AI Ethics

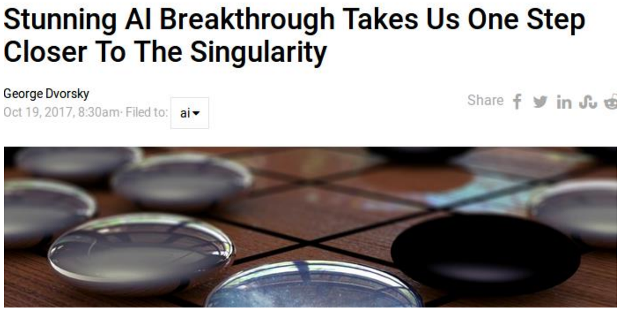
AI Ethics and Risks

- People might lose their **jobs**
  - AI creates wealth and does dangerous and boring jobs for us
- Accountability loss: **who is responsible**, AI, owner, creator?
  - Similar issues elsewhere (medicine, software, plane crash)
- AI reproducing our **negative biases** and attitudes (e.g. racism)
  - AI should share our positive values

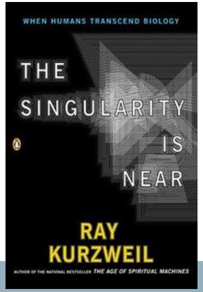
- Use of AI as **weapon** (e.g. drones)
  - Can also save lives? Every beneficial invention can be misused

AI Ethics and Risks

- AI Success might end of the human era
  - Kurtzweil, Musk, Hawking!
  - Once machine surpasses human intelligence it can design smarter machines.
  - Intelligence explosion and **singularity** at which human era ends



- Many counter arguments
  - limits to intelligence
  - nothing special about human intelligence
  - computational complexity
  - "intelligence to do a task" ≠ "ability to improve intelligence to do a task"



Robotics Laws

The Three Laws of Robotics [Azimov 1942]

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 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

- A robot may not injure humanity, or, through inaction, allow humanity to come to harm

UK Principles of Robotics [EPSRC 2011]

1. Robots are multi-use tools. Robots should not be designed solely or primarily to kill or harm humans, except in the interests of national security.
2. Humans, not robots, are responsible agents. Robots should be designed & operated as far as is practicable to comply with existing laws & fundamental rights freedoms, including privacy.
3. Robots are products. They should be designed using processes which assure their safety and security.
4. Robots are manufactured artefacts. They should not be designed in a deceptive way to exploit vulnerable users; instead their machine nature should be transparent.
5. The person with legal responsibility for a robot should be attributed.

Summary

- How to think or how to behave? Being like humans or being rational?
  - **This course about acting rationally**

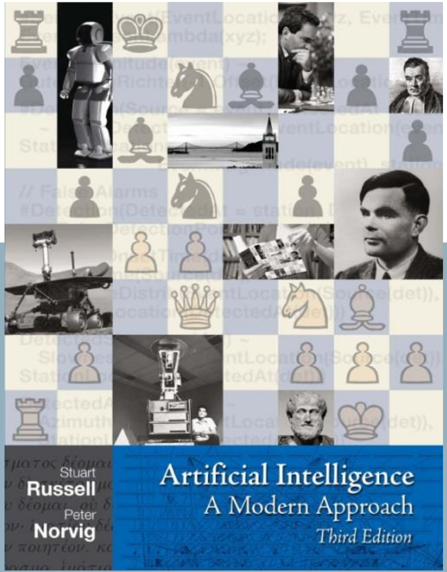
- AI related to many fields including philosophy, mathematics, economics, neuroscience, psychology, computer sci. and control theory
- 50+ years of progress along many different paradigms: logic, expert systems, neural nets, learning, probabilities
- Increasingly scientific: focus on experimental comparisons and theoretical foundations

