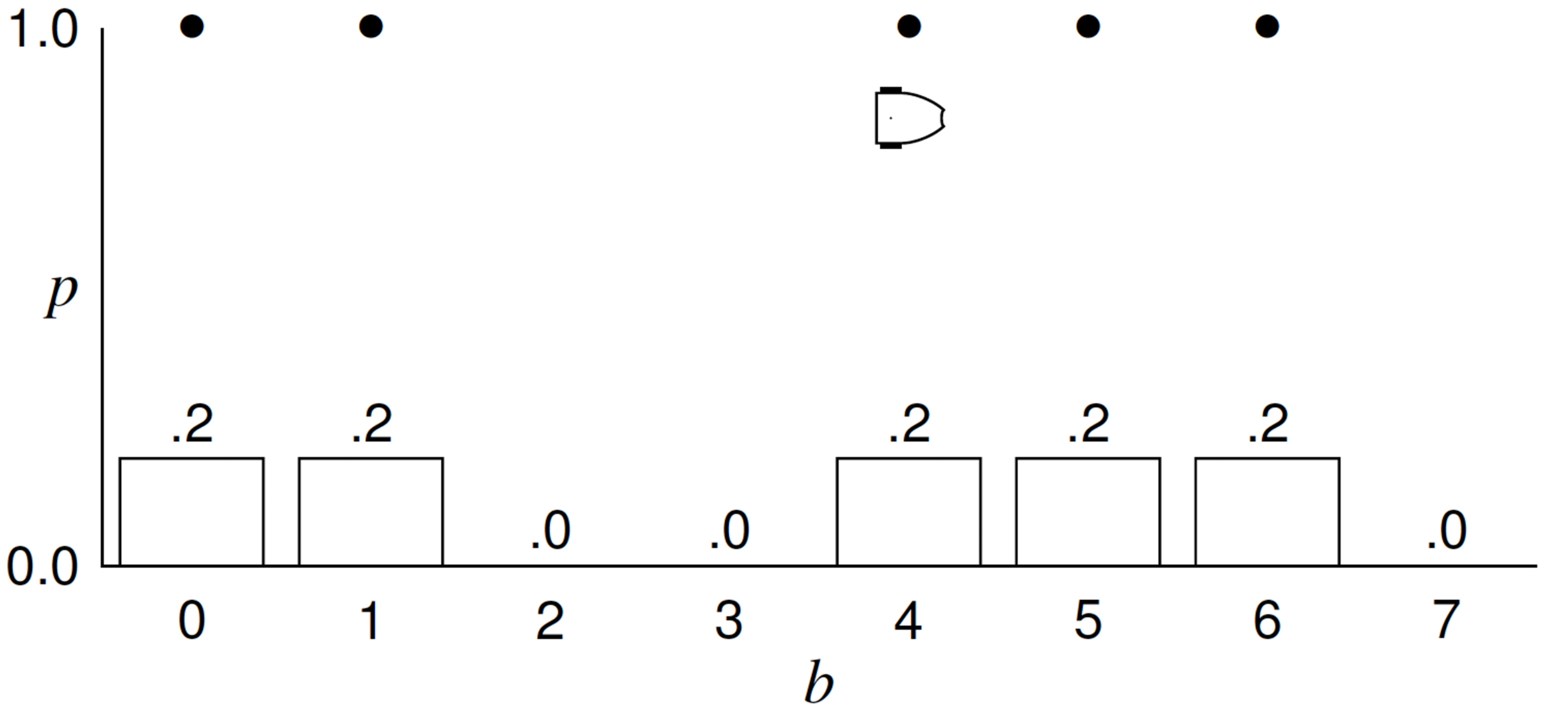
Sensing

Start, no info on position. Probability p same in all positions b.



