Introduction to Artificial Intelligence

Lecture 1 – Introduction

CS/CNS/EE 154

Andreas Krause

Think like humans

Learning

AI

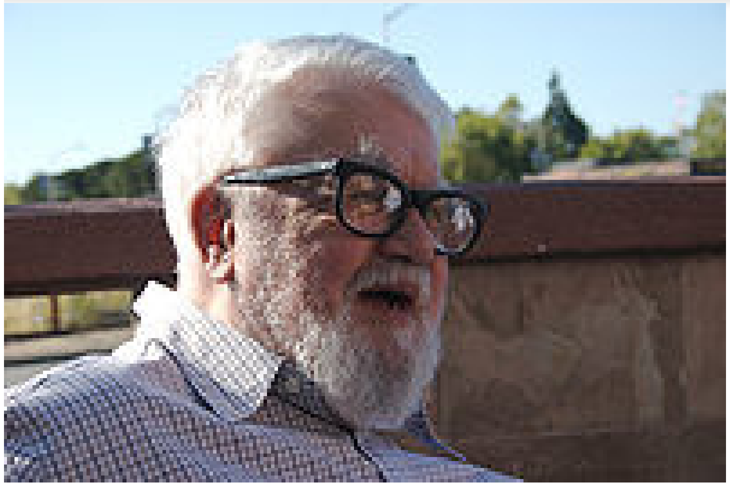
Vision

Games

Speech

NLP

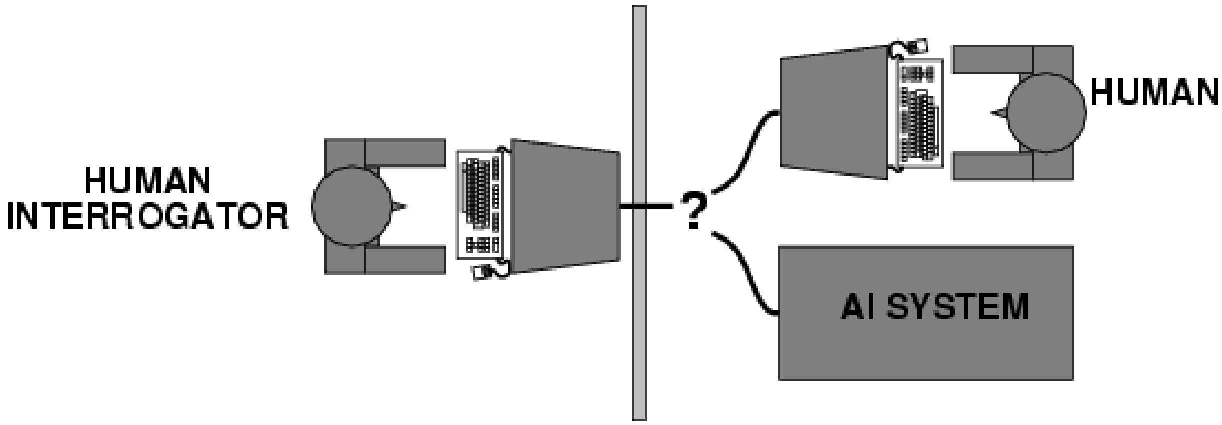
What is AI?



"The science and engineering of making intelligent machines" (McCarthy, '56)

What does "intelligence" mean??

The Turing test



- Turing ('50): Computing Machinery and Intelligence
- Predicted that by 2000, machine has 30% of fooling a lay person for 5 minutes

- Currently, human-level AI not within reach

What if we had intelligent machines?

- Will machines surpass human intelligence?
- Should intelligent machines have rights?
- What will we do with superintelligent machines?
- What will they do with us?
- …

AI today

- ~~Build systems that act intelligently~~
- Build systems that act rationally
- Act rationally = "perform well on some task"

- Amenable to mathematical analysis, empirical evaluation
- Involves / builds on
  - optimization, control theory, statistics, game theory, engineering, …
- This is what this course is about!

- General AI still inspiration for the field!

