Introduction to AI

ROBOTICS

Pedro Meseguer

Institut d'Investigació en Intel·ligència Artificial (IIIA)

Consejo Superior de Investigaciones Científicas (CSIC)

1

Contents

2

1.A Tentative Definition

- What is a robot? A robot is…
  - "An active artificial agent whose environment is the physical world"

    --Russell and Norvig
  - "A programmable, multifunction manipulator designed to move material, parts, tools or specific devices through variable programmed motions for the performance of a variety of tasks"

    --Robot Institute of America

3

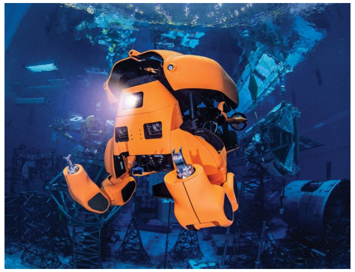
1.Industrial robots

4

1.Unmanned aerial vehicles (drons)

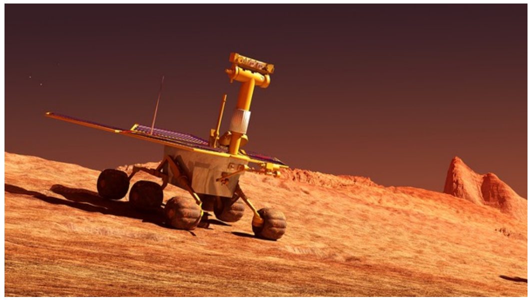
5

1.Underwater vehicles



6

1.Rovers for espace exploration



7

1.Self-driving cars

8

1.Agriculture robots

9

1.Humanoid robots

10

1.Robot types

- Manipulators
  - Robotic arms
- Mobile
  - Unmanned ground vehicles
    - Self-driving cars // Rovers for espace exploration
  - Unmanned aerial vehicles (UAVs)
  - Autonomous underwater vehicles
- Mobile manipulators
  - Humanoid robots

11

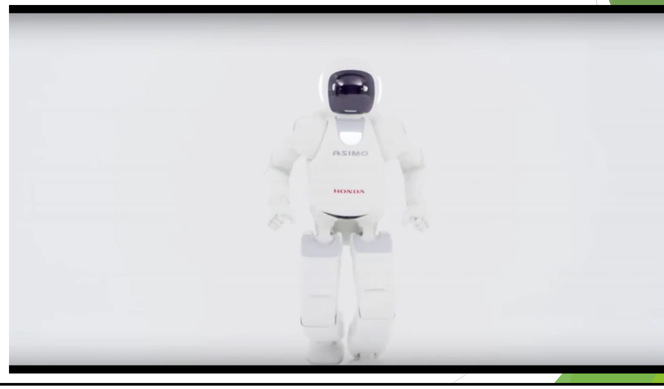
1.Manipulators

12

1.Autonomous cars

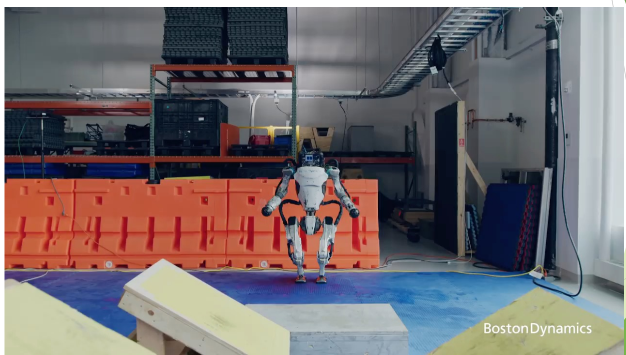
13

1.Humanoid robots: ASIMO



14

1.Humanoids: ATLAS from Boston Dynamics



15

2.Robot hardware

- Sensors:
  - Active: sending energy into the envoronment
  - Passive: observing the environment

  - Range finders: sonar (land, underwater), laser range finder, radar (aircraft), tactile sensors, GPS
  - Imaging sensors: cameras (visual, infrared)
  - Proprioceptive sensors: shaft decoders (joints, wheels), inertial sensors, force sensors, torque sensors

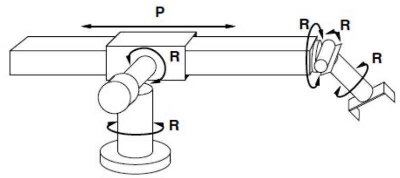
16

2.Robot hardware

- Effectors:
  - Degree of freedom
    - Of a point: the three coordinates
    - Of a rigid body: the three coordinates of its mass centre, plus three orientations
  - Kinematic state
  - Dynamic state
  - Holonomic and non-holonomic
  - Stability

17

2.Manipulators



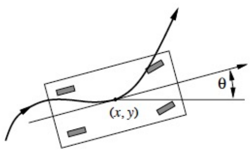
The configuration of this robotic arm is specified by 6 numbers
6 degrees of freedom (DOF)

6 is the minimum number required to position end-effector arbitrarily.
For dynamical systems, add velocity for each DOF.

