COMS W4701: Artificial Intelligence

Lecture 1: Introduction, Intelligent Agents

Tony Dear, Ph.D.
Department of Computer Science
School of Engineering and Applied Sciences

Today

- Course syllabus and logistics
- Definition, foundations, and modern capabilities of AI
- Properties of **task environments**
- Structure and types of **intelligent agents**

Course Expectations

- MS-level (4XXX) CS course

- Your peers: Mostly CS undergrads, CS grads, and SEAS grads
- Some taking first 4000-level course, others taking first CS course

- Coursework: Both **programming** and **quantitative analysis**

- Must be able to learn independently, keep up with course

Course Expectations

- Attendance not required, but **try your best** to attend live
- Recordings uploaded by CVN within 24 hours of each lecture

- We are covering material **twice as fast** as usual
- You should expect workload equivalent to **two** regular courses

- University course hour requirements for immersive courses: 6 hours in class, 12 hours out of class; **18 hours total weekly**

Enrollment

- Waitlist for section 001 closes at the end of the day on Friday 9/11

- If you're still on the waitlist now, you can either
  - Complete a short form to indicate interest (https://forms.gle/EtZUxk28fSkowraaA), or
  - Add yourself to the waitlist for Section 002 (Fall B) instead

- Waitlist for section 002 (Fall B) closes a week later on Friday 9/18
- If you're already enrolled in or waitlisted for Section 002, we will not enroll you into Section 001

Auditing

- Public access to our Courseworks page will close on Friday 9/11

- Students interested in auditing **and who do not intend to enroll in Fall B** may petition to audit by completing this form: https://forms.gle/x8xJugaNa4JcKE4G6

- If flexible, you're encouraged to audit the fall B section instead—fewer students all around, better experience for everyone

What is Artificial Intelligence?

- Two dimensions: thinking vs acting, humanly vs rationally

- Acting humanly, i.e. pass the **Turing test**
  - Capabilities: Natural language processing, knowledge representation, automated reasoning, machine learning, computer vision, robotics

- Example application areas of AI
- Modern AI extends well beyond "human" behaviors

What is Artificial Intelligence?

- Thinking humanly: Studied in **cognitive science**
  - AI models can be used in psychological experiments, but human thinking is not necessary to excel at different tasks

- Thinking rationally: The "laws of thought"
  - If we know all the rules of the world, **logic** and **inference** can help connect observations with understanding and predictions
  - But no clear connection to intelligent behavior

Acting Rationally: The Standard Model

- An **agent** autonomously interacts with an environment through *perception* and *action* to achieve pre-defined goals

- A **rational agent** tries to achieve the *best expected outcome*
- Focus on optimal **behavior**, not optimal reasoning or inference

An Interdisciplinary Field

- AI draws ideas and techniques from many other fields

- **Philosophy**: Logic, inference, theory of knowledge and the mind
- **Mathematics**: Formal logic, uncertainty, algorithms, computability
- **Economics**: Decision making, game theory, multiagent systems
- **Neuroscience** and **psychology**: Study of the brain, thoughts, behaviors
- **Control theory**: Autonomous control, feedback control, optimal control
- **Linguistics**: Knowledge representation, natural language processing

History of AI: Inception

- First AI work founded on neural networks
  - Boolean circuit model of brain (McCulloch and Pitts, 1943), neuron learning update rules (Hebb, 1949), neural network computer (Minsky and Edmonds, 1950)

- 1956: **Dartmouth meeting** of 10 influential researchers
  - First usage of and declared interest in “artificial intelligence”

- 1950s: Development of logic, math, and theorem-proving systems; games like checkers programs; planners; miniature worlds with limited domains

History of AI: Boom and Bust

- Early AI researchers were overly optimistic and overconfident
  - Herbert Simon: AI systems will solve problems “coextensive with the range to which the human mind has been applied” (1957)
- Challenges: Early systems relied too much on human methods; complex problems quickly became computationally intractable
- 1970s and 1980s: **Expert systems** utilized domain-specific knowledge
  - Advances in representation and reasoning tools, natural language understanding
- Late 1980s: **AI winter** as systems failed to deal with uncertainty and learning

Modern AI

- Late 1980s-1990s: Shift toward probability, experience, machine learning
- Less emphasis on philosophy, intuition, and symbolic computation

- 1988: Introduction of **Bayesian networks** (Pearl) for probabilistic reasoning; connection of decision theory and **reinforcement learning** (Sutton)

