Robotics

Erwin M. Bakker| LIACS Media Lab 21-2 2019

Universiteit Leiden

Bij ons leer je de wereld kennen

Organization and Overview

Period: February 15th - May 10th 2019
Time: Friday 09.00 – 10.45
Place: LIACS, Room 401 (Workshops Room 303)
Lecturer: Dr Erwin M. Bakker (erwin@liacs.nl)
Assistant: Andrius Bernatavicius

NB E-mail your name and student number to erwin@liacs.nl

Schedule:

15-2	Introduction and Overview
22-2	Locomotion and Inverse Kinematics
1-3	Sensors and Algorithms
8-3	*SLAM Workshop I and Yetiborg Introduction*
15-3	*Project Proposals (presentation by students)*
22-3	*ROS Workshop II* and Yetiborg Qualification
29-3	Robotics Image Processing
5-4	*Yetiborg Race and/or Nao Workshop III*
12-4	Robotics Image Processing and Understanding
19-4	No Class
26-4	Robotics Reinforcement Learning.
3-5	Robotics Reinforcement Learning Workshop IV
10-5	***Project Demos** (by students)*

Website: http://liacs.leidenuniv.nl/~bakkerem2/robotics/

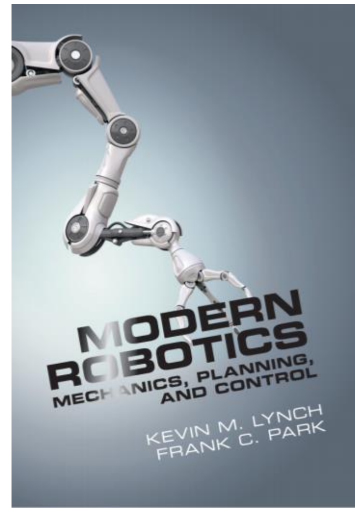
Grading (6 ECTS): Presentations and Robotics Project (60% of grade). Class discussions, attendance, workshops and assignments (40% of grade). It is necessary to be at every class and to complete every workshop.

Universiteit Leiden. Bij ons leer je de wereld kennen

Overview

- Robotic Actuators
- Configuration Space
- Rigid Body Motion
- Forward Kinematics
- Inverse Kinematics

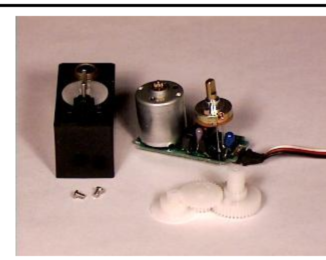
- Link: http://modernrobotics.org

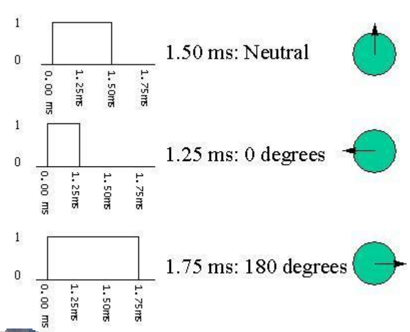


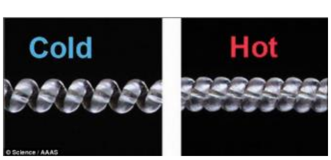
K.M. Lynch, F.C. Park, Modern Robotics: Mechanics, Planning and Control, Cambridge University Press, 2017

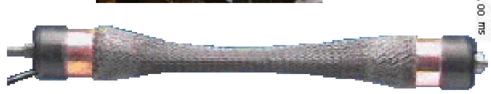
Robotics Actuators

- Electro motors
- Servo’s
- Stepper Motors
- Brushless motors
- Solenoids
- Hydraulic, pneumatic actuator’s
- Magnetic actuators
- Artificial Muscles
- Etc.