18

2.Holonomic vs non-holonomic robots

- Holonomic robot: it controls all its DOFs
- Non-holonomic robot: it has more DOFs than it controls



A car has more DOF (3) than controls (2), so is non-holonomic; cannot generally transition between two infinitesimally close configurations

19

3.Planning to move

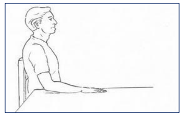
Two spaces:

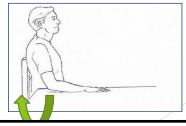
- Working space:
  - 3D space, coordinates of obstacles
- Configuration space:
  - Defined by robot location, orientation and joints: internal coordinates
  - Path planning:
    - A **free trajectory** in configuration space → a **path without obstacles** in working space

20

3.Planning to move

- Kinematics: from configuration to working space: easy (unique sol)
- Inverse Kinematics: from working to configuration space:
- Effector position
  - Hard (solutions are not unique)
- Hardness of inverse kinematics:
  - Fix the position of the effector (gripper…)
  - But joints may have different permitted configurations
  - Example: hand on table

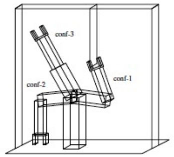


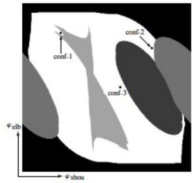


21

3.Planning to move

- Book example:
  - Working space: left – (x,y) elbow, (x,y) gripper
  - Configuration space: right – (ϕ_s, ϕ_e) joint angles
    - Free space: white
    - Occupied space: (collisions) dark

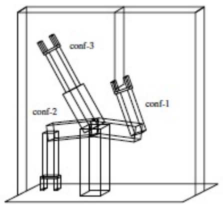


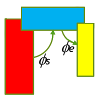


22

3.Planning to move

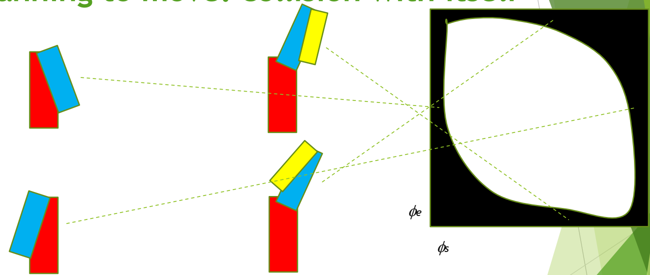
- Joint angles (ϕ_s, ϕ_e)





23

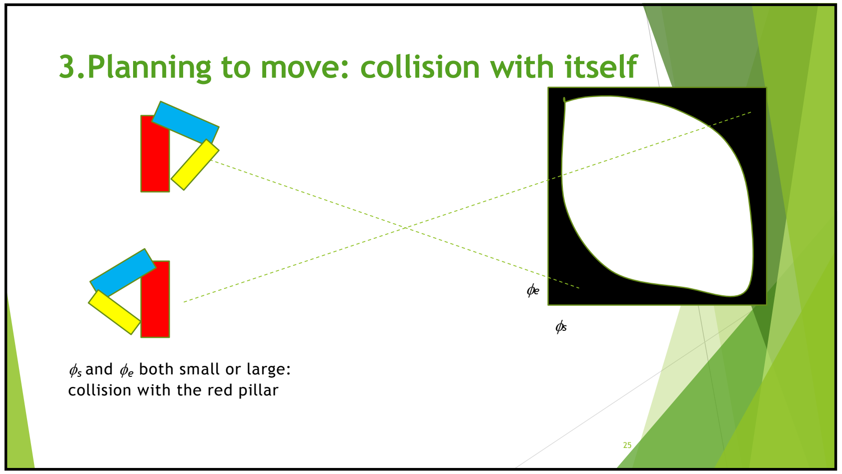
3.Planning to move: collision with itself