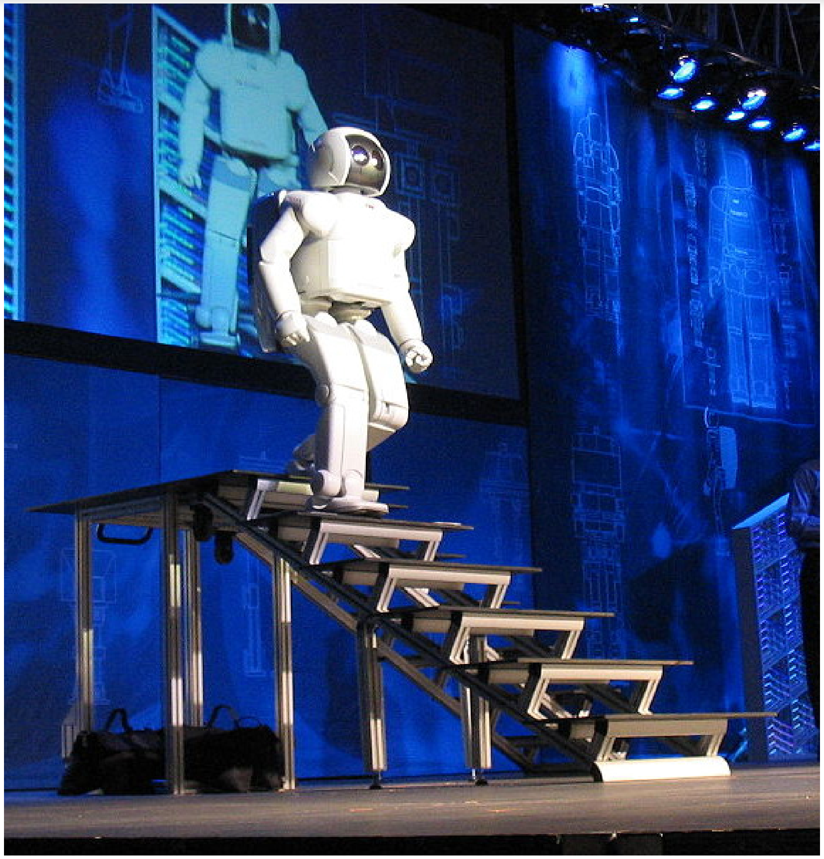
Autonomous driving

Caltech's Alice

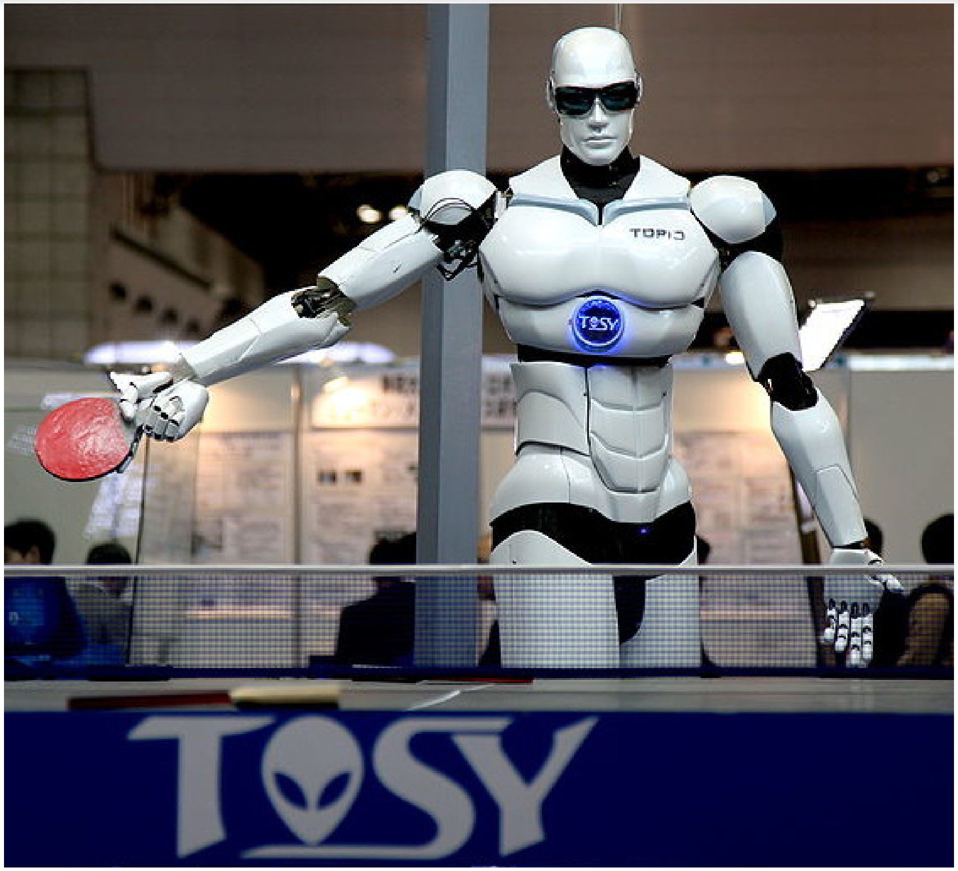
DARPA Grand Challenges:

- 2005: drive 150 mile in the Mojave desert
- 2007: drive 60 mile in traffic in urban environment

