Intelligent Robotics - part 1
F2RO/F21RO

Nick Taylor
N.K.Taylor@hw.ac.uk
Room EM1.62

Industrial Robotics
Course Content

- Manipulators
  - Geometries
  - Control
  - Programming
  - Kinematics
  - Kinetics
  - Path Control
- AGVs
  - Mapping
  - Navigation

Robot

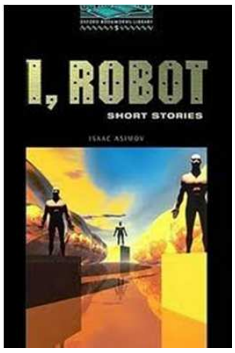
- Term comes from -

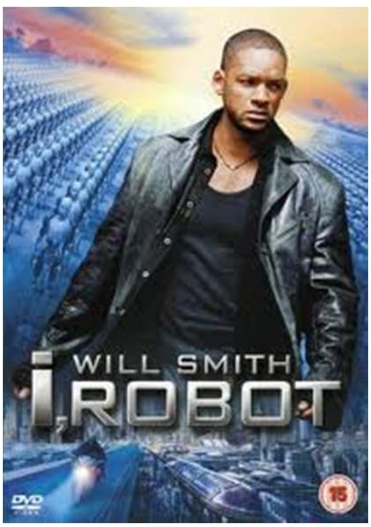
  "Rossum's Universal Robots"
  Karel Capek, 1923, Bohemia

  "Robota" = "worker"
  in Slavic languages

Robotics

- Isaac Asimov claims to have created this term in the 1940s





Industrial Robot

- A programmable articulated arm with a range of end-effectors (grippers, welders, etc.)

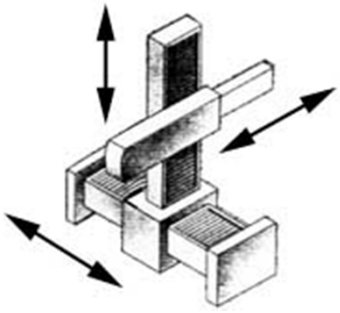
Industrial Robot Architecture

- A Controller and a jointed Arm
- The Arm is often referred to as a Manipulator
- The Manipulator will normally have a Wrist mounted on the end of it
- Wrists can contain 1, 2 or 3 further joints
- End-effectors (also known as Tools) are mounted on the end of the wrist
- There is a notional point called the Tool Centre Point (TCP) where the robot's work takes place

Manipulator Geometry (I)

- Cartesian (or gantry)

  3 orthogonal prismatic joints
  - 1 translation along x axis
  - 1 translation along y axis
  - 1 translation along z axis

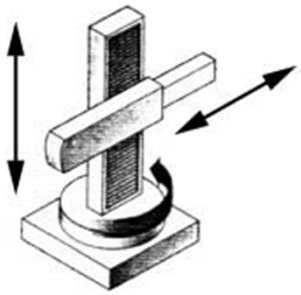


Manipulator Geometry (II)

- Cylindrical

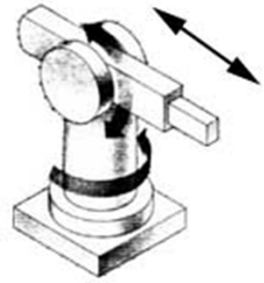
  2 prismatic joints
  - 1 vertical translation
  - 1 radial translation

  1 revolute joint
  - rotation in horizontal plane



Manipulator Geometry (III)

- Polar (or spherical)



1 prismatic joint
  - radial translation

2 revolute joints
  - 1 rotation in horizontal plane
  - 1 rotation in vertical plane

Manipulator Geometry (IV)

- SCARA (Selective Compliance Assembly Robot Arm)

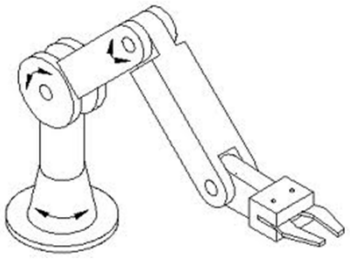
1 prismatic joint
  - vertical translation

