Representation & Reasoning, Representational Dimensions, Course Overview

Alan Mackworth

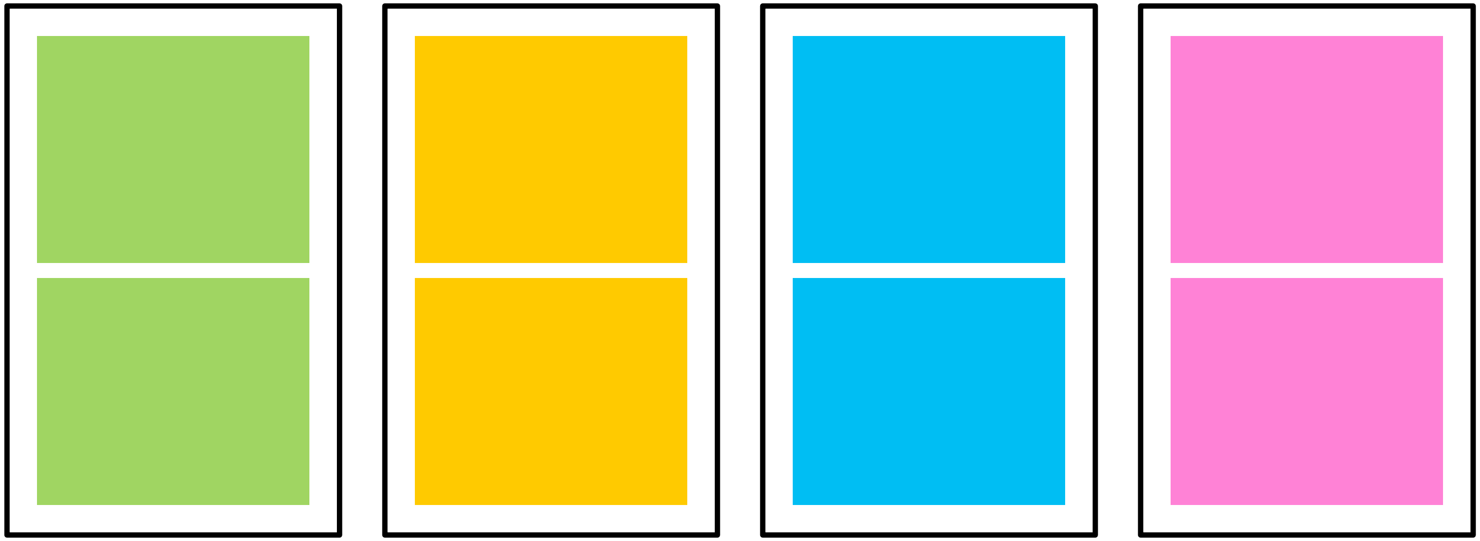
CS 322 – Intro 2

January 4, 2013

Textbook §1.4 - 1.5

Coloured Cards

- Please come to the front and pick up
  - 4 index cards
  - 2 Post-it stickers per colour (Blue, Yellow, Green, Pink)
- Use this material to make 4 voting cards as below
  - Cheap low tech variant of clickers
- Please bring them to class every time



Today's Lecture

- Recap from last lecture
- Representation and Reasoning
- An Overview of This Course
- Further Representational Dimensions

Teaching team & office hours

- Instructor
  - Alan Mackworth mack@cs.ubc.ca ICCS 121
    - Monday and Wednesday 4pm – 4:30pm in ICCS 121

- TAs: All office hours in the Demco Learning Center: ICCS X150 (behind Reboot Cafe) starting next week
  - Shafiq Joty rjoty@cs.ubc.ca
    - Monday, 1pm – 2pm

  - Mehran Kazemi smkazemi@cs.ubc.ca
    - Wednesday, 11am - 12am

  - Pooyan Fazli pooyanf@cs.ubc.ca
    - Friday, 12am – 1pm

Course Essentials

- Website: http://www.ugrad.cs.ubc.ca/~cs322
- Main Textbook
  - Artificial Intelligence: Foundations of Computational Agents. By Poole and Mackworth. (P&M)
  - Available electronically (free) http://artint.info/html/ArtInt.html
  - We will cover Chapters: 1, 3, 4, 5, 6, 8, 9
- Connect
  - Assignments posted there
  - Practice exercises (ungraded)
  - Learning goals
  - Discussion board
  - Check it often

What is Artificial Intelligence?

