G54ARS
Autonomous Robotic Systems
Lecture 1

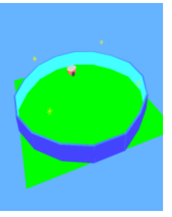
Overview

- Module Introduction
- (Autonomous) Robots = ?
- Foundations of Robotic Systems
- Architectures & Behaviours

Module Introduction

Module Delivery

- A bit about me…
  - Christian Wagner
    - Office: Computer Science, Jubilee Campus, B32
    - Email: christian.wagner@nottingham.ac.uk

Module Overview and Aims

- Course Details & Website:
  - http://moodle.nottingham.ac.uk
  - G54ARS-UK-AUT1718
  - **20 credits**
- Module Aims:
  - To provide a grounding in the basic principles of real-world sensors and actuators, autonomous mobile robots and to give experience in implementing robotic behaviour algorithms.
  - To provide a detailed knowledge of the problem of localisation of autonomous mobile robots and the use of this in mapping and navigation.

Learning Outcomes

- **Knowledge and Understanding:**
  - Experience in implementing algorithms in a real-world robotic context (e.g., robotic control, sensor data handling).
  - Understanding of current technology (e.g., sensors) and techniques in autonomous mobile robotics and an awareness of their limitations.
  - Understanding of how to handle and interpret uncertain sources of information.
- **Intellectual Skills:**
  - Apply knowledge of robotic control techniques to particular tasks.
  - Apply knowledge of uncertain data sources such as sensors within applications.
  - Evaluate and compare competing approaches to robotics and real-world sensor-driven applications.

Learning Outcomes ctd.

- **Professional Skills:**
  - Develop a working knowledge of real-world device programming (sensors and actuators) through implementing advanced robotic behaviour architectures.
  - Apply insights of hardware to developing software solutions.
- **Transferable Skills:**
  - Apply knowledge of the methods and approaches presented to other problem domains, in particular knowledge gained about computing with real-world information gathered from sensors, e.g., to mobile device programming.
  - Use the available resources (libraries, internet, etc.) to supplement the course material.

Module Outline

- Lectures cover theory
  - Formal material, often followed by examples
  - Lecture material covers all the topics, BUT you are expected to use available resources (libraries, the internet, etc.) to further develop your knowledge
  - Guest lectures by industry and/or international academics.
- Laboratories support practical learning
  - *Lab sheets (relate to all parts of the module)*
  - *LEGO Mindstorms EV3 &* Robot-C Programming
- Practicals
  - Self-study, Programming & Experimentation
- Feedback
  - continuous feedback on coursework during labs

Module Delivery

- Module Delivery Structure:
  - Overall, six hours per week:
    - 2 hours lecture (theory)
    - 2 hours lab (incl. lab sheets)
    - 2 hours practice (self-study & assignment development)
  - Complete Lab availability sheet at the end of today.
- **Your feedback is important and valued**
  - **How?**
  - **Email, Talk to me, etc.**

Module Delivery - Lectures

- Usual Schedule:
  - 1 x 2 hours each week:
    - Thursdays, 9-11am, Jubilee Campus, Business School South, Room A26
    - Lecture notes are made available on Moodle
    - Guest lectures are announced in prior lectures.
- Kahoot
  - Download the app or use web interface at *https://kahoot.it*
  - frequently used – please use real name.

Module Delivery - Labs

- Laboratories:
  - Computer Science Building, Room B85
  - 1 x lab (2 hours)
  - you will be working in teams (2 students per team)
  - teams are for whole term
  - Single student in a team as an exception only
    - E.g., uneven numbers, limited PC access
  - there will be two lab groups, student assignment to lab groups will be done this week

Module Delivery - Practicals

- Practicals:
  - 1 x practice session (2 hours)
  - you will be working in the same pairs
  - there will be two groups, student assignment to lab groups will be done this week
- Advice:
  - practical sessions provide vital time with the robots
  - 10 weeks pass very quickly
  - Lab sheets are here to help you. Progress on lab sheets is an expected minimum, you need to progress on the assignment **in parallel**.
  - frontload your progress, manage your team

Provisional Weekly Topics - 1

Week	Lecture
2	Introduction & Overview (Autonomous) Robots =? Foundations of Robotic Systems Architectures & Behaviours
3	Brooks' Subsumption Architecture - Theory Robot Hardware
4	DARPA Grand Challenge PID Control
5	Ultrasonic Sensor Models Localisation and Mapping
6	Fuzzy Control

Provisional Weekly Topics - 2

Week	Lecture
7	Sensor Fusion
8	Reading Week
9	Ethics and Robotics
10	Kalman Filters
11	Particle Filters
12	Revision

Note:
Weekly topics may change subject to timing and new material.
All lecture notes will be available online.