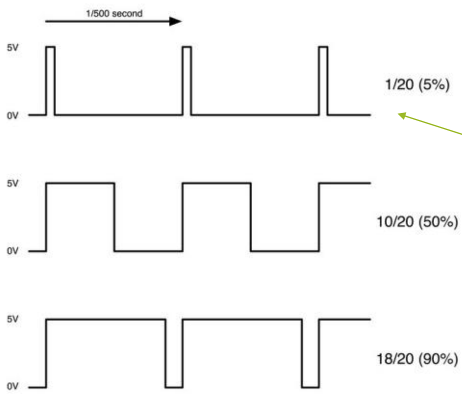


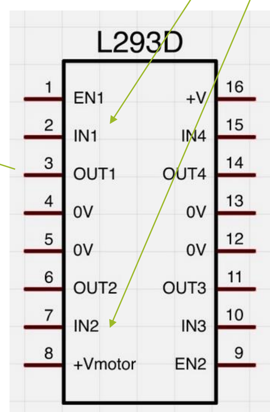


Universiteit Leiden. Bij ons leer je de wereld kennen

Robotics Actuators:

- Electro motors

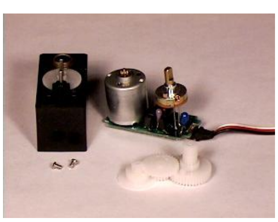




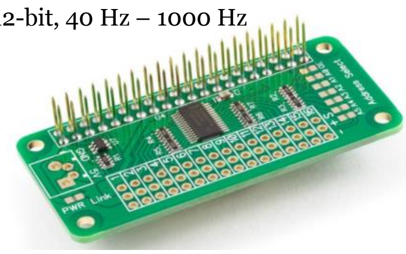

```
import RPi.GPIO as io
io.setmode(io.BCM)

in1_pin = 4
in2_pin = 17

io.setup(in1_pin, io.OUT)
io.setup(in2_pin, io.OUT)

def set(property, value):
    try:
        f = open("/sys/class/rpi-pwm/pwm0/" + property, 'w')
        f.write(value)
        f.close()
    except:
        print("Error writing to: " + property + " value: " + value)

set("delayed", "0")
set("mode", "pwm")
set("frequency", "500")
set("active", "1")

def clockwise():
    io.output(in1_pin, True)
    io.output(in2_pin, False)

def counter_clockwise():
    io.output(in1_pin, False)
    io.output(in2_pin, True)

clockwise()

while True:
    cmd = raw_input("Command, f/r 0..9, E.g. f5 :")
    direction = cmd[0]
    if direction == "f":
        clockwise()
    else:
        counter_clockwise()
    speed = int(cmd[1]) * 11
    set("duty", str(speed))
```

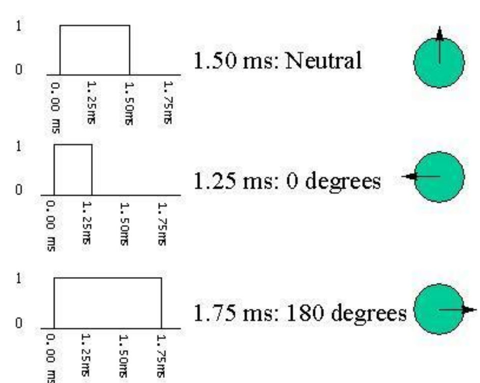
Universiteit Leiden. Bij ons leer je de wereld kennen

