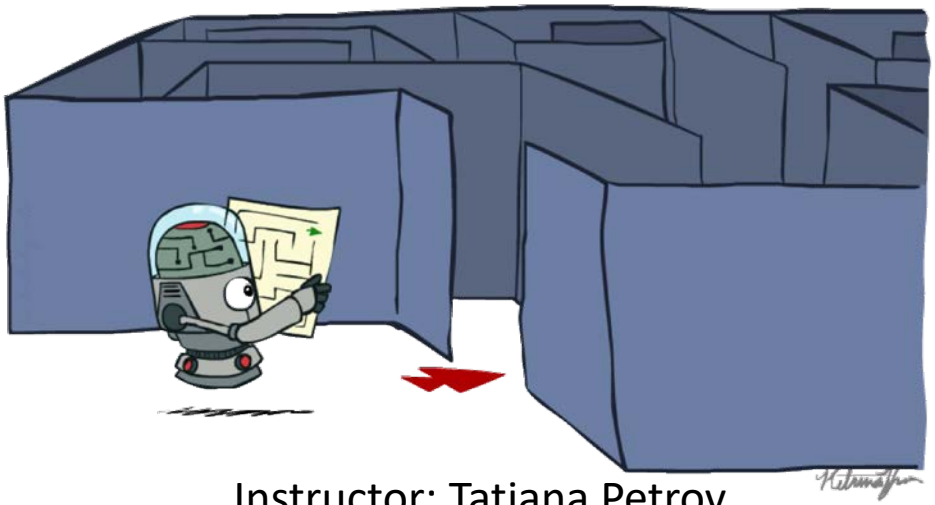
Introduction to Artificial Intelligence

Agents



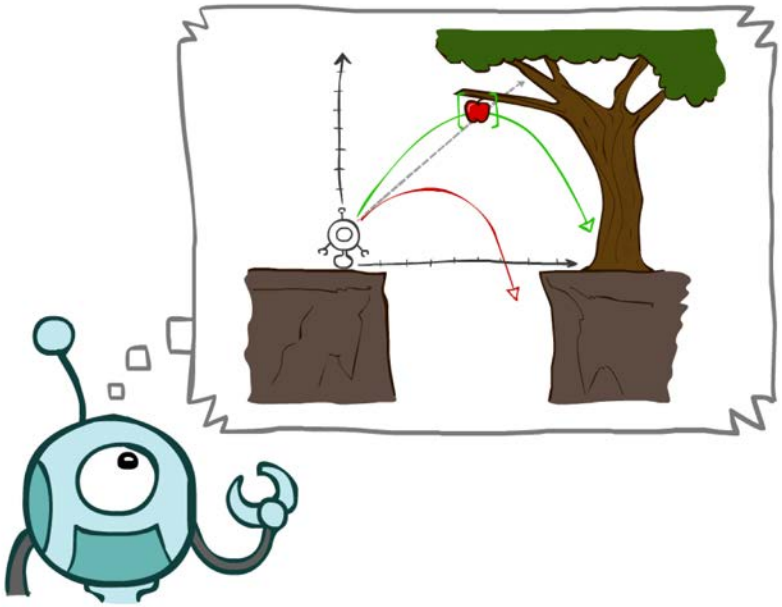
Instructor: Tatjana Petrov

University of Trieste, Italy

[slides adapted from Dan Klein, Pieter Abbeel, Stuart Russell, et al for CS188 Intro to AI at UC Berkeley. All materials available at http://ai.berkeley.edu. Thanks to Laura Nenzi for the first course edition in summer 2023]

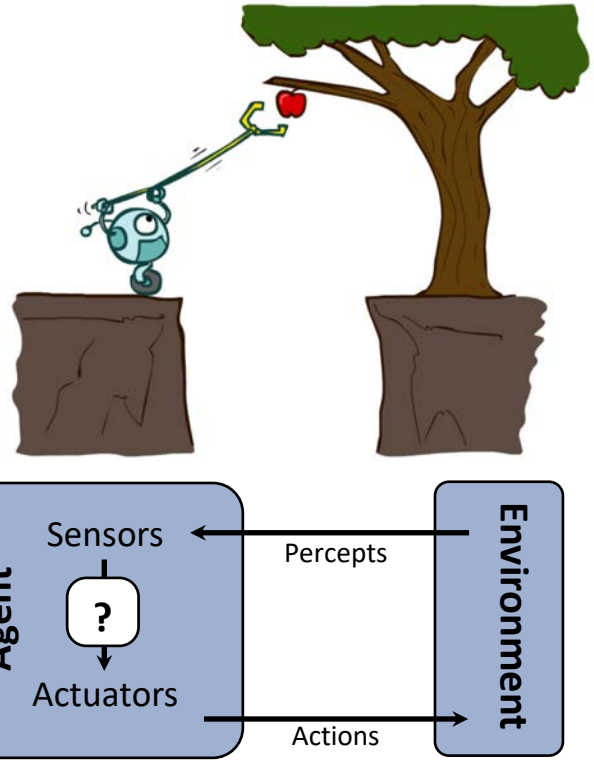
Today

- Agents that Plan Ahead

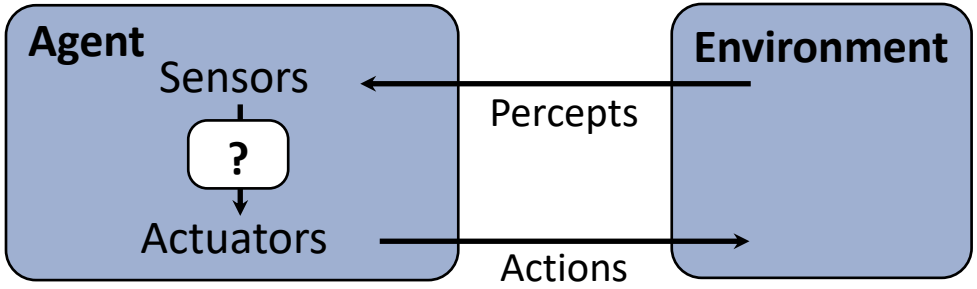


AI as Designing Rational Agents

- An **agent** is an entity that *perceives* and *acts*.
- A **rational agent** selects actions that maximize its (expected) **utility**.
- Characteristics of the **sensors, actuators, and environment** dictate techniques for selecting rational actions
- **This course** is about:
  - Classical AI techniques for many problem types
  - Learning to choose and apply the technique appropriate for each problem

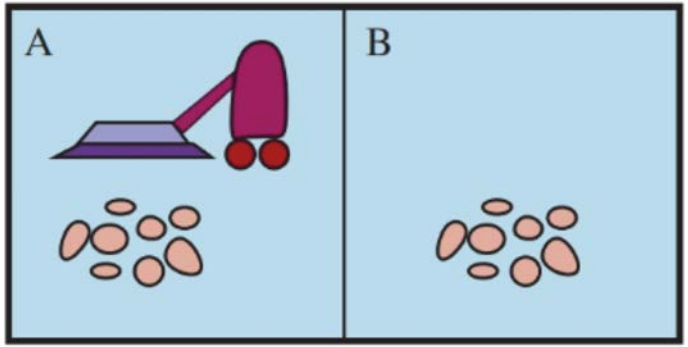


Agents and environments



- An agent ***perceives*** its environment through ***sensors*** and ***acts*** upon it through ***actuators*** (or *effectors*, depending on whom you ask)
- The ***agent function*** maps percept sequences to actions
- It is generated by an ***agent program*** running on a ***machine***

A vacuum-cleaner world



Agent function:
if the current square is dirty, then suck; otherwise, move to the other square.