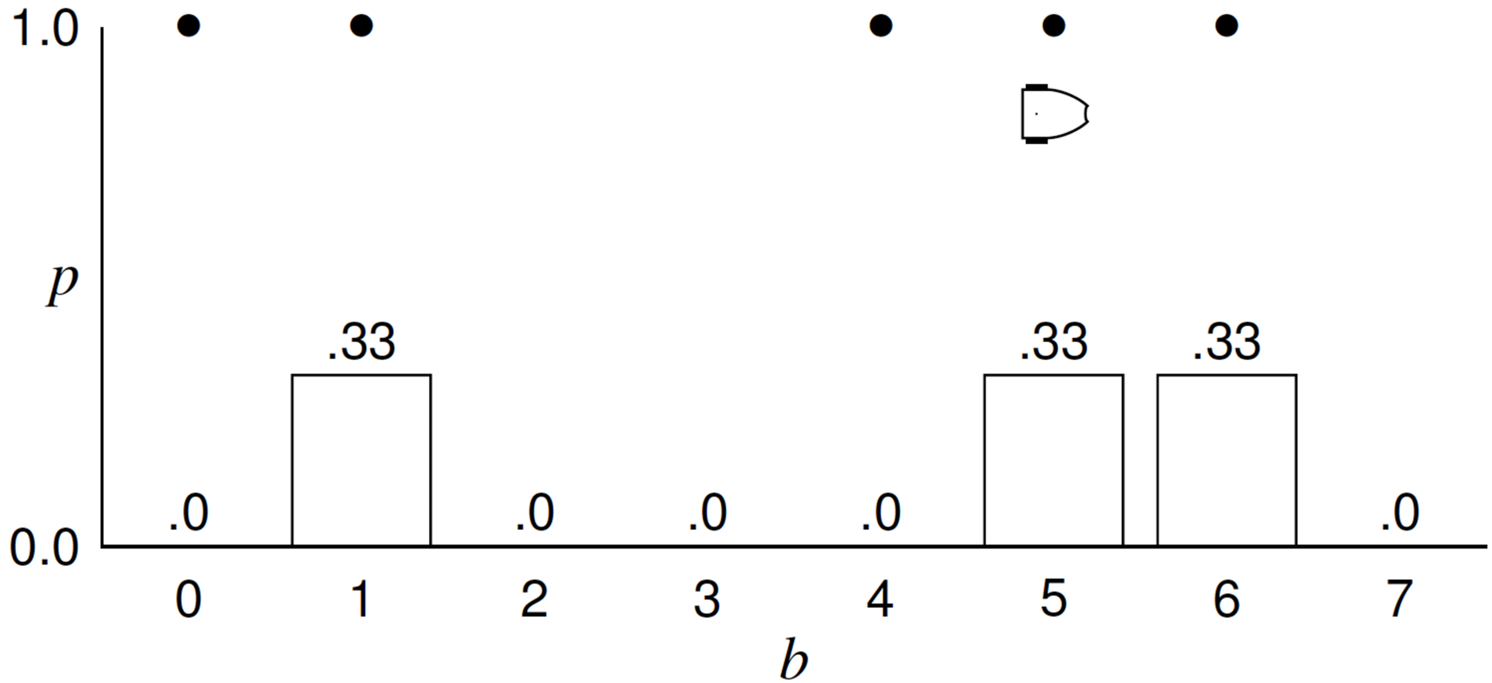
Sensing

Robot senses the ground and get an information on potential position:



