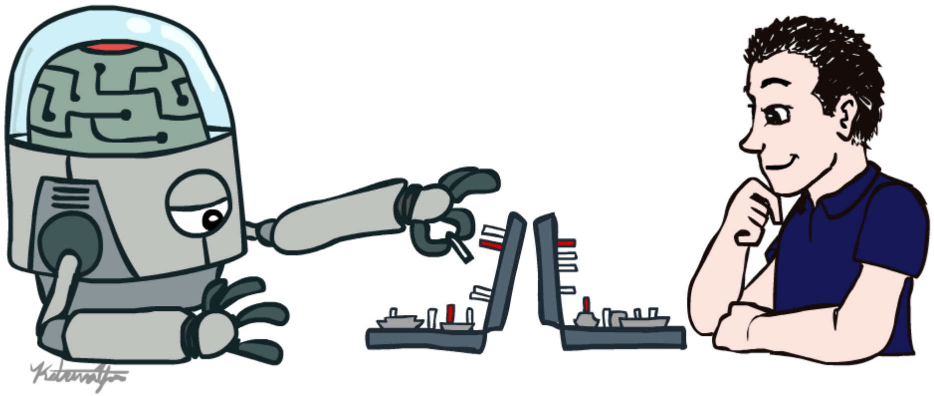
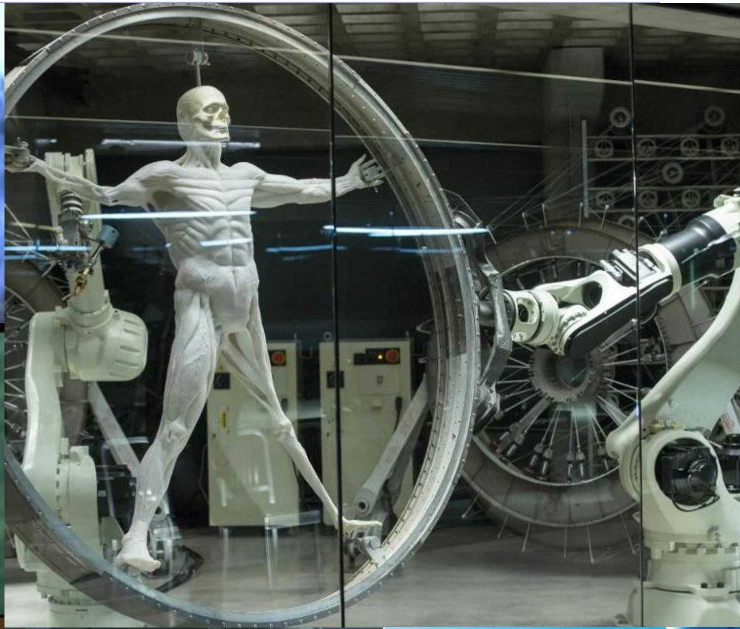
Artificial Intelligence

Introduction



Sci-Fi AI?




WAYMO

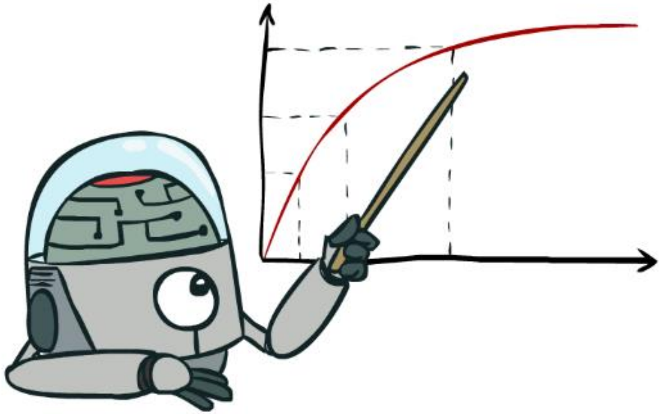

TUG
CAUTION
MAY CONTAIN
CHEMOTHERAPY DRUG
CAUTION
MAY CONTAIN
CHEMOTHERAPY DRUG
TUG

Rational Decisions

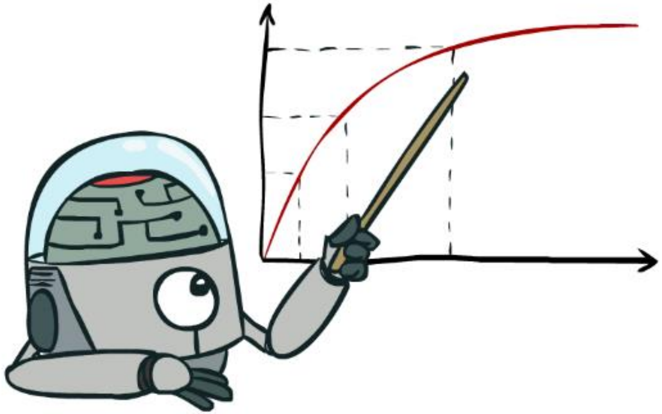
We'll use the term **rational** in a very specific, technical way:

- Rational: maximally achieving pre-defined goals
- Rationality only concerns what decisions are made
  (not the thought process behind them)
- Goals are expressed in terms of the **utility** of outcomes
- Being rational means **maximizing your expected utility**