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
⋮	⋮
[A, Clean], [A, Clean], [A, Clean]	Right
[A, Clean], [A, Clean], [A, Dirty]	Suck
⋮	⋮

A partial tabulation

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*

Performance measures

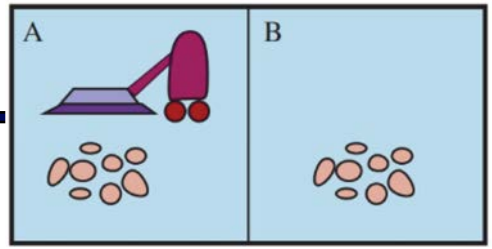
As a general rule, it is better to design a performance measure according to what one actually wants to be achieved in the environment, rather than according to how one thinks the agent should behave.

We assume that the performance measures can be specified correctly

Rational Agent

*For each possible percept sequence, a rational agent should select an **action** that is expected to maximize its **performance measures**, given the evidence provided by the **percept sequence** and whatever **built-in knowledge** the agent has.*

Example vacuum-cleaner agent



Is this a rational agent?

- Performance measure: 1 point, ∀ clean square, ∀ time step, for a “lifetime” of 1000 time steps.
- Known a priori: the “geography” of the environment but the dirt distribution and the initial location of the agent are not. Clean squares stay clean and sucking cleans the current square. The Right and Left actions move the agent one square except when this would take the agent outside the environment, in which case the agent remains where it is.
- Action: Right, Left, and Suck.
- Percepts: its location and whether that location contains dirt.

Rational Agent

*For each possible percept sequence, a rational agent should select an **action** that is expected to maximize its **performance measures**, given the evidence provided by the **percept sequence** and whatever **built-in knowledge** the agent has.*

Rational Agent

*For each possible percept sequence, a rational agent should select an **action** that is expected to maximize its (expected) **performance measures**, given the evidence provided by the **percept sequence** and whatever **built-in knowledge** the agent has.*

The sphex wasp

https://www.youtube.com/watch?v=YNvi_j2z96w&ab_channel=Pumpkinseedvalley

The sphex is unable to learn that its innate plan is failing, and thus will not change it.

Specifying the task environment

PEAS

Performance measure

Environment

Actuators

Sensors

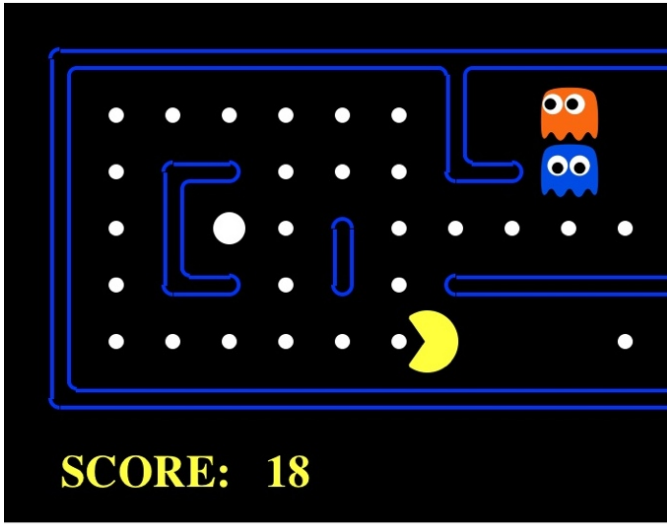
PEAS: Automated taxi

- Performance measure
  - Income, happy customer, vehicle costs, fines, insurance premiums
- Environment
  - streets, other drivers, customers, weather, police…
- Actuators
  - Steering, brake, gas, display/speaker
- Sensors
  - Camera, radar, accelerometer, engine sensors, microphone, GPS

Image: http://nypost.com/2014/06/21/how-google-might-put-taxi-drivers-out-of-business/

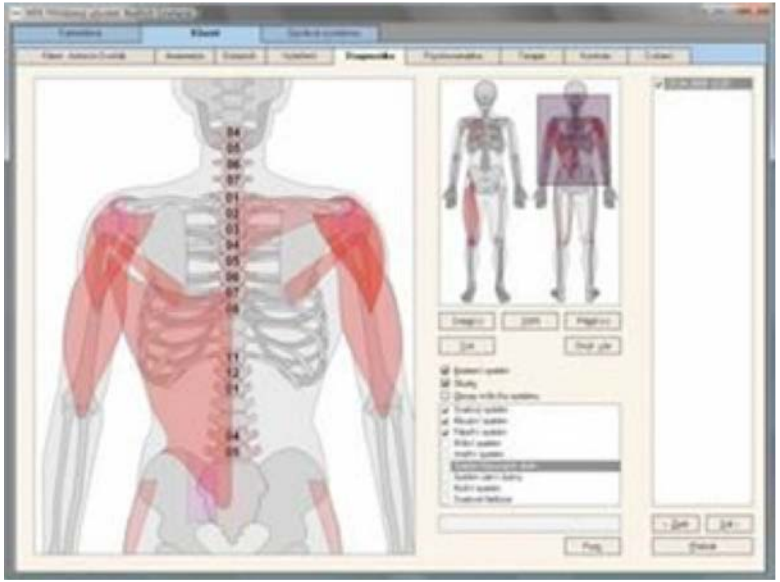
The task environment - PEAS

- Performance measure
  - -1 per step; + 10 food; +500 win; -500 die; +200 hit scared ghost
- Environment
  - Pacman dynamics (incl ghost behavior)
- Actuators
  - Left Right Up Down or NSEW
- Sensors
  - Entire state is visible (except power pellet duration)



PEAS: Medical diagnosis system

- Performance measure
  - Patient health, cost, reputation
- Environment
  - Patients, medical staff, insurers, courts
- Actuators
  - Screen display, email
- Sensors
  - Keyboard/mouse



Properties of task environments

	Pacman	Backgammon	Diagnosis	Taxi
Fully or partially observable				
Single-agent or multiagent				
Deterministic or non deterministic or stochastic				
Static or dynamic				
Discrete or continuous				
Known physics/rules?				
Known perf. measure?				