A Note on Math and other things…

- Many of the topics in robotics (i.e. in the module) involve mathematics, e.g. PID Control:

$$CA(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

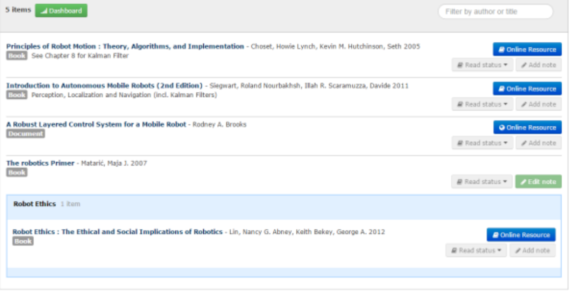
- Lectures do not include revisions in mathematics – it is your responsibility to read up on material which you are not familiar with.
- The course is (unfortunately necessarily) back-loaded, i.e. the harder stuff comes at the end – lesson: keep on top of the material week-by-week!

Module Assessment

- Examination – 50%
  - Covering all material covered in lectures and reading week. (This includes scientific papers, articles, videos, etc.)
- Coursework – 50%
  - Working in teams throughout term
  - 35% Laboratory assignment & individual associated report
  - 15% Five lab sheets/demos per team (3% each)
  - The lab is only accessible during your allocated lab and practice sessions.
    - i.e. **attend and prepare your sessions!**

Reading List

- Reading List on Moodle
- Materials accessible online where possible
- Updated throughout



Any Questions

?

(Autonomous) Robots = ?

What is a robot?

- What does the word “robot” mean?
- The word ‘robot’ originates from the 1921 play “Rossum’s Universal Robots” by the author Karel Capek.
- It is derived from the Czeck words ‘robota’, meaning “forced labour” and ‘robotnik’, meaning “serf” (someone who needs to render services to a lord or superior)

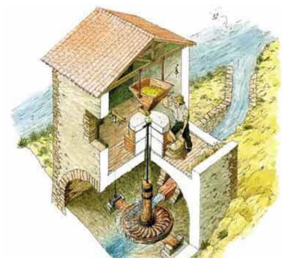
What is a Robot – robotics?

- ‘Robotics’ appeared in the 1942 novel ‘Runaround’ by Isaac Asimov
- Isaac Asimov also coined the three basic laws of robotics in the same novel.
  1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
  2. A robot must obey the orders given to it by human beings, except where such orders would conflict with the First Law.
  3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Laws.
- Today, robot ethics are a hot topic, we will delve into this more later in the module.

http://en.wikipedia.org/wiki/I,_Robot_%28film%29

What is a Robot?

- Definition = ?
- A machine that can help people execute tasks?



Source: http://www.top-alternative-energy-sources.com/water-wheel-design.html

What is a Robot?

- What about:

Source: http://www.yourlocalcinema.com

What is a Robot?

- The notion of what a robot is and is not has changed over time, today, one good current definition:
- “*A robot is an autonomous system which exists in the physical world, can sense its environment, and can act on it to achieve some goals.*” (M. J. Matarić)

What is a Robot?