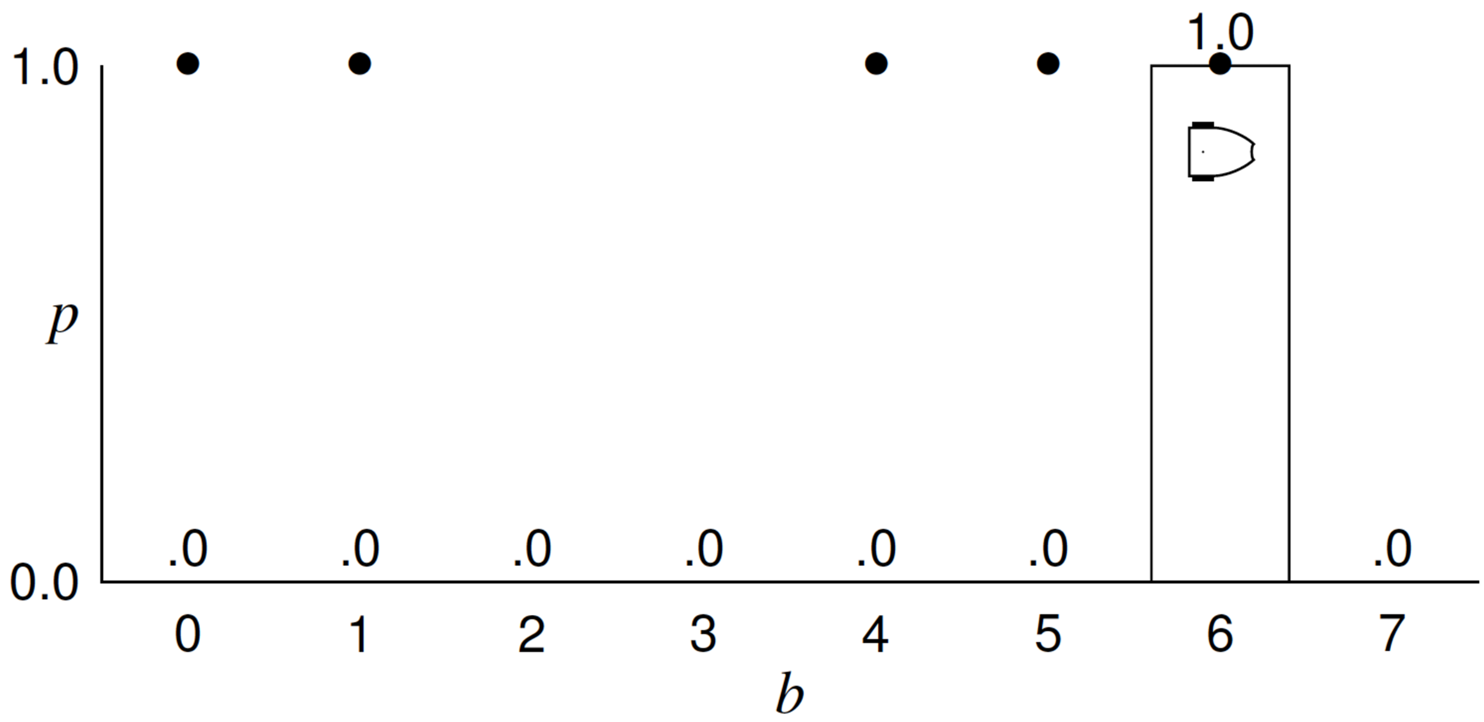
Action

Robot moves and senses again:



After Last Move

Robot moves and senses again:



How to Compute the Probability?

Sense:

p(b|z)=p(z|b).p(b)

