Computational Principles of Mobile Robotics

Overview and motivation

The Encyclopedia Galactica defines a robot as a mechanical apparatus designed to do the work of a man. The marketing division of the Sirius Cybernetics Corporation defines a robot as “Your Plastic Pal Who’s Fun to Be With”

- Douglas Adams

What are robots?

- The word robot was introduced by the writer Karl Capek in his play Rossums’ Universal Robots (RIR) in 1920.
    - The term was used to describe artificial people (factory workers).
- Autonomous and automation have their roots in the Greek for *self-willed*.

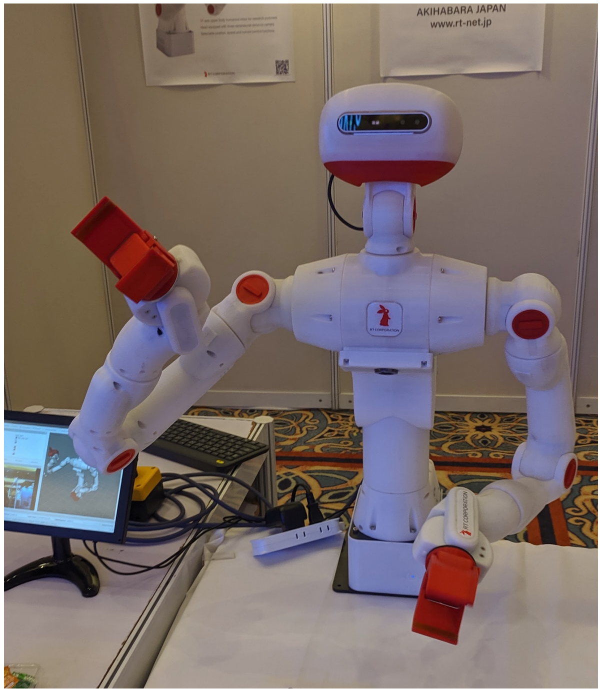
What is a robot?

- It is a machine designed to perform a complex sequence of actions.
- Critical aspects
  - Ability to perform actions (act on the environment).
  - Ability to perceive/understand the world sufficiently well to perform these actions (sensing).
  - Ability to interact with the environment (planning, representation).
- Can (perhaps artificially) break robots into two primary groups
  - Manipulators (robot arms, typically fixed).
  - Mobile robots (robots that move about their environment).