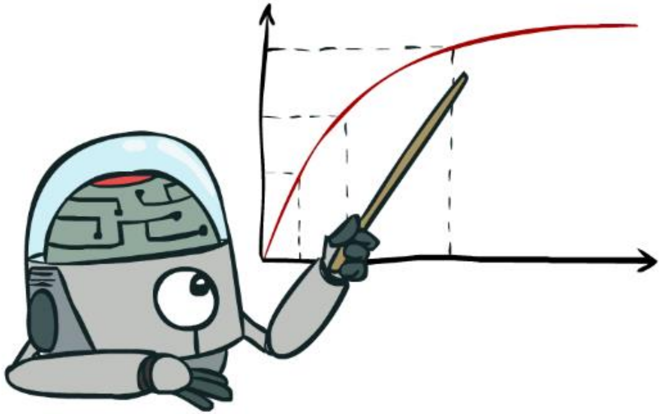
Maximize Your Expected Utility



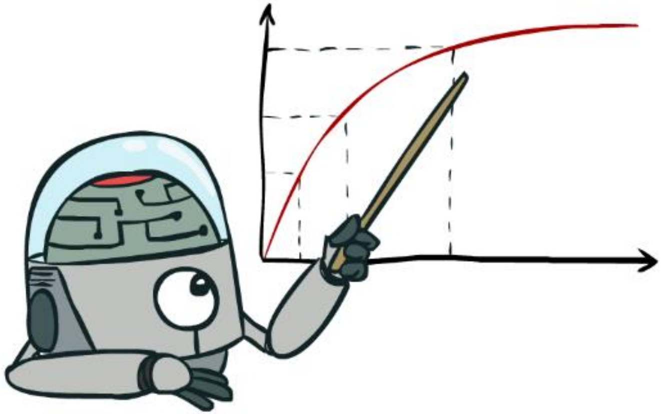
Maximize Your Expected Utility



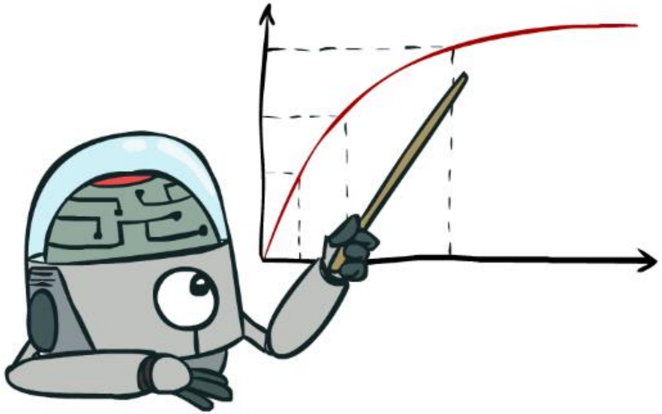
Maximize Your Expected Utility



Maximize Your Expected Utility



Maximize Your Expected Utility

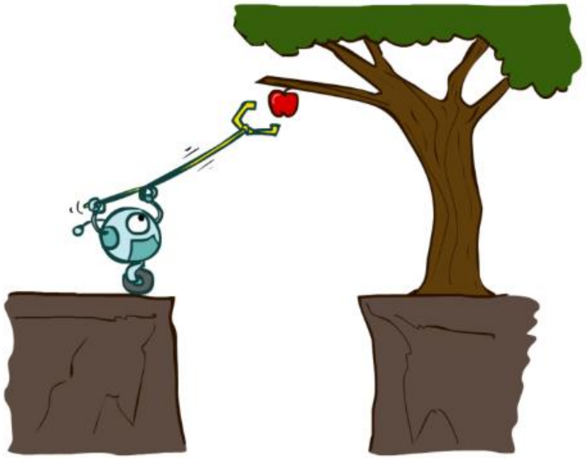


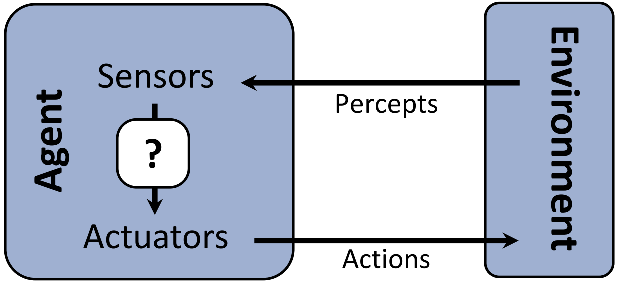
What About the Brain?

- Brains (human minds) are very good at making rational decisions, but not perfect
- Brains aren’t as modular as software, so hard to reverse engineer!
- “Brains are to intelligence as wings are to flight”
- Lessons learned from the brain: memory and simulation are key to decision making

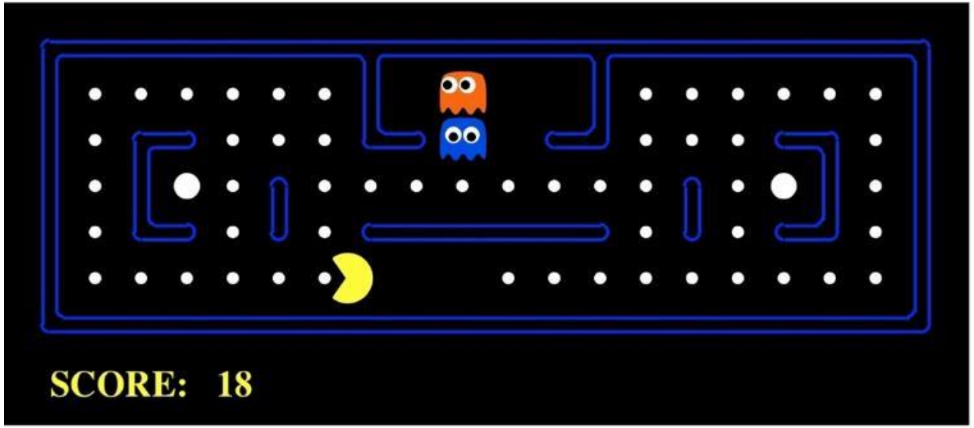
Designing Rational Agents

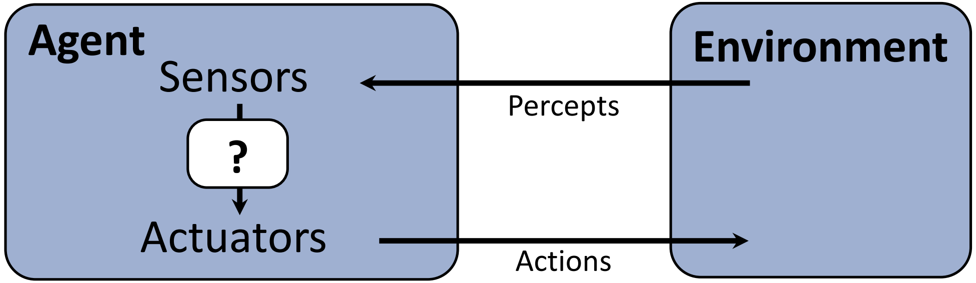
- An **agent** is an entity that *perceives* and *acts*.
- A **rational agent** selects actions that maximize its (expected) **utility**.
- Characteristics of the **percepts, environment,** and **action space** dictate techniques for selecting rational actions
- **This course** is about:
  - General AI techniques for a variety of problem types
  - Learning to recognize when and how a new problem can be solved with an existing technique