Humanoid robotics

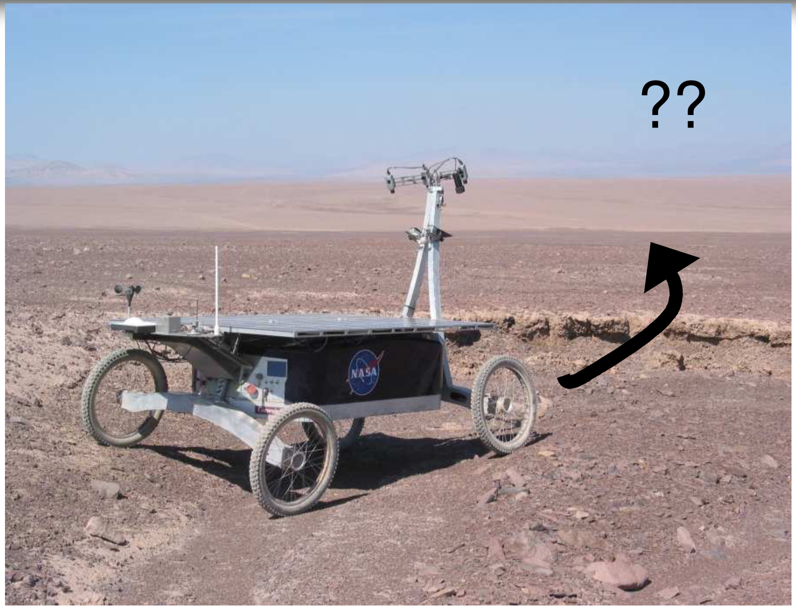


Honda ASIMO



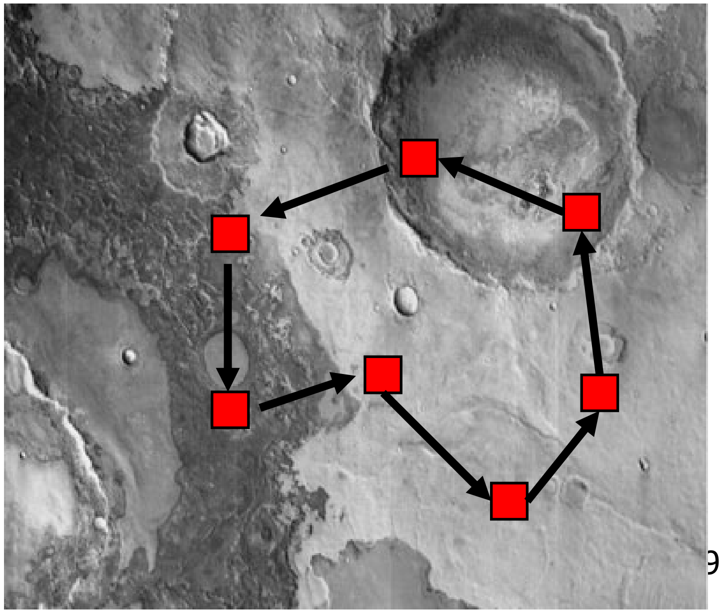
TOSY TOPIO

Autonomous robotic exploration

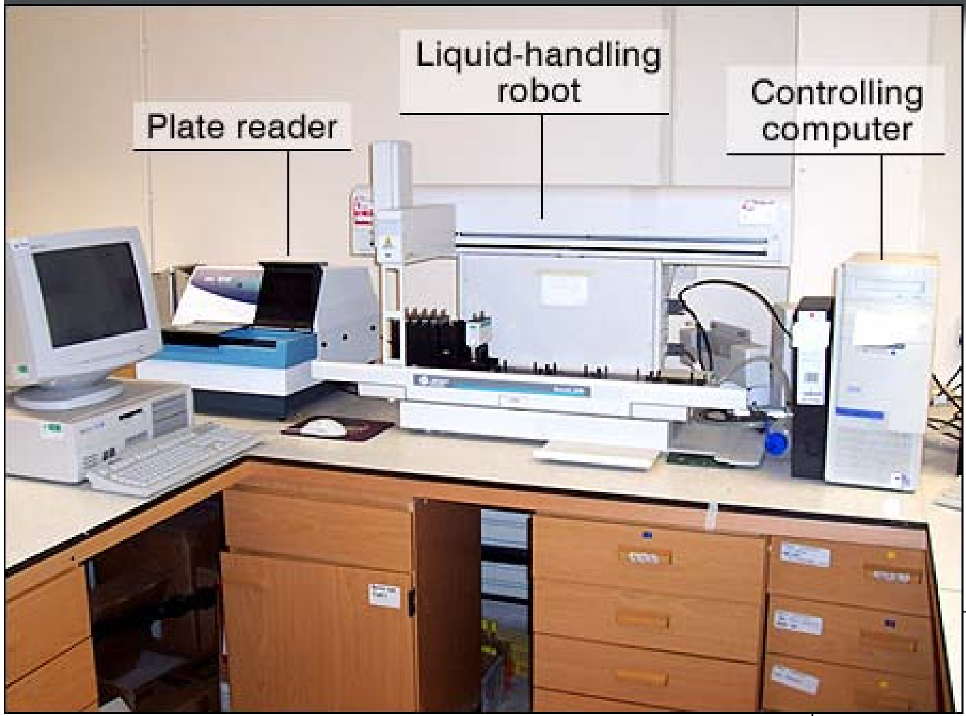


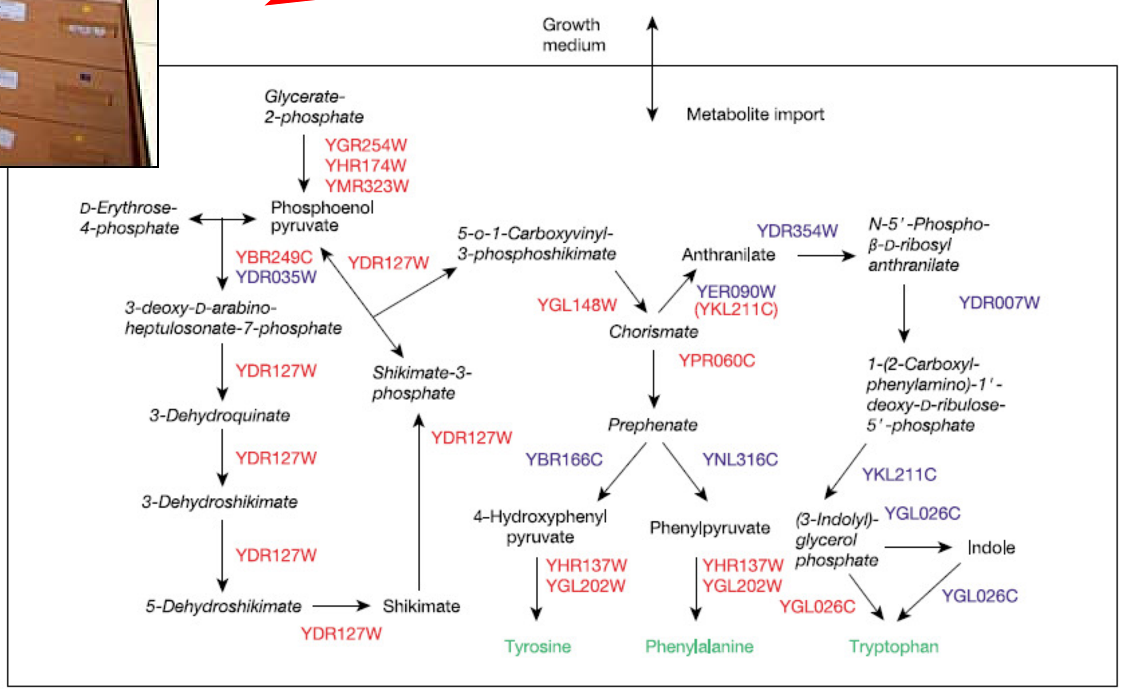
- Limited time for measurements
- Limited capacity for rock samples

- **Need optimized information gathering!**



A robot scientist
[King et al, Nature '04, Science '09]





Games

IBM's Deep Blue wins 6 game match against Garry Kasparov ('97)

Games

- Go: 2008: MoGo beats Pro (8P) in 9-stone game
- Poker: Next big frontier for AI in games

Computer games

NLP / Dialog management
[Bohus et al.]

Reading the web
[Carlson et al., AAAI 2010]

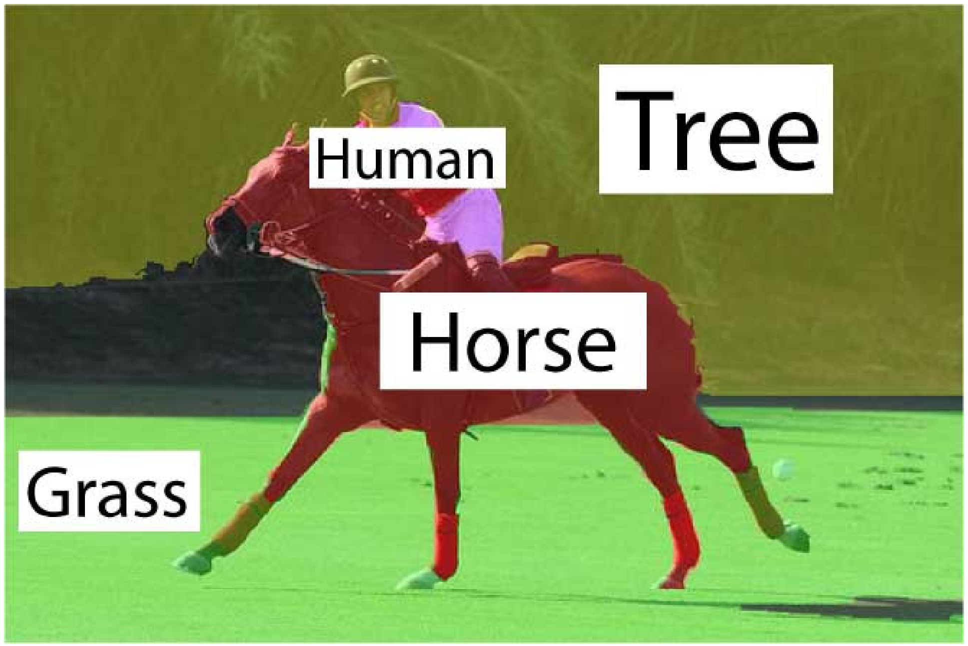
- Never-Ending Language Learner
- After 67 days, built ontology of 242,453 facts
- Estimated precision of 73%

