# Representational Dimensions

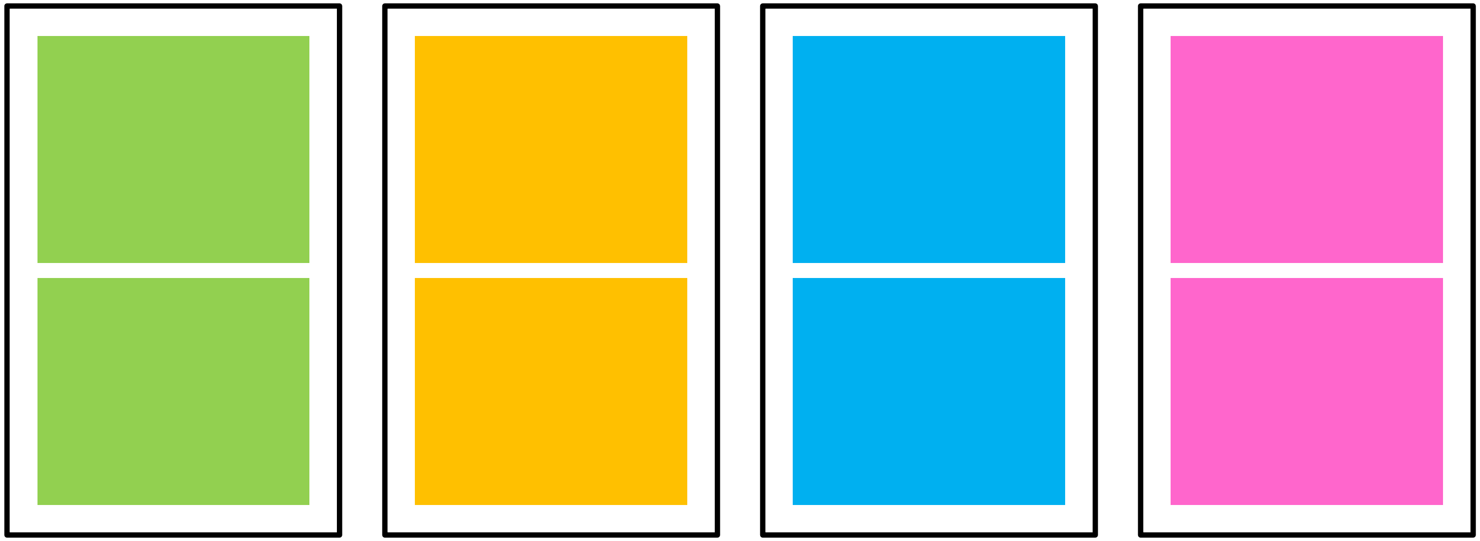
CPSC 322 – Intro 2

January 7, 2011

Textbook §1.4 - 1.5

# Colored Cards

- Please come to the front and pick up
  - 4 index cards
  - 2 Post-it per colour (Blue, Yellow, Green, Pink)
- Use this material to make 4 “voting cards” as below
  - Low budget 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
  - Frank Hutter (hutter@cs.ubc.ca; Beta lab, ICICS X560)
    - Monday, Wednesday, Friday, 4-4:30pm in ICICS X530

- TAs
  - All office hours in the Demco Learning Center: ICICS X150 (behind Reboot Cafe)

  - Simona Radu (sradu@cs.ubc.ca)
    - Monday, 11am-12pm.

  - Vasanth Rajendran (vasanthr@cs.ubc.ca)
    - Thursday, 3pm-4pm

  - Mike Chiang (mchc@cs.ubc.ca)
    - Wed 1- 2pm

# 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
- WebCT
  - 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