Pac-Man as an Agent

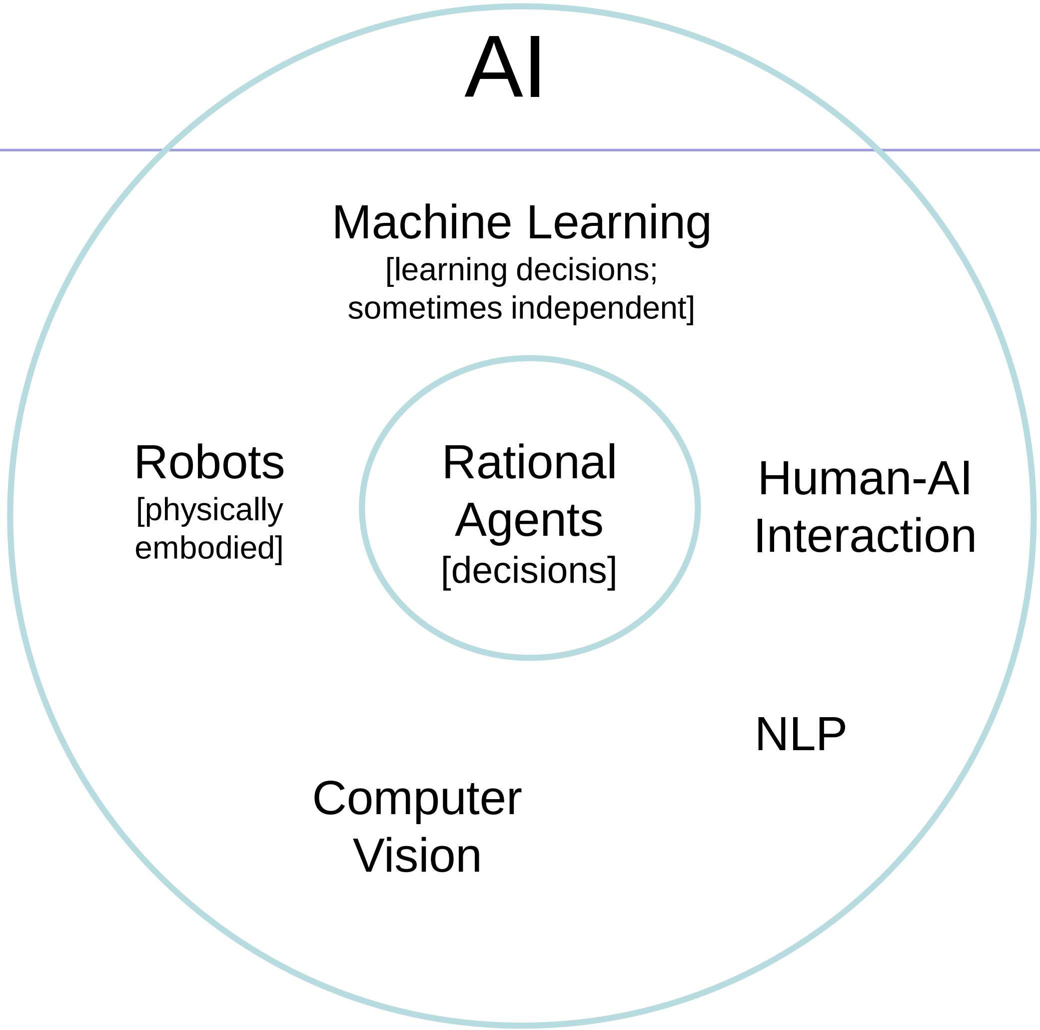




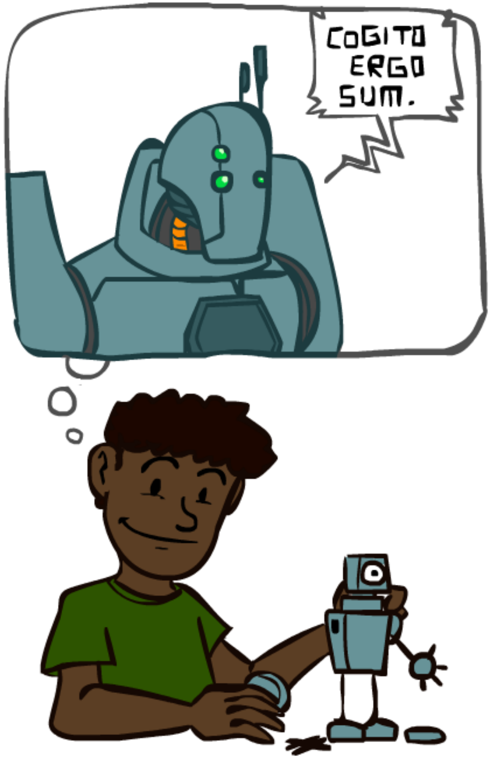
Pac-Man is a registered trademark of Namco-Bandai Games, used here for educational purposes

Demo1: pacman-l1.mp4

AI

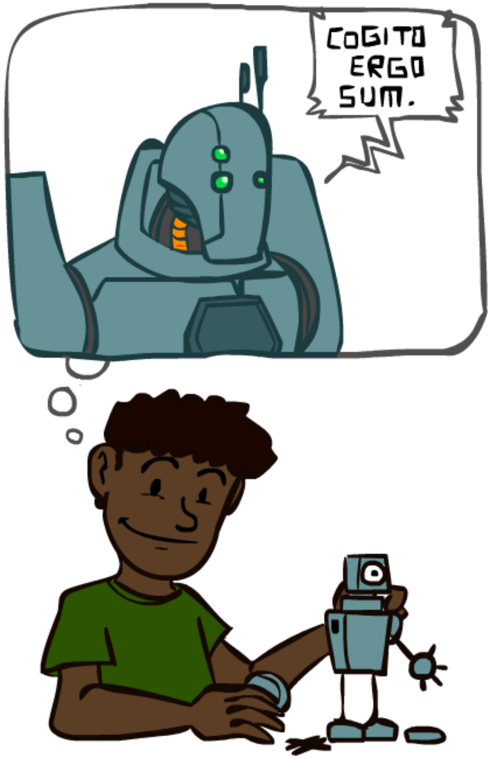

Machine Learning
[learning decisions;
sometimes independent]
Robots
[physically
embodied]
Rational
Agents
[decisions]
Human-AI
Interaction
NLP
Computer
Vision

A (Short) History of AI



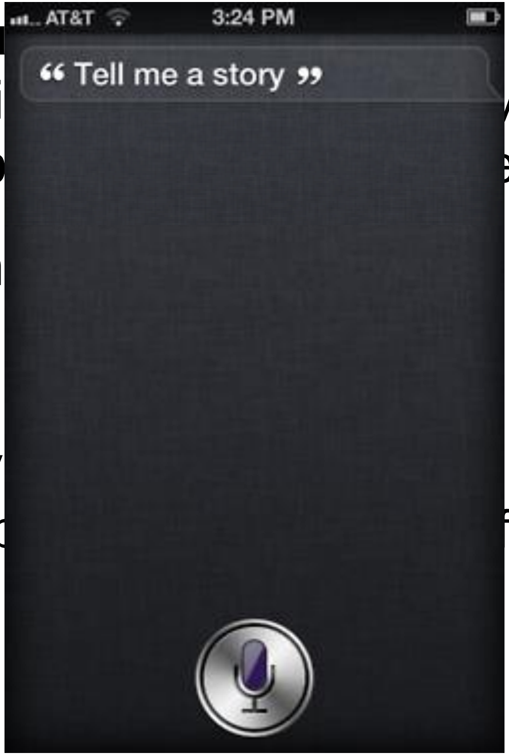
A (Short) History of AI

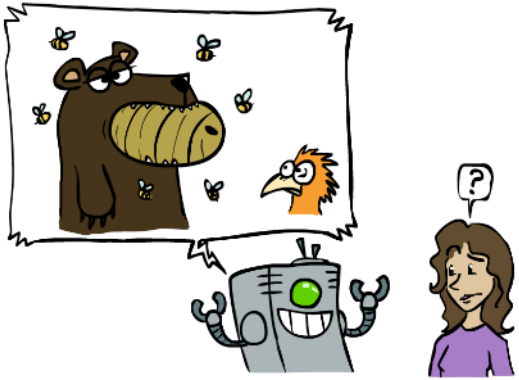
- 1940-1950: Early days
  - 1943: McCulloch & Pitts: Boolean circuit model of brain
  - 1950: Turing's “Computing Machinery and Intelligence”
- 1950—70: Excitement: Look, Ma, no hands!
  - 1950s: Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
  - 1956: Dartmouth meeting: “Artificial Intelligence” adopted
  - 1965: Robinson's complete algorithm for logical reasoning
- 1970—90: Knowledge-based approaches
  - 1969—79: Early development of knowledge-based systems
  - 1980—88: Expert systems industry booms
  - 1988—93: Expert systems industry busts: “AI Winter”
- 1990—: Statistical approaches
  - Resurgence of probability, focus on uncertainty
  - General increase in technical depth
  - Agents and learning systems… “AI Spring”?
- 2000—: Where are we now?



Unintentionally Funny Stories

- One day Joe Bear was hu
He asked his friend Irving Bi … was.
Irving told him there was a b … e.
Joe walked to the oak tree.
He ate the beehive. The En



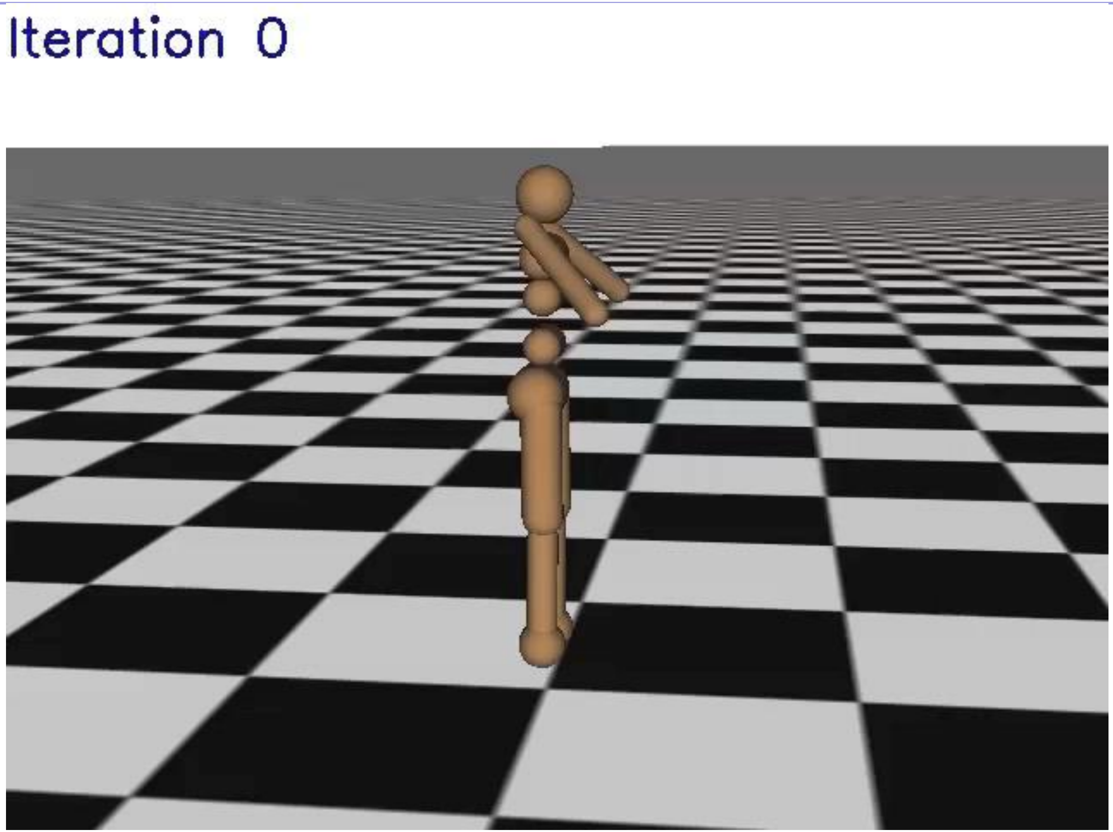


- Henry Squirrel was thirsty
He walked over to the river b … friend Bill Bird was sitting.
Henry slipped and fell in the
Gravity drowned. The End.