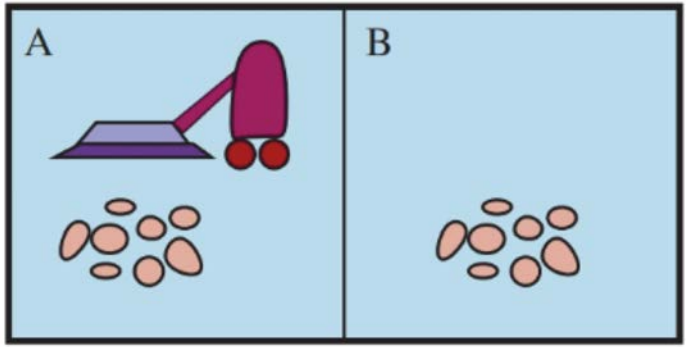
Properties of task environments

	Pacman	Backgammon	Diagnosis	Taxi
Fully or partially observable	F/P	F	P	P
Single-agent or multiagent	S	M	S	M
Deterministic or non deterministic or stochastic	D/N	S	S	S
Static or dynamic	S/D	S	D	D
Discrete or continuous	D/C	D	C	C
Known physics/rules?	Yes	Yes	Yes	Yes
Known perf. measure?				

Agent design

- **The environment type largely determines the agent design**
  - ***Partially observable*** => agent requires ***memory*** (internal state)
  - ***Stochastic*** => agent may have to prepare for ***contingencies***
  - ***Multi-agent*** => agent may need to behave ***randomly***
  - ***Static*** => agent has time to compute a rational decision
  - ***Continuous time*** => continuously operating ***controller***
  - ***Unknown physics*** => need for ***exploration***
  - ***Unknown perf. measure*** => observe/interact with ***human principal***

A vacuum-cleaner world



Agent function:
if the current square is dirty, then suck; otherwise, move to the other square.

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
⋮	⋮
[A, Clean], [A, Clean], [A, Clean]	Right
[A, Clean], [A, Clean], [A, Dirty]	Suck

$$\sum_{t=1}^{T} |\mathcal{P}|^t$$

function TABLE-DRIVEN-AGENT(*percept*) **returns** an action
  persistent: *percepts*, a sequence, initially empty
    table, a table of actions, indexed by percept sequences, initially fully specified

  append *percept* to the end of *percepts*
  action ← LOOKUP(*percepts*, *table*)
  return *action*

Figure 2.7 The TABLE-DRIVEN-AGENT program is invoked for each new percept and returns an action each time. It retains the complete percept sequence in memory.

The structure of the Agents

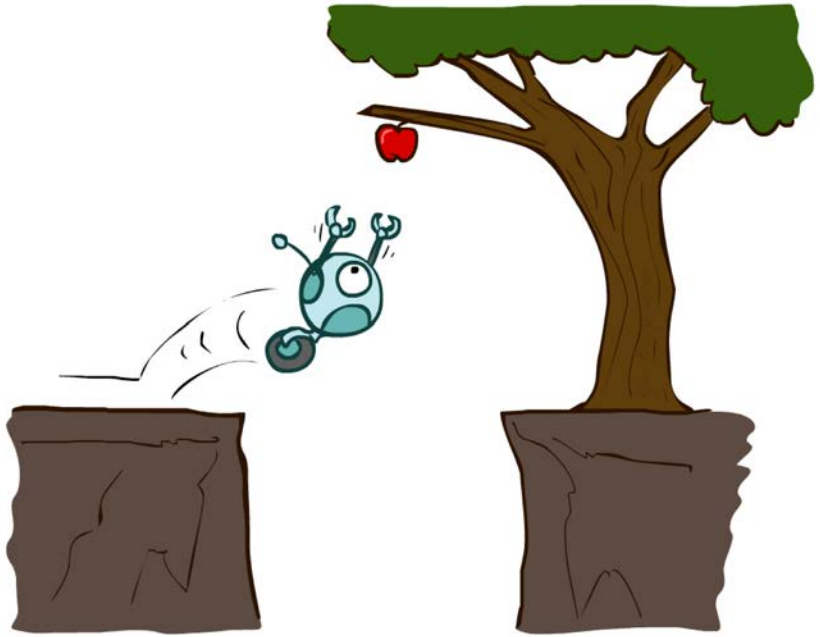
- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents.

Agent types

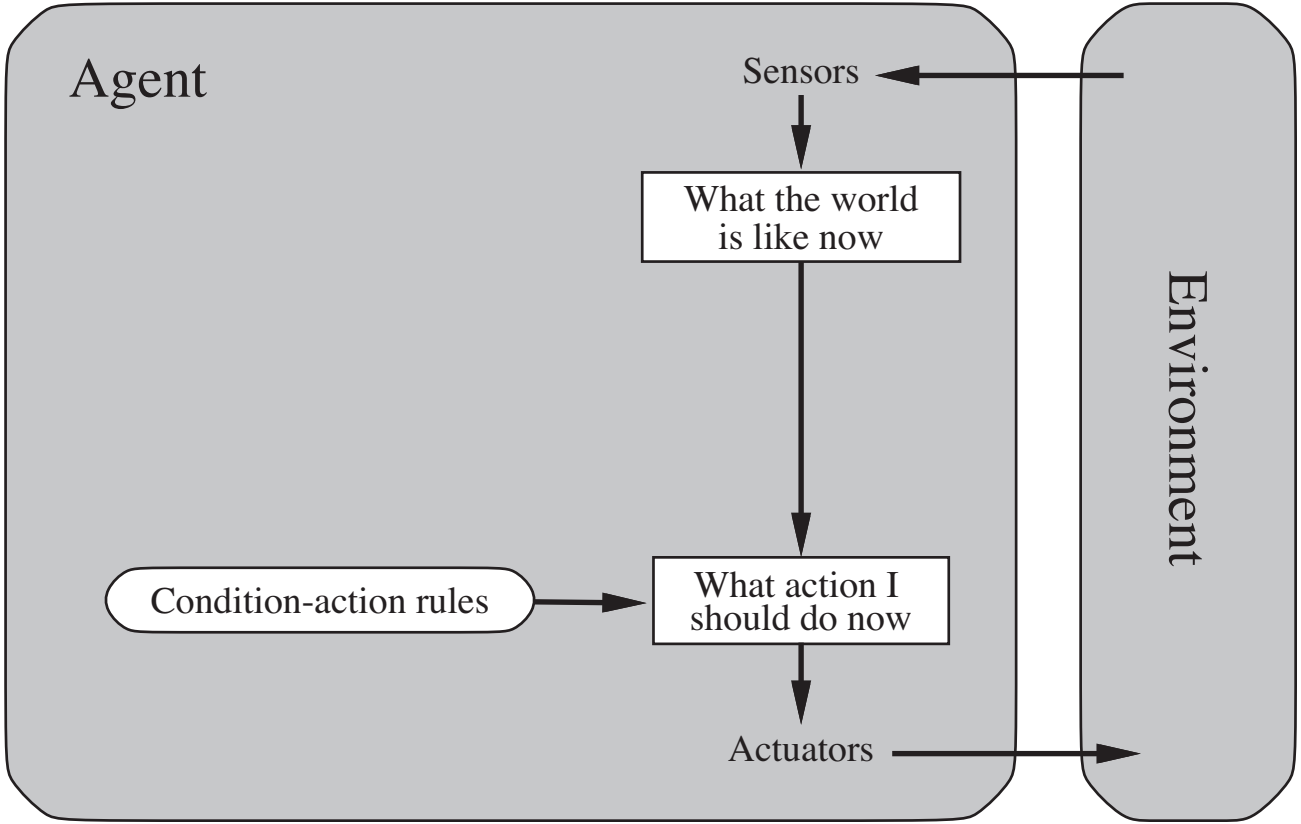
- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents.