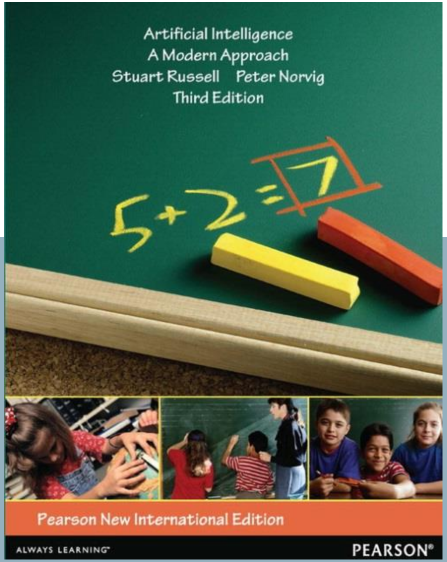
- **AI is a high-risk high-gain area with major ethical implications**

Australian National University

Intelligent Agents

Chapter 2

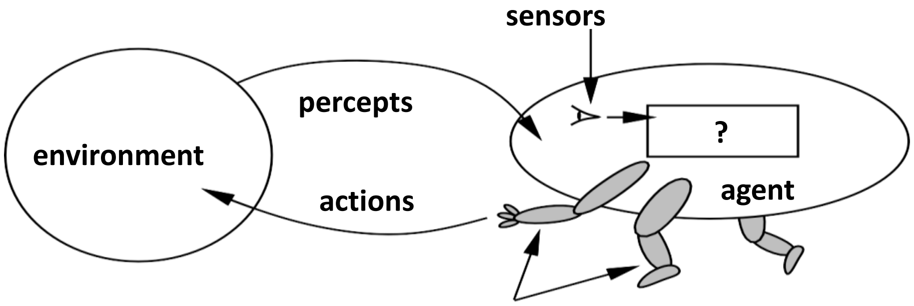




Outline

- Agents and environments
- Rationality
- PEAS (Performance measure, Environment, Actuators, Sensors)
- Environment types
- Agent types

Agents and Environments

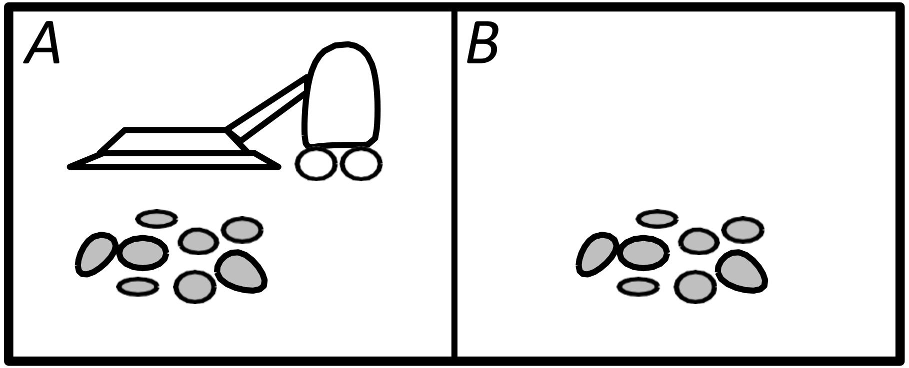


- **Agents** include humans, robots, softbots, thermostats, etc.
- **Percept** refers to the agent perceptual input at any given instant
- The **agent function** maps from percept histories to actions:

$$f: P^* \rightarrow A$$

- The **agent program** implements f on the physical **architecture.**

Vacuum-cleaner World



- Percepts: current location and its content, e.g., *(A, Dirty)*
- Actions: *Left, Right, Suck, NoOp*

A Vacuum-cleaner Agent

Percept sequence	Action
(A, Clean)	*Right*
(A, Dirty)	*Suck*
(B, Clean)	*Left*
(B, Dirty)	*Suck*
(A, Clean), (A, Clean)	*Right*
(A, Clean), (A, Dirty)	*Suck*
...	...