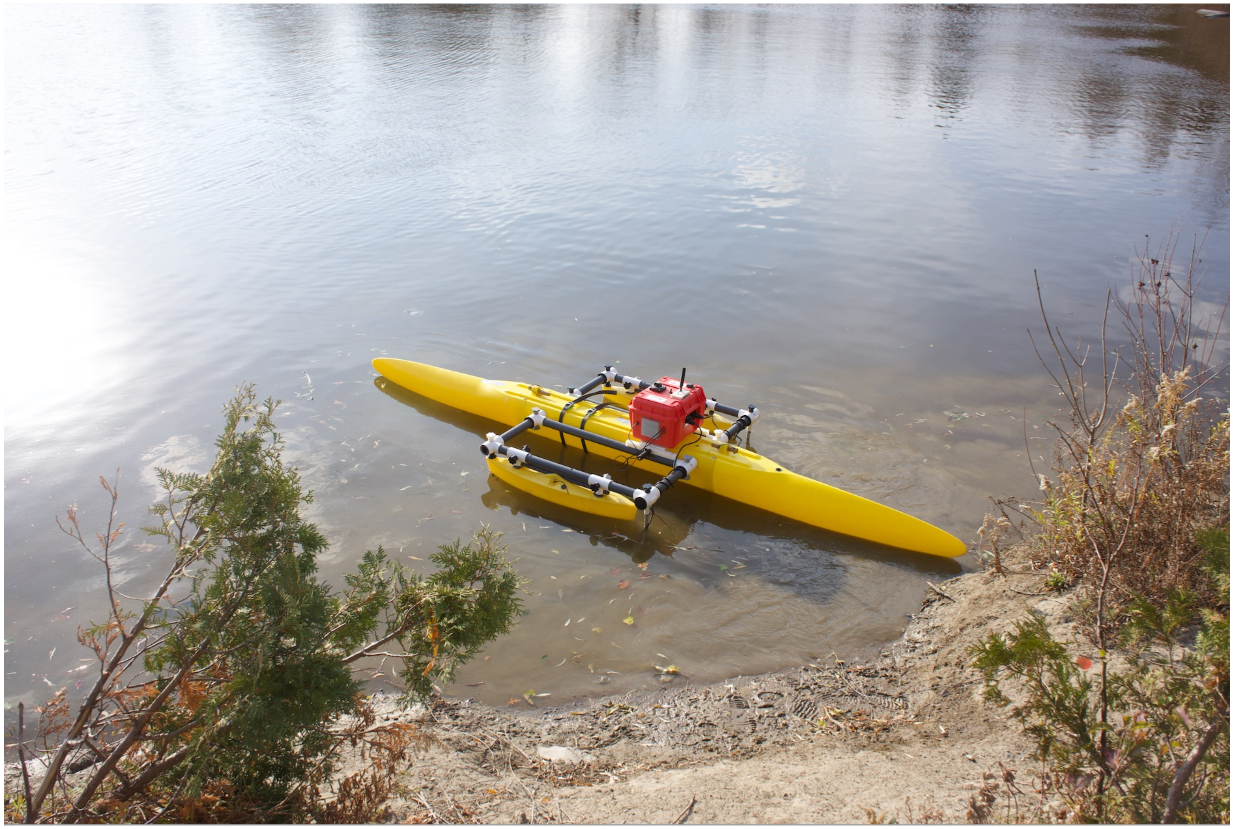
From mechanisms to computation

- Environments well suited for robotic tasks are –
  - Inhospitable.
  - Remote.
  - Tasks have a high duty cycle or high fatigue factor.
  - Tasks are highly disagreeable to a human.

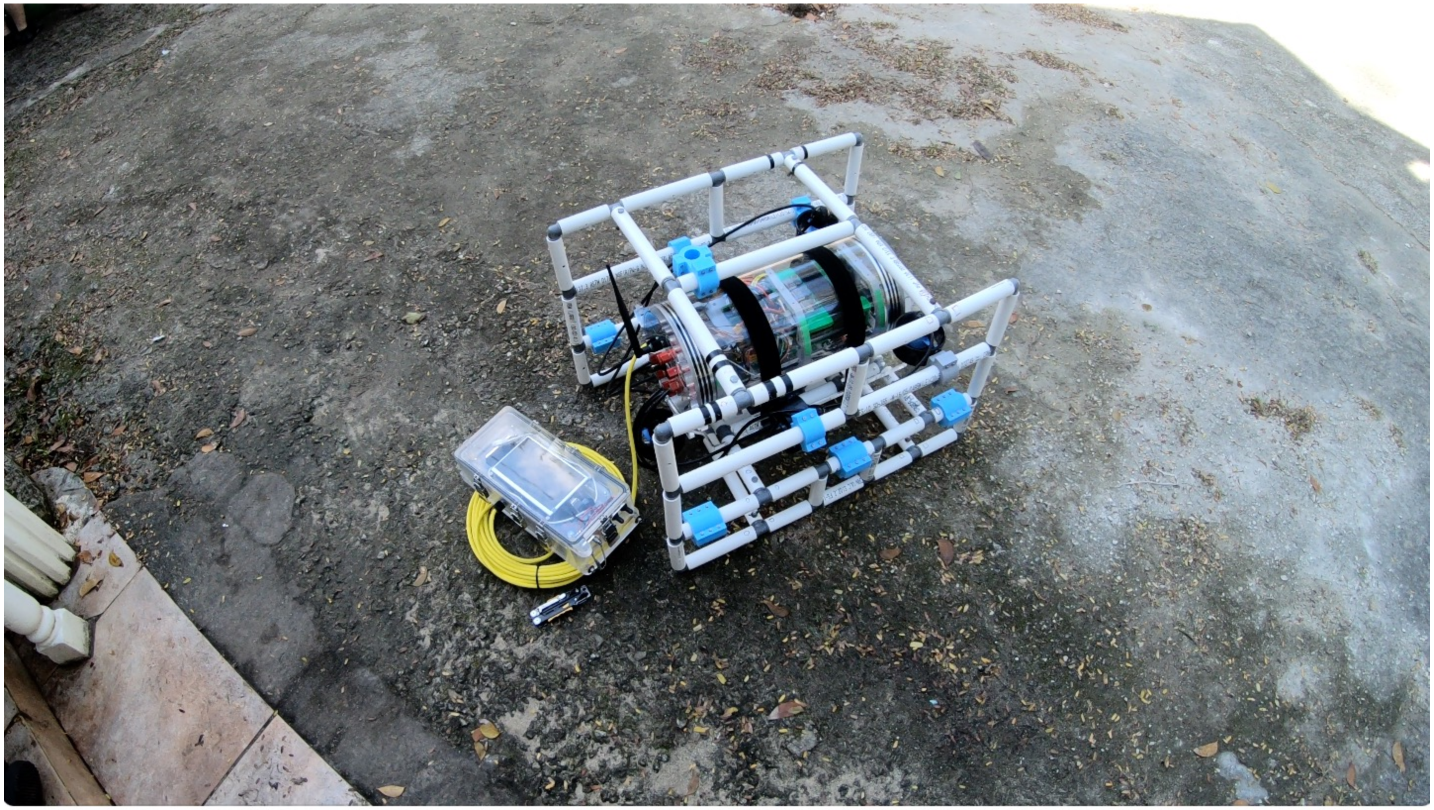
Some examples



Sciurus17 robot (anthropomorphic manipulator robot)