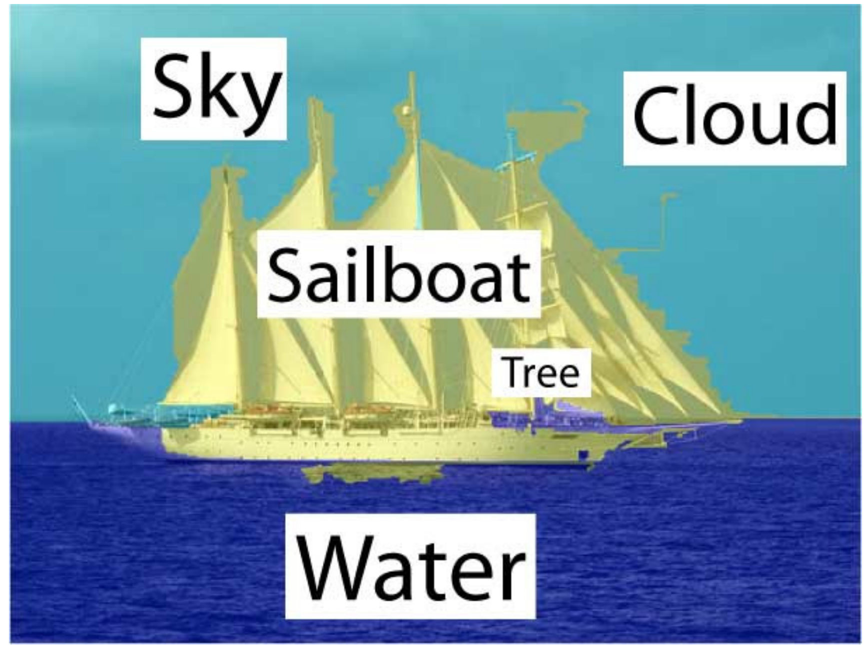
Recently-Learned Facts

Refresh

instance	iteration	date learned	confidence
laura_burkett is a fashion model	149	09-sep-2010	92.8
samsung_b110 is a product	151	15-sep-2010	100.0
tetrapropyl_orthotitanate is a chemical	153	23-sep-2010	100.0
mvp_records is a record label	149	09-sep-2010	100.0
buy_levitra_online is a drug	153	23-sep-2010	100.0

Scene understanding
[Li et al., CVPR 2009]





Topics covered

- Agents and environments
- Search
- Logic
- Games
- Uncertainty
- Planning
- Learning
- Advanced topics
- Applications

Overview

- *Instructor*:
  Andreas Krause (krausea@caltech.edu) and

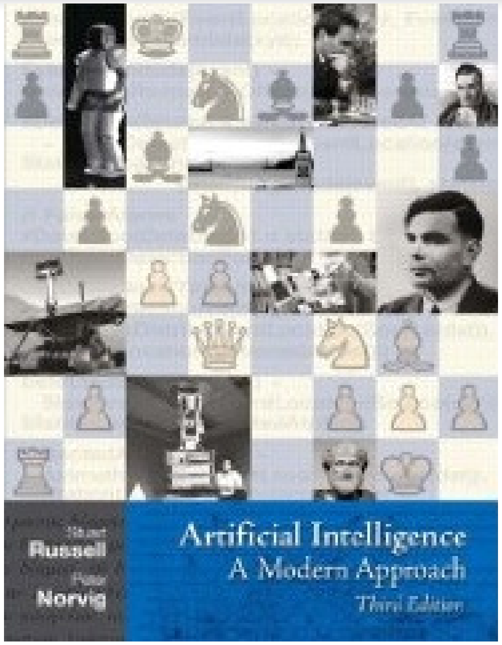
- *Teaching assistants*:
  Pete Trautman (trautman@cds.caltech.edu)
  Xiaodi Hou (xiaodi.hou@gmail.com)
  Noah Jakimo (njakimo@caltech.edu)

- *Administrative assistant*:
  Lisa Knox (lisa987@cs.caltech.edu)

Course material

- Textbook:

  S. Russell, P. Norvig: Artificial Intelligence, A Modern Approach (3rd edition)



- Additional reading on course webpage: http://www.cs.caltech.edu/courses/cs154/

Background & Prequisites

- Formal requirements:
  - Basic knowledge in probability and statistics (Ma 2b or equivalent)
  - Algorithms (CS 1 or equivalent)
- Helpful: basic knowledge in complexity (e.g., CS 38)

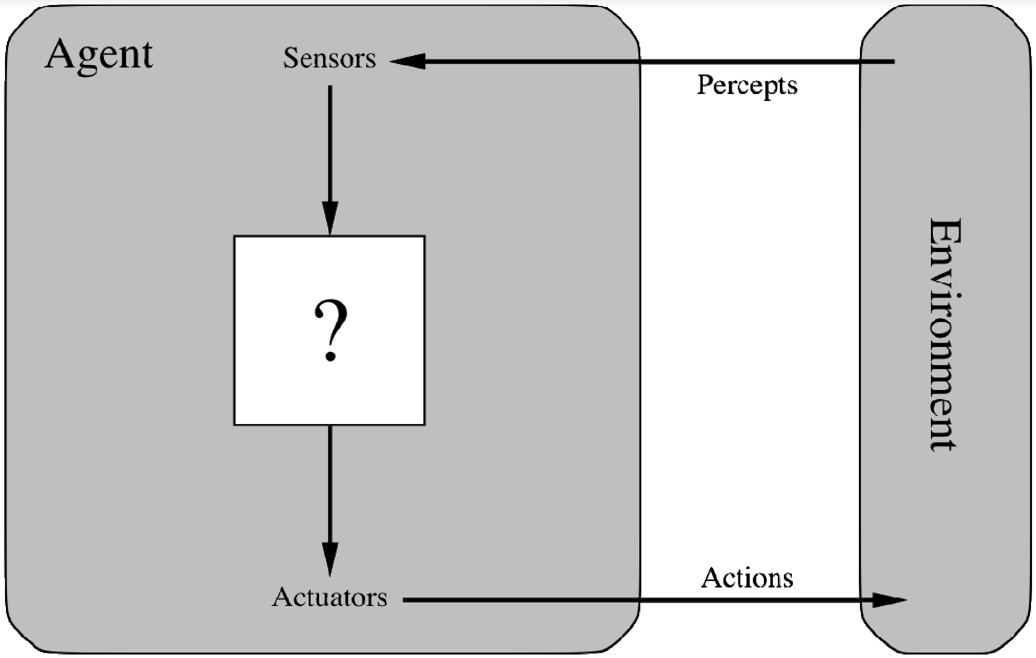
Coursework

- Grading based on
  - 3 homework assignments (50%)
  - Challenge project (30%)
  - Final exam (20%)
- 3 late days, for homeworks only
- Discussing assignments allowed, but everybody must turn in their own solutions
- Exam will be take home open textbook. No other material or collaboration allowed for exam.
- Start early! ☺