- Autonomous system
- Can sense its environment
- Exists in the physical world
- Can act on its environment to achieve some goals





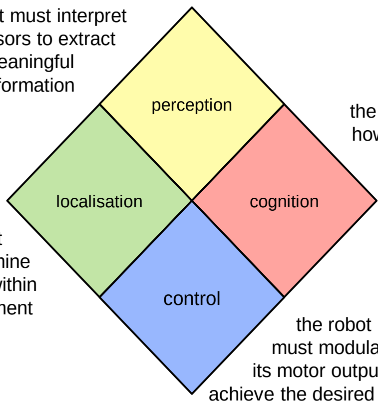


Autonomous Robots

- Dictionary definitions of 'autonomous'
  - *undertaken without outside control*
    - carry on-board sensors, controllers and power supplies
    - for example, *automated guided vehicles* (AGV's) that operate by following tracks to move parts & equipment
      - 'weak autonomy'
  - *having the power of self-government*
    - able to adapt to changing environments
    - determine its course of action by its own reasoning process
    - the ability to build internal representations of the world
    - the ability to learn from experience and plan new actions
      - 'strong autonomy' / 'intelligent mobile robots'

Autonomous Mobile Robots - Navigation

the robot must interpret its sensors to extract meaningful information



the robot must decide how to act to achieve its goals

the robot must determine its position within the environment

the robot must modulate its motor outputs to achieve the desired trajectory

Autonomous Robots = ? ✓

But why do robotics?

Basics

… from the motivation to robotics to intelligence…

Motivation

- Why study autonomous robotic systems?
- Applied:
  - to create robots to be used in hostile environments
    - underwater
    - planetary exploration
    - nuclear power stations
    - bomb disposal
  - Increasingly: domestic chores like…?
- Theoretical
  - to investigate intelligent behaviour
    - artificial intelligence
    - Psychology, cognitive science

Robotics and Intelligent Agents

- The word ‘agent’ means “to do”
  - an entity that produces an effect
- Consequently, ‘agent’ is used to describe both software *simulations* and / or actual hardware *implementations* of robots
  - robot
    - physical machine
  - agent
    - numerical computer model
    - physical machine

Simulation v. Implementation

- Perhaps software agents could be used as the primary mechanism to investigate robots
  - Advantages
    - **cheap**
    - **flexible**
  - Disadvantages
    - **flexible** simulated worlds not identical to real worlds
- Many people (e.g., Brooks) believe that true intelligent behaviour only emerges <u>when a physical agent interacts with its environment</u>
  - the *symbol grounding problem* (e.g. meaning of words on this page.)

By the way - what is Intelligence?

- *The extent to which we regard something as behaving in an intelligent manner is determined as much by our own state of mind and training as by the properties of the object under consideration.*

- *If we are able to explain and predict its behaviour or if there seems to be little underlying plan, we have little temptation to imagine intelligence. With the same object, therefore, it is possible that one person would consider it as intelligent and another would not; the second would have found out the rules of its behaviour.*
  - Alan Turing, 1947

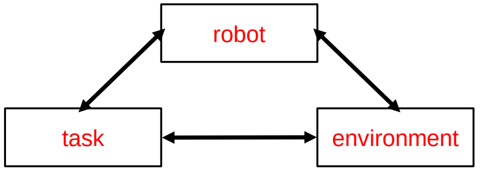
Robots

… and what they are made of…

Components

- A robot comprises three main component classes
  - sensors
    - *a device giving a signal for the detection or measurement of a physical property to which it responds* (Oxford English Dictionary)
    - provides the inputs to the robot
  - control hardware & software
    - programmed behaviour(s); data and 'memory'
    - makes "decisions" for the robot
  - actuators
    - *a thing which moves to mechanical action, communicates motion to, or impels (an instrument, machine, or agent)* (Oxford English Dictionary)
    - effects the outputs from the robot

Linkage



- A robot, its task and the environment all depend on, and influence, each other
  - e.g., a spider in the bath!

General Purpose Robots?

- A general purpose robot is not possible
  - a general purpose living thing does not exist
    - humans are the most intelligent (???)
    - but humans are poor at:
      - flying (c.f. swallow, swift, Arctic tern, housefly)
      - swimming (c.f. tuna, sperm whale)
      - surviving (c.f. scorpions, ants)

but humans are excellent generalisers!

- A robot's function and operation are defined by its own behaviour within a specific environment, taking into account a specific task
  - only the simultaneous description of a robot, its task and the environment describes the robot completely

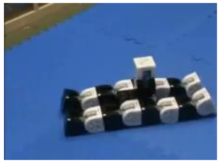
Example

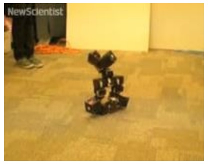
- Outdoor transport robot
- Task = ?
- How would you build it?
- Inspiration in nature?
    - Task: in groups, sketch such a robot
- Example: Still – task specific! (e.g., does not swim!)

