Foundations of Artificial Intelligence

3. Introduction: Rational Agents

Thomas Keller and Florian Pommerening

University of Basel

February 22, 2023

Foundations of Artificial Intelligence

February 22, 2023 — 3. Introduction: Rational Agents

3.1 Systematic AI Framework

3.2 Example

3.3 Rationality

3.4 Summary

Introduction: Overview

Chapter overview: introduction

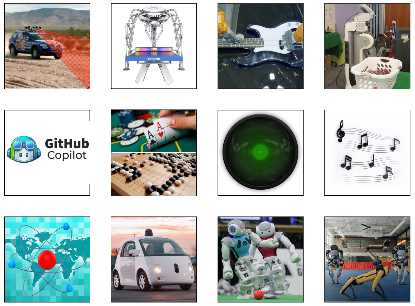
- 1. What is Artificial Intelligence?
- 2. AI Past and Present
- 3. Rational Agents
- 4. Environments and Problem Solving Methods

3.1 Systematic AI Framework

Systematic AI Framework

so far we have seen that:

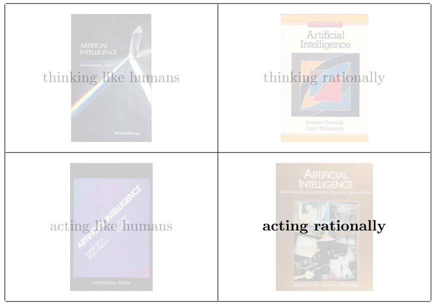
- AI systems applied to wide variety of challenges



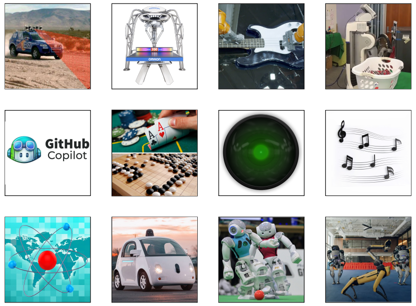
Systematic AI Framework

so far we have seen that:

- AI systems act rationally



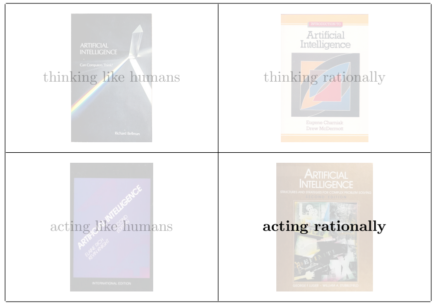
- AI systems applied to wide variety of challenges



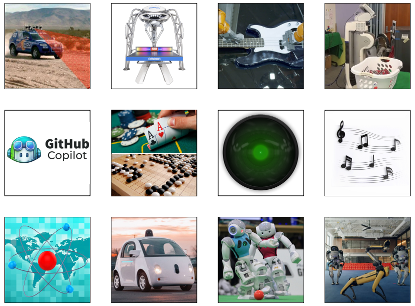
Systematic AI Framework

so far we have seen that:

- AI systems act rationally



- AI systems applied to wide variety of challenges



now: describe a systematic framework that

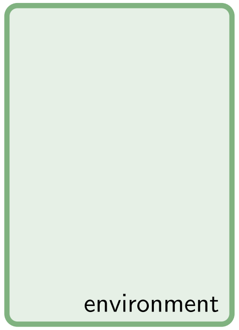
- captures this diversity of challenges
- includes an entity that is acting in the environment
- determines if the agent acts rationally in the environment

Systematic AI Framework

so far we have seen that:

- AI systems act rationally

- AI systems applied to wide variety of challenges



now: describe a systematic framework that

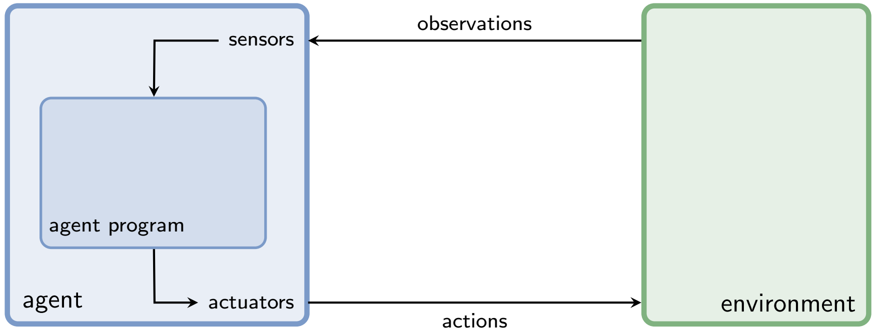
- captures this diversity of challenges
- includes an entity that is acting in the environment
- determines if the agent acts rationally in the environment

Systematic AI Framework

so far we have seen that:

