Artificial Intelligence – Chapter 2

August 24, 2020

Outline

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

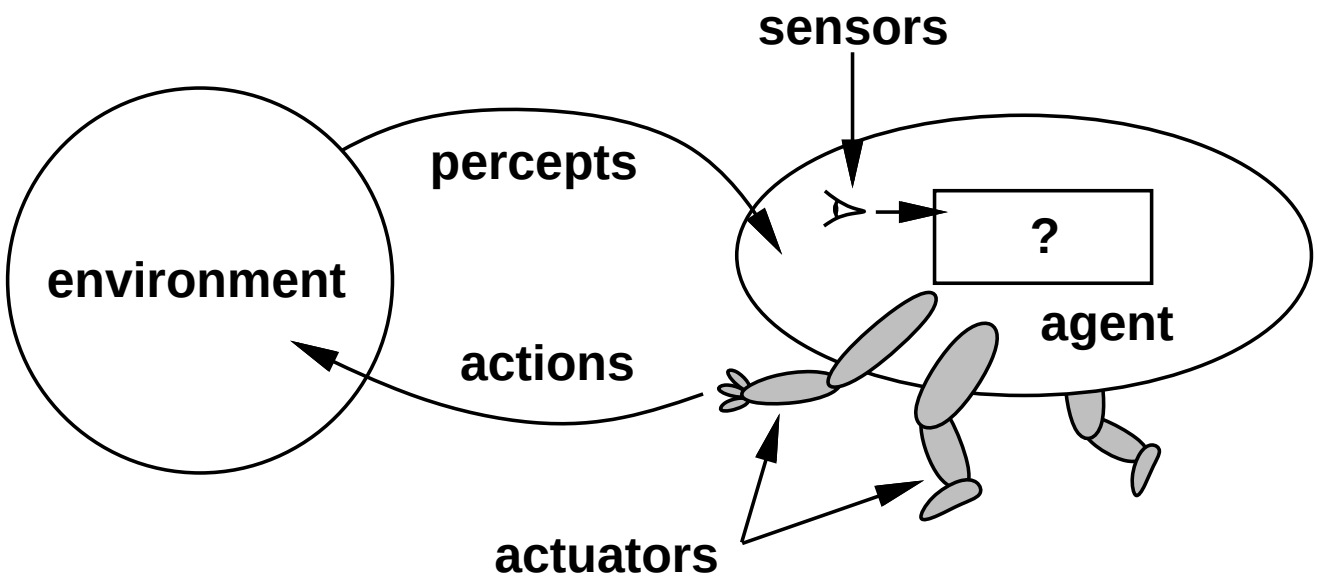
Agents and environments

Agents: humans, robots, softbots, thermostats, etc.

The agent function maps from percept histories to actions:

$$f : \mathcal{P}^* \rightarrow \mathcal{A}$$

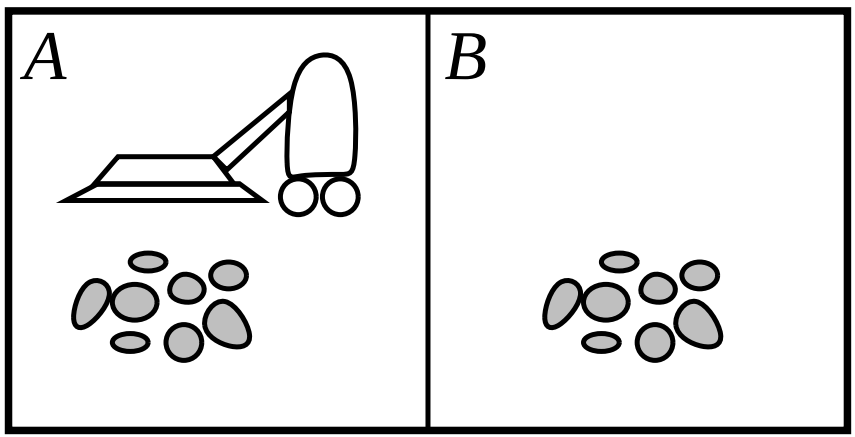
The agent program runs on the physical architecture to produce f



Vacuum-cleaner world

Percepts: location and contents, 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*
⋮	⋮

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

function Reflex-Vacuum-Agent([*location*, *status*]) returns an action

1. **if** status = Dirty **then return** *Suck*
2. **else if** location = A **then return** *Right*
3. **else if** location = B **then return** *Left*
4. **else ...**

Rationality

Fixed performance measure evaluates the environment sequence

- one point per square cleaned up in time T?
- one point per clean square per time step, minus one per move?
- penalize for $> k$ dirty squares?