Challenge project

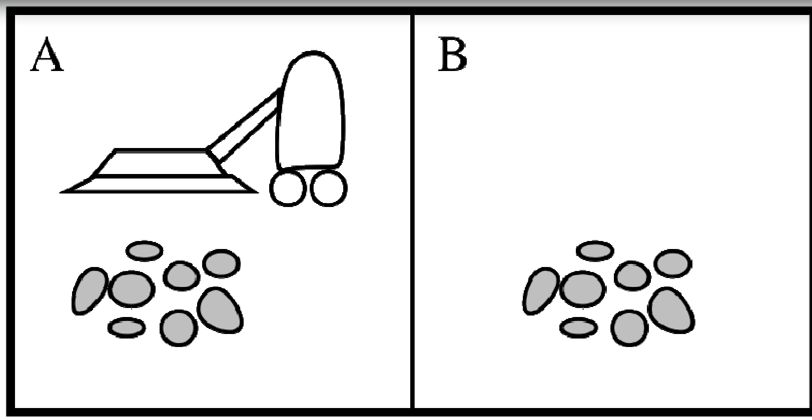
- “Get your hands dirty” with the course material
- More details soon
- Groups of 2-3 students
- Can opt to do independent project
  (with instructors permission)

Agents and environments



- **Agents**: Alice, Poker player, Robo receptionist, …
  - Agent maps sequence of percepts to action
  - Implemented as algorithm running on physical architecture
- **Environment** maps sequence of actions to percept

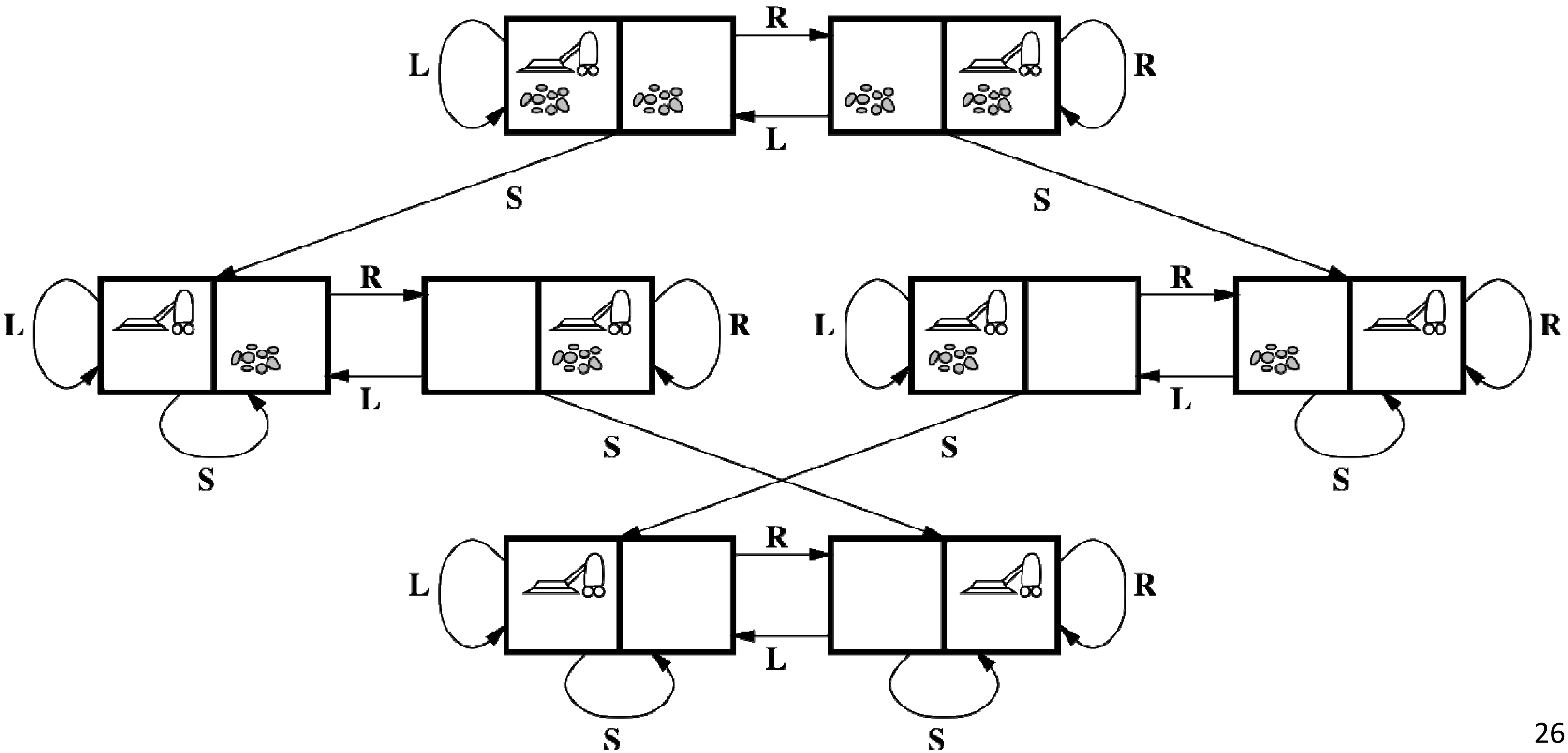
Example: Vacuum cleaning robot



- Percepts P = {[A,Clean], [A,Dirty], [B,Clean], [B,Dirty]}
- Actions A = {Left, Right, Suck, NoOp}
- Agent function: $f : P^* \rightarrow A$
- Example: $$f([A, dirty]) = Suck$$ $$f([A, clean]) = Right$$ $$f([A, clean], [B, dirty]) = Suck$$

Modeling the environment

- Set of states S (not necessarily finite)
- State transitions depend on current state and actions (can be stochastic or nondeterministic)

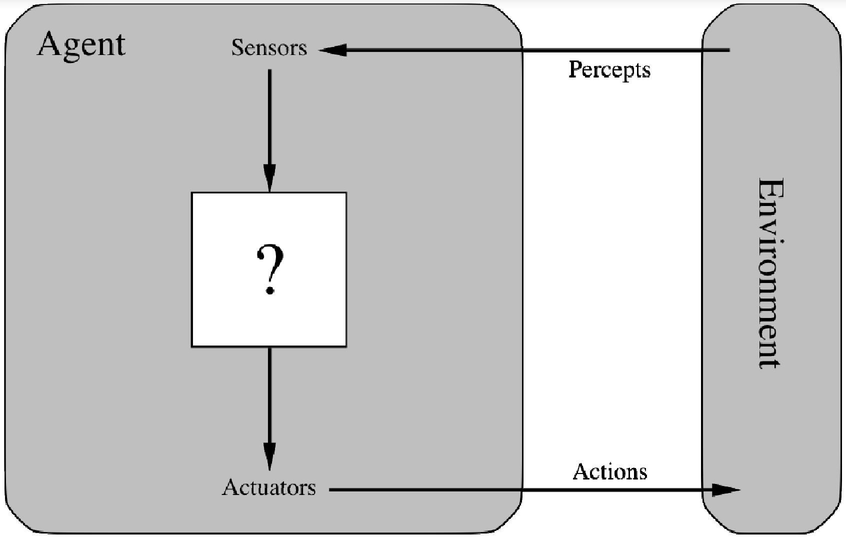


Rationality: Performance evaluation

- Fixed performance measure

$$R : S^* \to \mathbb{R}$$

evaluates environment seq.