[Shank, Tale-Spin System, 1984]

Photo: Google / Getty Images

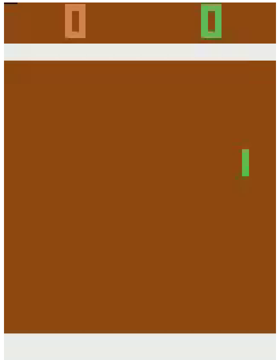
Simulated Agents



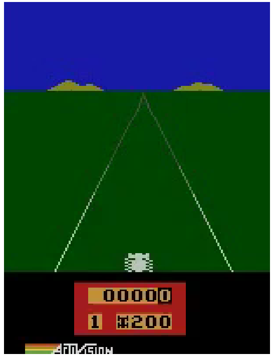
[Schulman, Moritz, Levine, Jordan, Abbeel, ICLR 2016]

Game Agents

- Reinforcement learning

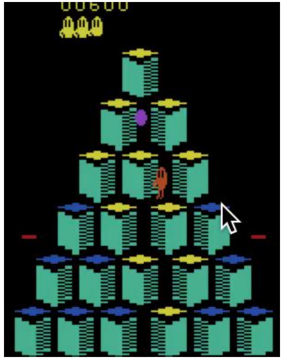


Pong



Enduro

Beamrider

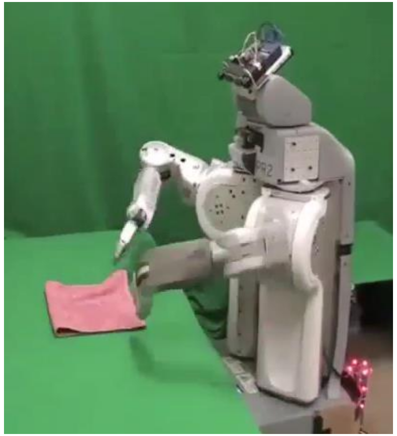


Q*bert

Robotics

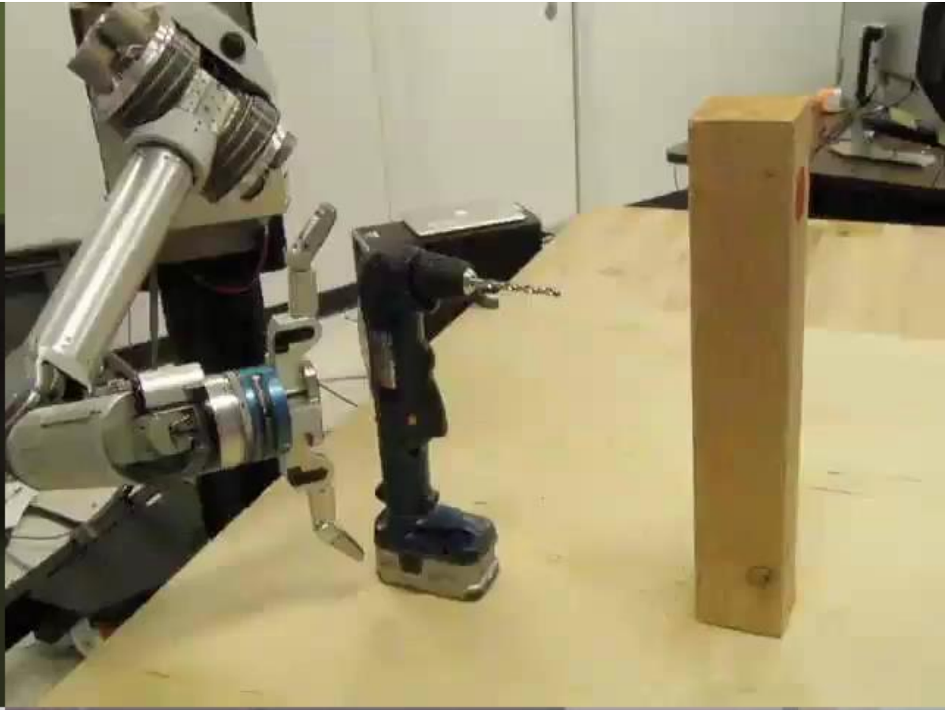
Demo 1: ROBOTICS – soccer.avi
Demo 2: ROBOTICS – soccer2.avi
Demo 3: ROBOTICS – gcar.avi
Demo 4: ROBOTICS – laundry.avi
Demo 5: ROBOTICS – petman.avi

- Robotics
  - Part mech. eng.
  - Part AI
  - Reality much harder than simulations!
- Technologies
  - Vehicles
  - Rescue
  - Help in the home
  - Lots of automation…
- In this class:
  - We ignore mechanical aspects
  - Methods for planning
  - Methods for control