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

Rational $\neq$ omniscient

- percepts may not supply all relevant information

Rational $\neq$ clairvoyant

- action outcomes may not be as expected $\rightarrow$ rational $\neq$ successful

Rational $\Longrightarrow$ exploration, learning, autonomy

PEAS

To design a rational agent, we must specify the task environment
Consider, e.g., the task of designing an automated taxi:
Performance measure??
Environment??
Actuators??
Sensors??

PEAS - automated taxi

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

Consider, e.g., the task of designing an automated taxi:

Performance measure?? safety, destination, profits, legality, comfort, ...

Environment?? US streets/freeways, traffic, pedestrians, weather, ...

Actuators?? steering, accelerator, brake, horn, speaker/display, ...

Sensors?? video, accelerometers, gauges, engine sensors, keyboard, GPS, ...

Internet shopping agent

- *Performance measure:* price, quality, appropriateness, efficiency
- *Environment:* current and future WWW sites, vendors, shippers
- *Actuators:* display to user, follow URL, fill in form
- *Sensors:* HTML pages (text, graphics, scripts)

Environment types – definitions

Deterministic	**Nondeterministic**
The current state and the agent's actions determine the next state; if the environment is *partially observable*	Otherwise
Episodic	**Sequential**
Agent's experience is divided into episodes; in each episode the agent perceives and acts; the next episode does not depend on previous episodes. Example: classification tasks.	The current decision affects ALL future decisions; Example: chess, taxi driving
Static	**Dynamic**
The environment does NOT change while the agent is deliberating	Otherwise
Discrete	**Continuous**
The distinction between discrete and continuous applies to a state in the way *time* is handled and to the *percepts* and *actions* of the agent.	

Agent – example

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe, fast, legal, comfortable trip, maximize profits, minimize impact on other road users	Roads, other traffic, police, pedestrians, customers, weather	Steering, accelerator, brake, signal, horn, display, speech	Cameras, radar, speedometer, GPS, engine sensors, accelerometer, microphones, touchscreen

Agents & PEAS – examples

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, reduced costs	Patient, hospital, staff	Display of questions, tests, diagnoses, treatments	Touchscreen/voice entry of symptoms and findings
Satellite image analysis system	Correct categorization of objects, terrain	Orbiting satellite, downlink, weather	Display of scene categorization	High-resolution digital camera
Part-picking robot	Percentage of parts in correct bins	Conveyor belt with parts; bins	Jointed arm and hand	Camera, tactile and joint angle sensors
Refinery controller	Purity, yield, safety	Refinery, raw materials, operators	Valves, pumps, heaters, stirrers, displays	Temperature, pressure, flow, chemical sensors
Interactive English tutor	Student's score on test	Set of students, testing agency	Display of exercises, feedback, speech	Keyboard entry, voice

Tasks & Environment types

Task Environment	Observable	Agents	Deterministic	Episodic	Static	Discrete
Crossword puzzle	Fully	Single	Deterministic	Sequential	Static	Discrete
Chess with a clock	Fully	Multi	Deterministic	Sequential	Semi	Discrete
Poker	Partially	Multi	Stochastic	Sequential	Static	Discrete
Backgammon	Fully	Multi	Stochastic	Sequential	Static	Discrete
Taxi driving	Partially	Multi	Stochastic	Sequential	Dynamic	Continuous
Medical diagnosis	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
Image analysis	Fully	Single	Deterministic	Episodic	Semi	Continuous
Part-picking robot	Partially	Single	Stochastic	Episodic	Dynamic	Continuous
Refinery controller	Partially	Single	Stochastic	Sequential	Dynamic	Continuous
English tutor	Partially	Multi	Stochastic	Sequential	Dynamic	Discrete

More on environment types

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??	Yes	Yes	No	No
Deterministic??	Yes	No	Partly	No
Episodic??	No	No	No	No
Static??	Yes	Semi	Semi	No
Discrete??	Yes	Yes	Yes	No
Single-agent??	Yes	No	Yes (except auctions)	No

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 in order of increasing generality:

