Applications of AI

CPSC 322 - Intro 3
January 10, 2011

Textbook §1.5 - 1.6
Today’s Lecture

Recap from last lecture

- Further Representational Dimensions
- Applications of AI
Representation and Reasoning (R&R) System

Problem $\Rightarrow$ representation $\Rightarrow$ 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

• How should an agent act given the current state of its environment and its goals?
• How should the environment be represented in order to help an agent to reason effectively?
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
Features vs. States, another example

$T_{11}$: student 1 takes course 1
$T_{12}$: student 1 takes course 2
$T_{21}$: student 2 takes course 1
$T_{22}$: student 2 takes course 2

Does student 2 take course 2?
- Feature-based: Is $T_{22}$ true?
- State-based: are we in one of the red states?

<table>
<thead>
<tr>
<th></th>
<th>$T_{11}$</th>
<th>$T_{12}$</th>
<th>$T_{21}$</th>
<th>$T_{22}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_0$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S_1$</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$S_2$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$S_3$</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$S_4$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S_5$</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$S_6$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$S_7$</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$S_8$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S_9$</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$S_{10}$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$S_{11}$</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$S_{12}$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$S_{13}$</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$S_{14}$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$S_{15}$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
## Course overview

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
<td></td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
<td></td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
<td></td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
<td></td>
</tr>
</tbody>
</table>

**Example problems:**

- “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
• Further Representational Dimensions
• Applications of AI
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
  - Hierarchical 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 for 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
Today’s Lecture

• Recap from last lecture
• Further Representational Dimensions

Applications of AI
Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td><strong>Sequential</strong></td>
<td>States or Features</td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td><strong>Stochastic</strong></td>
<td>Static</td>
<td>Features</td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td><strong>Stochastic</strong></td>
<td><strong>Sequential</strong></td>
<td>Features</td>
</tr>
</tbody>
</table>
Search: Checkers

• Early work in 1950s by Arthur Samuel at IBM

• Chinook program by Jonathan Schaeffer (UofA)
  – Search to explore the space of possible moves and their consequences
    – 1994: world champion
    – 2007: declared unbeatable
Search: Chess

- In 1997, **Gary Kasparov**, the chess grandmaster and reigning world champion played against **Deep Blue**, a program written by researchers at IBM.

Source: IBM Research
Search: Chess

- Deep Blue’s won 3 games, lost 2, tied 1

- 30 CPUs + 480 chess processors
- Searched 126.000.000 nodes per sec
- Generated 30 billion positions per move reaching depth 14 routinely
# Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
<td></td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
<td></td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
<td></td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
<td></td>
</tr>
</tbody>
</table>
CSP: Sudoku

Sudoku rules are extremely easy: Fill all empty squares so that the numbers 1 to 9 appear once in each row, column and 3x3 box.
Constraint optimization

• Optimization under side constraints (similar to CSP)
• E.g. mixed integer programming (software: IBM CPLEX)
  - Linear program: max. linear objective subject to linear constraints
  - Mixed integer program: additional constraint: some variables integer
  - NP-hard, widely used in operations research and in industry

Transportation/Logistics:
SNCF, United Airlines
UPS, United States Postal Service, ...

Supply chain management software:
Oracle, SAP, ...

Production planning and optimization:
Airbus, Dell, Porsche, Thyssen Krupp, Toyota, Nissan, ...
# Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
</tr>
</tbody>
</table>

- **Deterministic vs. Stochastic**: Deterministic vs. Stochastic
- **Static vs. Sequential**: Static vs. Sequential
- **States vs. Features vs. Relations**: States vs. Features vs. Relations
Planning: Spacecraft Control

NASA: Deep Space One spacecraft

- operated autonomously for two days in May, 1999:
  - determined its precise position using stars and asteroids
    - despite a malfunctioning ultraviolet detector
  - planned the necessary course adjustment
  - fired the ion propulsion system make this adjustment

Source: NASA
## Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
<td></td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
<td></td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
<td></td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>6. Decision</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>Theory</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Logic: Cyc

• AI project that started 1984 with the objective
  - to codify, in machine-usable form, millions of pieces of knowledge that comprise human common sense