function REFLEX-VACUUM-AGENT($(location, status)$) **returns** an action

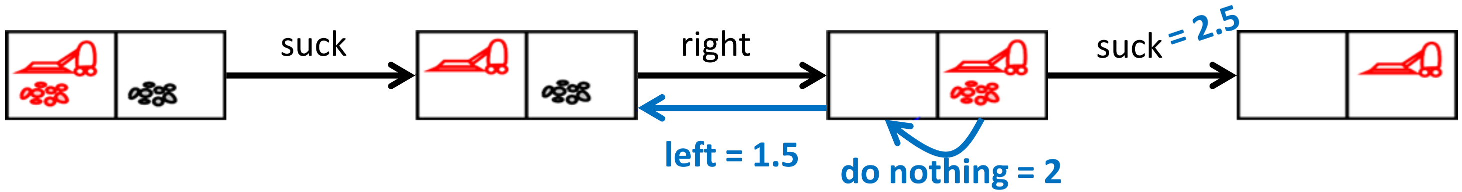
  if $status = Dirty$ **then return** $Suck$
  else if $location = A$ **then return** $Right$
  else if $location = B$ **then return** $Left$

- What is the **right** function f ?
- Can it be implemented in a small agent program?

Rationality

The **performance measure** evaluates the **environment sequence**

- one point per room cleaned up within T time steps?
- one point per clean room per time step, minus half a point per action?
- penalize for > k dirty rooms?



A **rational agent** chooses whichever action maximizes the **expected** value of the performance measure **given the percept sequence to date**

- Rational ≠ omniscient
  - percepts may not supply all relevant information
- Rational ≠ clairvoyant
  - action outcomes may not be as expected
- Hence, rational ≠ successful

PEAS

To design a rational agent, we must specify the **task environment**

Consider, e.g., the task of designing a **driverless taxi**:

- **P**erformance measure:
  –
- **E**nvironment:
  –
- **A**ctuators:
  –
- **S**ensors:
  –

Internet shopping agent

Consider, e.g., the task of designing an **internet shopping bot**:

- **P**erformance measure:
  -
- **E**nvironment:
  -
- **A**ctuators:
  -
- **S**ensors:
  -

Properties of Task Environments

- **Fully vs partially observable**
  - do the agent sensors give access to all relevant information about the environment state?
- **Deterministic vs stochastic**
  - is the next state completely determined by the current state and executed action?
- **Known vs unknown**
  - does the agent know the environment's laws of physics?
- **Episodic vs sequential**
  - is the next decision independent of the previous ones?
- **Static vs dynamic**
  - can the environment change whilst the agent is deliberating?
  - **Semi-dynamic:** only the performance score changes.
- **Discrete vs continuous**
  - can time, states, actions, percepts be represented in a discrete way?
- **Single vs multi-agent**
  - is a single agent making decisions, or do multiple agents need to compete or cooperate to maximise interdependent performance measures?

Environment types

	Crossword	Poker	Part picking robot	Taxi
Observable	Yes	No		
Deterministic	Yes	No		
Known	Yes	Yes		
Episodic	No	No		
Static	Yes			No
Discrete	Yes	Yes	No	No
Single-agent	Yes		Yes	

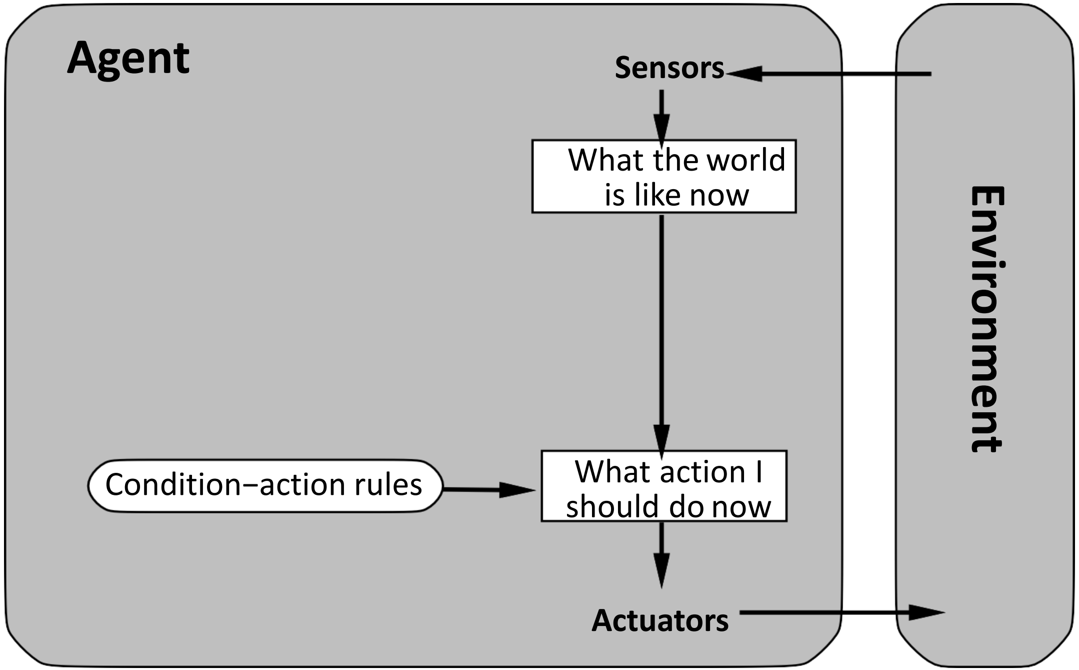
The environment type largely determines the agent design