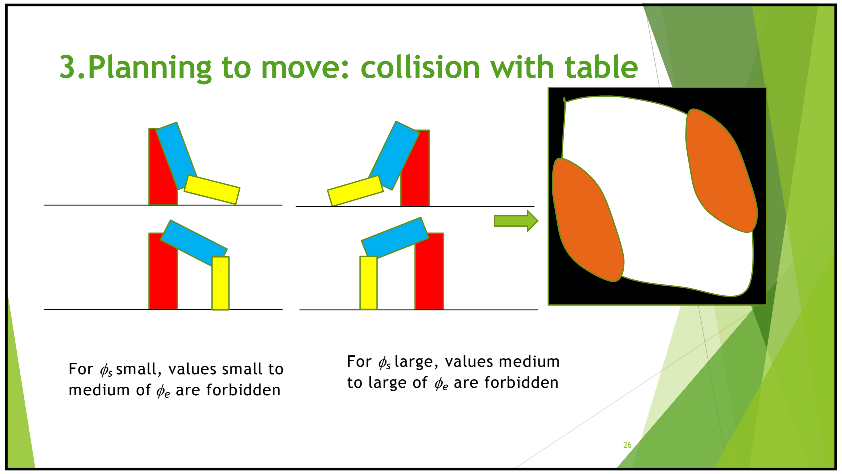
ϕ_s very small or very large: collision with the red pillar (no matter the value of ϕ_e, the yellow part is not shown)

ϕ_e very small or very large: collision with the blue part (no matter the value of ϕ_s)

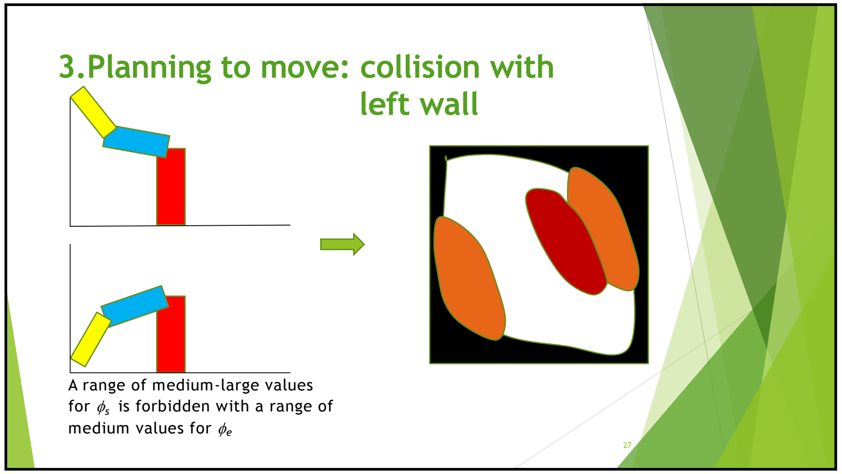
24


3.Planning to move: collision with itself
ϕ_e
ϕ_s
ϕ_s and ϕ_e both small or large:
collision with the red pillar

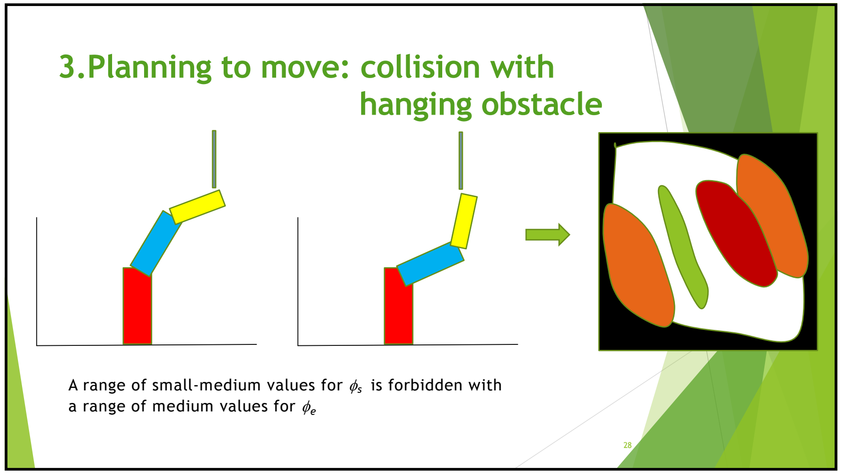
25


3.Planning to move: collision with table
For ϕ_s small, values small to
medium of ϕ_e are forbidden
For ϕ_s large, values medium
to large of ϕ_e are forbidden