Servo’s



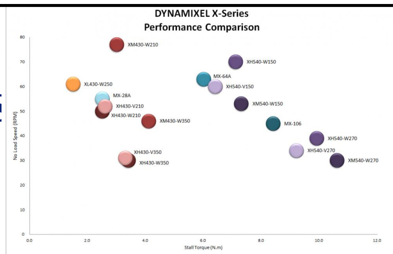
16-ch, 12-bit, 40 Hz – 1000 Hz

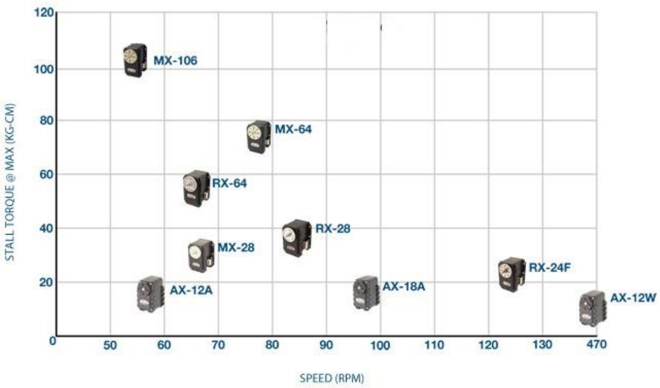




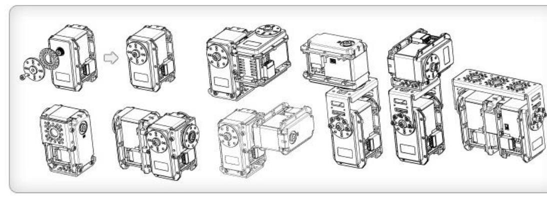
Universiteit Leiden. Bij ons leer je de wereld kennen

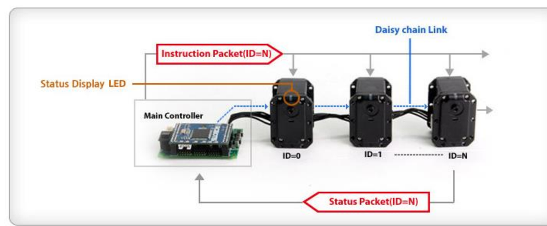
Servo's





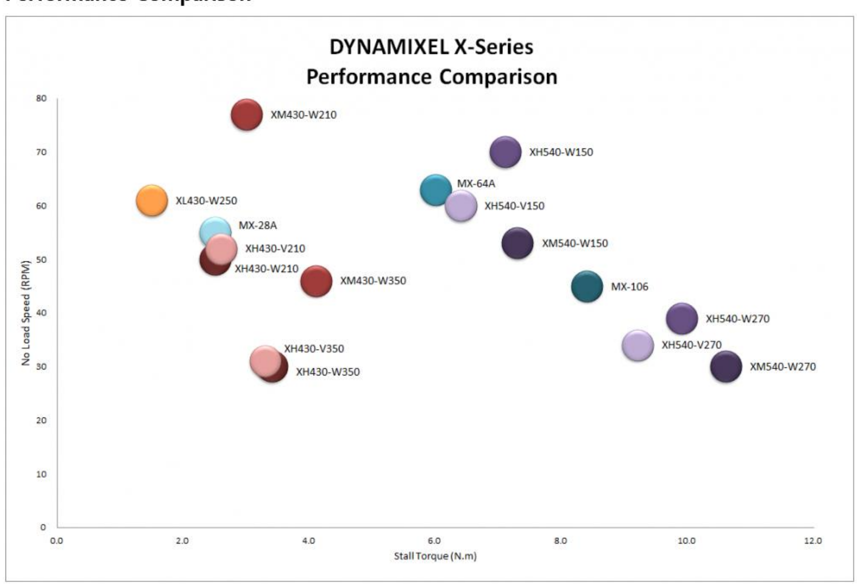
Flexible Construction and Modular Structures