Boston Dynamics Big Dog http://www.youtube.com/watch?v=W1czBcnX1Ww

Beyond purpose-built robots

- Self-assembling robots
- One/multiple agents?
- Still – no true multi-purpose





http://www.youtube.com/watch?v=v6W-sEpJEqY http://www.youtube.com/watch?v=uIn-sMq8-Ls http://www.youtube.com/watch?v=6aZbJS6LZbs

Environment

Environment Types

- There are many different types of environment in which a robot may be required to operate
- Environments are typically categorised by their degree of *structure*
- Although there is no solidly accepted definition of structures, environments can be split into one of the following categories
  - structured
  - unstructured
  - partially structured

Structured Environments

- A structured environment has been specially designed for the robot to operate in
    - e.g., an artificial maze
    - a factory floor with in-built 'tracks' to follow
  - an exact description of the environment can be supplied to the robot during its design phase
    - very little or perhaps no sensor data may be required
- There are usually no **unexpected** or **unplanned** dynamic aspects to the environment
  - the environment does not change
  - the robot has been 'told' in advance of how and when the environment will change, and how to deal with it

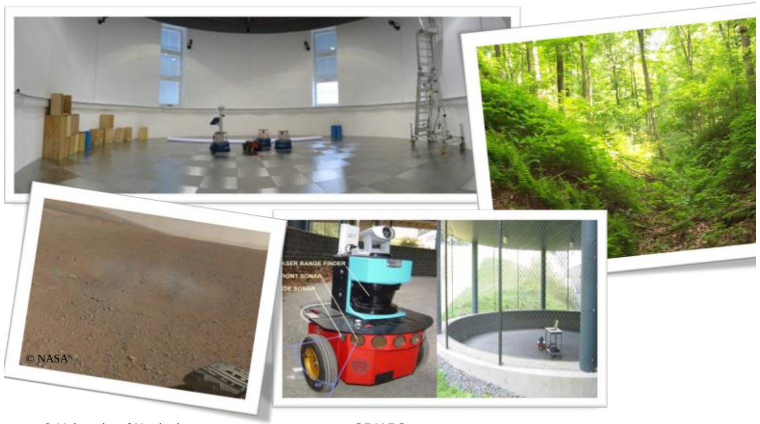
Unstructured Environments

- Complex environments for which no models or maps exist, or can even be accurately generated
  - robots operate in response to **(real-time) sensor data**
- Such environments usually have significant dynamic changes
  - natural, real-world as opposed to artificially created
  - may have unknown attributes
    - e.g., deep-sea exploration
  - or my be almost entirely unknown
    - e.g., planetary probes

Partially Structured

- Somewhere between the previous two extremes!
  - an environment which may be modelled to a certain extent, but with insufficient model detail to fully support task completion
- Possibly, the static component of the environment has been modelled, but the dynamic changes are unpredictable and must be sensed, for example:
  - a factory floor with in-built 'tracks' to follow, but with unpredictable (e.g., human) obstacles to avoid
  - actual conditions encountered by Curiosity on Mars
  - the city centre, weather conditions, leaf litter, people(!),….

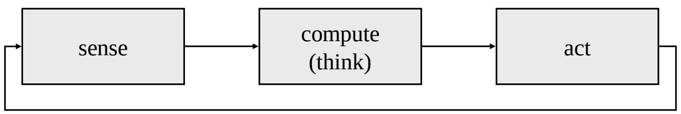
Environments - Examples



…how to act in real world environments?…
→Control Models

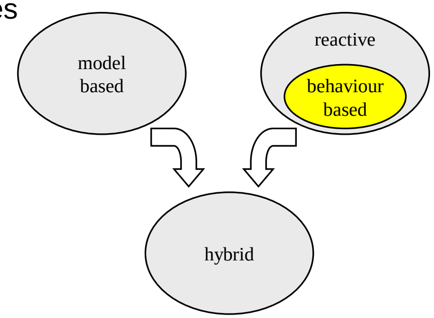
The Control Cycle

- A fundamental methodology derived in the early days of robotics from engineering principles is the *sense-think-act* cycle
  - the principle is to continuously attempt to minimise the error between the actual state and the desired state
    - based on control theory