- We use the following definition

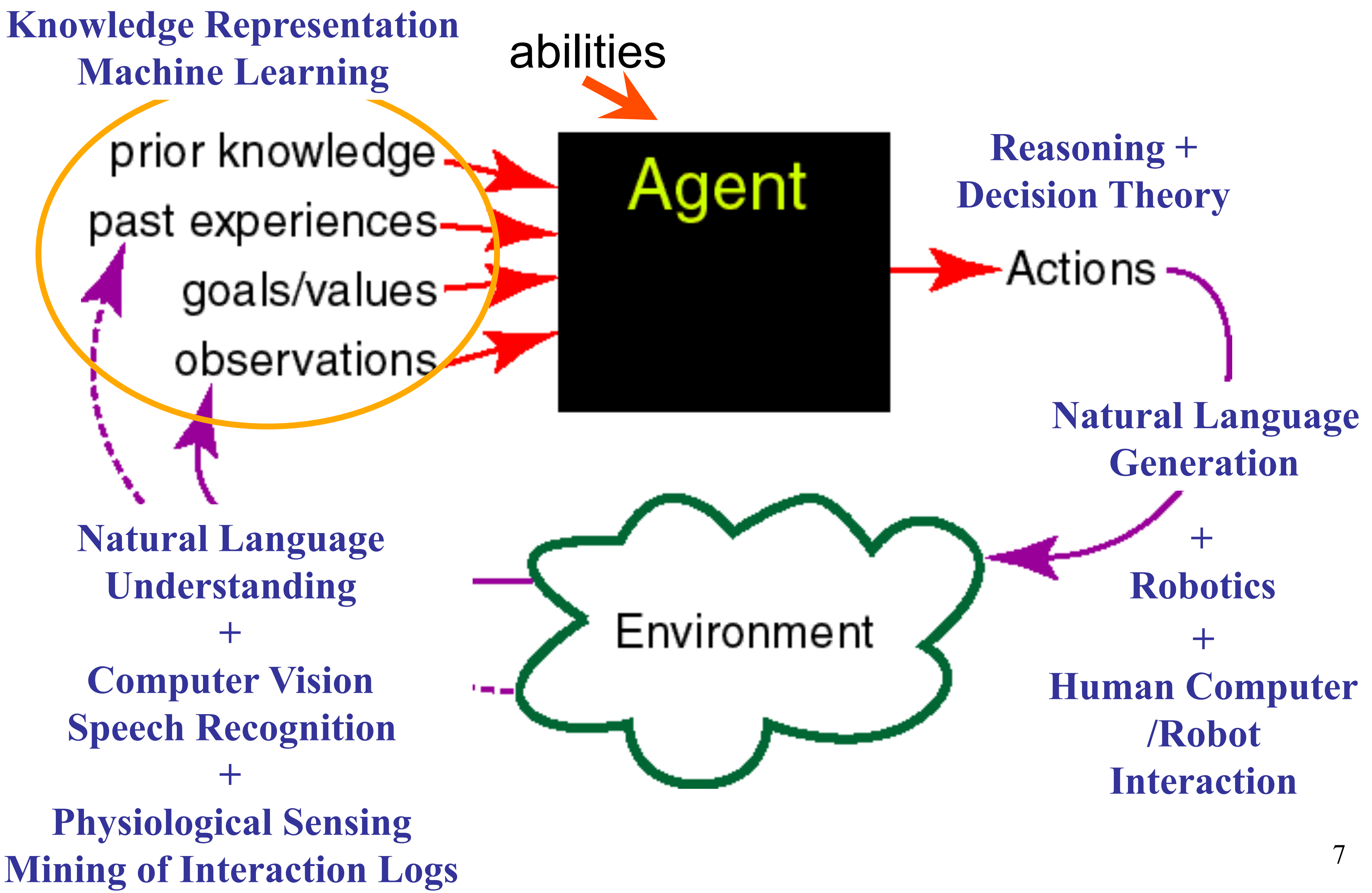
Systems that think rationally

Systems that act like humans

Systems that act rationally

Systems that think like humans

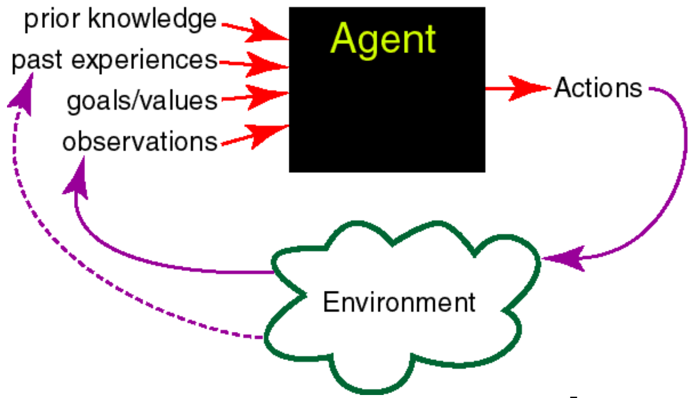
Intelligent Agents in the World



Today's Lecture

- Recap from last lecture
- ➡ Representation and Reasoning
- An Overview of This Course
- Further Representational Dimensions

Representation and Reasoning



To use these inputs an agent needs to represent them

⇒ knowledge

One of AI's goals: specify how a system can

- Acquire and represent knowledge about a domain (representation)
- Use the knowledge to solve problems in that domain (reasoning)

Representation and Reasoning (R&R) System

problem $\Longrightarrow$ representation $\Longrightarrow$ computation $\Longrightarrow$ representation $\Longrightarrow$ solution

- A representation language that allows description of
  - The environment and
  - Problems (questions/tasks) to be solved
- Computational reasoning procedures to
  - Compute a solution to a problem
  - E.g., an answer/sequence of actions
- Choice of an appropriate R&R system depends on
  - Various properties of the environment, the agent, the computational resources, the type of problems, etc.

What do we want from a representation?

We want a representation to be:

- rich enough to express the knowledge needed to solve the problem
- as close to the problem as possible: compact, natural and maintainable
- amenable to efficient computation; able to express features of the problem we can exploit for computational gain