The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent.

Agent types

- Four basic types of agents in order of increasing generality:
  - simple reflex agents
  - reflex agents with state
  - goal-based agents
  - utility-based agents
- All these can be turned into learning agents

Simple reflex agents



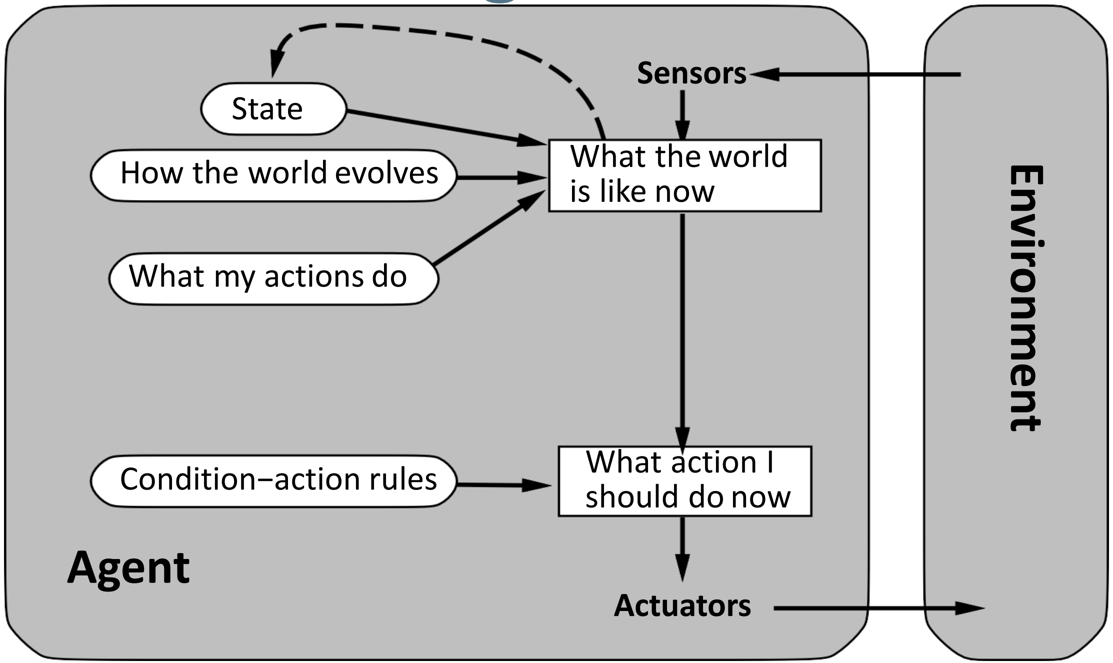
Decisions are made based on the current percept only. Raises issues for partially observable environments.