26


3.Planning to move: collision with
left wall
A range of medium-large values
for ϕ_s is forbidden with a range of
medium values for ϕ_e

27


3.Planning to move: collision with
hanging obstacle
A range of small-medium values for ϕ_s is forbidden with
a range of medium values for ϕ_e

28

3.Planning to move: solutions

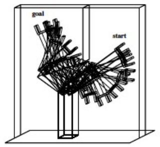
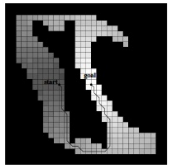
- For any effector coordinates
  - In the example there are two inverse kinematics solutions
  - For industrial robots: very large (even infinite) set of solutions
- A path free of obstacles in working space corresponds to a free trajectory in configuration space
  - Configuration space contains an infinite number of points
  - Shapes could be irregular
- Two basic strategies:
  - Cell decomposition
  - Skeletonization

29

29

3.Cell decomposition

- Decompose the free space into contiguous cells, search for a shortest path
- Inside a cell, path-planning is easy (move in straight line)
- Trajectory: computed by discrete graph search
- Simplest strategy: uniform grid (every cell has the same shape)

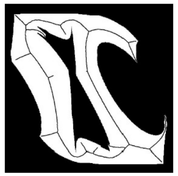
  - Issues with mixed cells that contain parts of free and occupied space
    - If path enters occupied space, the planner could be unsound
    - If only complete free cells are used, the planner could be incomplete
- Issue if the path goes very close to obstacles

30

30

3.Skeletonization

- Voronoi diagram: points that are equidistant of two or more obstacles



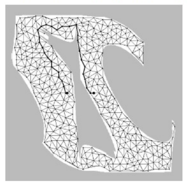
- Strategy:
  - Move straight from start to the closest point in Voronoi diagram
  - Move along Voronoi diagram until the closest point to target (with a shortest path st.)
  - Move straight from Voronoi diagram to target
- Issues with wide open areas

31

31

3.Skeletonization

- Generate a large “random” number of configurations: populate the free configuration space
- Two nodes are joined if it is “easy” to go between them: discrete graph



- Plus start and goal configurations
- Path planning: discrete graph search
- In theory incomplete, in practice works well

32

32

4.Subsumption architecture

- Reactive, bottom-up approach
- Simple and fast movements, avoid deliberation
- Example: Genghis, an hexapod robot
  - Every leg of each side moves independently
  - If it fails an step, it raises higher
- Augmented finite state machines (AFSMs)
- Achieve complex behaviours composing simpler ones
- Behaviours are executed in parallel

- Issues:
  - Rely on sensors: what if they are non trustable?
  - Only for simple tasks: changing is very difficult
  - Hard to understand and maintain

33

4.Three-layer architecture

- Hybrid: reactive + deliberation
- Three layers:
  - Reactive
    - Low level control, tight sensor-action loop
    - Time cycle: milliseconds
  - Executive
    - In-between reactive and deliberative layers (e.g. accepts commands from the deliberative layer, sequences them toward the reactive layer)
    - Time cycle: seconds
  - Deliberative
    - Complex tasks using planning, decision making
    - Time: minutes

34

4.Pipeline architecture

- All layers are executed in parallel

- Sensor interface
  - Collect data from sensors
- Perception
  - Updates internal state based on these data
- Planning & Control
  - Updates robot's internal plans
- Vehicle interface
  - Physical actions on effectors (in self-driving car: turning the wheel, increasing speed, etc.)
- User interface
  - Connection with the user

35

5.Application domains

- Industry and Agriculture
- Transportation
- Self-driving cars
- Health care
- Hazardous environments
- Exploration
- Services
- Entertainment
- Human augmentation

36

Rehabilitation robotics

37

Assistive robotics

38

Wrap up

- Several robots on very different tasks: from industrial arms in car factories to assistive robotics for the elderly.
- Robot hardware
  - Sensors/effectors
  - Degrees of freedom: holonomic/non-holonomic
- Spaces:
  - Working space
  - Configuration space
- Free path in configuration space
- Moving strategies: cell decomposition / skeletonization
- Software architectures

39

Further reading

- Russel & Norvig 3rd ed:
  25.1, 25.2, 25.4, 25.7, 25.8 plus
  Bibliographical Notes chapter 25

- Many videos on robotics in YouTube

40