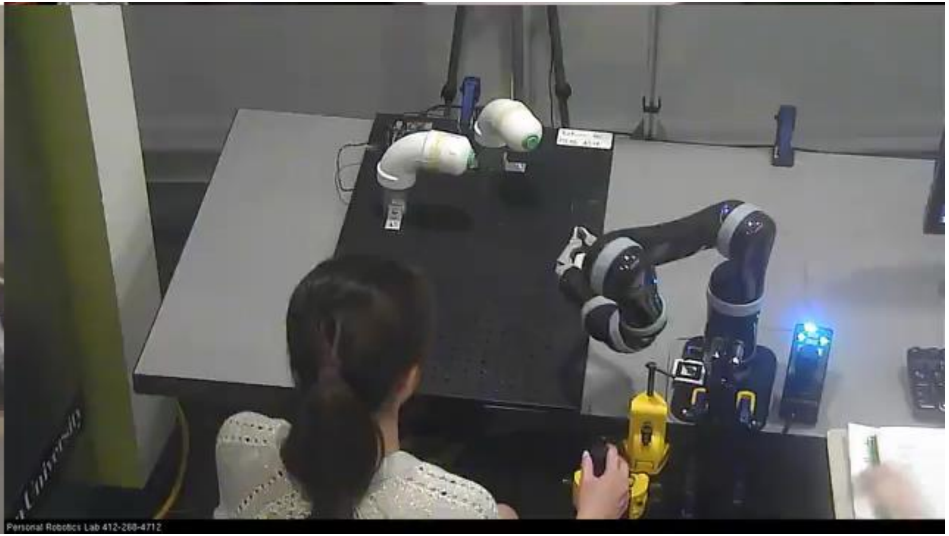


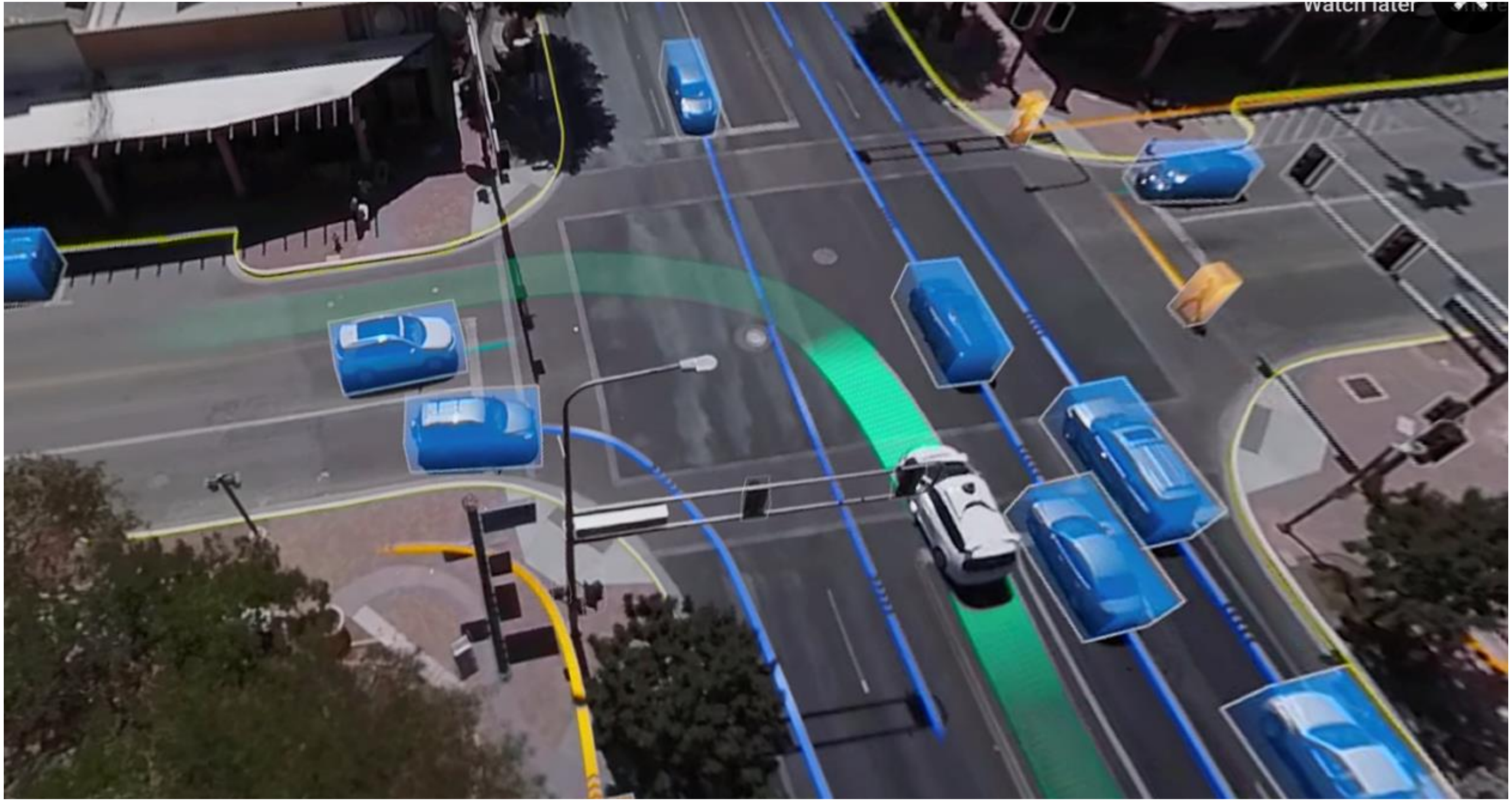
Images from UC Berkeley, Boston Dynamics, RoboCup, Google