- AI systems act rationally

- AI systems applied to wide variety of challenges



now: describe a systematic framework that

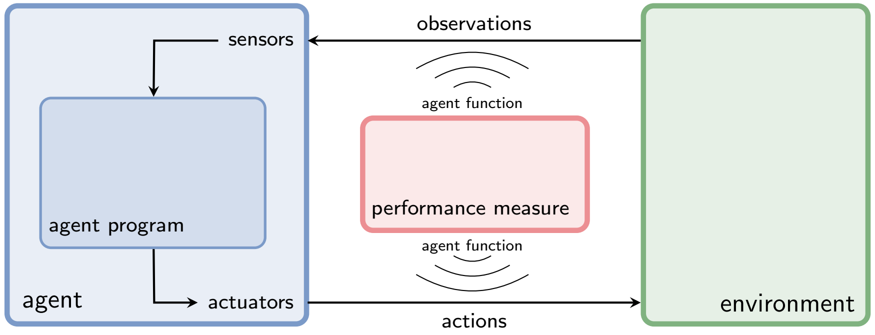
- captures this diversity of challenges
- includes an entity that is acting in the environment
- determines if the agent acts rationally in the environment

Systematic AI Framework

so far we have seen that:

- AI systems act rationally

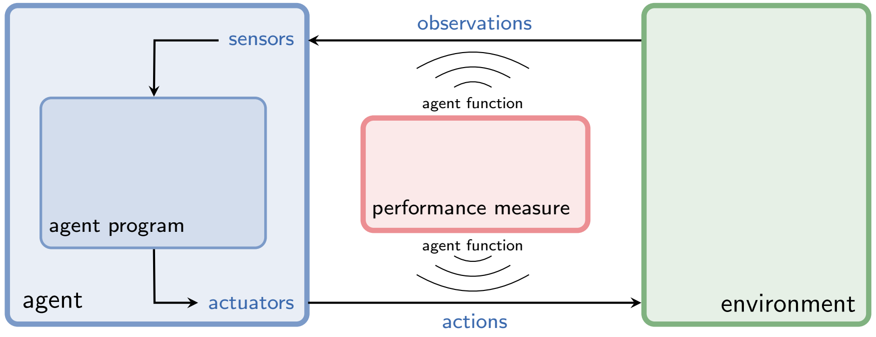
- AI systems applied to wide variety of challenges



now: describe a systematic framework that

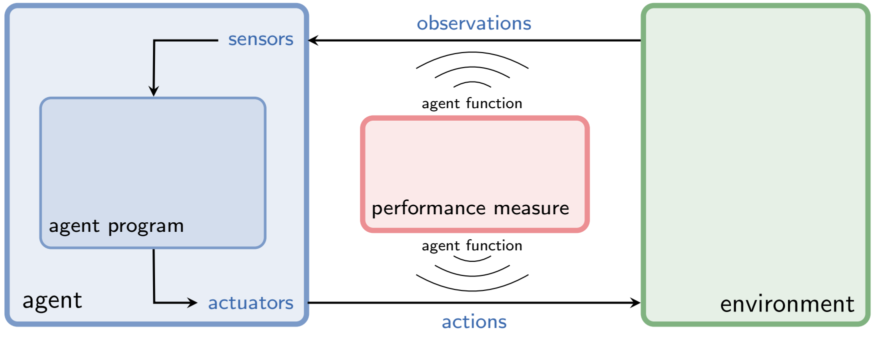
- captures this diversity of challenges
- includes an entity that is acting in the environment
- determines if the agent acts rationally in the environment

Agent-Environment Interaction



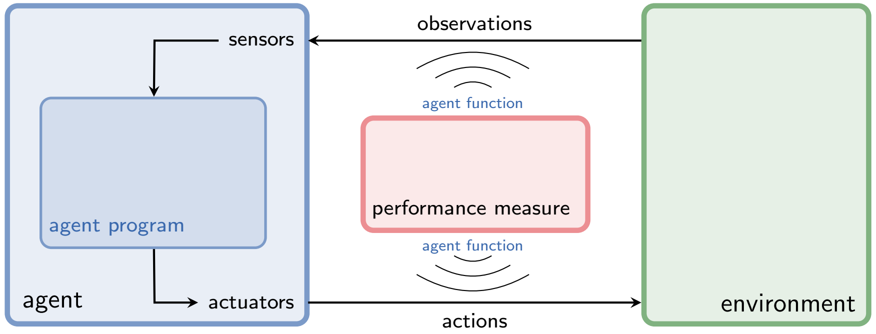
- sensors: phyisical entities that allow the agent to observe
- observation: data perceived by the agent's sensors
- actuators: phyisical entities that allow the agent to act
- action: abstract concept that affects the state of the environment

Agent-Environment Interaction



- sensors and actuators are not relevant for the course (⇝ typically covered in courses on robotics)
- observations and actions describe the agent's capabilities (the agent model)