Reflex Agents

- Reflex agents:
  - Choose action based on current percept (and maybe memory)
  - May have memory or a model of the world's current state
  - Do not consider the future consequences of their actions
  - Consider how the world IS
- Can a reflex agent be rational?



Simple reflex agents



Simple reflex agents

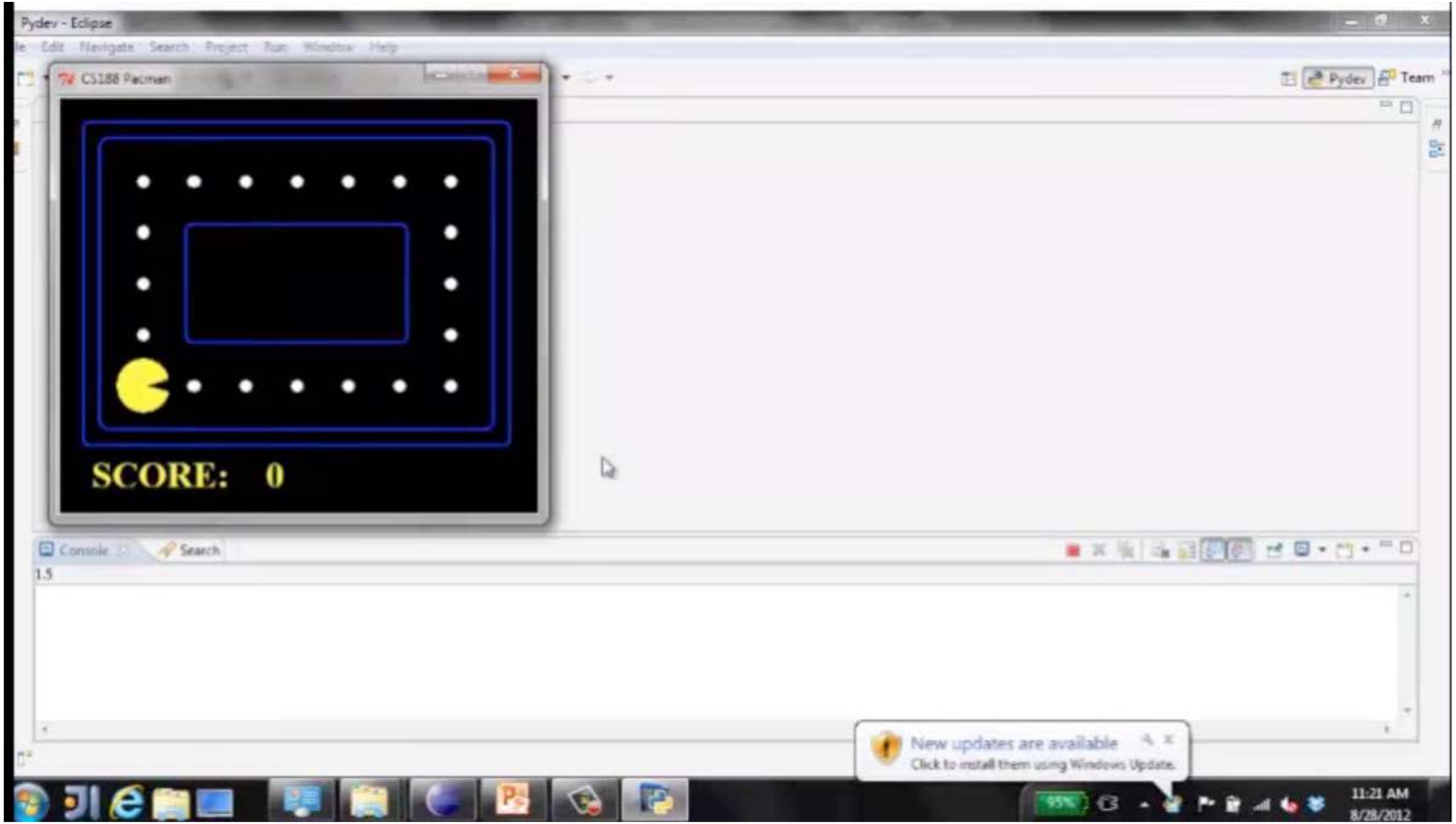
function SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
  persistent: *rules*, a set of condition–action rules

  state ← INTERPRET-INPUT(*percept*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← *rule*.ACTION
  return *action*

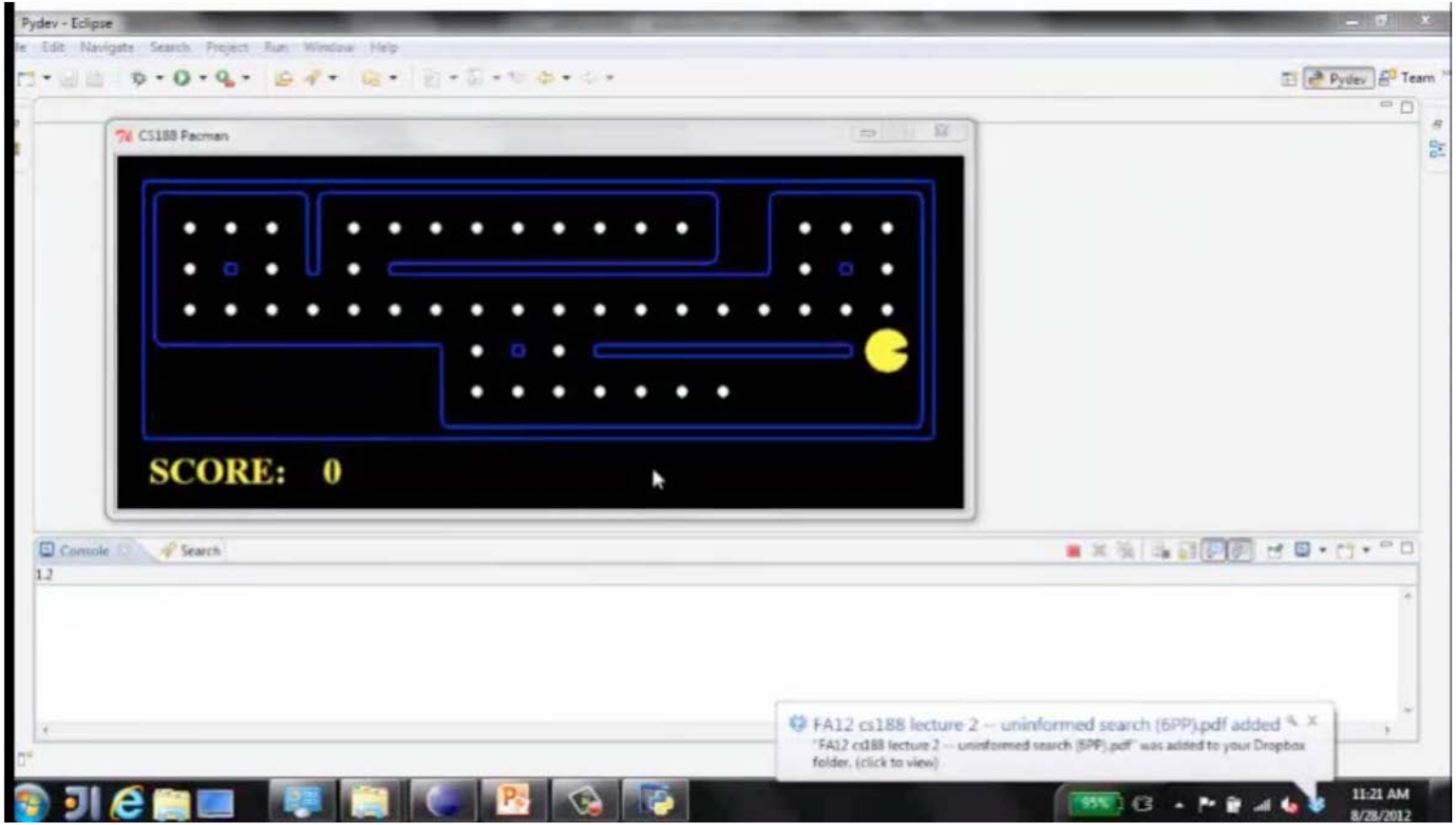
Based on condition-action rule:

if *happen X* ***then*** *action A*

Eat adjacent dot, if any



Eat adjacent dot, if any



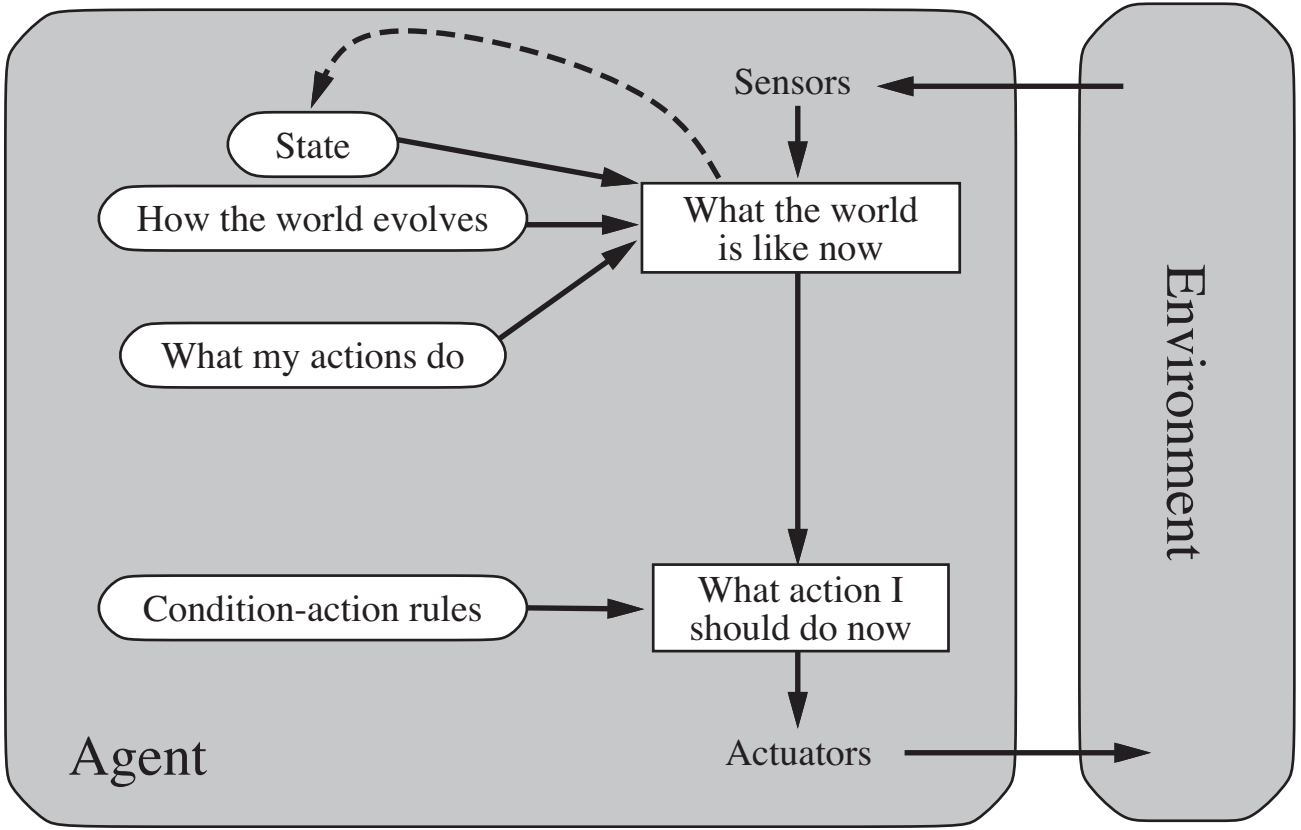
Pacman agent contd.

- Can we (in principle) extend this reflex agent to behave well in all standard Pacman environments?
  - No – Pacman is not quite fully observable (power pellet duration)
  - Otherwise, yes – we can (*in principle*) make a lookup table…..
  - *How large would it be?*

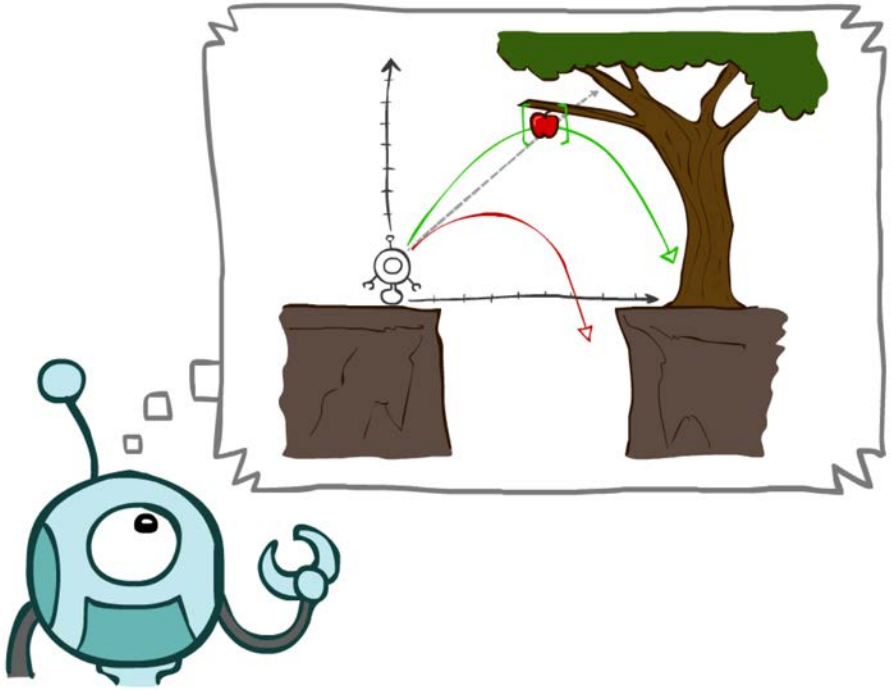
Agent types

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents.