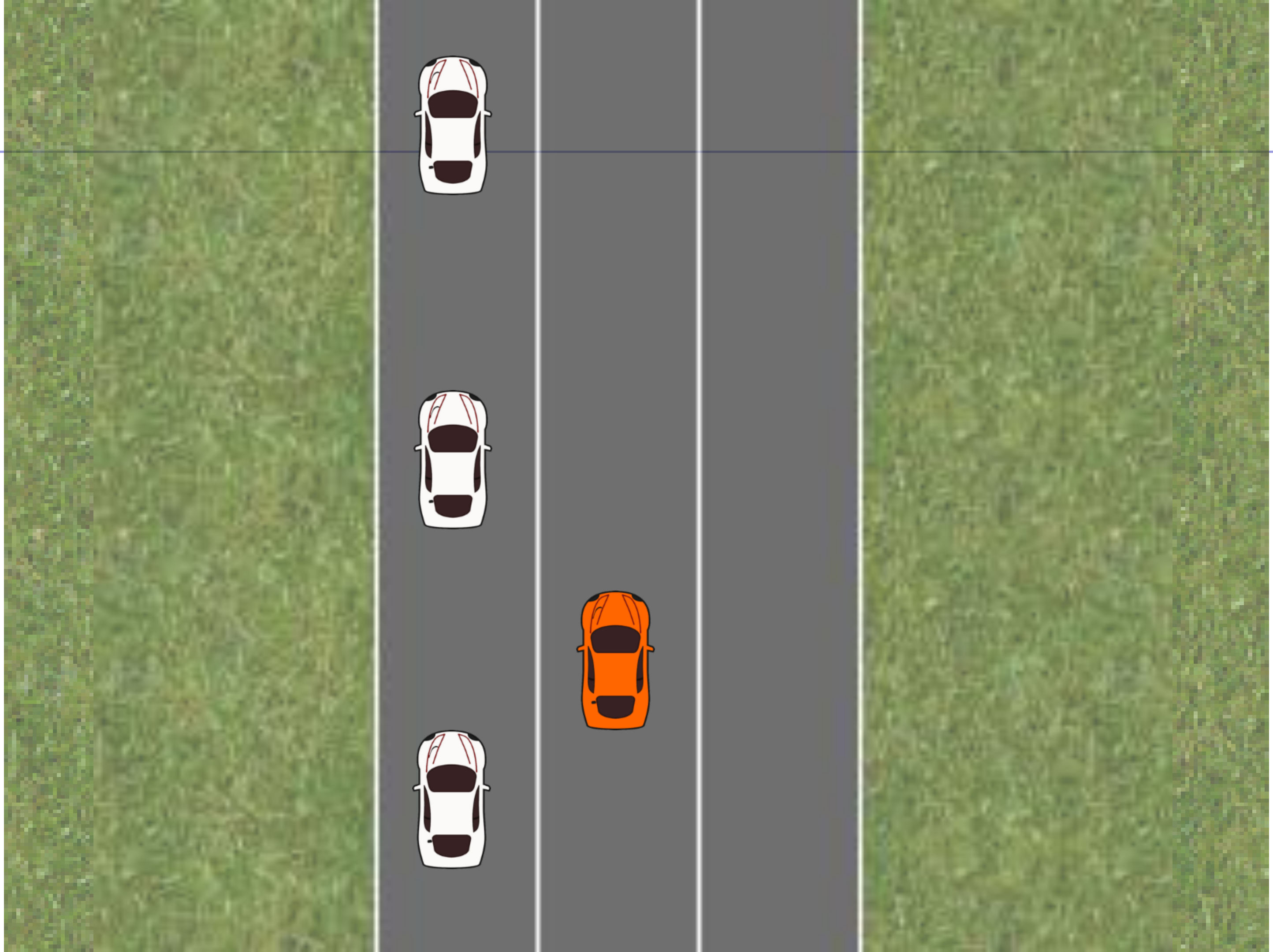
Robots

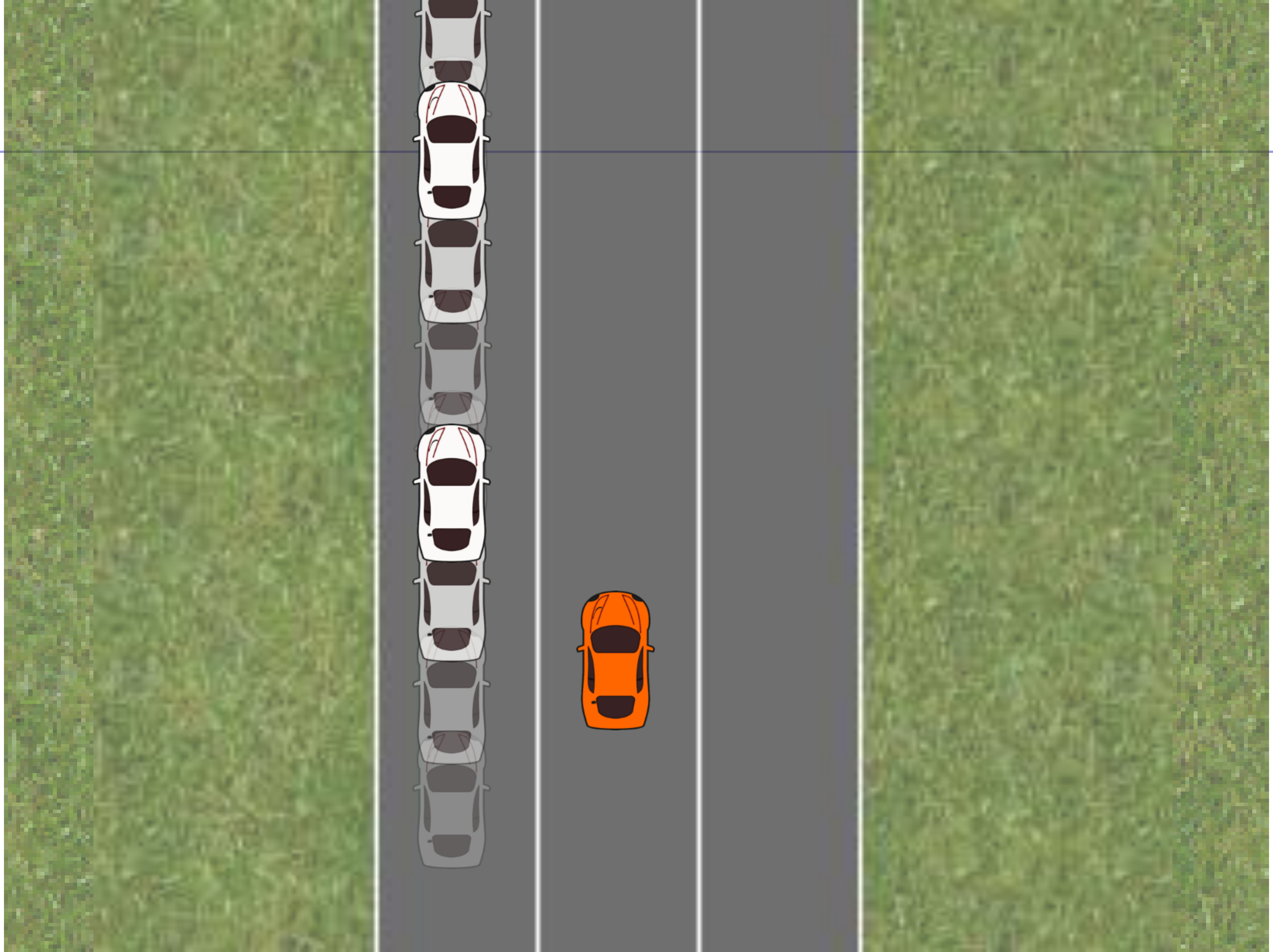


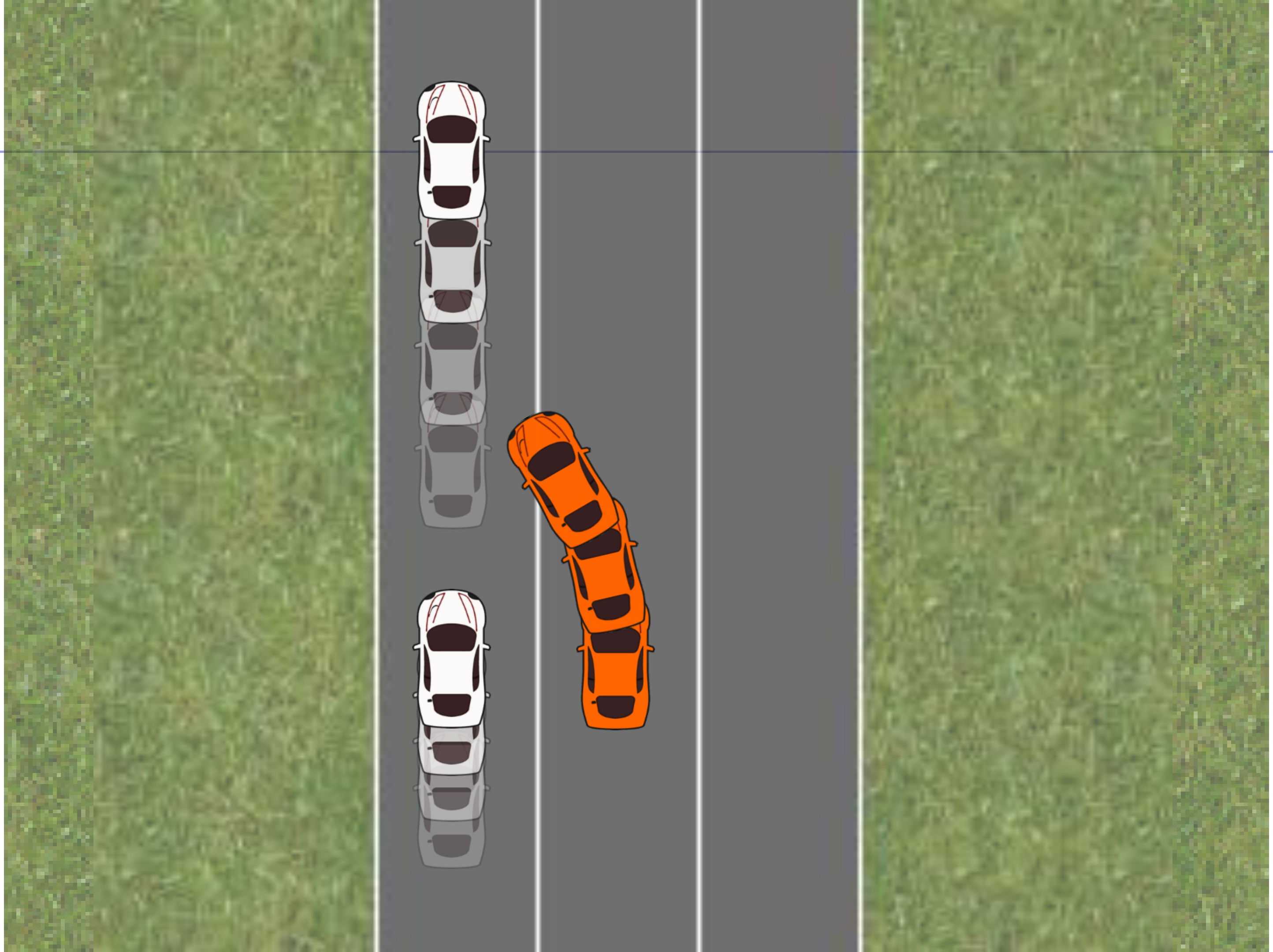
Human-AI Interaction



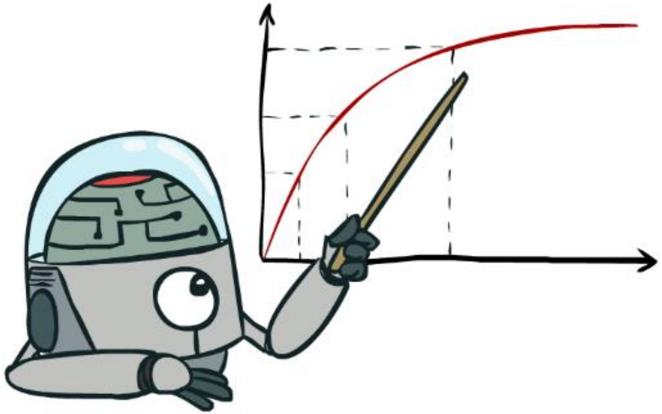

Watch later







Maximize Your Expected Utility



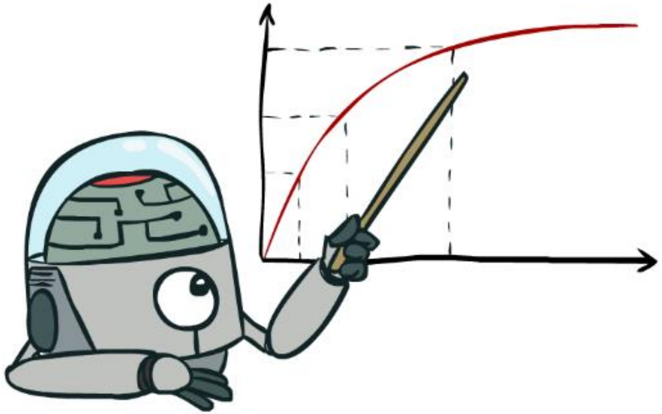
Utility?