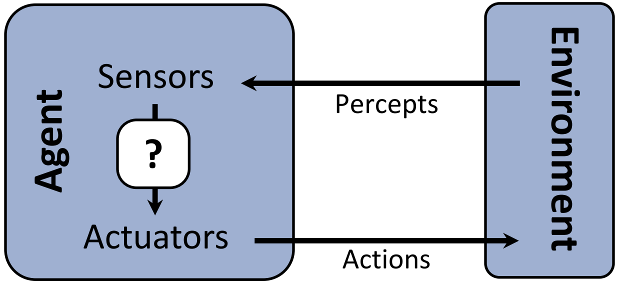
- 2000s-present: **Big data** facilitated success of new ML algorithms
- 2010s-present: **Deep learning** using multiple-layer neural networks facilitated by hardware improvements, starting in speech and visual recognition

Modern AI Applications

- Robotics: Autonomous vehicles, drones, legged systems
- Planning and scheduling (mapping directions)
- Machine translation and speech recognition (Skype, Alexa, Siri)
- Recommender systems (Amazon, YouTube, Spotify, Netflix, Instagram)
- Game playing (AlphaGo and AlphaZero, Atari, StarCraft)

Agents and Environments

- An agent uses **sensors and actuators** to interact with its environment
- Anything can be an agent, e.g. humans, robots, software systems



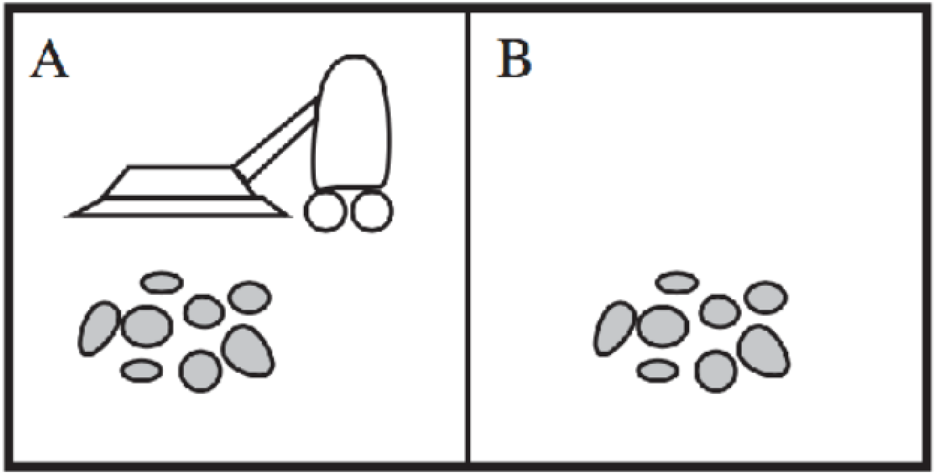
- Agent's actions may depend on its **percepts**
- May even depend on entire **percept sequence**

- **Agent function** maps percept sequence to action

- **Environment** is (smallest) part of the universe directly interacting with agent

Example: Vacuum Cleaner World

- Agent: Vacuum cleaner
- Environment: Square A and square B



Example: Vacuum Cleaner World

- Percepts: Current square; is the square dirty?
- Actions: Move left, move right, clean, do nothing

- Agent function:

[A, IsClean]	Move right
[A, IsDirty]	Clean
[B, IsClean]	Move left
[B, IsDirty]	Clean
[[A, IsDirty], [A, IsClean]]	Move right
[[B, IsDirty], [A, IsClean]]	Move right
[[B, IsClean], [A, IsClean]]	Do nothing
……	……

Rational Agents

- Sequence of environment states evaluated by a **performance measure**
- Performance measures usually based on *desired outcomes*, not behaviors

- A **rational agent** selects an action to *maximize* its performance measure given percept sequence and in-built knowledge.
  - What performance measure makes our vacuum cleaner rational or irrational?

- Rational agents maximize *expected* performance, are not omniscient
- Rationality may involve info gathering, exploration, learning

Task Environments

- **PEAS**: Performance measure, environment, actuators, sensors
- A rational agent is a *solution* to a given task environment

- Vacuum cleaner task environment
  - P: Cleanliness, power usage, time taken
  - E: The small grid world
  - A: Wheels to move, filter to clean
  - S: “GPS”, cleanliness sensor

Example: Self-Driving Taxi

- **P**: Safe, fast, legal, comfortable, profit-maximizing
  - Performance measures can be complementary or contradictory!
- **E**: Roads, other traffic, pedestrians, customers, weather
  - Not just “physical” environment, but also everything taxi interacts with
- **A**: Steering, acceleration, brakes, turn signals, horn, etc.
- **S**: Cameras, GPS, speedometer, accelerometer, odometer, etc.