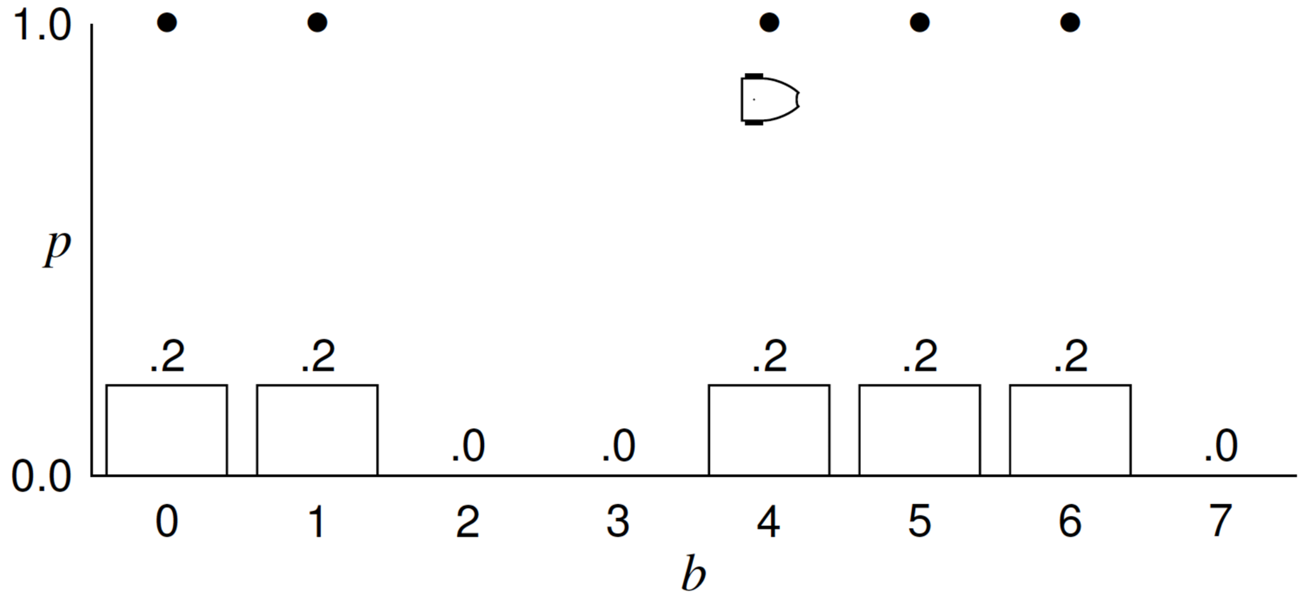
z: measurement

b: state

p(z|b) = 1
if map = measure

p(z|b) = 0
if map ≠ measure

then normalization



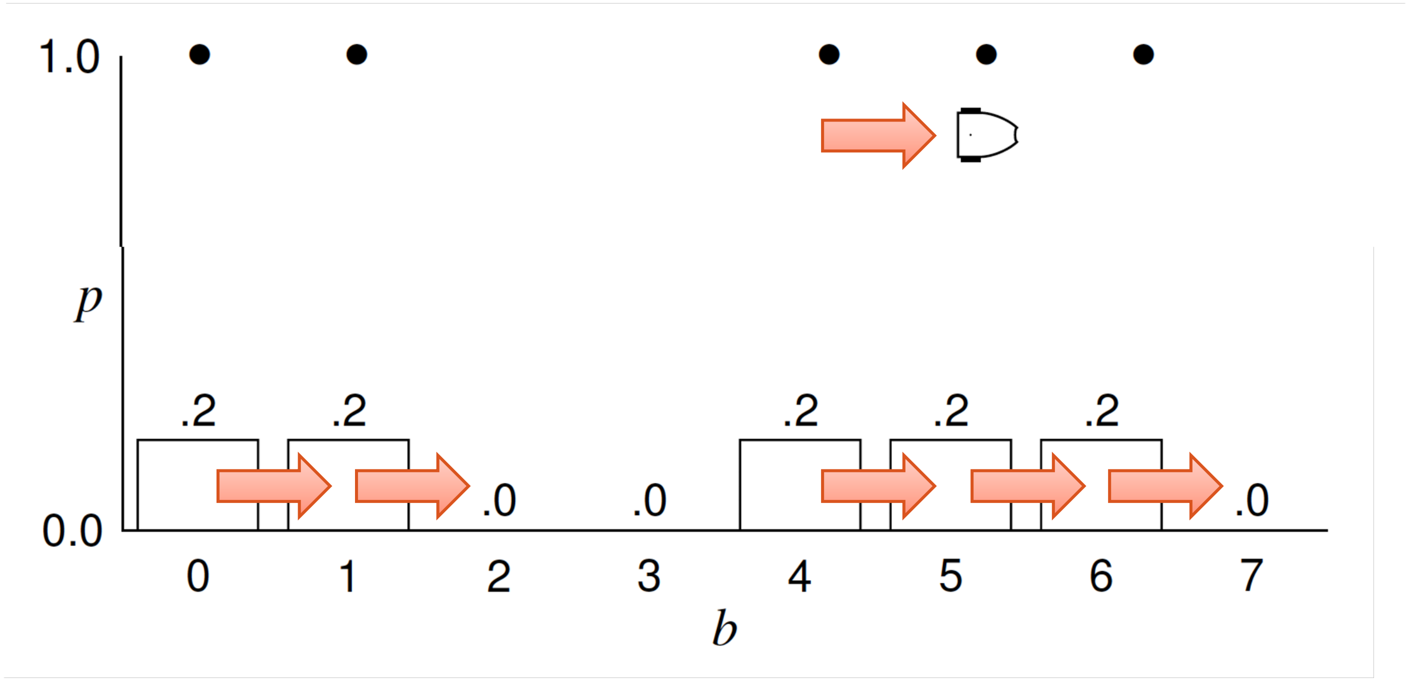
How to Compute the Probability?

Movement:

$p(b_t)=p(b_t|u,b_{t-1}).p(b_{t-1})$

u: action

$p(b_t|u,b_{t-1})$: shift



Uncertainty in Sensing

Measure:

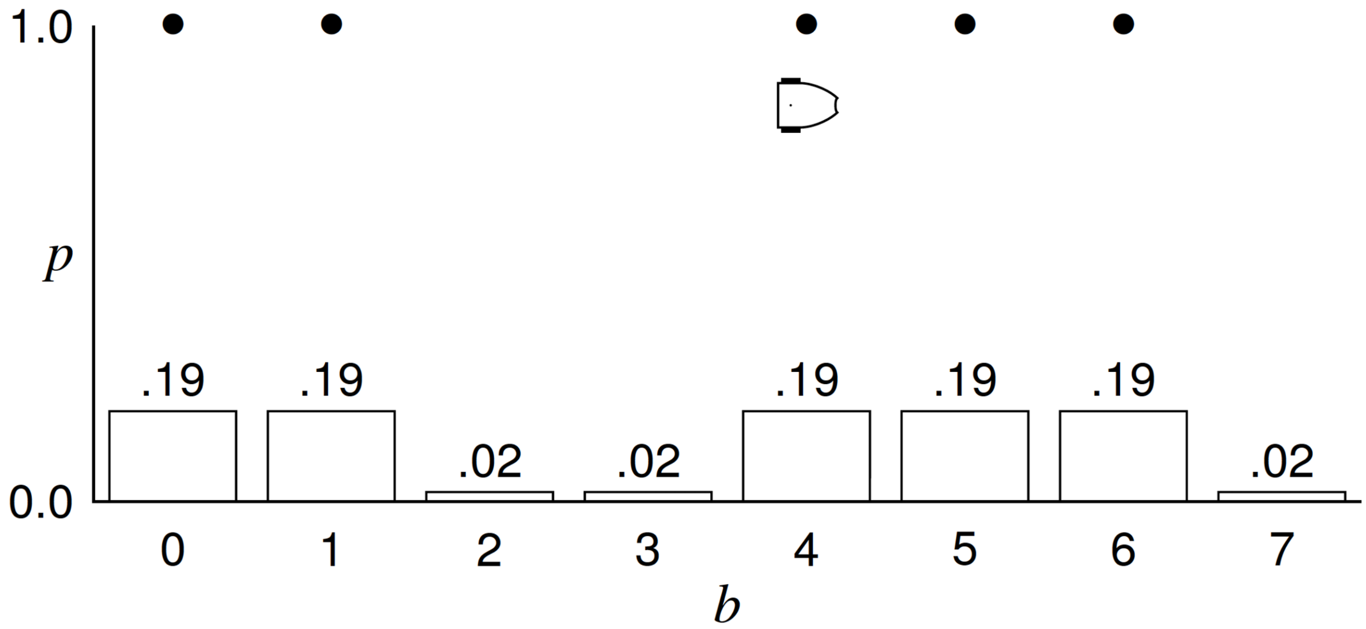
p(b|z)=p(z|b).p(b)