Model-based Reflex agents

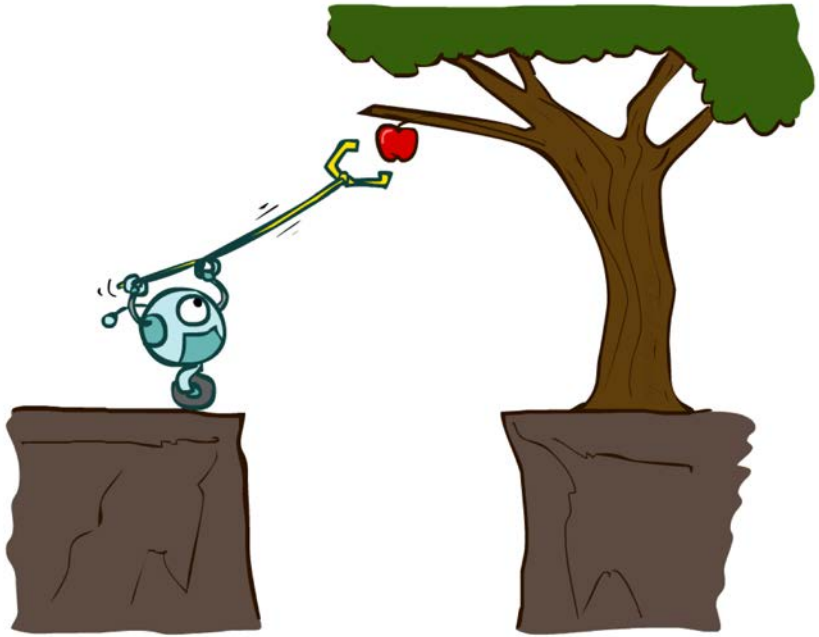


Agents that Plan

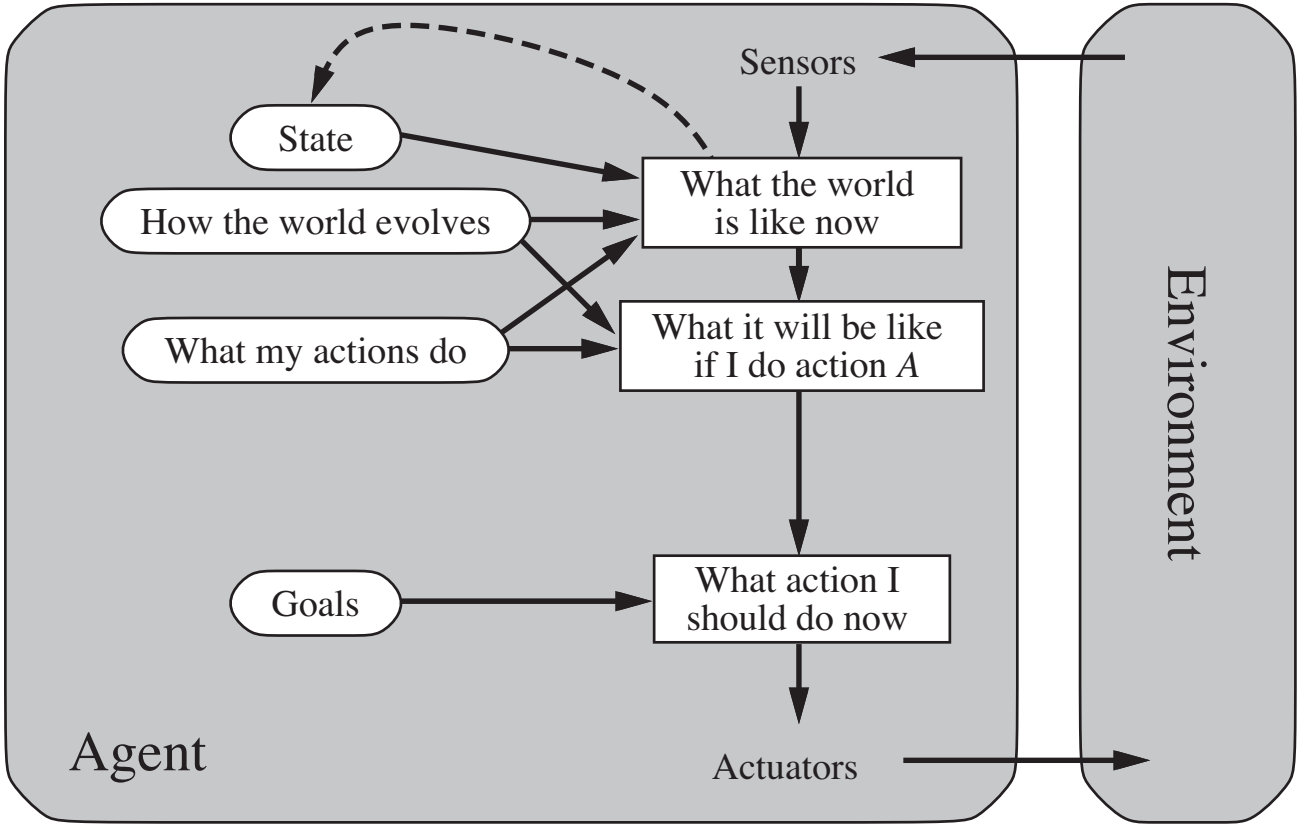


(model-based), goal-based agents

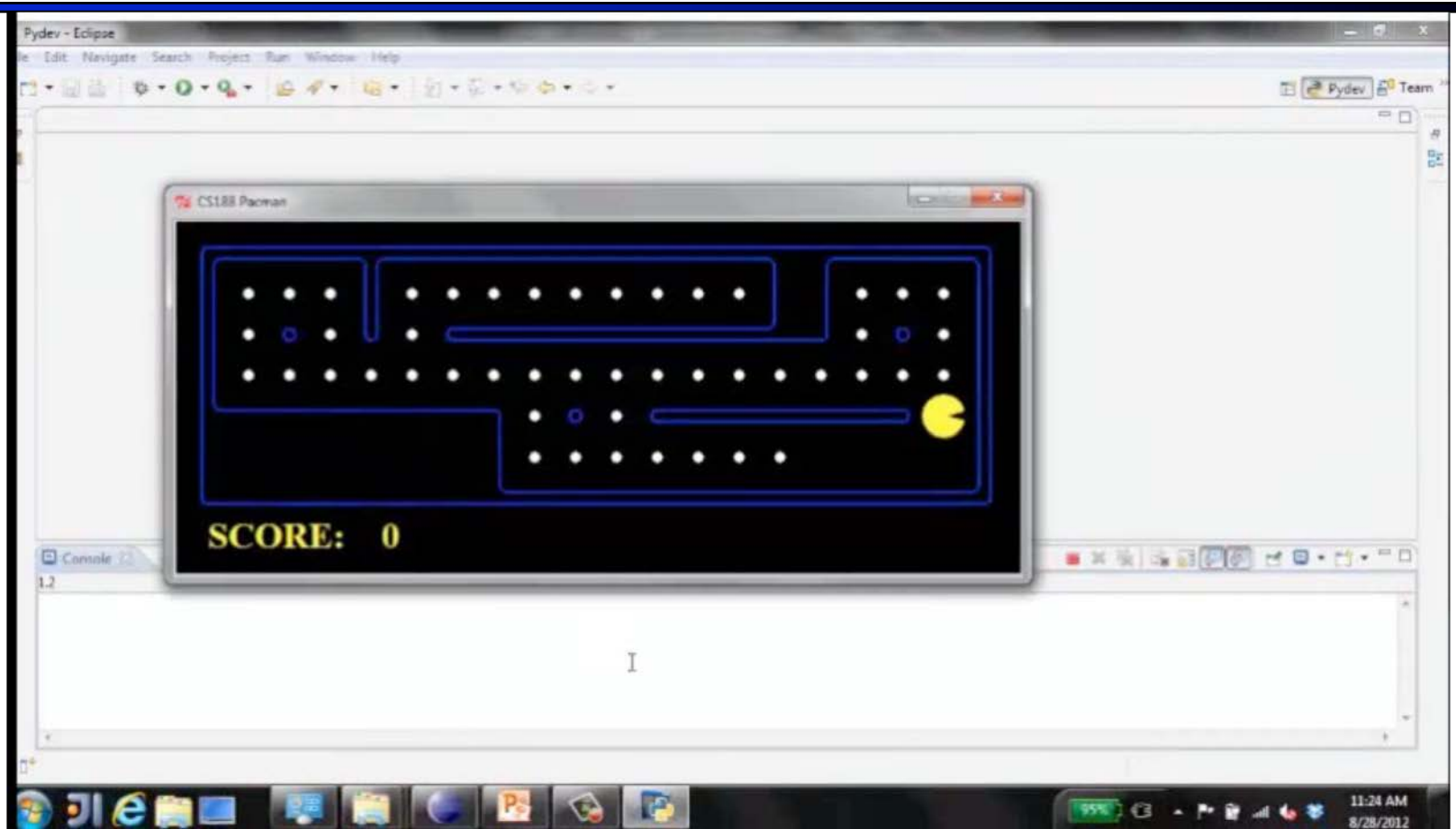
- goal-based agents:
  - Ask “what if”
  - Decisions based on (hypothesized) consequences of actions
  - Must have a model of how the world evolves in response to actions
  - Must formulate a goal (test)
  - Consider how the world WOULD BE