Clear utility function

Not so clear utility function

START
START
SCORE
0
LAPS
-/3
TIME
0:01
TURBO
MORE GAMES

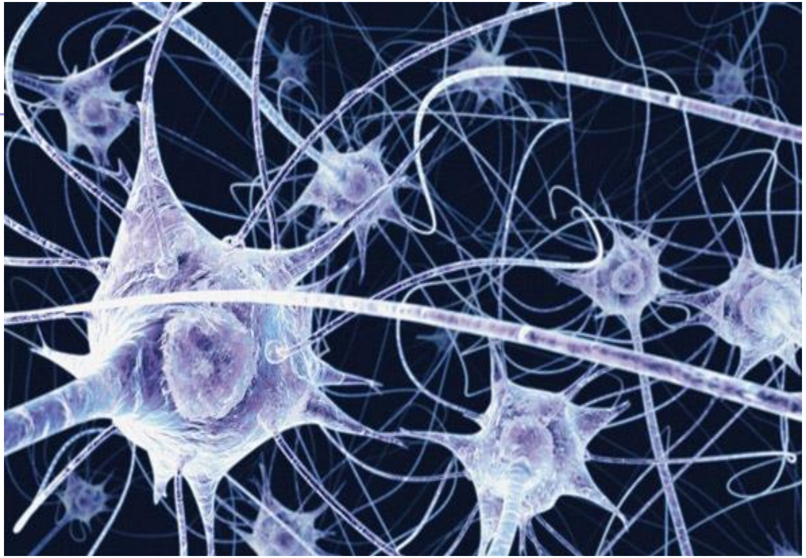
Maximize Your Expected Utility



What is Intelligence???

❖Intelligence is the ability to learn ,

Understand & think.



What Is Artificial Intelligence???

- ❖ Artificial Intelligence (AI) is a branch of computer science concerned with the study & creation of computer systems that exhibit some form of intelligence.

- ❖ A.I deals with the:

1) Systems that learn new concepts & tasks .

2) Systems that can reason & draw conclusions about the world around us.

Human Intelligence VS Artificial Intelligence

Human Intelligence VS Artificial Intelligence

Pros

Human Intelligence

- Intuition, Common sense, Judgement, Creativity, Beliefs etc
- The ability to demonstrate their intelligence by communicating effectively
- Plausible Reasoning and Critical thinking

Artificial Intelligence

- Ability to simulate human behavior and cognitive processes
- Capture and preserve human expertise
- Fast Response. The ability to comprehend large amounts of data quickly.

Human Intelligence VS Artificial Intelligence

Cons

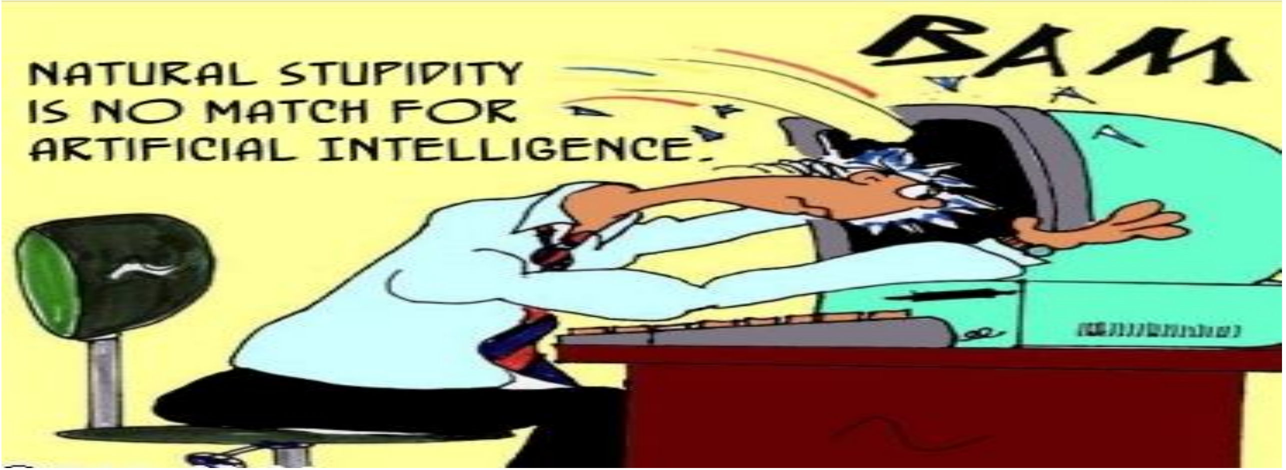
Human Intelligence

- Humans are fallible
- They have limited knowledge bases
- Information processing of serial nature proceed very slowly in the brain

Artificial Intelligence

- No “common sense”
- Cannot readily deal with “mixed” knowledge
- May have high development costs
- Raise legal and ethical concerns

Human Intelligence VS Artificial Intelligence



We achieve more than we know. We know more than we understand. We understand more than we can explain (Claude Bernard, 19th C French scientific philosopher)

Unit 1

1. Introduction
2. Foundations of AI
3. Criteria for Success :Turing Test
4. AI Problems
5. AI Task Domain
6. Underlying Assumption
7. What is an AI Technique?

1. Introduction

- Intelligence: “ability to learn, understand and think” (Oxford dictionary)
- AI is the study of how to make computers make things which at the moment people do better.
- Examples: Speech recognition, Face, Object, Intuition, Inferencing, Learning new skills, Decision making, Abstract thinking

- AI is a branch of computer science concerned with the **study and creation of computer systems that exhibit some form of intelligence**:

Systems that learn new concepts and tasks

Systems that can reason and draw conclusions about the world around us

Systems that can understand natural language or perceive and comprehend a visual scene

& systems that perform other types of feats that require human types of intelligence

The State of the Art

- Computer beats human in a chess game.
- Computer-human conversation using speech recognition.
- Expert system controls a spacecraft.
- Robot can walk on stairs and hold a cup of water.
- Home appliances use fuzzy logic.
- ......

The Foundations of AI

- Philosophy (423 BC – present):
  - Logic, methods of reasoning.
  - Mind as a physical system.
  - Foundations of learning, language, and rationality.

- Mathematics (c.800 – present):
  - Formal representation and proof.
  - Algorithms, computation, decidability, tractability.
  - Probability.