z: measurement

p(z|b) = **0.9**
if map = measure

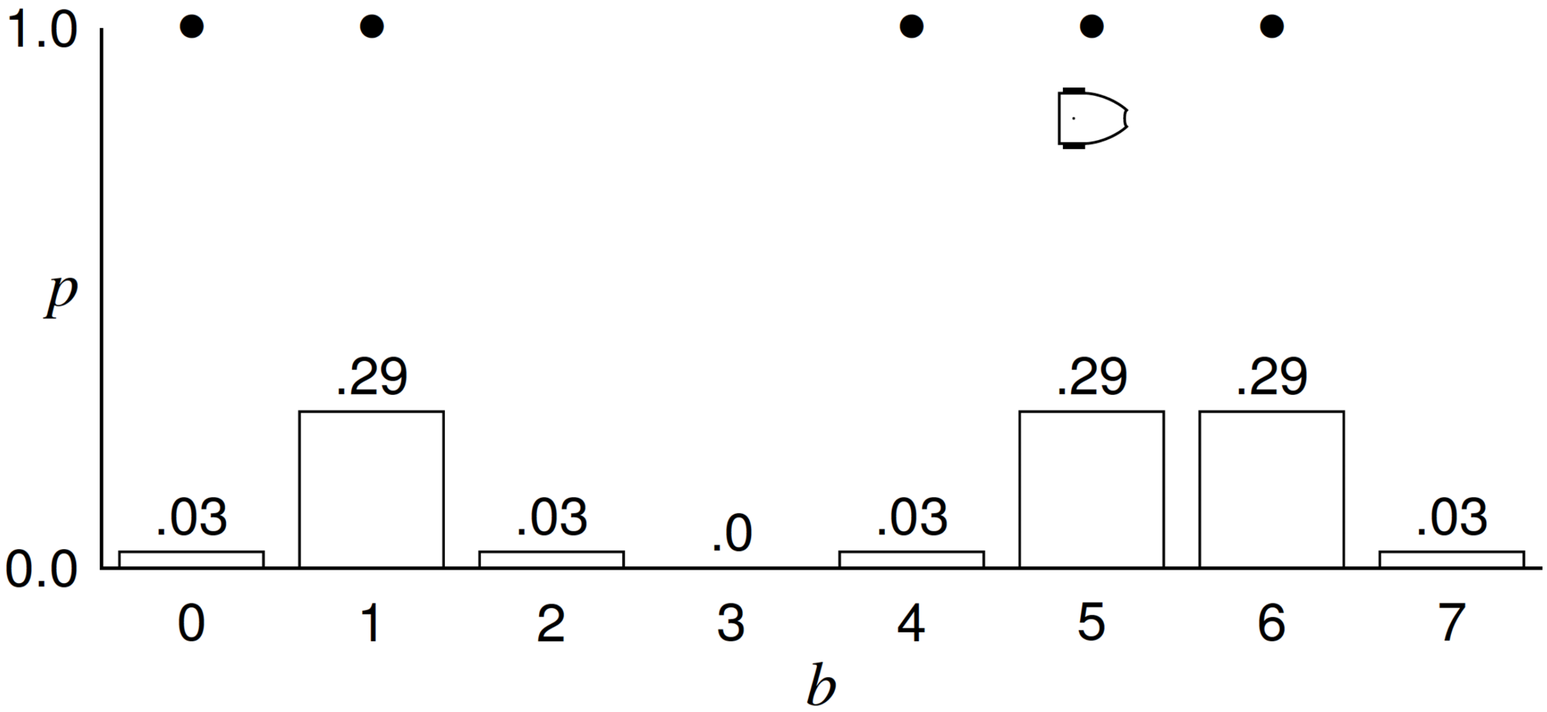
p(z|b) = **0.1**
if map ≠ measure

then normalization