Formalizing an Agent's Behavior



❶ as agent program:

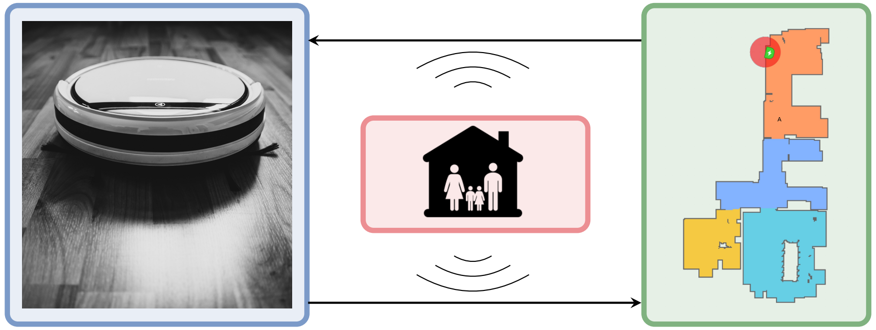
- internal representation
- specifics possibly unknown to outside
- takes observation as input
- outputs an action
- computed on physical machine (the agent architecture)

❷ as agent function:

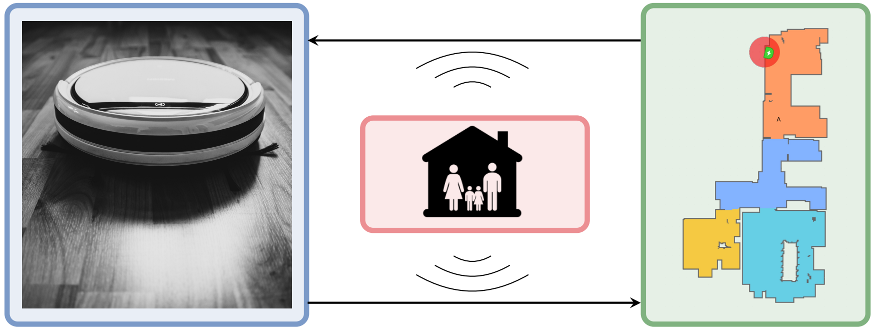
- external characterization
- maps sequence of observations to (probability distribution over) actions
- abstract mathematical formalization

3.2 Example

Vacuum Domain

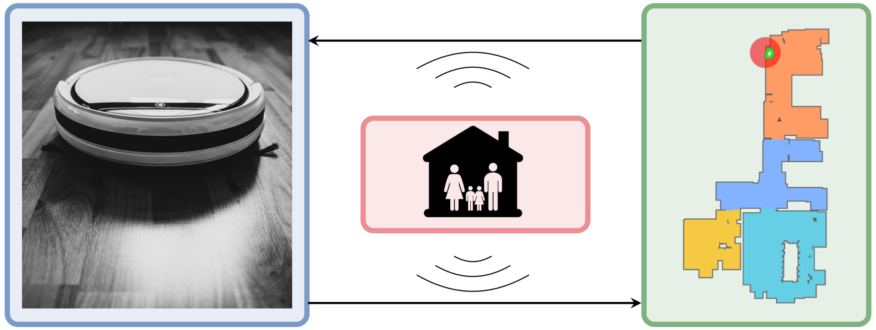


Vacuum Agent: Sensors and Actuators



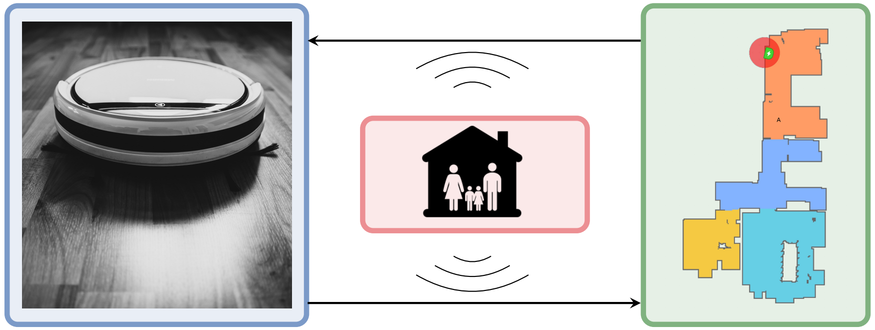
- sensors: cliff sensors, bump sensors, wall sensors, state of charge sensor, WiFi module
- actuators: wheels, cleaning system