Control Architectures

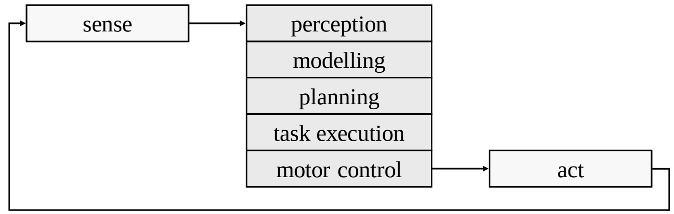
- A variety of different approaches have been tried for implementing the *sense-think-act* control cycle
- These approaches can be categorised as
  - model-based
  - reactive
  - hybrid



Model Based
…think hard first, then act…

Model Based

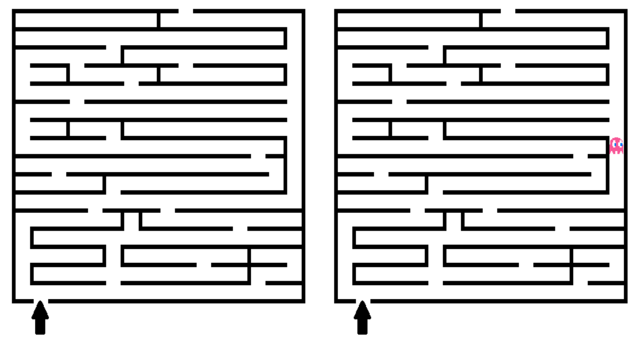
- A symbolic internal 'world-model' is maintained
  - the sub-tasks are *decomposed* into functional layers
  - similar to 'classical' artificial intelligence approach (i.e. take/create perfect knowledge representation (a model) and search the best solution, then act→ chess)



Problems with Models

- An adequate, accurate and up-to-date model must be maintained at all times
  - this is very difficult in practice!
  - suppose, for example, the sensors detect an object that we have not got a symbol for (a novel object)
- A model-based system is extremely brittle
  - if one of the functional layers fails (e.g., hardware problems, software bugs), then the whole system fails
- Significant processing power is required
  - maintaining the model takes time, so slow responses!?
  - reflexes?
- Rapid progress, but, still a challenge!

Problems with Models



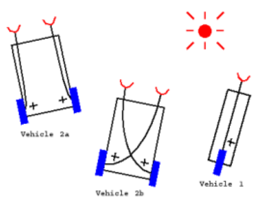
Reactive Robotics
…don't think – react…

Reactive Controllers

- In order to try to overcome the shortcomings of model-based robots, modern approaches have centred on simple *reactive* systems with minimal amounts of computation
  - 'model-free systems'
- More correctly, the models are *simple* and *implicit*
  - the systems do not use symbolic models but, for example, a rule-set which tells a robot how to react to a corner when following a wall may be considered to be a simple, implicit model fragment
    - it implicitly encodes assumptions about the environment
    - no explicit symbolic model for a wall or a corner exists

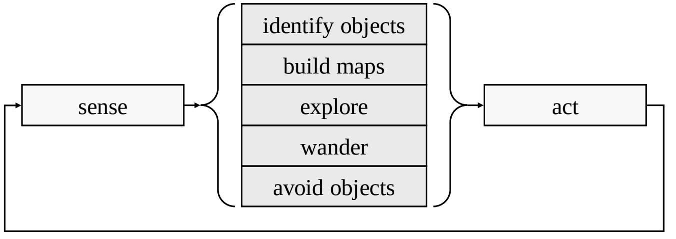
Reactive Controllers

- Tight coupling between sensing and acting
- No symbolic modelling.
- Little to no "thinking" (e.g., rules)
- Examples:
  - Many animals, especially insects
  - Braitenberg Vehicles
  http://www.youtube.com/watch?v=
  - (More explanations:
    https://www.youtube.com/watch
- Problems with Reactive Control?