– simple reflex agents

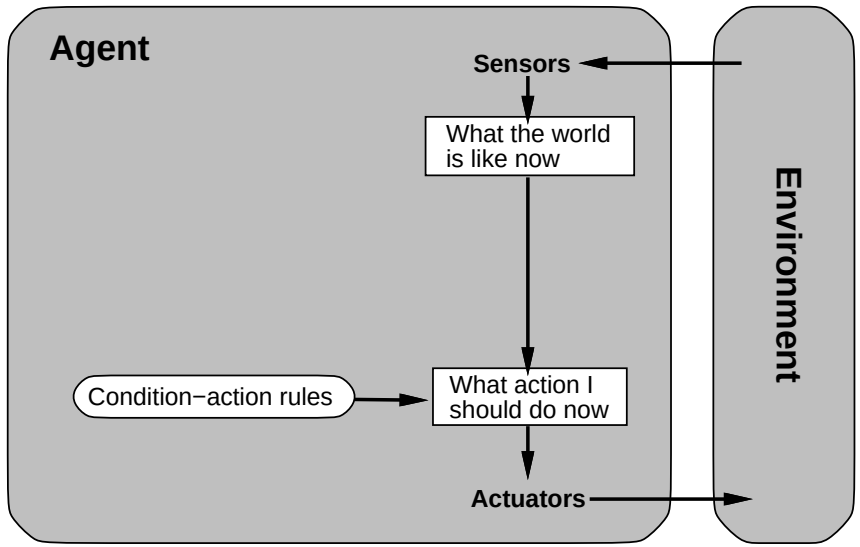
– reflex agents with state

– goal-based agents

– utility-based agents

All these can be turned into learning agents

Simple reflex agents



Example: reflex-vacuum-agent-algorithm

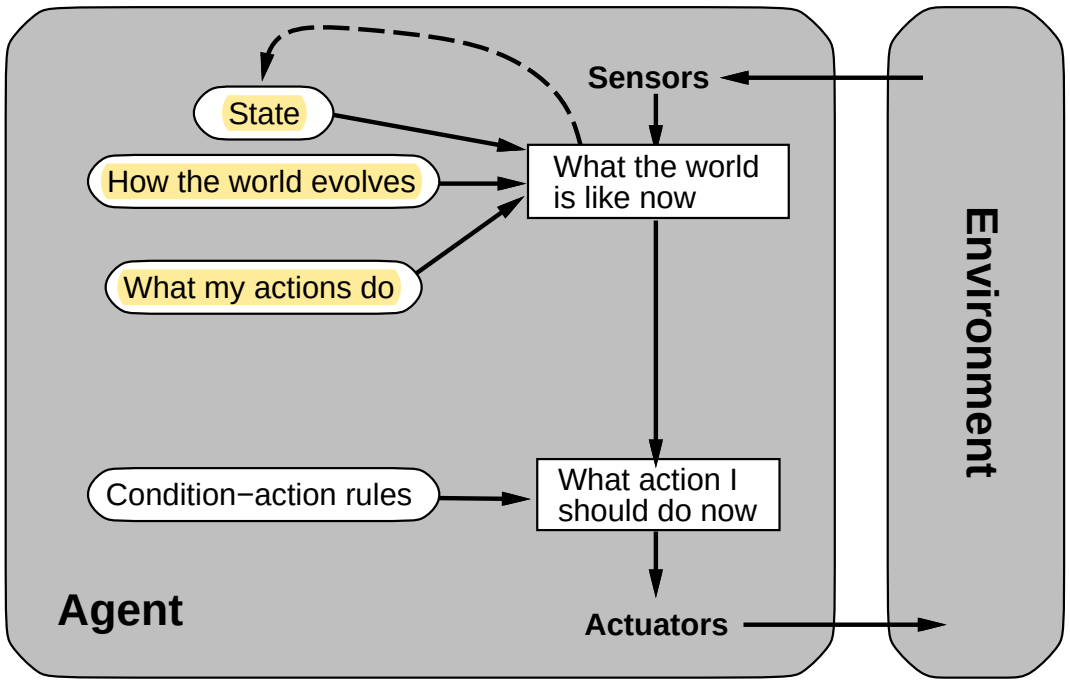
function REFLEX-VACUUM-AGENT([location, status]) returns an action
if status = Dirty **then** return Suck
else if location = A **then** return Right
else if location = B **then** return Left

/ ((reflexVacuumAgent ((LOcation STatus)) ACTION)) holds when the Agent takes action ACTION under percept LOcation STatus */*
((reflexVacuumAgent ((_ Dirty)) Suck)(!))
((reflexVacuumAgent ((a _)) RightTurn)(!))
((reflexVacuumAgent ((b _)) LeftTurn))