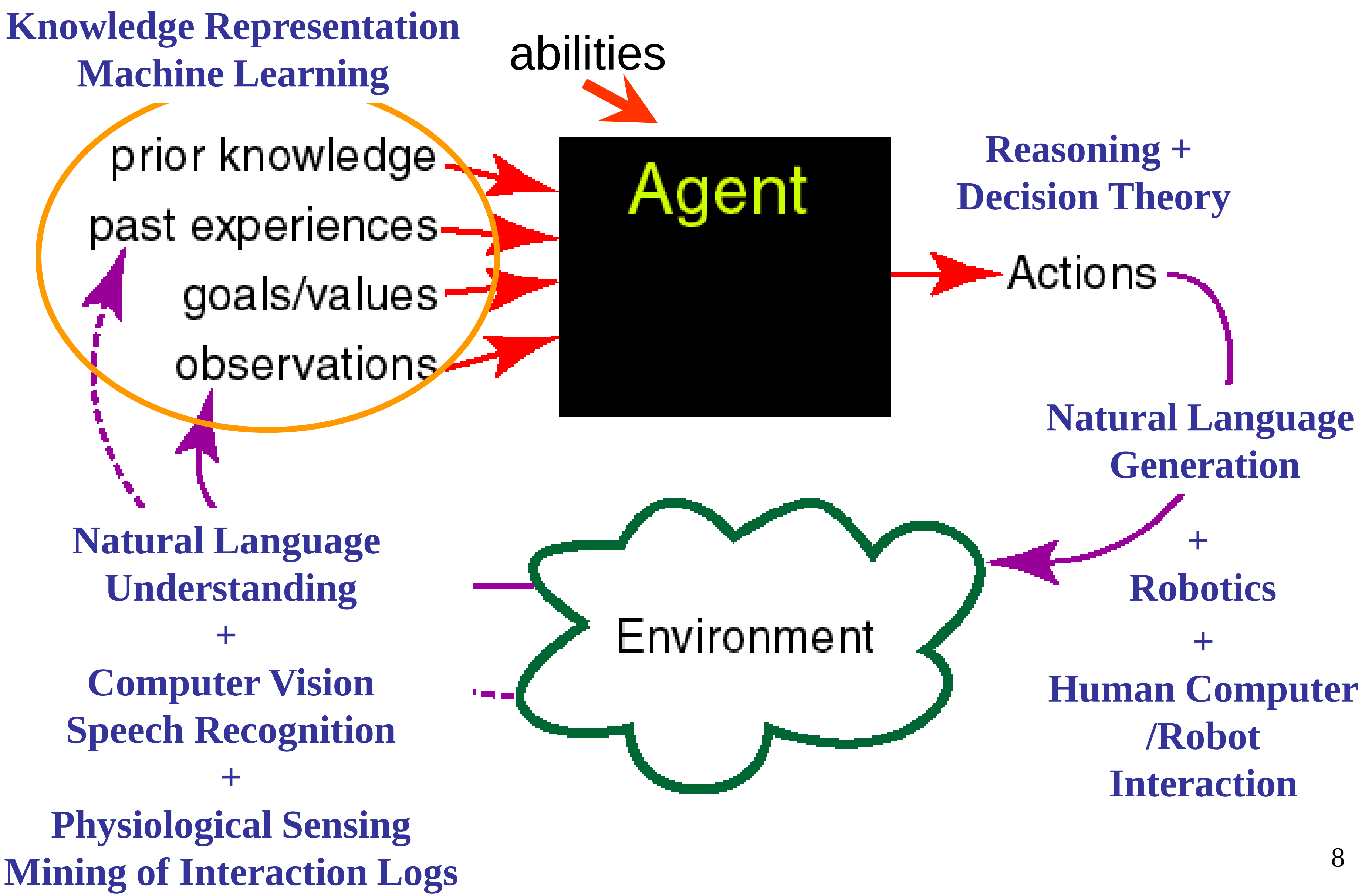
Systems that think like humans

# What is Artificial Intelligence?

- We use the following definition

Systems that act rationally

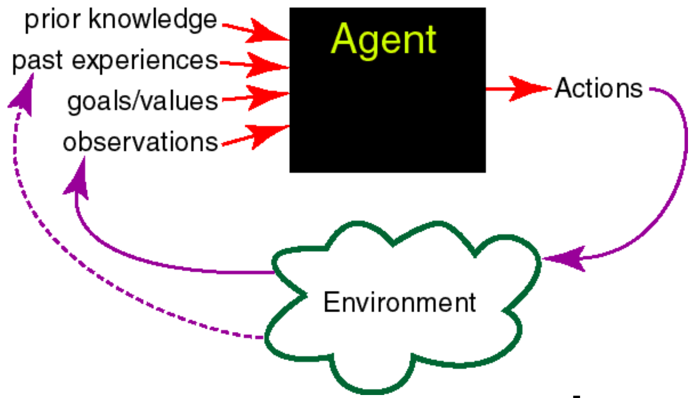
# Intelligent Agents in the World



# 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 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

- A representation language that allows to describe
  - 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 of 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?
  - Poker vs. chess
- Is the environment deterministic or stochastic?
  - Is the outcome of an action certain? E.g. slippage in a robot

- Some of the most exciting current research in AI is actually 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

# 3. Explicit State vs. Features

How do we model the environment?

- You can enumerate the possible states of the world
- 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* | [0, 179] |

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

Number of possible states (mutually exclusive)

2 x 81 x 360 x 180

# 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 | 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

| Dimensions / Course Modules | Deterministic vs. Stochastic | Static vs. Sequential | States vs. Features vs. Relations | |
|---|---|---|---|---|
| 1. Search | Deterministic | Static | States | "find path in known map" |
| 2. CSPs | Deterministic | Static | Features | "are deliveries feasible?" |
| 3. Planning | Deterministic | Sequential | States or Features | "what order to do things in to finish jobs fastest?" |
| 4. Logic | Deterministic | Static | Relations | "HasCoffee(Person) if InRoom(Person, Room) ∧ DeliveredCoffee(Room)" |
| 5. Uncertainty | Stochastic | Static | Features | "probability of slipping" |
| 6. Decision Theory | Stochastic | Sequential | Features | "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, Mexico*

- Delivery robot: Plan on level of cities, districts, buildings, …
- This course: only flat representations
  - Hieararchical representations pose mainly engineering problems

# 5. Knowledge given vs. knowledge learned from experience

- The agent is provided with a model of the world once and far all
- The agent can learn how the world works based on experience
  - in this case, the agent often still does start out with some prior knowledge
- Delivery robot: Known/learned map, prob. of slipping, …
- This course: mostly knowledge given
  - Learning: CPSC 340

# 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. Multiagent 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
  - Multiagent problems tend to be complex
  - 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 |

# TODOs

- For Monday: carefully read Section 1.6
  - Prototypical applications
- For Wednesday: Assignment 0
  - Available on WebCT
  - This class should have covered all you need to know for the assignment
  - Section 1.5 & 1.6 in the textbook will also be particularly helpful