Behaviour Based

- The control system is broken down into horizontal modules, or *behaviours*, that run in parallel
  - each behaviour has direct access to sensor readings and can control the robot's motors directly



Behaviour Advantages

- It supports multiple goals and is more efficient
  - there is no functional hierarchy between layers
    - one layer does not call another layer
  - each layer can work on different goals in parallel
  - communication between layers is achieved via message passing which need not be synchronised
- The system is easier to design, debug and extend
  - each module can be designed and tested individually
- The system is robust
  - if one module fails, e.g. *wander*, then other layers, e.g. *avoid obstacles*, still function and behave correctly

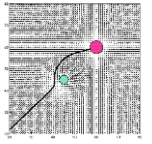
Behaviour Limitations

- It is extremely difficult to implement plans
  - in pure form a behaviour-based robot has no memory (not even an internal state memory) and so is unable to follow an externally specified sequences of actions
- It can be very hard to predict how a large number of multiple behaviours may interact
  - *emergent behaviour* is the term given to unexpected behaviour that comes about through these interactions
    - sometimes it is useful, sometimes it is not!
- The robot can get trapped in a *limit cycle*
  - trapped in a dead-end, repeatedly turning left then right

Other Approaches

Other Reactive Approaches

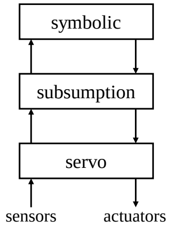
- Two other reactive approaches that are popular
- Potential field methods
  - a potential field is a concept from physics
    - two examples are the *gravitational field*
      - you do not need to be told which way to fall
      - planets do not need to plan how to move around the sun
    - and *electromagnetic fields*
      - mobile phones, television, radio, etc.
    - e.g., obstacles exert hypothetical repulsive forces on the robot



- Motor schema navigation
  - multiple, concurrent schema generate separate behaviours which are summed to produce output
    - schema are dynamically created/destroyed as needed

Hybrid Approaches

- The *SSS* three-layer architecture
  - the *servo-subsumption-symbolic* architecture combines Brooks' architecture with a lower-level servo control level and a higher-level symbolic system [Connell]



- Fuzzy logic and neural network controllers
  - fuzzy logic rule-base(s), neural network(s) and combinations of both take inputs from sensors and process the data to generate output to actuators

Learning Approaches

- Traditional learning techniques
  - rather than attempt to predefine and predict a symbolic model of the 'real-world', the robot learns how to operate and how to behave by:
    - supervised learning
      - desired output is known for each set of input settings (e.g. ANN's, DL)
    - reinforcement learning
      - learning by trial and error through performance feedback
- Evolutionary algorithms
  - using genetic algorithms to find good network weights
  - significant problems with evolving real solutions in reasonable time on current mobile robot hardware

Further Reading

- Further reading for some of today's topics:
  - Braitenberg Vehicles
    - Braitenberg, V. :Vehicles: Experiments in Synthetic Psychology, MIT 1986
    - Start here: http://en.wikipedia.org/wiki/Braitenberg_vehicle
  - Motor Schemas and navigation:
    - Arkin, R.; , "Motor schema based navigation for a mobile robot: An approach to programming by behavior," *Robotics and Automation. Proceedings. 1987 IEEE International Conference on* , vol.4, no., pp. 264- 271, Mar 1987
  - Potential Field Approach
    - Khatib, O.; , "Real-time obstacle avoidance for manipulators and mobile robots," *Robotics and Automation. Proceedings. 1985 IEEE International Conference on* , vol.2, no., pp. 500- 505, Mar 1985
  - SSS:
    - Connell, J. HJ., "SSS: A Hybrid Architecture Applied to Robot Navigation," Proc. IEEE Int'l Conf. Robotics and Automation, IEEE Comp. Soc. Press, Los Alamitos, Calif., 1992, pp. 2,719- 2,724.

Kahoot!

https://kahoot.it/

Lecture Summary

- Summary
  - Module outline
  - What is an autonomous robot?
  - Foundations of robotics
    - agents, simulation and intelligence
    - robots and their components
    - tasks in robotics & general purpose robots?
    - environments (structured, unstructured, and partially structured)
  - Architectures and Behaviours
    - control models
      - the *sense-think-act* control cycle
      - model-based controllers
    - reactive robotics
      - behaviour-based controllers
      - other approaches, hybrid controllers & learning robots
- Next lecture: Brooks' Subsumption Architecture & Robot Hardware