Example

function REFLEX-VACUUM-AGENT($(location, status)$) **returns** an action

if $status = Dirty$ **then return** $Suck$
else if $location = A$ **then return** $Right$
else if $location = B$ **then return** $Left$

Reflex agents with state



The internal state keeps track of relevant unobservable aspects of the environment. The environment model describes how the environment works (how the environment state is affected by actions)

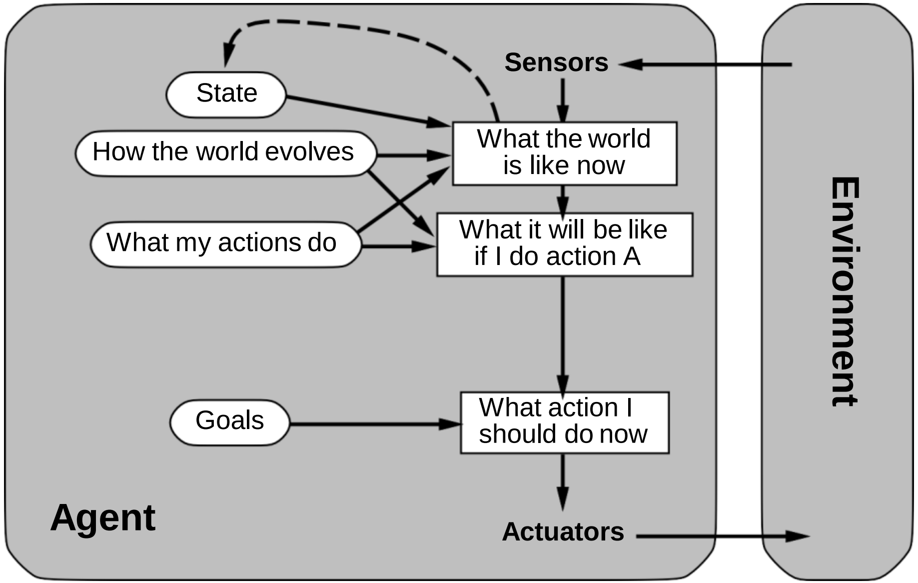
Example

function VACUUM-AGENT-WITH-STATE($(location, status)$) **returns** an action
static: $last\_A$, $last\_B$, numbers, initially ∞

increment $last\_A$ and $last\_B$
if $location = A$ **then** $last\_A = 0$
else $last\_B = 0$
case
  $status = Dirty$:
    return $Suck$
  $location = A$:
    if $last\_B > 3$ **then return** $Right$
    else return $NoOp$
  $location = B$:
    if $last\_A > 3$ **then return** $Left$
    else return $NoOp$

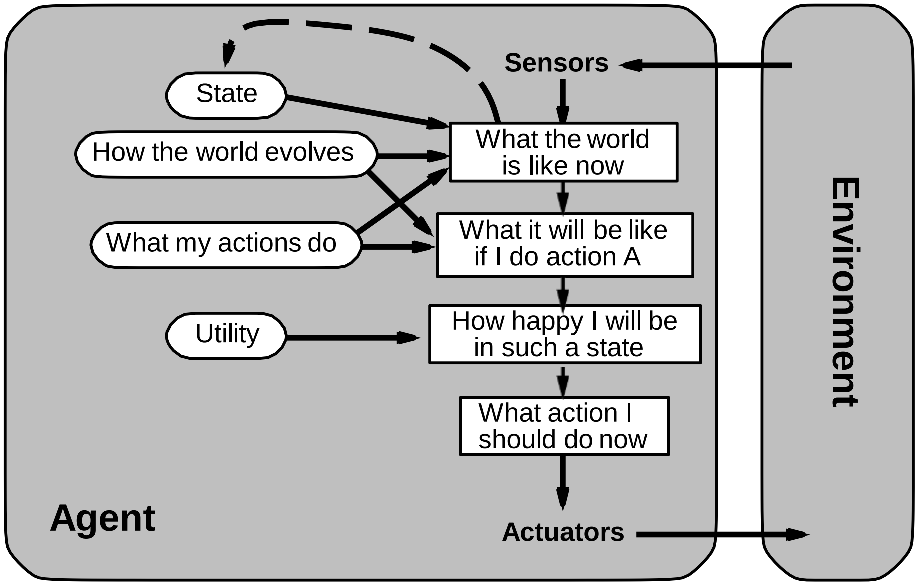
The time passed since a location was visited is a proxy for the likelihood of this location's status changing from clean to dirty.