- learnable from data and past experiences
- able to trade off accuracy and computation time

We want a representation for a problem to be…

… as general as possible

… as close to the problem as possible

Today's Lecture

- Recap from last lecture
- Representation and Reasoning
- ⇨ An Overview of This Course
- Further Representational Dimensions

High-level overview of this course

This course will emphasize two main themes:

- Reasoning
  - How should an agent act given the current state of its environment and its goals?
- Representation
  - How should the environment be represented in order to help an agent to reason effectively?

Main Representational Dimensions Considered

Domains can be classified by the following dimensions:

- 1. Uncertainty
  - Deterministic vs. stochastic domains
- 2. How many actions does the agent need to perform?
  - Static vs. sequential domains

An important design choice is:

- 3. Representation scheme
  - Explicit states vs. propositions vs. relations

1. Deterministic vs. Stochastic Domains

Historically, AI has been divided into two camps:

- those who prefer representations based on logic
- those who prefer probability

- Is the agent's knowledge certain or uncertain?
  - Chess vs. poker
- Is the environment deterministic or stochastic?
  - Is the outcome of an action certain?

  E.g. Filling in Sudoku vs. slippage in a robot, coin toss, ball kick, …

- Some of the most exciting current research in AI is building bridges between these camps

2. Static vs. Sequential Domains

How many actions does the agent need to select?