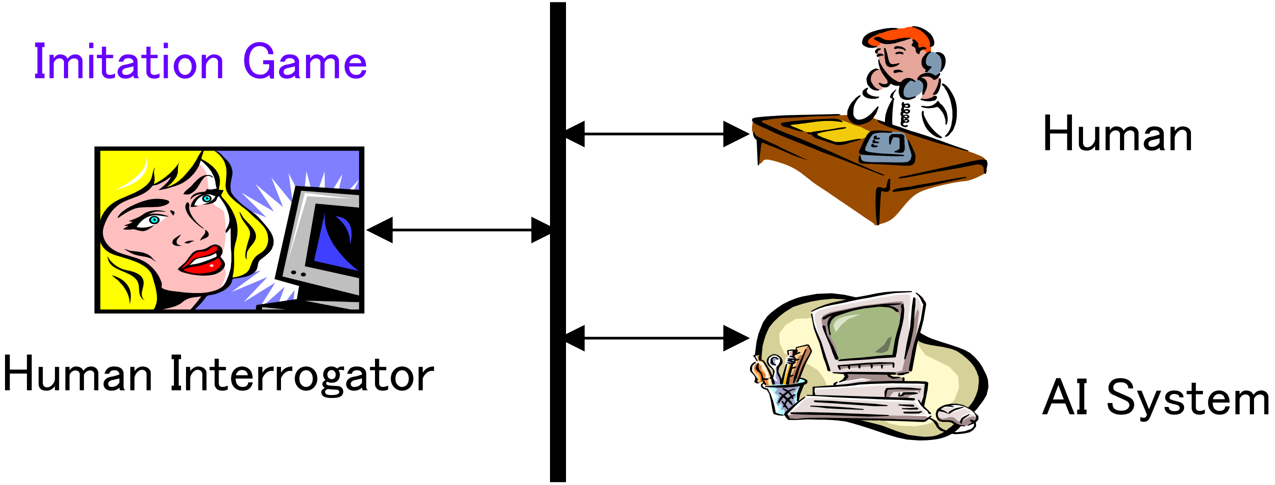
The Foundations of AI

- Psychology (1879 – present):
  - Adaptation.
  - Phenomena of perception and motor control.
  - Experimental techniques.

- Linguistics (1957 – present):
  - Knowledge representation.
  - Grammar.

Criteria for Success: The Turing Test

- Alan Turing (1912-1954)
- "Computing Machinery and Intelligence" (1950)



To pass Turing Test computer needs..

- Natural language processing
- Knowledge Representation
- Automated Reasoning
- Machine learning
- Computer Vision
- Robotics

AI Problems

1) Common sense knowledge problem

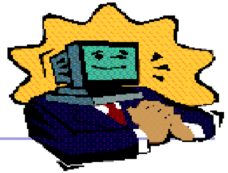
2) Making Simple Problems complex

3) Lack of Intuition

AI Task Domains

- Mundane Tasks:
  - Perception
    - Vision
    - Speech
  - Natural Languages
    - Understanding
    - Generation
    - Translation
  - Common sense reasoning
  - Robot Control
- Formal Tasks
  - Games : chess, checkers etc
  - Mathematics: Geometry, logic,Proving properties of programs
- Expert Tasks:
  - Engineering (Design, Fault finding, Manufacturing planning)
  - Scientific Analysis
  - Medical Diagnosis

Applications Of Artificial Intelligence



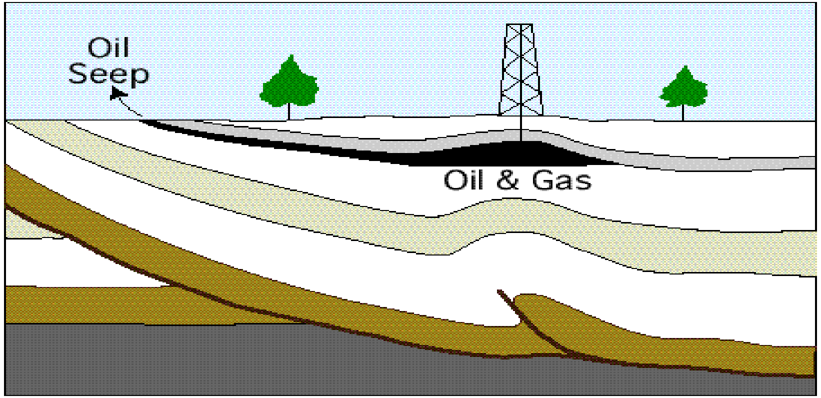
Expert Systems!!

- An expert system is a computer program that is designed to hold the accumulated knowledge of one or more domain experts

- It reasons with knowledge of some specialist subject with a view to solving problems or giving advice

- They are tested by being placed in the same real world problem solving situation

Applications of Expert Systems

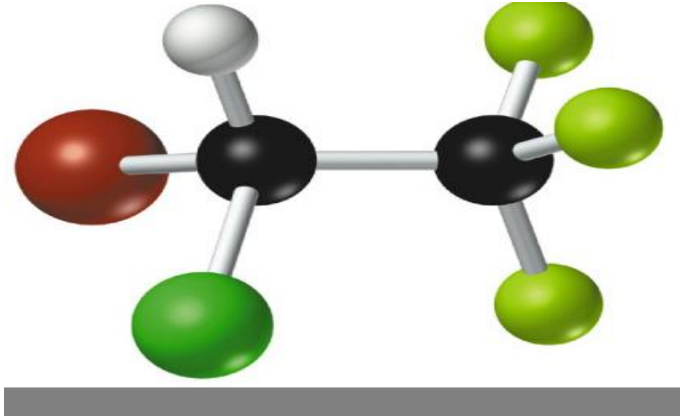


PUFF:
Medical system
for diagnosis of respiratory conditions

PROSPECTOR:
Used by geologists to identify sites for drilling or mining

Applications of Expert Systems

DENDRAL: Used to identify the structure of chemical compounds. First used in 1965



LITHIAN: Gives advice to archaeologists examining stone tools

Tic Tac Toe

- Three programs are presented :
  - Series increase
  - Their complexity
  - Use of generalization
  - Clarity of their knowledge
  - Extensability of their approach

Introductory Problem: Tic-Tac-Toe

X		X
	o	

Introductory Problem: Tic-Tac-Toe

Program 1:

Data Structures:

- Board: 9 element vector representing the board, with 1-9 for each square. An element contains the value 0 if it is blank, 1 if it is filled by X, or 2 if it is filled with a O
- Movetable: A large vector of 19,683 elements (3^9), each element is 9-element vector.

Algorithm:

1. View the vector as a ternary number. Convert it to a decimal number.
2. Use the computed number as an index into Move-Table and access the vector stored there.
3. Set the new board to that vector.

Introductory Problem: Tic-Tac-Toe

Comments:

This program is very efficient in time.

1. A lot of space to store the Move-Table.

2. A lot of work to specify all the entries in the Move-Table.

3. Difficult to extend.

Introductory Problem: Tic-Tac-Toe

1	2	3
4	5	6
7	8	9

Introductory Problem: Tic-Tac-Toe

Program 2:

Data Structure: A nine element vector representing the board. But instead of using 0,1 and 2 in each element, we store 2 for blank, 3 for X and 5 for O

Functions:

Make2: returns 5 if the center sqaure is blank. Else any other balnk sq

Posswin(p): Returns 0 if the player p cannot win on his next move; otherwise it returns the number of the square that constitutes a winning move. If the product is 18 (3x3x2), then X can win. If the product is 50 (5x5x2) then O can win.

Go(n): Makes a move in the square n

Strategy:

Turn = 1 Go(1)

Turn = 2 If Board[5] is blank, Go(5), else Go(1)

Turn = 3 If Board[9] is blank, Go(9), else Go(3)

Introductory Problem: Tic-Tac-Toe

Comments:

1. Not efficient in time, as it has to check several conditions before making each move.
2. Easier to understand the program's strategy.
3. Hard to generalize.

Introductory Problem: Tic-Tac-Toe

8	3	4
1	5	9
6	7	2

$$15 - (8 + 5)$$

Introductory Problem: Tic-Tac-Toe

Comments:

1. Checking for a possible win is quicker.

2. Human finds the row-scan approach easier, while computer finds the number-counting approach more efficient.

Introductory Problem: Tic-Tac-Toe

Program 3:

1. If it is a win, give it the highest rating.

2. Otherwise, consider all the moves the opponent could make next. Assume the opponent will make the move that is worst for us. Assign the rating of that move to the current node.

3. The best node is then the one with the highest rating.

Introductory Problem: Tic-Tac-Toe

Comments:

1. Require much more time to consider all possible moves.

2. Could be extended to handle more complicated games.