Task Environment Properties

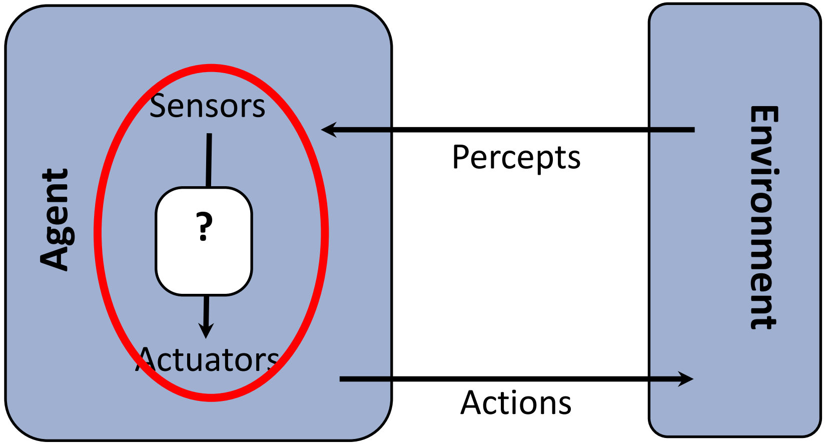
- Fully observable vs partially observable vs unobservable
  - Can agent access all *relevant* information?
- Single-agent vs multi-agent
  - Does behavior of other agents depend on what we do?
- Deterministic vs stochastic
  - Can we determine environment changes completely based on our actions?
- Episodic vs sequential
  - Do future decisions depend on what we do now?
- Static vs dynamic
  - Does the environment change while the agent is thinking or doing nothing?
- Discrete vs continuous
  - Is number of states, actions, percepts, time, etc. finite?

Examples of Environments

Environment	Partially / Fully Observable	Single- / Multi-Agent	Deterministic / Stochastic	Sequential / Episodic	Dynamic / Static	Continuous / Discrete
Vacuum cleaner world	Partially	Single	Deterministic	Sequential	Static	Discrete
Chess	Fully	Multi (adversarial)	Deterministic	Sequential	Static	Discrete
Self-driving car*	Partially	Multi (cooperative)	Stochastic	Sequential	Dynamic	Continuous
Image classification	Fully	Single	Deterministic	Episodic	Static	Depends

Agent Programs

- **Agent programs** (percept to action) *implement* agent functions (percept sequence to action)
- Simplest idea: Lookup table indexed by all possible percept sequences
- What's the problem?



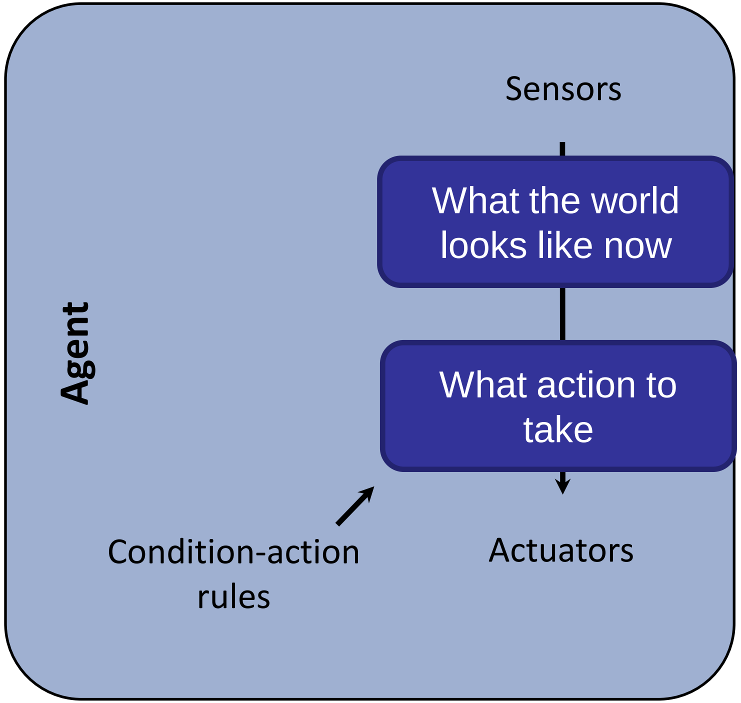
Simple Reflex Agents

- Simple reflex agent: Use current percept only
- Can be implemented using if-then rules

- Environment must be fully observable!