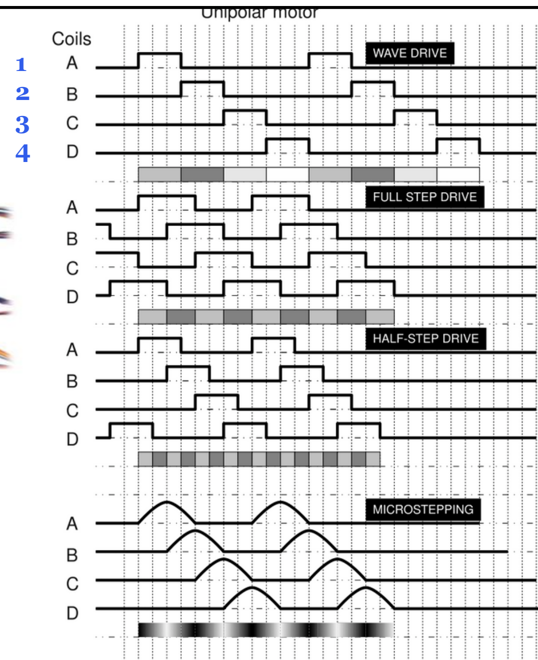
Servo's

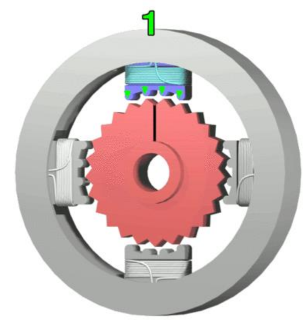
Performance Comparison



Robotics Actuators

- Stepper motors
- Drivers: low-level, high level

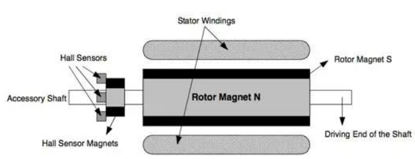


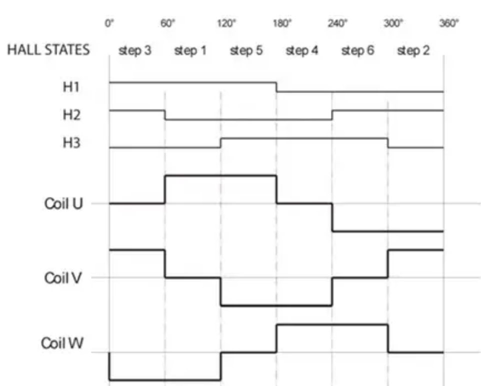


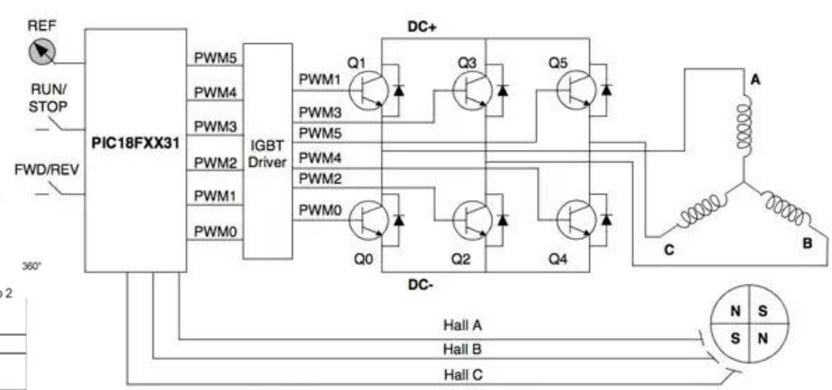
By Wapcaplet; Teravolt. (Wikipedia)

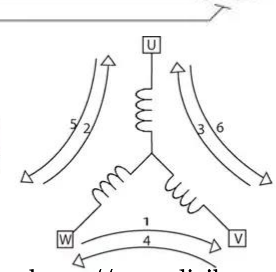
Universiteit Leiden. Bij ons leer je de wereld kennen