- The agent needs to take a single action
  - solve a Sudoku
  - diagnose a patient with a disease
- The agent needs to take a sequence of actions
  - navigate through an environment to reach a goal state
  - bid in online auctions to purchase a desired good
  - decide sequence of tests to enable a better diagnosis of the patient

Caveat:

- Distinction between the two can be a bit artificial
  - In deterministic domains, we can redefine actions (e.g., fill in individual numbers in the Sudoku vs. solving the whole thing)
  - Not in stochastic domains (Why?)

3. Explicit State vs. Features

How do we model the environment?

- You can enumerate the possible states of the world OR
- A state can be described in terms of features
  - Often the more natural description
  - 30 *binary* features can represent 2^{30} =1,073,741,824 states

3. Explicit State vs. Features (cont'd)

Mars Explorer Example

Weather	{S, C}
Temperature	[-40, 40]
Longitude	[0, 359]
Latitude	[1, 179]

(North Pole and South Pole excluded.Why?)

One possible state: {S, -30, 320, 110}

Number of possible states (mutually exclusive)

2 x 81 x 360 x 179

3. Explicit State vs. Features vs. Relations

- States can be described in terms of objects and relationships
- There is a proposition for each relationship on each tuple of objects
- University Example:
  - Students (S) = {s1, s2, s3, …, s200)
  - Courses (C) = {c1, c2, c3, …, c10}
  - Relation: Registered (S, C)
  - E.g. one proposition: Registered(s73, c4). In each state a proposition can be true or false: it is a binary feature.
  - Number of Relations: 1
  - Number of Propositions: 200*10 | 200+10 | 10^{200} | 200^{10}

3. Explicit State vs. Features vs. Relations

- States can be described in terms of objects and relationships
- There is a proposition for each relationship on each tuple of objects

- University Example:
  - Students (S) = {s1, s2, s3, …, s200)
  - Courses (C) = {c1, c2, c3, …, c10}
  - Registered (S, C)

  - Number of Relations: 1
  - Number of Propositions: 200*10 = 2000
  - Number of States: $2000*2$ $2000+2$ 2000^2 2^{2000}

Course Map

Dimen-sions / Course Modules	Deterministic vs. Stochastic	Static vs. Sequential	States vs. Features vs. Relations
1. Search	Deterministic	Static	States
2. CSPs	Deterministic	Static	Features
3. Planning	Deterministic	Sequential	States or Features
4. Logic	Deterministic	Static	Relations
5. Uncertainty	Stochastic	Static	Features
6. Decision Theory	Stochastic	Sequential	Features

Example reasoning tasks for delivery robot

Dimen-sions / Course Modules	Deterministic vs. Stochastic	Static vs. Sequential	States vs. Features vs. Relations
1. Search	Deterministic	Static	States
2. CSPs	Deterministic	Static	Features
3. Planning	Deterministic	Sequential	States or Features
4. Logic	Deterministic	Static	Relations
5. Uncertainty	Stochastic	Static	Features
6. Decision Theory	Stochastic	Sequential	Features

"find path in known map"

"are deliveries feasible?"

"what order to do things in to finish jobs fastest?"

"HasCoffee(Person) if InRoom(Person, Room) ∧ DeliveredCoffee(Room)"

"probability of slipping"

"given that I may slip and the utilities of being late and of crashing, should I take a detour?"

Today's Lecture

- Recap from last lecture
- Representation and Reasoning
- An Overview of This Course
- ➡ Further Representational Dimensions

Further Dimensions of Representational Complexity

We've already discussed:

1. Deterministic versus stochastic domains
2. Static vs. Sequential domains
3. Explicit state or features or relations

Some other important dimensions of complexity:

4. Flat vs. hierarchical representation
5. Knowledge given vs.
   knowledge learned from experience
6. Goals vs. complex preferences
7. Single-agent vs. multi-agent
8. Perfect rationality vs. bounded rationality

4. Flat vs. hierarchical

- Should we model the whole world on the same level of abstraction?
  - Single level of abstraction: flat
  - Multiple levels of abstraction: hierarchical

- *Example: Planning a trip from here to a resort in Cancun*
  - *Going to the airport*
    - *Take a cab*
      - *Call a cab*
        - *Lookup number*
        - *Dial number*
      - *Ride in the cab*
      - *Pay for the cab*
  - *Check in*
  - *….*

- Delivery robot: Plan on levels of cities, districts, buildings, …
- This course: mainly flat representations
  - Hierarchical representations required for scaling up.