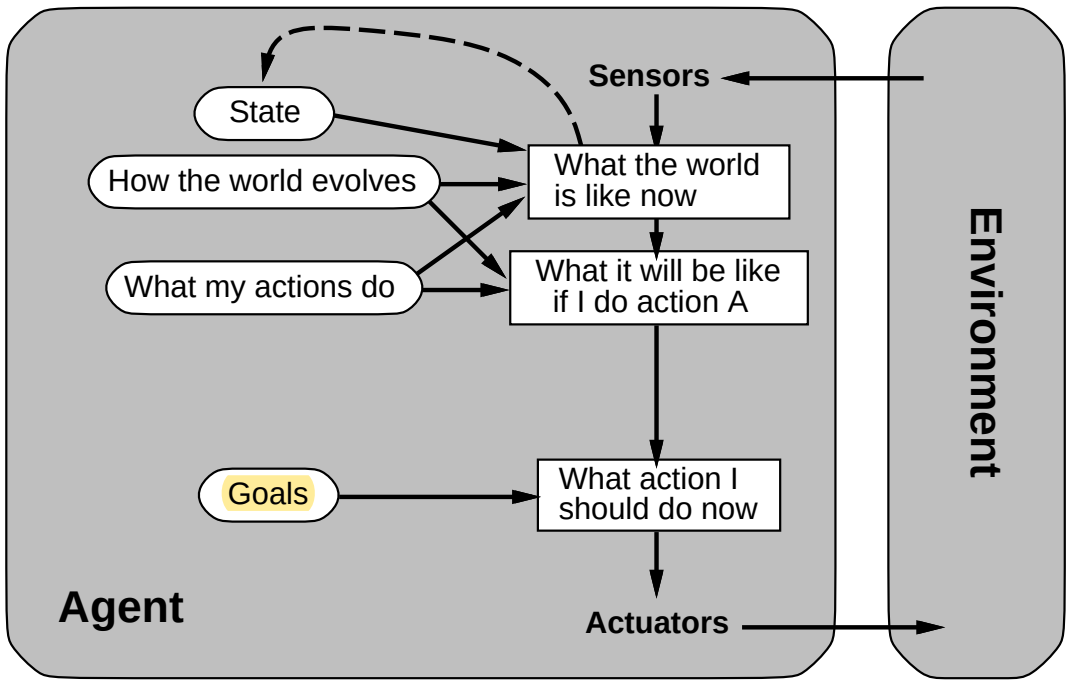
Notes:

1. "!" (the cut in prolog) encodes "else"
2. a, b, instead of A, B respectively because upper case denotes variables
3. _ denotes anonymous variables