- For example:
  - One point for each clean square after 10 rounds?
  - Time it takes until all squares clean?
  - One point per clean square per round, minus one point per move
- **Goal**: find agent function (program) to maximize performance

PEAS: Specifying tasks

To design a rational agent, we need to specify **P**erformance measure, **E**nvironment, **A**ctuators, **S**ensors.

Example: Chess player

Performance measure:	2 points/win, 1 points/draw, 0 for loss
Environment:	Chess board, pieces, rules, move history
Actuators:	move pieces, resign
Sensors:	observe board position

PEAS: Specifying tasks

Example: Autonomous taxi

Performance measure:	safety, fare, fines, satisfaction, …
Environment:	road network, traffic rules, other cars, lights, pedestrians, …
Actuators:	steer, gas, brake, pick up, …
Sensors:	cameras, LIDAR, weight sensor, ..

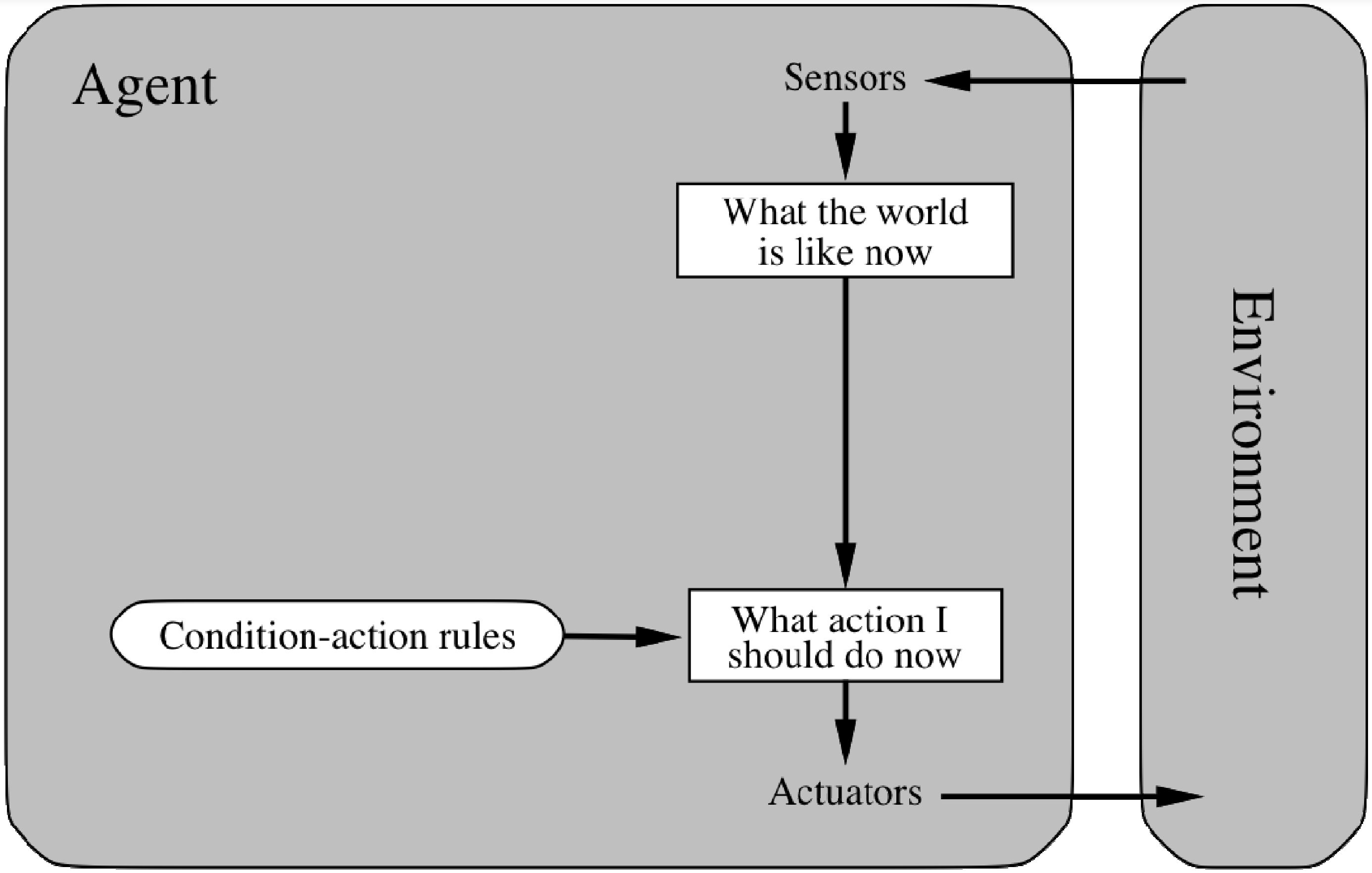
Environment types

	Sudoku	Poker	Spam Filter	Taxi
Observable?	Y	N	N	N
Deterministic?	Y	N	N	N
Episodic?	N	N	Y ?	N
Static?	Y	Y	Y ?	N
Discrete?	Y	Y	Y	N
Single-agent?	Y	N	Y ?	N

Agent types

- In principle, could specify action for any possible percept sequence
  - Intractable
- Different types of agents
  - Simplex reflex agent
  - Reflex agents with state
  - Goal based agents
  - Utility based agents