5. Knowledge given vs. knowledge learned from experience

- The agent is provided with a model of the world once and for all OR

- The agent can learn how the world works based on experience
  - in this case, the agent almost always starts out with some prior knowledge (no *tabula rasa*!)

- Delivery robot: Known/learned map, prob. of slipping, …
- This course: *mostly* knowledge given
  - Learning: CS 340 and CS 422

6. Goals vs. (complex) preferences

- An agent may have a goal that it wants to achieve
  - E.g., there is some state or set of states of the world that the agent wants to be in
  - E.g., there is some proposition or set of propositions that the agent wants to make true
- An agent may have preferences
  - E.g., a preference/utility function describes how happy the agent is in each state of the world
  - Agent's task is to reach a state which makes it as happy as possible
- Preferences can be complex
  - E.g., diagnostic assistant faces multi-objective problem
    - Life expectancy, suffering, risk of side effects, costs, …
- Delivery robot: "deliver coffee!" vs "mail trumps coffee, but Chris needs coffee quickly, and don't stand in the way"
- This course: goals and simple preferences
  - Some scalar, e.g. linear combination of competing objectives

7. Single-agent vs. Multi-agent domains

- Does the environment include other agents?
- If there are other agents whose actions affect us
  - It can be useful to explicitly model their goals and beliefs, and how they react to our actions

- Other agents can be: cooperative, competitive, or a bit of both

- Delivery robot: Are there other agents?
  - Should I coordinate with other robots?
  - Are kids out to trick me?
- This course: only single agent scenario
  - Multi-agent problems tend to be complex (soccer)
  - Exception: deterministic 2-player games can be formalized easily

8. Perfect rationality vs. bounded rationality

We've defined rationality as an abstract ideal

- Is the agent able to live up to this ideal?
  - Perfect rationality:
    the agent can derive what the best course of action is
  - Bounded rationality:
    the agent must make good decisions
    based on its perceptual, computational and memory limitations
- Delivery robot:
  - ”Find perfect plan” vs.
  - “Can’t spend an hour thinking (thereby delaying action) to then deliver packages a minute faster than by some standard route”
- This course: *mostly* perfect rationality
  - But also consider anytime algorithms for optimization problems

Summary(1)

Would like most general agents possible, but to start we need to restrict ourselves to:

4. Flat representations (vs. hierarchical)
5. Knowledge given (vs. knowledge learned)
6. Goals and simple preferences (vs. complex preferences)
7. Single-agent scenarios (vs. multi-agent scenarios)
8. Perfect rationality (vs. bounded rationality)

Extensions we will cover:

1. Deterministic versus stochastic domains
2. Static vs. Sequential domains
3. Representation: Explicit state or features or relations

Summary (2)

- Right representation: Rich enough but close to the problem
- Course Map:

Dimen-sions / Course Modules	Deterministic vs. Stochastic	Static vs. Sequential	States vs. Features vs. Relations
1. Search	Deterministic	Static	States
2. CSPs	Deterministic	Static	Features
3. Planning	Deterministic	Sequential	States or Features
4. Logic	Deterministic	Static	Relations
5. Uncertainty	Stochastic	Static	Features
6. Decision Theory	Stochastic	Sequential	Features

For You TO DO!

- For Monday: carefully read Section 1.6
  - Prototypical applications
- For next Friday: Assignment 0
  - Available on Connect
  - This class has covered all you need to know for the assignment
  - Sections 1.5 & 1.6 in the textbook will also be particularly helpful