Brushless Motors





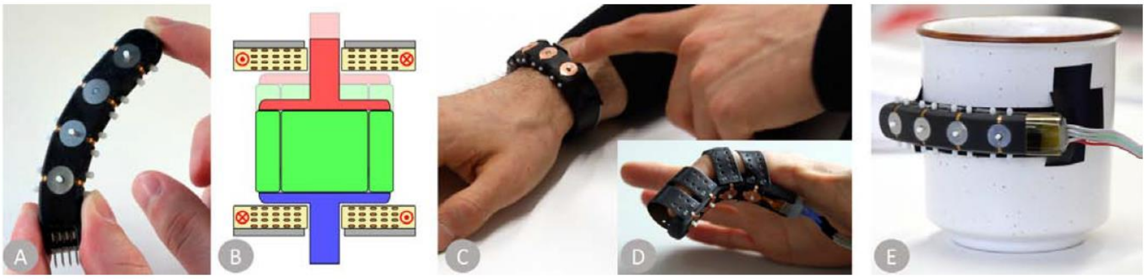


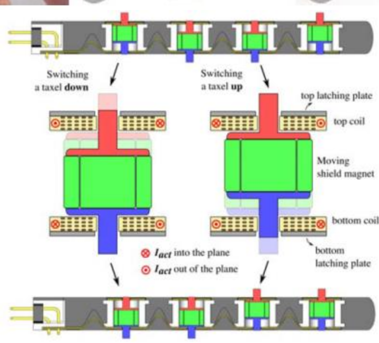


https://www.digikey.com

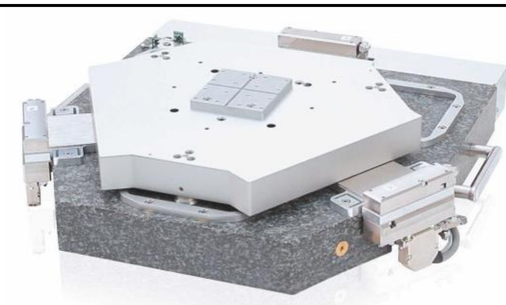
Universiteit Leiden. Bij ons leer je de wereld kennen

Flexible Magnetic Actuators



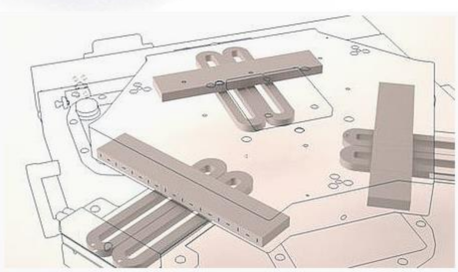


F. Pece et al., MagTics: Flexible and Thin Form Factor Magnetic Actuators for Dynamic and Wearable Haptic Feedback, UIST 2017, Oct. 22–25, 2017, Québec City, Canada