2 revolute joints
  - 2 rotations in horizontal plane

Manipulator Geometry (V)

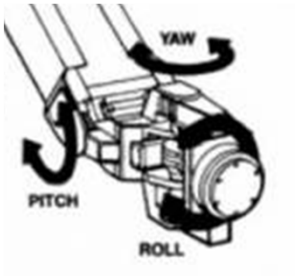
- Articulated (or revolute)

  3 revolute joints
  - Waist rotation in horizontal plane
  - Shoulder rotation in vertical plane
  - Elbow rotation in vertical plane



Wrist Geometry

- Wrist controls orientation of end effector
- Normally composed of revolute joints
- Up to 3 in number
  - Roll
  - Pitch
  - Yaw



Complete Arm Geometry

- Normally no more than 6 joints
- Described by degrees of freedom
  - N joints can be arranged to give less than N d.o.f. but never more than N

Actuators

- Electric
  - Stepping motors; DC motors
- Hydraulic
  - Liquid pressure
- Pneumatic
  - Gas pressure
- Accuracy decreases down the list
- Speed increases down the list

Controllers

- Limited sequence
  - Mechanical stops
- Servo control
  - Point-to-point
- Continuous path
  - Multi-point

Robot Programming (I)

- Physical Set-up
  - As in limited sequence control
  - Actually setting the stops

Robot Programming (II)

- Walk-through
  - Physically dragging the arm (or proxy training arm) through the required positions recording them a suitable places
  - The De Vilbiss Trallfa robot was advertised as *learning the skills of your best operator on his best day*

Robot Programming (III)

- Lead-through
  - Driving the arm remotely from a hand-held controller called a “Teach Pendant” recording positions as desired
  - 3 ways of doing the driving
    - (a). Individual joint direction
    - (b). Relative tool direction (up, down, etc.)
    - (c). Absolute tool direction (x,y,z,orientation)

      Easiest to implement is (a) and hardest is (c)

      But easiest to use is (c) and hardest is (a)

Robot Programming (IV)

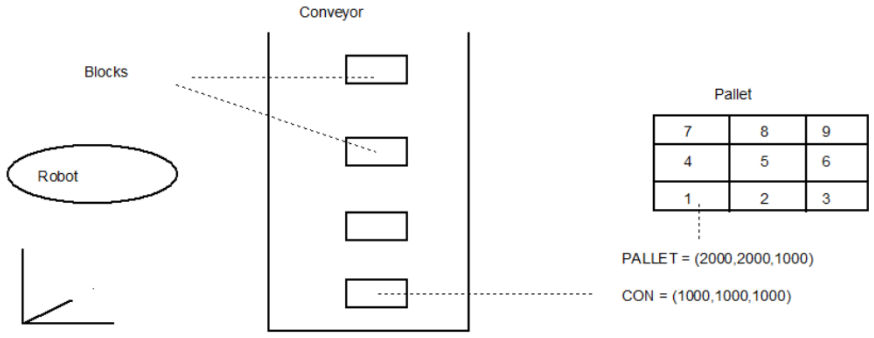
- Off-line
  - Prepare and test a program of instructions off-line
  - Download to controller for execution
  - Preferable approach to programming but difficulties

Robot Programming Languages

- Robotic devices always incur errors
  - Not simply because of mechanical tolerances but also because of the kinetics of motion when loads are added to the end of the arm
  - Repeatability
    - Degree of error in repeating a previously taught sequence
    - Invariably very good
  - Accuracy
    - Degree of error in moving TCP (Tool Centre Point) to a position in space
    - Invariably poor
- So off-line programming not really fully automatic

Programming Example

- VAL I
  - Unimation PUMA robot
  - Superseded by VAL II



TASK: Unload 9 blocks from the conveyor onto the pallet as indicated