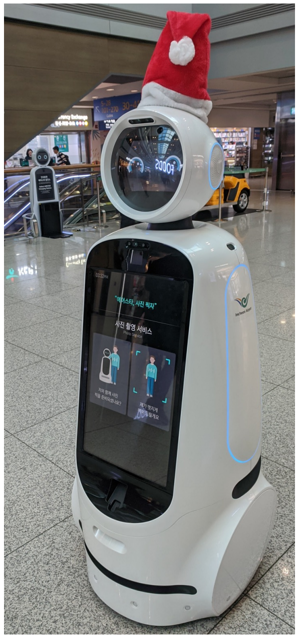
Unmanned surface vessel (Eddy)



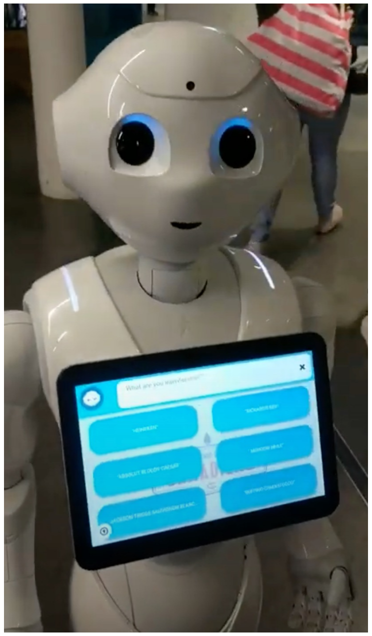
Milton: Unmanned underwater vehicle

Dragonball – a spherical rolling robot

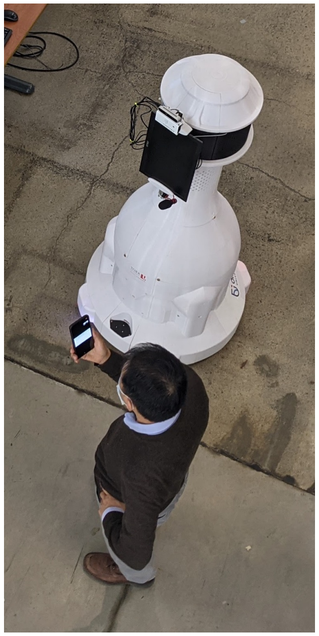
Mindar (Buddhist priest)



LG Airport Guide Robot



Pepper



Sentrynet Robot

Social Interaction Robots

AVIDBOTS floor cleaning robot

And of course, many, many more examples

- There exist robots that crawl, walk, float, swim, fly, …
- There exist robots that are very large (gantry robots) and very small (microrobots).
- There exist robots that are made of rigid parts and robots that are highly malleable (soft).
- We will encounter many such examples throughout the course.

1.1 From mechanisms to computation

- Can consider robots at various different levels
  - Physical level (power, motors, computers, sensors, displays, communication infrastructure).
  - System level (abstracts of the physical components, software infrastructure).
  - Computational level (algorithms that enable a robot to move, sense, reason).
  - Cognitive level (as an intelligent abstract agent able to reason and act on its environment).
- Here we will concentrate on the computational level.

1.2 Historical context

- Robots can be found in fiction in many cultures
  - Talos from ancient Greece.
  - Golem in Jewish folklore.
  - Robotic elephants in the stories of Jules Verne.
  - And so on. See Appendix A of te text.
- And as actual devices
  - Karakuri tea robots of Japan.
  - Leonardo’s efforts in Europe.
  - Ingenious devices in Arabia as recorded by Ibn Al-Razzaz Al-Jasari.
  - And so on…