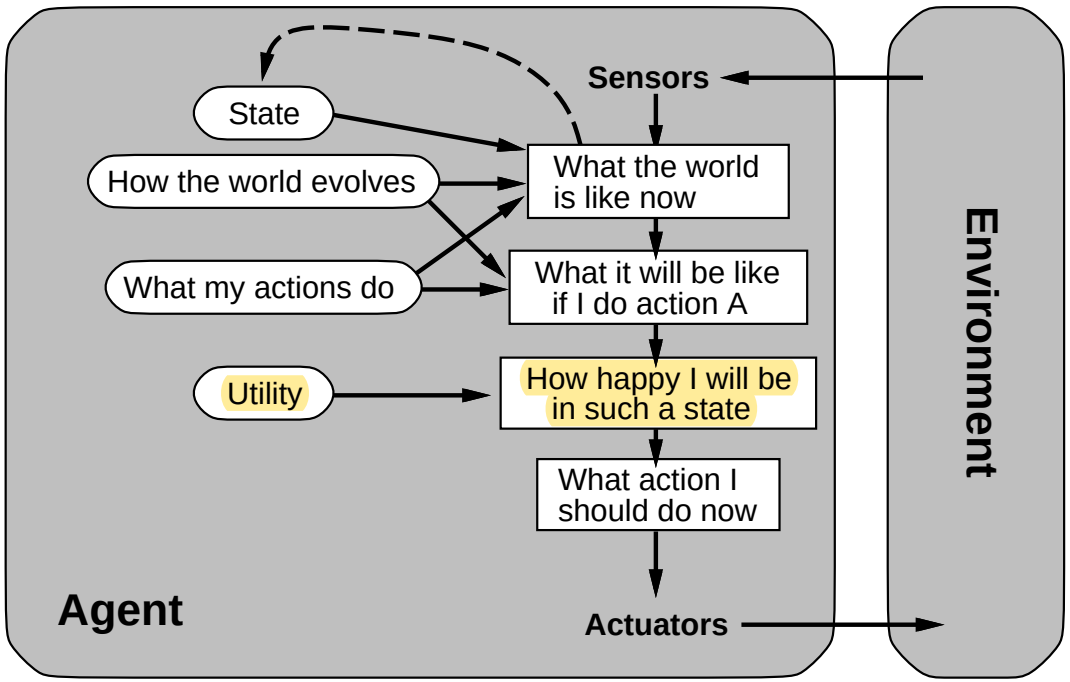
Reflex agents with state



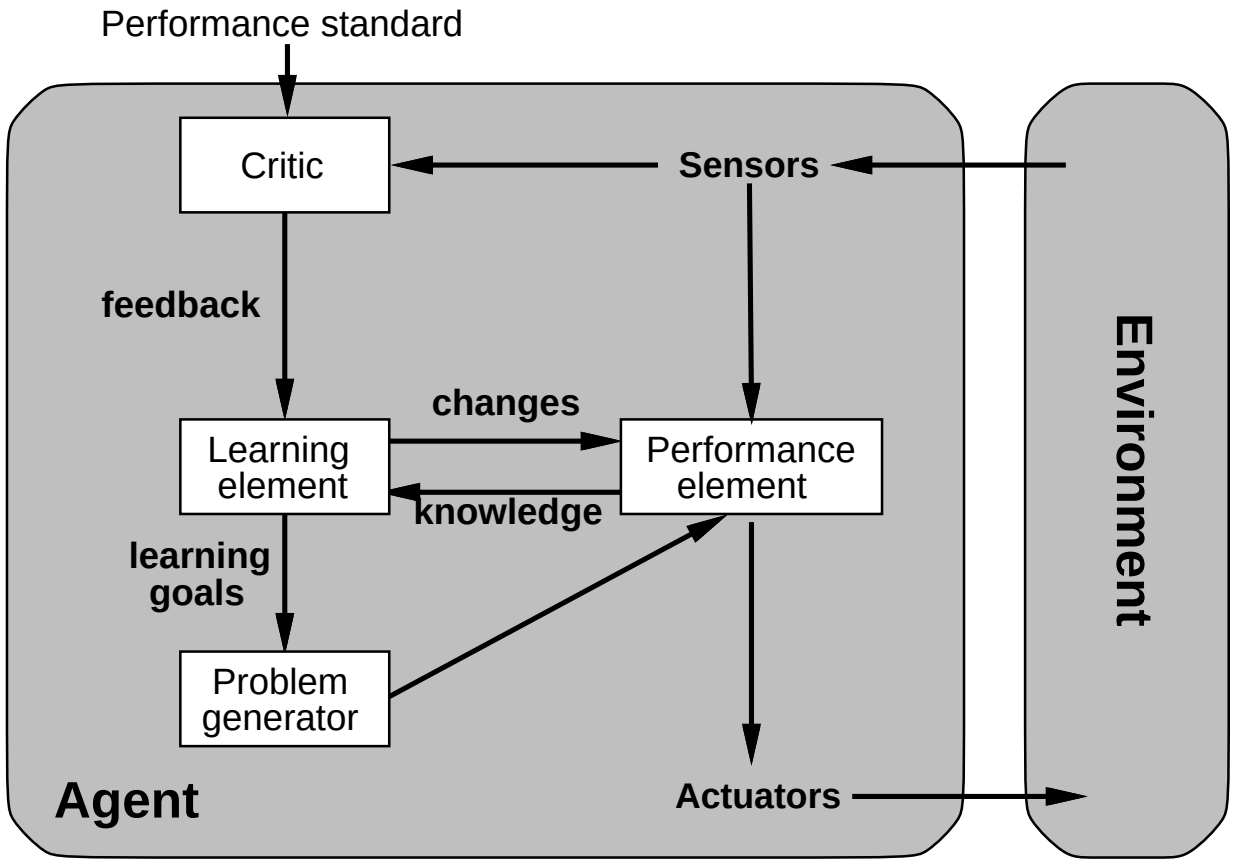
Goal-based agents



Utility-based agents



Learning agents



Summary

Agents interact with environments through actuators and sensors. The agent function describes what the agent does in all circumstances

The performance measure evaluates the environment sequence. A perfectly rational agent maximizes expected performance . Agent programs implement (some) agent functions. PEAS descriptions define task environments. Environments are categorized by:

- observable?
- deterministic?
- episodic?
- static?
- discrete?
- single-agent?

Several basic agent architectures exist:

- reflex
- reflex with state
- goal-based
- utility-based