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*



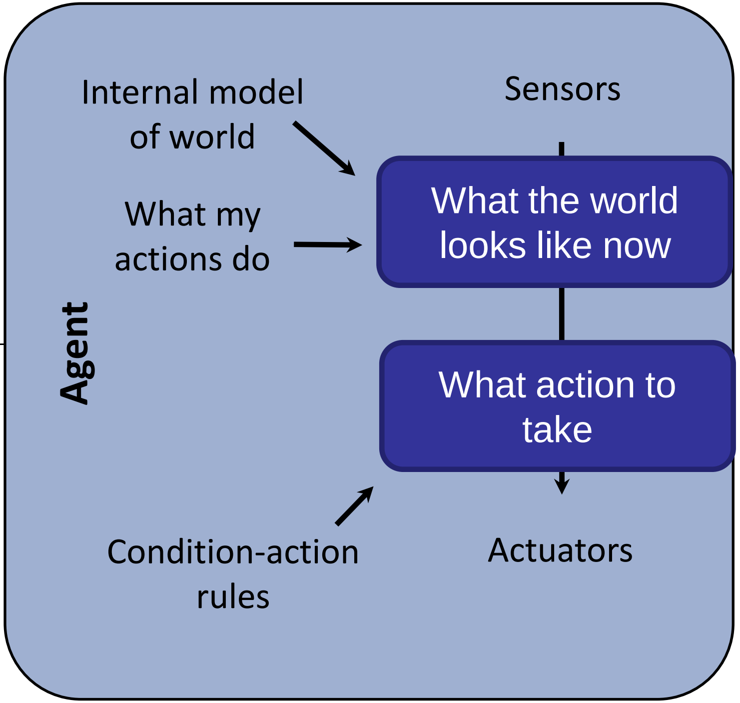
Model-Based Reflex Agents

- What about partially observable environments?

- Maintain an **internal state** of the world!
- **Transition model**: How the world changes

```
function MODEL-BASED-REFLEX-AGENT(percept) returns an action
   persistent: state, the agent's current conception of the world state
               model, a description of how the next state depends on current state and action
               rules, a set of condition–action rules
               action, the most recent action, initially none

   state ← UPDATE-STATE(state, action, percept, model)
   rule ← RULE-MATCH(state, rules)
   action ← rule.ACTION
   return action
```

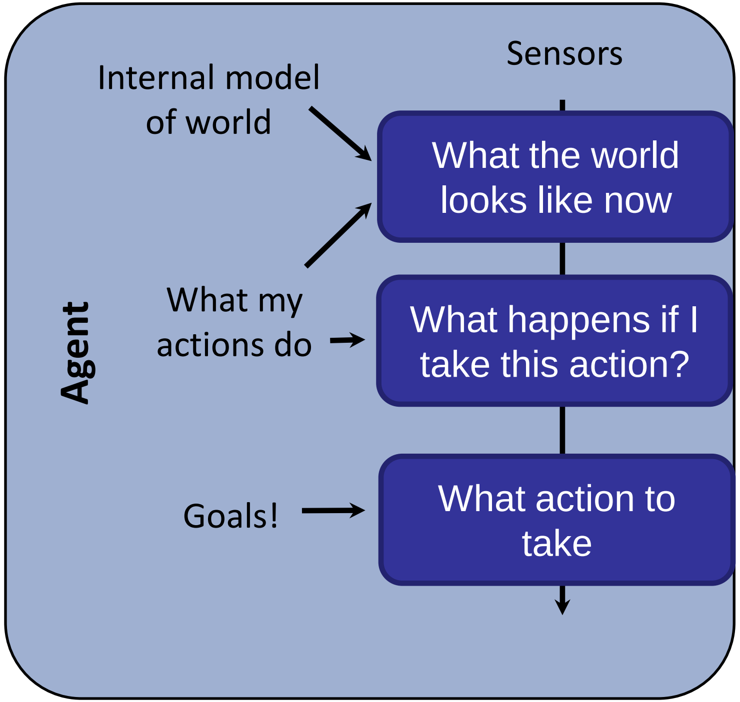


Goal-Based Agents

- Reflex agents are very rigid and predictable
- How to change a reflex agent's behavior?

- Include *goal* information in program
- Goals may be encoded using **utilities**
- Internalize the overall performance measure

- Utilities specify tradeoffs for competing goals
- Also useful in face of uncertainty



Learning Agents

- How to change or improve an existing agent program?
- Incorporate as *performance element* into a **learning agent**

- A *learning element* can make changes in the performance element
- Does so using information from a *critic*, which evaluates current environment according to performance measure

- A *problem generator* may suggest exploratory actions to obtain new information

Summary

- AI is the interdisciplinary study of designing rational agents
- Lots of ups and downs from inception in 1940s
- Modern-day AI: Probabilistic methods, big data, machine learning

- Rational agents maximize expected performance measure
- Difficulty of a task environment depends on specific properties

- Agent programs may use current percept only, keep a model around, and/or try to achieve certain goals quantified by utilities

To-Do List: First Week

- If you’re not yet enrolled, fill out interest form or auditing form
- Sign up for Piazza and fill out introductory polls

- Obtain textbook, start getting familiar with Jupyter Notebook

- HW 1 to be released tomorrow—start looking over this weekend!
- Quiz 1 out on Sunday, due before lecture on Monday 9/14