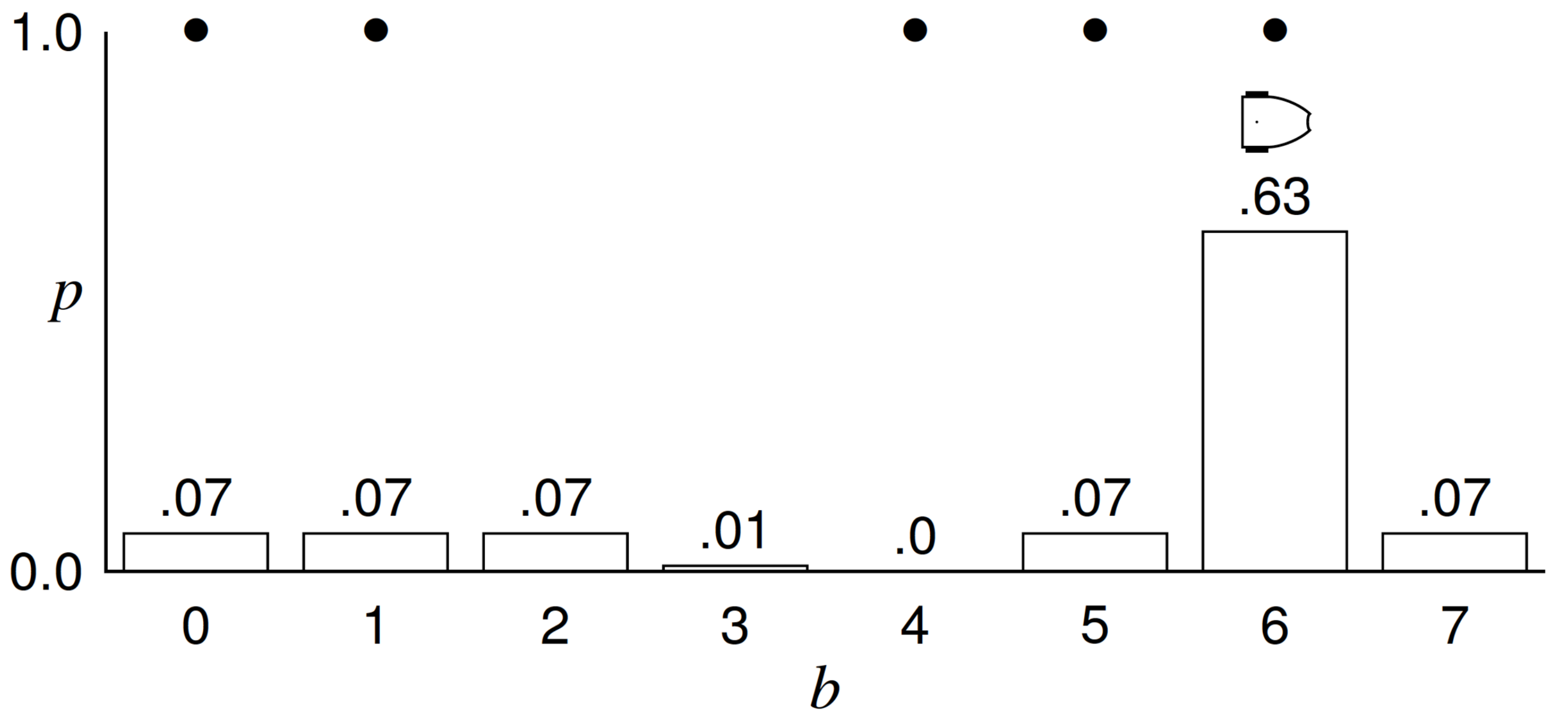
Uncertainty in Sensing

After move:



Uncertainty in Sensing

After move:



Uncertainty in Motion

Movement:

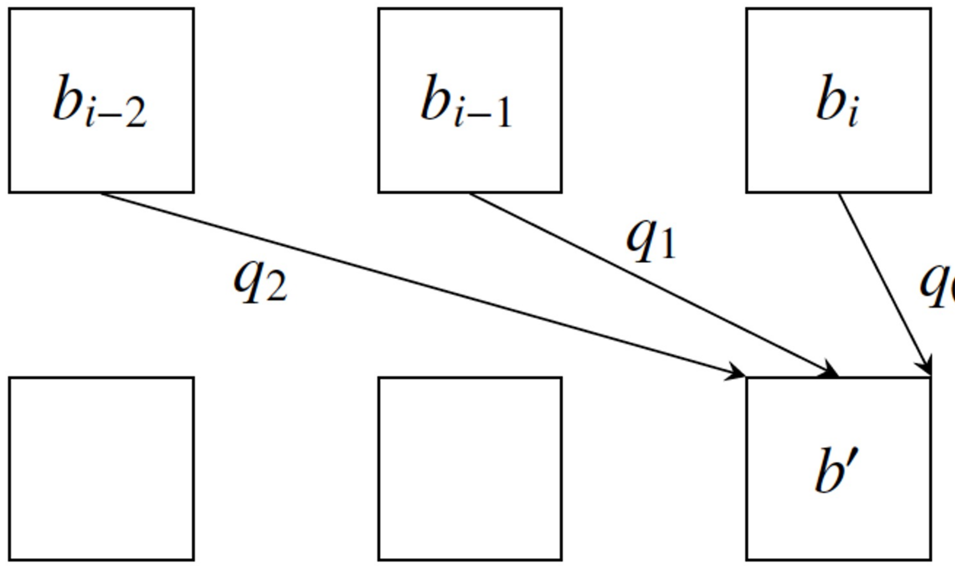
$p(b_t)=p(b_t|u,b_{t-1}).p(b_{t-1})$

u: action

q_1 = **0.8**

$q_0 = q_2$ = **0.1**

$$b'_i = p_i\,(b_{i-2}\,q_2 + b_{i-1}\,q_1 + b_i\,q_0)$$



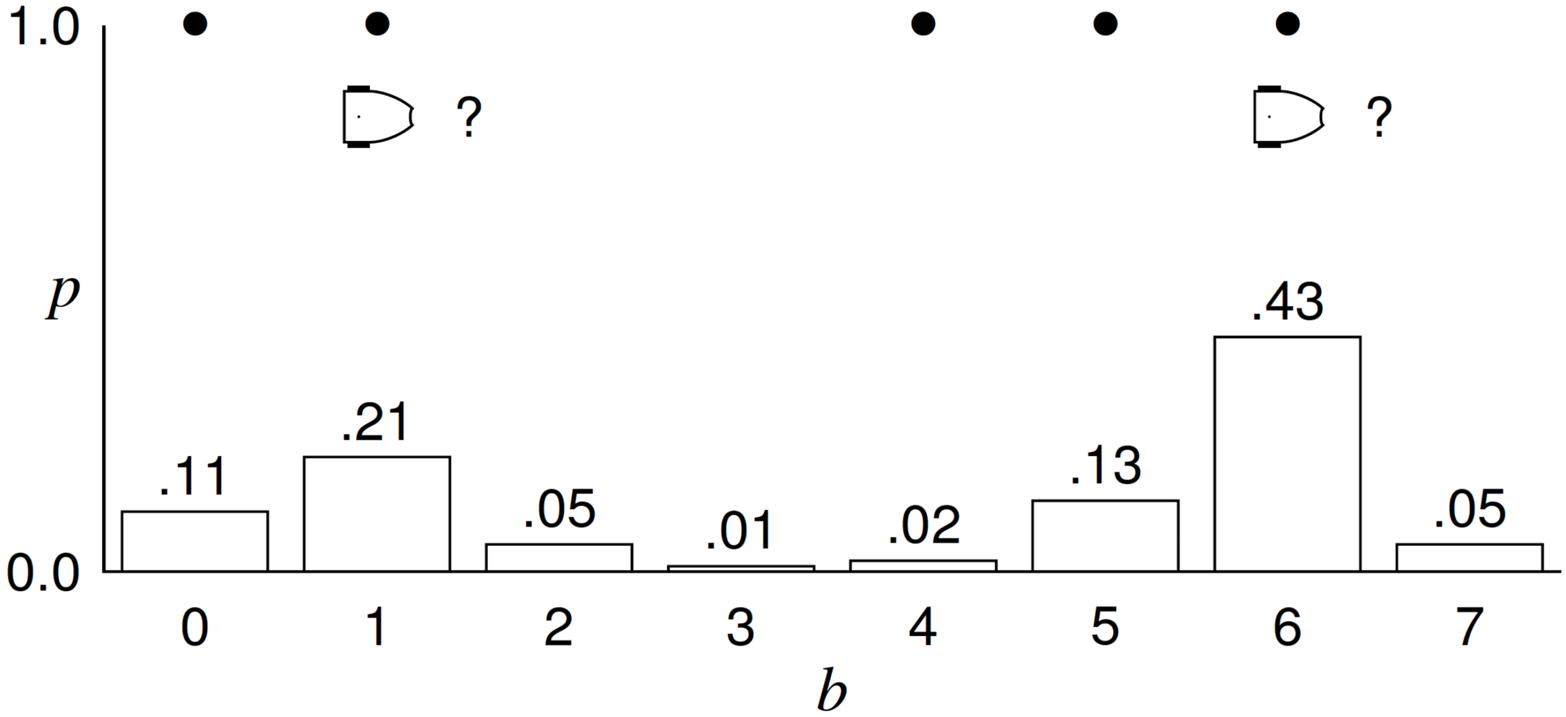
Uncertainty in Motion & Sensing

Table 8.2 Localization with uncertainty in sensing and motion
sensor=after multiplying by the sensor uncertainty
norm=after normalization
right=after moving right one position

	position	0	1	2	3	4	5	6	7
	door?	•	•			•	•	•	
	initial	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13
sensing	sensor	0.11	0.11	0.01	0.01	0.11	0.11	0.11	0.01
	norm	0.19	0.19	0.02	0.02	0.19	0.19	0.19	0.02
moving	right	0.05	0.17	0.17	0.04	0.04	0.17	0.19	0.17
sensing	sensor	0.05	0.17	0.02	0.00	0.03	0.15	0.17	0.02
	norm	0.08	0.27	0.03	0.01	0.06	0.25	0.28	0.03
moving	right	0.06	0.12	0.23	0.05	0.01	0.07	0.23	0.25
	sensor	0.05	0.10	0.02	0.01	0.01	0.06	0.21	0.02
	norm	0.11	0.21	0.05	0.01	0.02	0.13	0.43	0.05

Uncertainty in Motion & Sensing

Final state:



What should I remember

- Definition of the pose of the robot
- Taxonomy, definition of:
  - position tracking
  - global localization
  - kidnapping
- Sensors for pose estimation, features and weaknesses
  - Global positioning system (GPS)
  - Beacon based system (lateral or ceiling, with projection)
  - Natural beacons:
    - vision-based systems
    - object mapping based systems
- Sensors for displacement estimation, features and weaknesses
  - IMU
  - wheel sensors
- Introduction to handling uncertainty in localisation
  - simple 1D example, ability to compute it