• Logic reasoning procedures, e.g.
  - Every tree is a plant
  - Plants die eventually
  - Therefore, every tree dies eventually

• Criticisms include
  - Difficulty of adding knowledge manually
  - Non-scalability
  - Empirical evaluation - no benchmarks
Logic: Cyc

Cyc contains:
- 10,000 Predicates
- 100,000 Concepts
- 1,400,000 Assertions

Represented in:
- First Order Logic
- Higher Order Logic
- Context Logic
- Micro-theories
CSP/logic: formal verification

Hardware verification  
(e.g., IBM)

Software verification  
(small to medium programs)

Most progress in the last 10 years based on:  
Encodings into propositional satisfiability (SAT)
# Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
<td></td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
<td></td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
<td></td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
<td></td>
</tr>
</tbody>
</table>
Reasoning Under Uncertainty

E.g. motion tracking: track a hand and estimate activity:
- drawing, erasing/shading, other

Source: Kevin Murphy, UBC
Reasoning under Uncertainty

Sample application: Microsoft Kinect
- Sensors: 3 cameras for depth perception
- Noise: no fixed reference points; movements in the background

Source: Microsoft & youtube
# Course Map

<table>
<thead>
<tr>
<th>Course Modules</th>
<th>Dimensions</th>
<th>Deterministic vs. Stochastic</th>
<th>Static vs. Sequential</th>
<th>States vs. Features vs. Relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Search</td>
<td>Deterministic</td>
<td>Static</td>
<td>States</td>
<td></td>
</tr>
<tr>
<td>2. CSPs</td>
<td>Deterministic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>3. Planning</td>
<td>Deterministic</td>
<td>Sequential</td>
<td>States or Features</td>
<td></td>
</tr>
<tr>
<td>4. Logic</td>
<td>Deterministic</td>
<td>Static</td>
<td>Relations</td>
<td></td>
</tr>
<tr>
<td>5. Uncertainty</td>
<td>Stochastic</td>
<td>Static</td>
<td>Features</td>
<td></td>
</tr>
<tr>
<td>6. Decision Theory</td>
<td>Stochastic</td>
<td>Sequential</td>
<td>Features</td>
<td></td>
</tr>
</tbody>
</table>
Decision Theory: Decision Support Systems

E.g., Computational Sustainability

- New interdisciplinary field, **AI is a key component**
  - Models and methods for **decision making** concerning the management and allocation of resources
  - to solve most challenging problems related to **sustainability**

- Often **constraint optimization problems**. E.g.
  - **Energy**: when are where to produce green energy most economically?
  - Which parcels of land to purchase to **protect endangered species**?
  - **Urban planning**: how to use budget for best development in 30 years?

Source: http://www.computational-sustainability.org/
Planning Under Uncertainty

Helicopter control: MDP, reinforcement learning

Source: Andrew Ng
Planning Under Uncertainty

Autonomous driving: DARPA Grand Challenge

Dr. Sebastian Thrun
Stanford Racing Team Leader & Director
Stanford Artificial Intelligence Lab

Source: Sebastian Thrun
Military applications: ethical issues

- Robot soldiers
  - Existing: robot dog carrying heavy materials for soldiers in the field
  - The technology is there
- Unmanned airplanes
- Missile tracking
- Surveillance
- ...
Multiagent Systems: Robot Soccer

Source: RoboCup web site
Robot Soccer: Penalty Shooting

Source: Darmstadt Dribbling Dackels
Robot Soccer: Goal of the Month

Source: Darmstadt Dribbling Dackels
Summary(1)

We would like most general agents possible, but to start with we need to restrict scope:

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)

• Huge diversity of applications
• More than I could possibly show here
• We will focus on their common foundations
Coming up …

• For Wednesday: Assignment 0
  - Available on WebCT
  - Section 1.5 & 1.6 in the textbook will be particularly helpful

• We’ll start the search module: read Sections 3.0-3.4
• Please continue bringing coloured cards (didn’t work today)
Reasoning Under Uncertainty

- Texture classification using SVMs
  - foliage, building, sky, water