Vacuum Agent: Observations and Actions



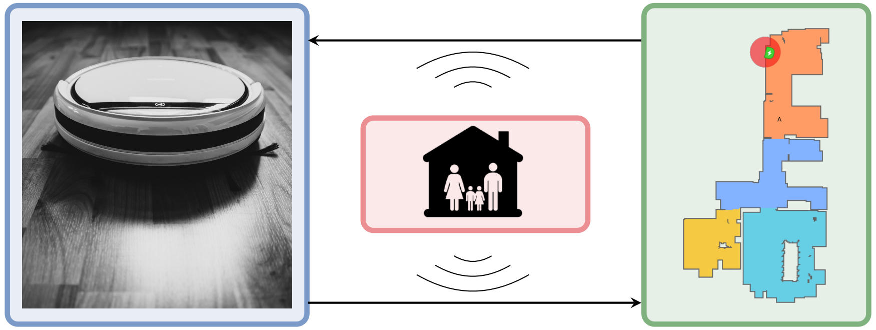
- observations: current location, cleanness of current room
  state of battery charge, presence of humans
- actions: move-to-next-room, move-to-base, vacuum, wait

Vacuum Agent: Agent Program



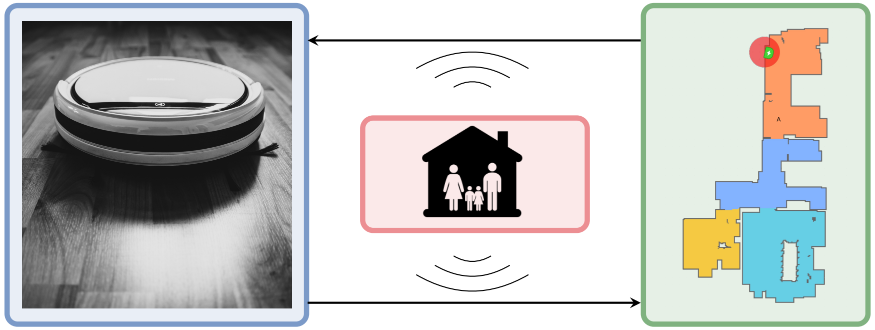
```
def vacuum-agent([$cleanness$, $owner\text{-}present$, $battery$]):
    if $battery \leq 10\%$: return $move\text{-}to\text{-}base$
    else if $owner\text{-}present$ = True: return $move\text{-}to\text{-}next\text{-}room$
    else if $cleanness$ = dirty: return $vacuum$
    else: return $move\text{-}to\text{-}next\text{-}room$
```

Vacuum Domain: Agent Function



observation sequence	action
⟨[clean, False, 100%]⟩	*move-to-next-room*
⟨[dirty, False, 100%]⟩	*vacuum*
⟨[clean, True, 100%]⟩	*move-to-next-room*
...	...
⟨[clean, False, 100%], [clean, False, 90%]⟩	*move-to-next-room*
⟨[clean, False, 100%], [dirty, False, 90%]⟩	*vacuum*
...	...

Vacuum Domain: Performance Measure



potential influences on performance measure:

- cleanliness
- times vacuum-cleaned
- distance travelled
- safety
- energy consumption
- disturbance of owners

3.3 Rationality

Evaluating Agent Functions

What is the right agent function?

Rationality

rationality of an agent depends on performance measure
(often: utility, reward, cost) and environment

Perfect Rationality
- for each possible observation sequence
- select an action which maximizes*
- expected value of future performance
- given available information on observation history
- and environment

*sometimes minimize, e.g. in case of costs

Perfect Rationality of Our Vacuum Agent

Is our vacuum agent perfectly rational?

depends on performance measure and environment, e.g.:
- Do actions reliably have the desired effect?
- Do we know the initial situation?
- Can new dirt be produced while the agent is acting?

Performance Measure

- usually specified by developer
- sometimes clear,
  sometimes not so clear
- significant impact on
  - desired behavior
  - difficulty of problem

Perfect Rationality of Our Vacuum Agent

consider performance measure:

- +1 utility for cleaning a dirty room −1 utility for each dirty room in each step

consider environment:

- actions and observations reliable
- world only changes through actions of the agent
- non-zero probability that yellow room becomes dirty

our vacuum agent is perfectly rational our vacuum agent is not perfectly rational

Rationality: Discussion

- perfect rationality ≠ omniscience
  - incomplete information (due to limited observations) reduces achievable utility
- perfect rationality ≠ perfect prediction of future
  - uncertain behavior of environment (e.g., stochastic action effects) reduces achievable utility
- perfect rationality is rarely achievable
  - limited computational power ⇝ bounded rationality

3.4 Summary

Summary (1)

common metaphor for AI systems: rational agents

agent interacts with environment:

- sensors perceive observations about state of the environment
- actuators perform actions modifying the environment
- formally: agent function maps observation sequences to actions
- reflexive agent: agent function only based on last observation

Summary (2)

rational agents:

- try to maximize performance measure (utility)
- perfect rationality: achieve maximal utility in expectation given available information
- for “interesting” problems rarely achievable ⇝ bounded rationality