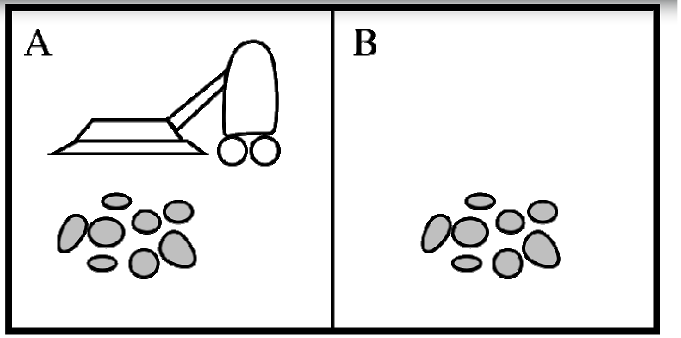
Simple reflex agent



Action only function of last percept

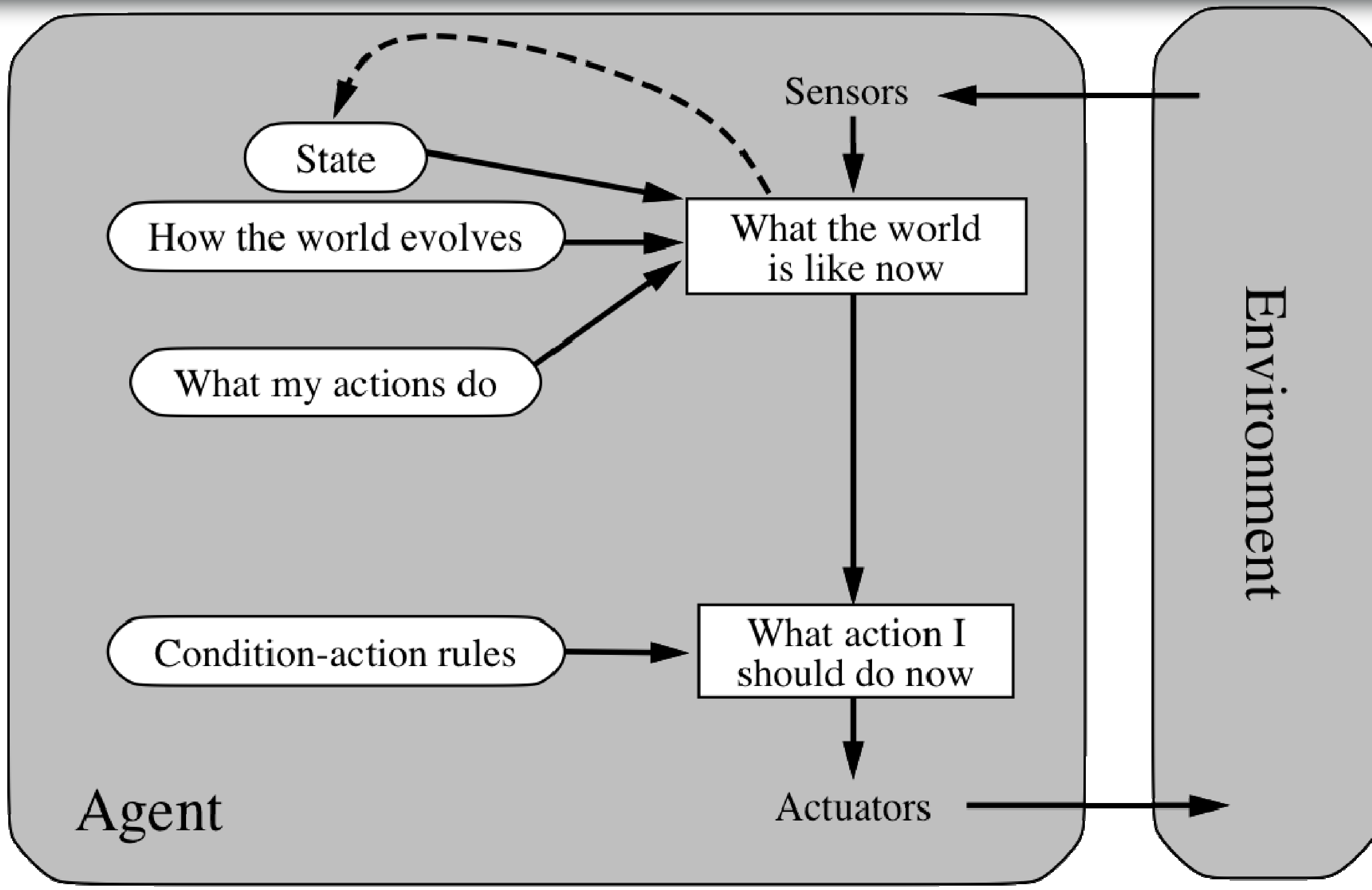
Example

Percept	Action
[A,dirty]	Suck
[B,dirty]	Suck
[A,clean]	Right
[B,clean]	Left



- Will never stop (noop), since we can't remember state
- This is a fundamental problem of simple reflex agents in partially observable environments!