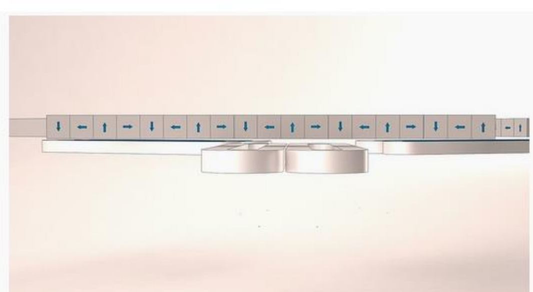


Robotics Actuators

- 6D Magnetic Control
- https://www.pi-usa.us
- pimag-6d-magnetic-levitation

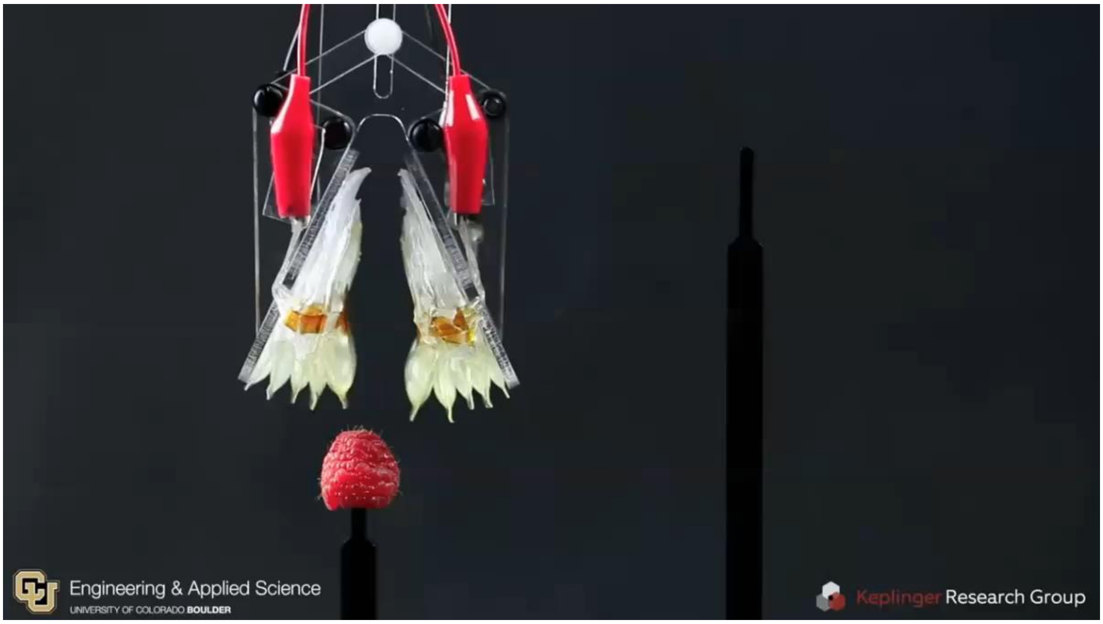


Simple structure: The platform levitates on a magnetic field generated by only six planar coils in the stator



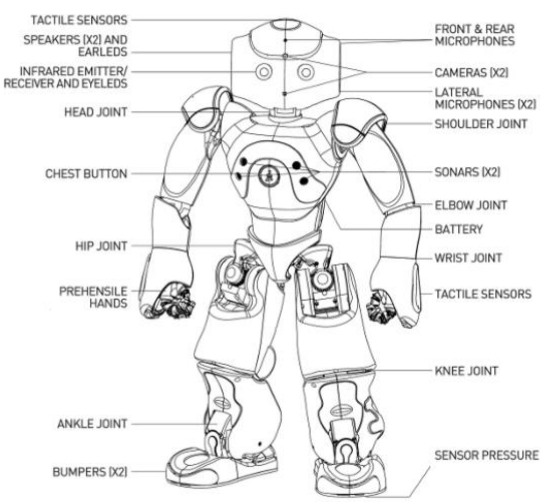
The Halbach arrangement of the magnets makes it possible, to minimize the energy required by the active coils in the stator for carrying the platform, to increase the load carrying capacity and to reduce thermal load

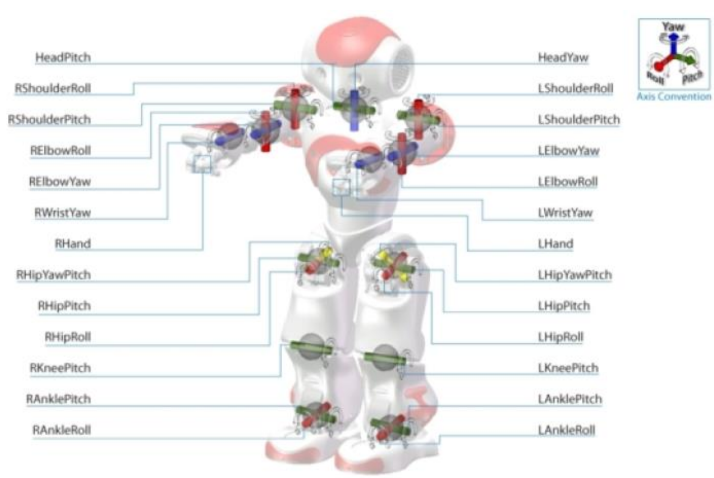
Universiteit Leiden. Bij ons leer je de wereld kennen



Universiteit Leiden. Bij ons leer je de wereld kennen

NAO





http://doc.aldebaran.com/2-1/family/nao_dcm/actuator_sensor_names.html

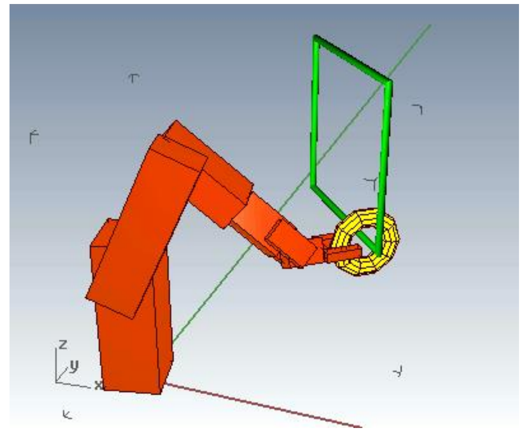
How to move to a goal?