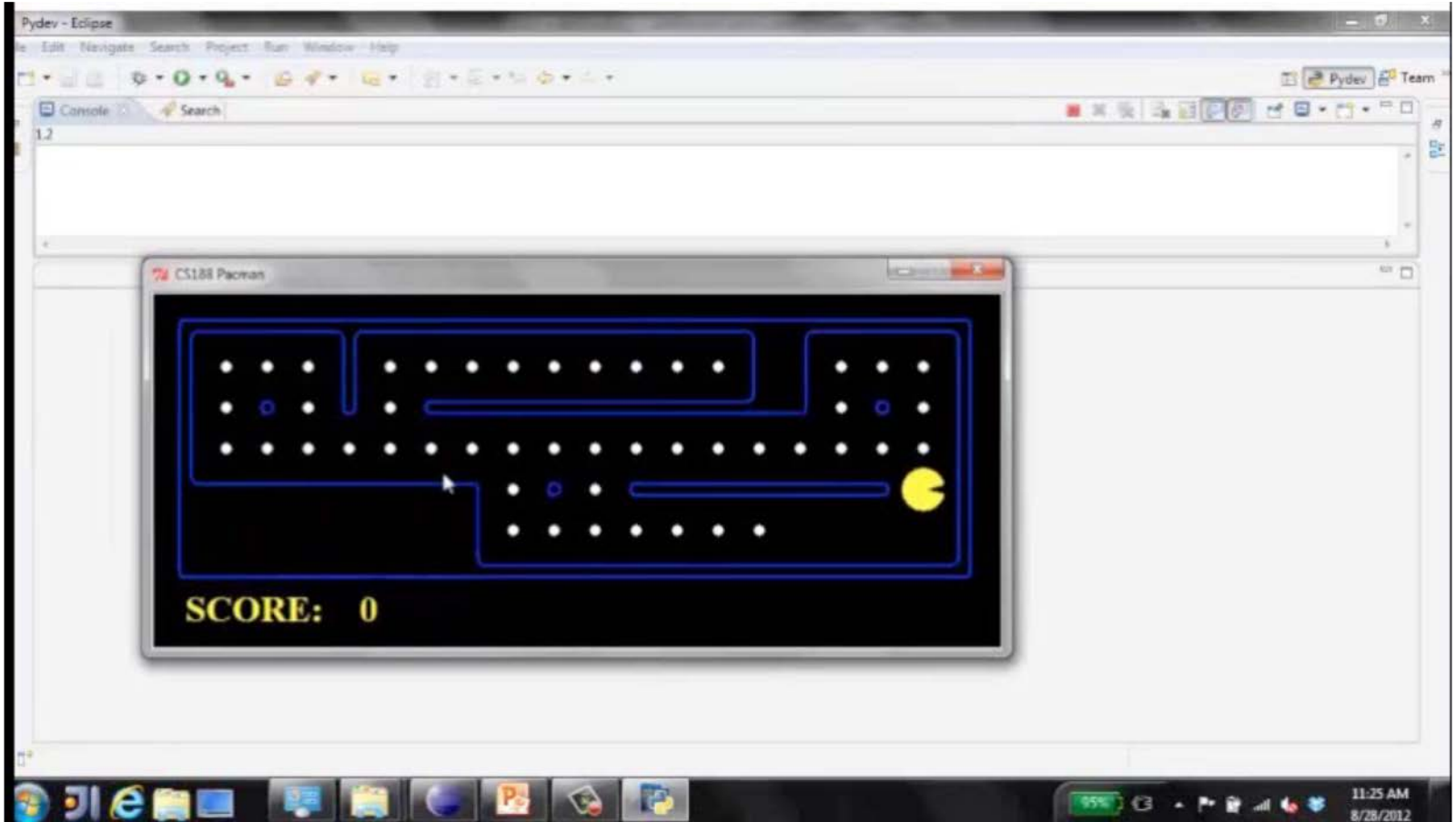
(model-based), goal-based agents



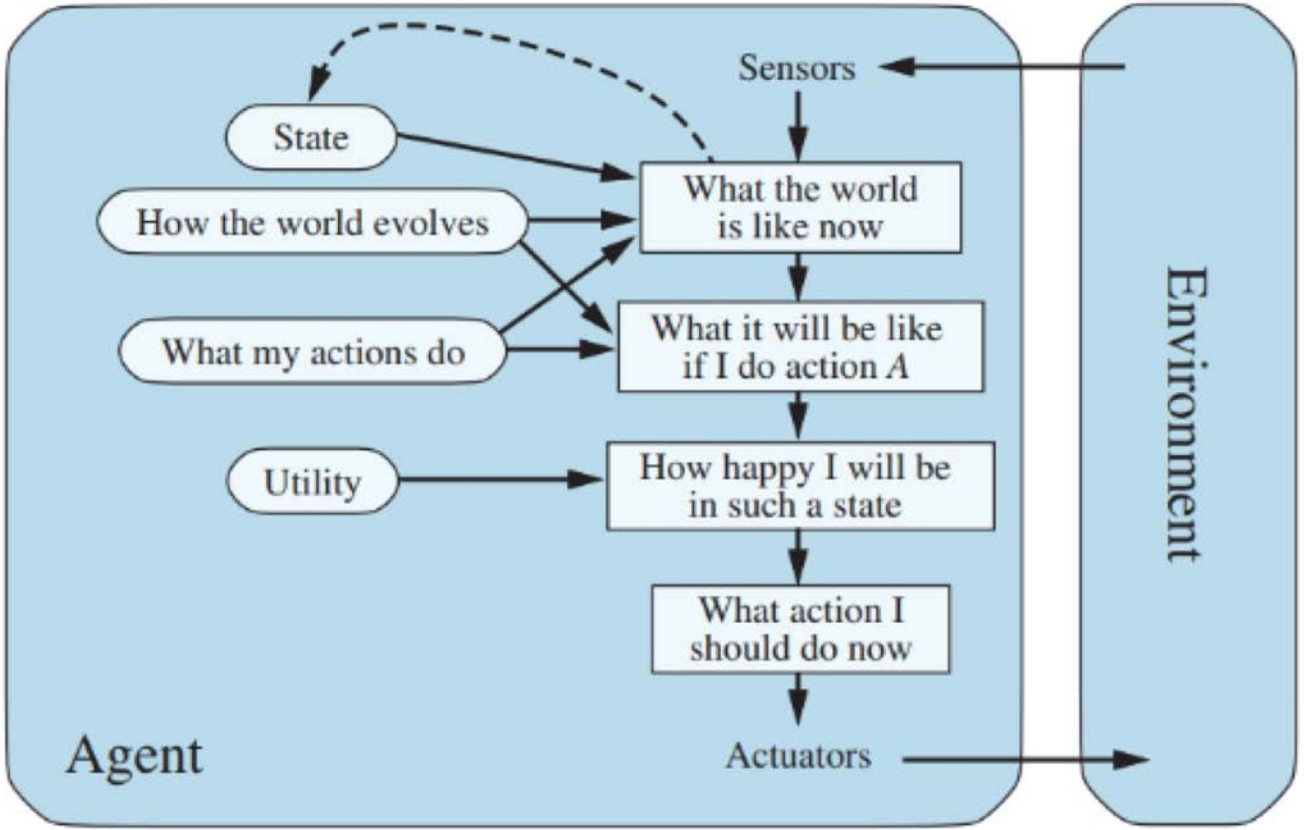
Video of Demo Replanning