Goal-Based agents



- The goal describes desirable situations.
- The agent combines goal and environment model to choose actions.
- Planning and search are AI subfields devoted to building goal-based agents.

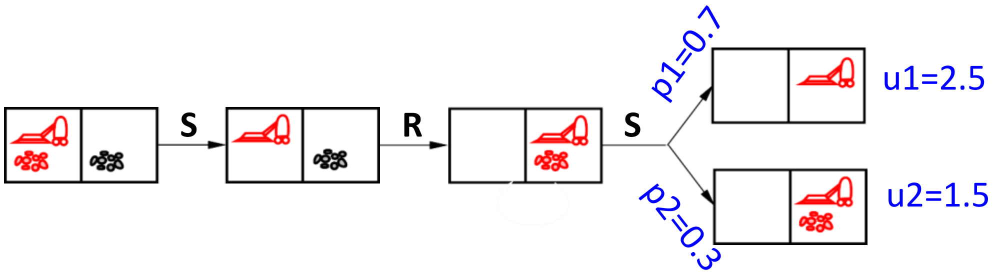
Utility-based agents



- The utility function internalises the performance measure.
- Under uncertainty, the agent chooses actions that maximise the expected utility.

Utility-based agents

Rational agent: chooses the action that maximises expected utility:

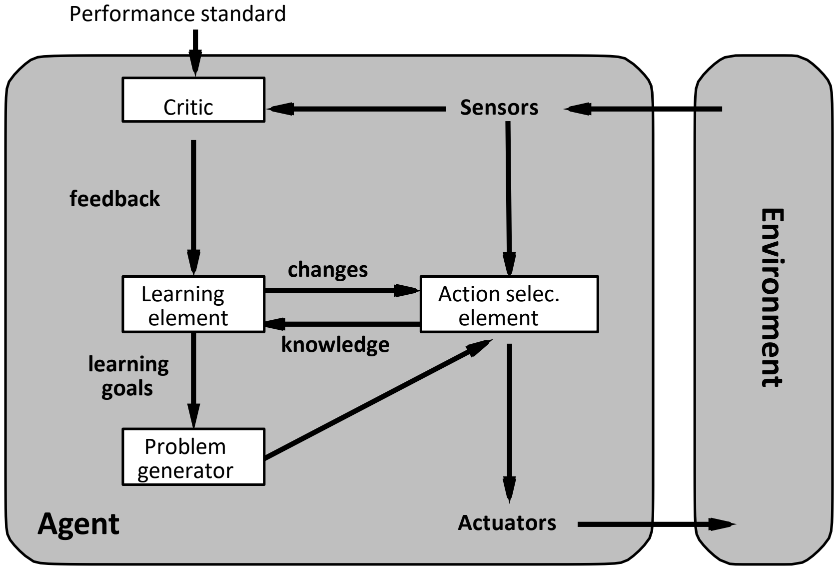


Expected utility of $Suck$:

$$p1 \times u1 + p2 \times u2 = 0.7 \times 2.5 + 0.3 \times 1.5 = 2.2$$

- $Suck$ has an expected utility of 2.2
- $NoOp$ has an expected utility of 2
- $Left$ has an expected utility of 1.5

Learning agents



- The action selection element is what we described earlier.
- The learning element uses feedback from the critic to modify the action selection.
- The problem generator suggests actions that lead to new informative experience.

Exploration vs Exploitation

A fundamental dilemma for learning agents:

- **Exploitation**: greedily uses what the agent has learnt to select the action that will, in the light of the current knowledge, have the best outcome

- **Exploration**: taking some other (possibly random) action to learn more, hoping to find something even better than what is currently known

In practice, agents must explore to avoid getting stuck in severely sub-optimal behaviour, but exploration has a cost.

Typically, a smart agent explores more in early stages than later on