Problem: How to move to a goal?

- Grasp, Walk, Stand, Dance, Follow, etc.

Solution:

- Program step by step.
- Inverse kinematics: take end-points and move them to designated points.
- Trace movements by specialist, human, etc.
- **Learn the right movements:**
  Reinforcement Learning, give a reward when the movement resembles the designated movement.



https://pybullet.org/wordpress/

Configuration Space

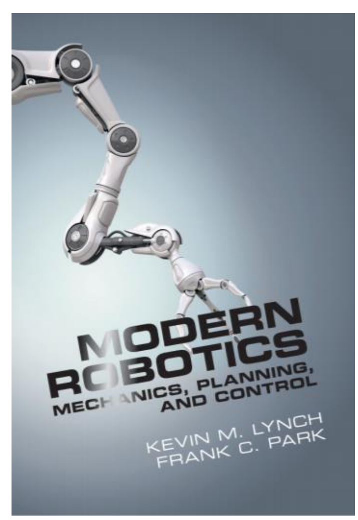
Robot Question: Where am I?

Answer:

The robot's configuration: a specification of the positions of all points of a robot.

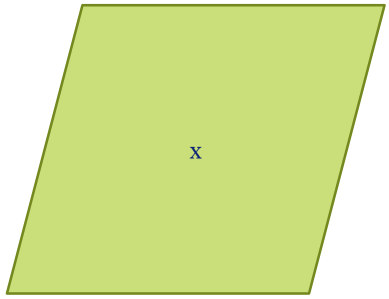
Here we assume:

Robot links and bodies are rigid and of known shape => only a few variables needed to describe it's configuration.



K.M. Lynch, F.C. Park, Modern Robotics: Mechanics, Planning and Control, Cambridge University Press, 2017

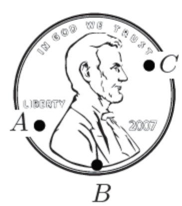
Configuration Space

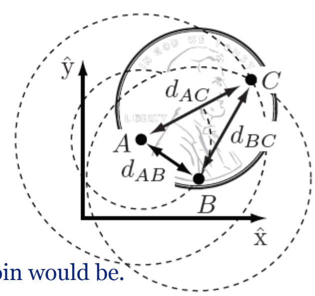


- Degrees of Freedom of a Rigid Body: the smallest number of real-valued coordinates needed to represent its configuration

Universiteit Leiden. Bij ons leer je de wereld kennen

Configuration Space





Assume we have a coin with 3 points A, B, C on it.

In the plane A,B,C by themselves would have
6 degrees of freedom: (x_A,y_A) , (x_B,y_B) , (x_C,y_C)
A coin is rigid => 3 constraints: d_{AB}, d_{AC}, d_{BC} are fixed, wherever the location of the coin would be.

1. The coin and hence A can be placed everywhere => (x_A,y_A) free to choose.
2. B can only be placed under the constraint that its distance to A would be equal to d_{AB}. => We have the freedom to turn the coin around A with angle φ_{AB} => (x_A, y_A, φ_{AB}) are free to choose.
3. Now the constraints that C should be placed at distance d_{AC}, d_{BC} from A and B, respectively gives only 1 possibility once head or tail is determined => no degree of freedom added.

Degrees of Freedom (DOF) of a Coin
= sum of freedoms of the points – number of independent constraints
= number of variables – number of independent equations = 6 – 3 = 3

Universiteit Leiden. Bij ons leer je de wereld kennen