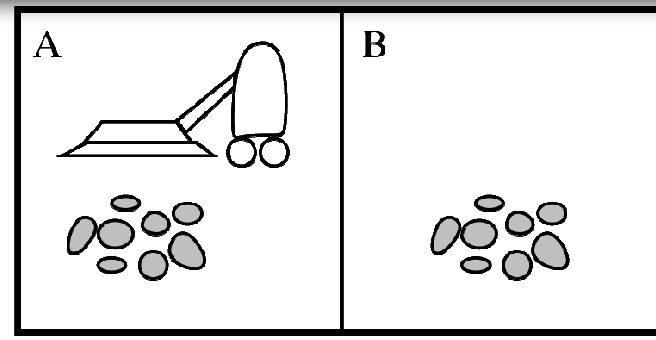
Reflex agent with state



Action function of percept and internal state

Example

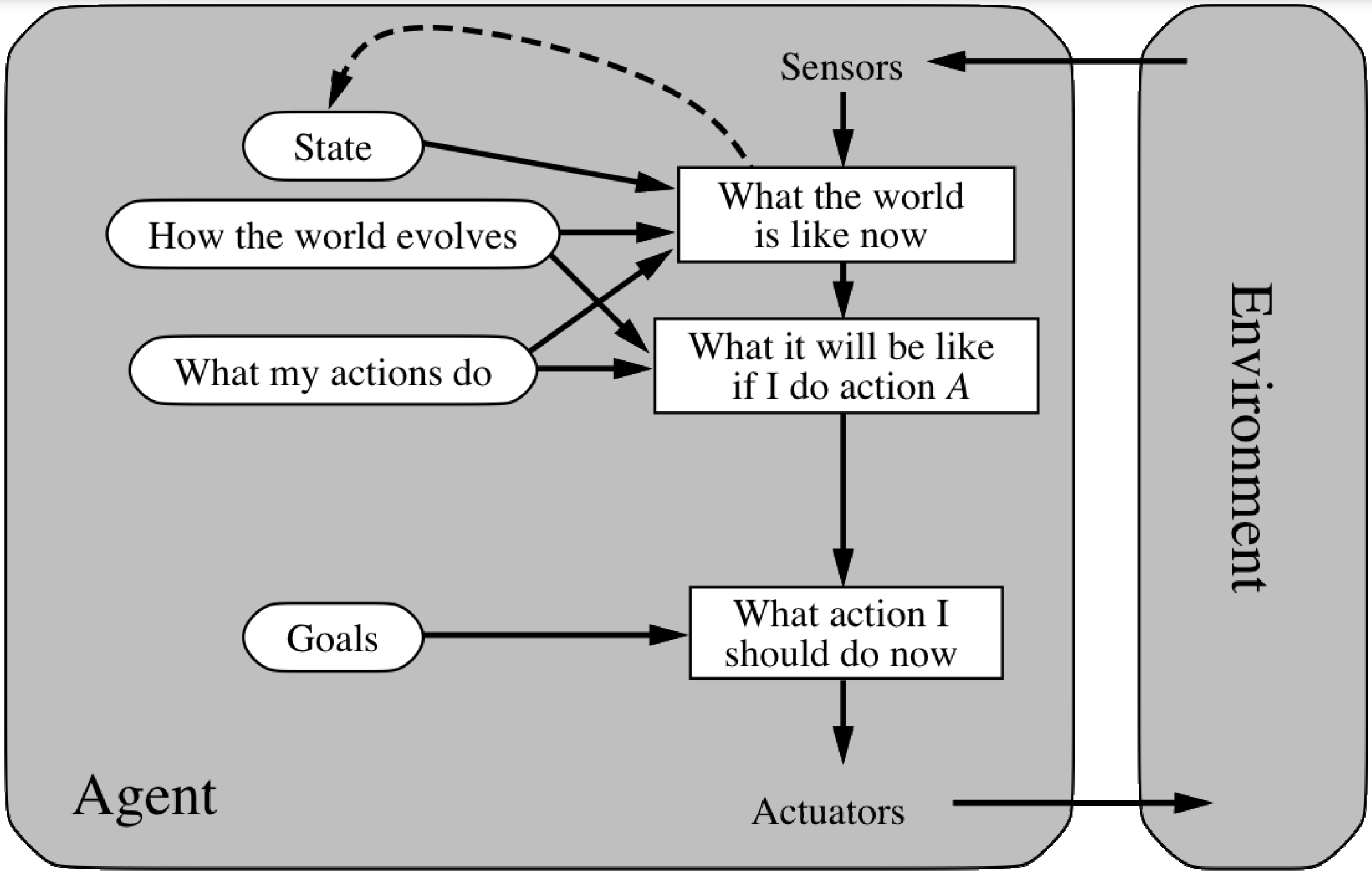
State vars: cleanA = cleanB = false



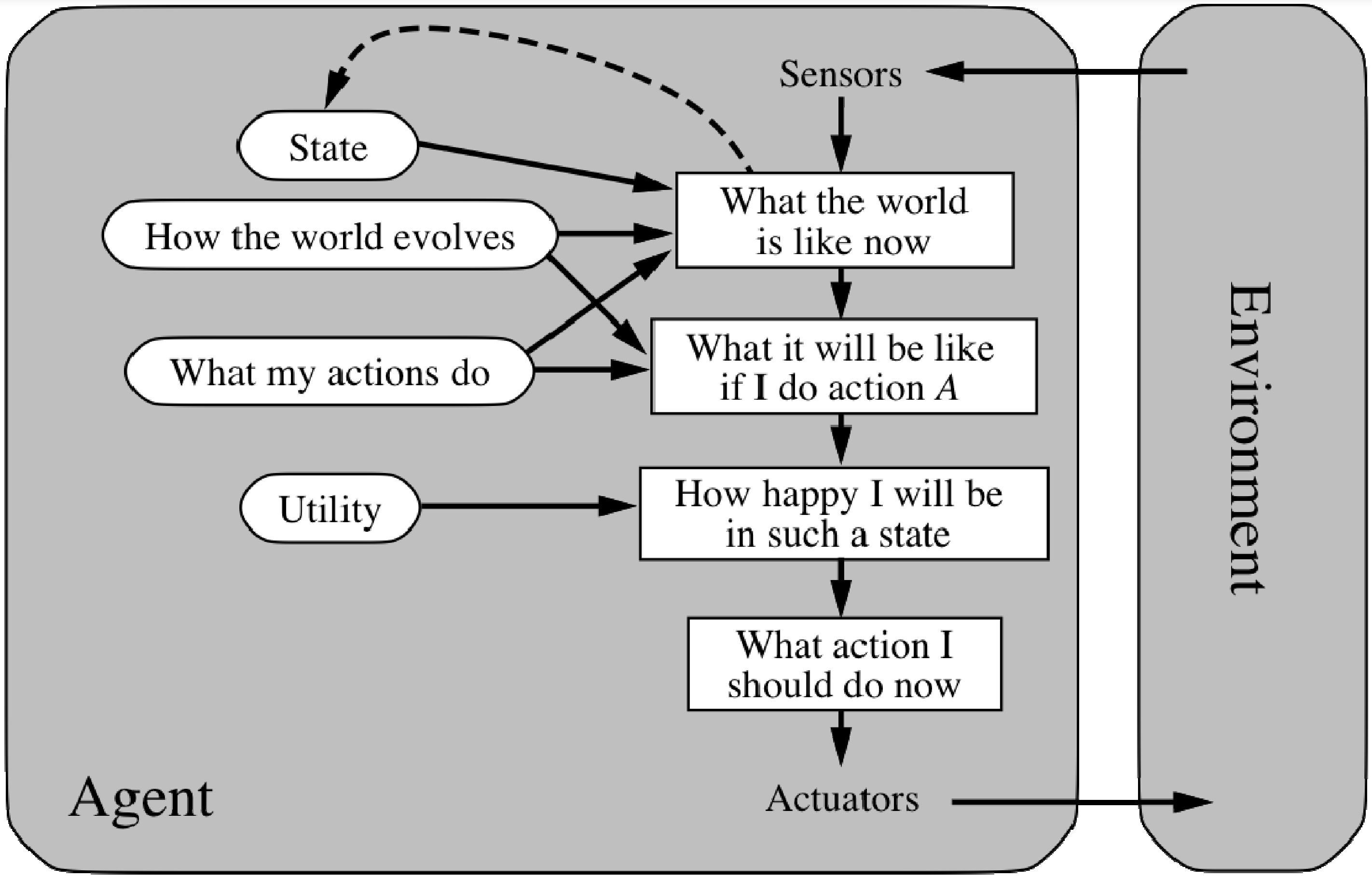
Percept	cleanA	cleanB	Action	State change
[X,dirty]	?	?	Suck	cleanX = true
[A,clean]	?	true	NoOp	
[A,clean]	?	false	Right	
[B,clean]	true	?	NoOp	
[B,clean]	false	?	Left	

? means “don’t care”

Goal-based agents



Utility-based agents



What you need to know

- Agents interact with the environment using sensors and actuators
- Performance measure evaluates environment state sequence
- A perfectly rational agent maximizes (expected) performance
- PEAS descriptions define task environments
- Environments categorized along different dimensions
  - Observable? Deterministic? …
- Basic agent architectures
  - Simple reflex, reflex with state, goal-based, utility-based, …