Asimov’s three laws of robotics

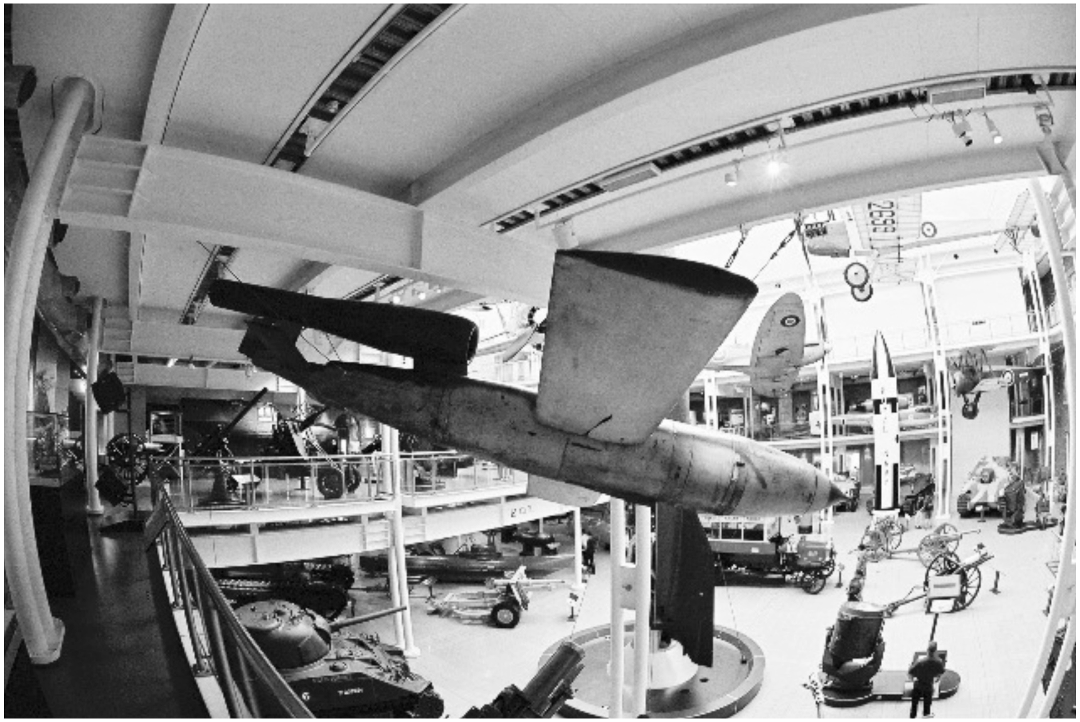
1. A robot may not injure a human being or, through inaction, allow human being to come to harm.
2. A robot must obey the order given it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

First appeared in Runaround, in the March 1942 edition of Astounding Magazine

It has long been clear to me that the modern ultra-rapid computing machine was in principle an ideal central nervous system to an apparatus for automatic control; and its input and output need not be in the form of numbers or diagrams, but might very well be, respectively, the readings of artificial sensors such as photoelectric cells or thermometers, and the performance of motors or solenoids.

Norbert Wiener, January 1949, *Electronics*

Norbert Wiener is considered to be the founding father of *cybernetics.*



V1 – hanging in the Imperial War Museum, London
Built in the 1940's, the V1 could travel over 150 miles autonomously with an accuracy of about 7 miles

Historical context

- Late 1960’s to early 1970’s saw the development of a number of modern autonomous systems –
  - Shakey, Stanford Cart, CMU Rover, Hilare – wheeled systems
  - General Electric Quadruped – walking systems
- From the mid 1980’s there has been a huge explosion in system design and the emergence of standard robotic platforms.

1.3 Biological inspiration

- Robots (often) take inspiration from biological systems
  - It must be possible to do it this way, ants do.
  - Many different motivations for this, but one practical one: robots must operate under the same basic constraints as their biological counterparts.
- Robots (often) provide inspiration to biologists
  - We can validate a particular model from biology by building a machine that implements the model and testing it in the real world.

1.4 Operational regimes

- Fully autonomous
  - The device operates on its own.
- Semi-autonomous
  - The device operates autonomously, under the active supervision of a human operator.
- Teleoperated
  - The device operates under the direct control of a human operator.

1.5 Operational modes

- Single robot
  - Self contained for computation.
  - Shares computation with offboard resources.
  - Relies primarily on offboard resources.
- Multiple robots (a swarm or collective)
  - Can be centrally controlled.
  - Can have distributed control.

1.6 Software support for physical robots

- Early efforts to develop robot systems typically utilized hand-built libraries to control their robot.
- Manufacturers developed their own middleware and these were typically incompatible with each other.
- Willow Garage developed ROS to provide support for their PR2 robot and ROS 1.0 was released in January 2010. It has become the de facto standard for experimental robots.
- ROS is currently moving from ROS1 to ROS2. We will use ROS2 for examples here.

1.6 Software support Foundation Models

- As of 2025, the preferred large language model software suite for thr course is “Ollama” which can be used to load and execute various different LLM models (see http://ollama.com).
- Various assignments or activities may used or expect a specific models or system. The current preferred test suites models are “llama3.2-vision” and “llama3.2”, but at the rate things are going these may be obsolete by the time you read this (this was written Jan 27, 2025).
- Ollama can be used on almost any architecture and you are encouraged to try it immediately.

ROS 2

- Although one can run ROS (the 2 assumed throughout) on a number of different platforms, Ubuntu is the standard and will be assumed.
- The current ROS 2 standard is ROS Jazzy Jalisco and the primary underling OS is Ubuntu 24.04 (Noble). It is possible to run it on other platforms, with more or less effort.
- ROS support software development with a range of tools. We will use python3.
- A number of good references for ROS exist, as does a large on-line set of tutorials.