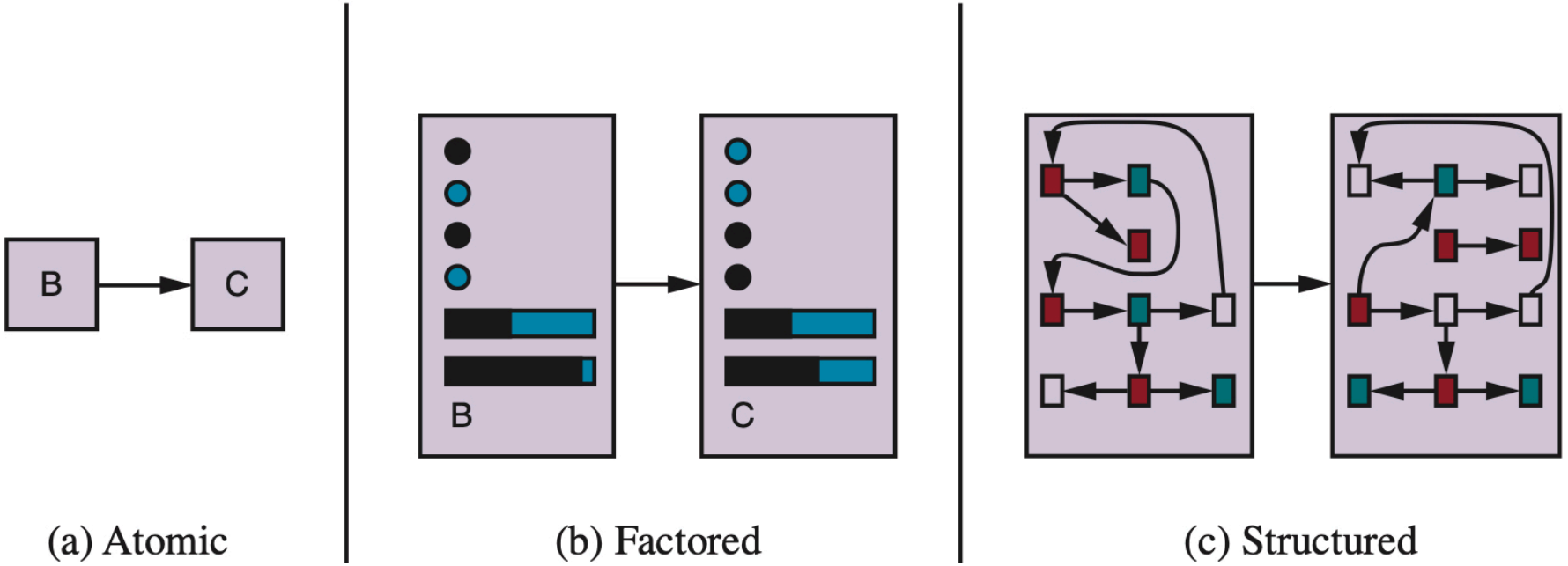
Video of Demo Mastermind



(model-based, goal-based), utility-based agents



Spectrum of representations



Outline of the course