

An Introduction to Reinforcement Learning

Instructor: Shuguang Cui
shuguangcui@cuhk.edu.cn

(Based on Prof. Sutton's book material and notes)

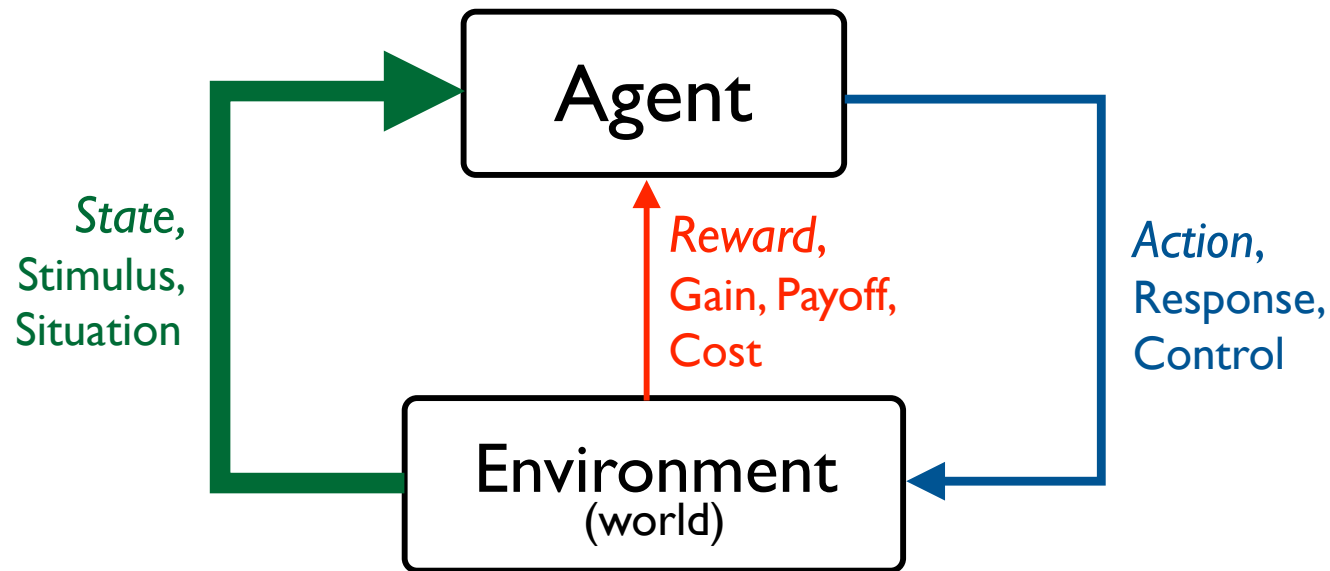
Machine Learning

- Supervised Learning
 - Learning from a training set with labels
 - Labels specify the correct vs. wrong action
 - Not able to learn in uncharted territory
 - Actions do not directly affect the environment
- Unsupervised Learning
 - Without labels, learning the hidden structure of the data
 - Not clear goals and correct vs. wrong actions defined
 - Actions do not directly affect the environment
- **Reinforcement Learning**

What is Reinforcement Learning?

- Agent vs. Environment
- Learning by interacting with an environment to achieve a goal
 - more **realistic** and **ambitious** than other kinds of machine learning
 - actions affect the environment
- Learning in uncharted territory: by trial and error **experience**, with only delayed evaluative feedback (reward)
 - the kind of machine learning most like natural learning
 - learning that can tell for itself when it is right or wrong

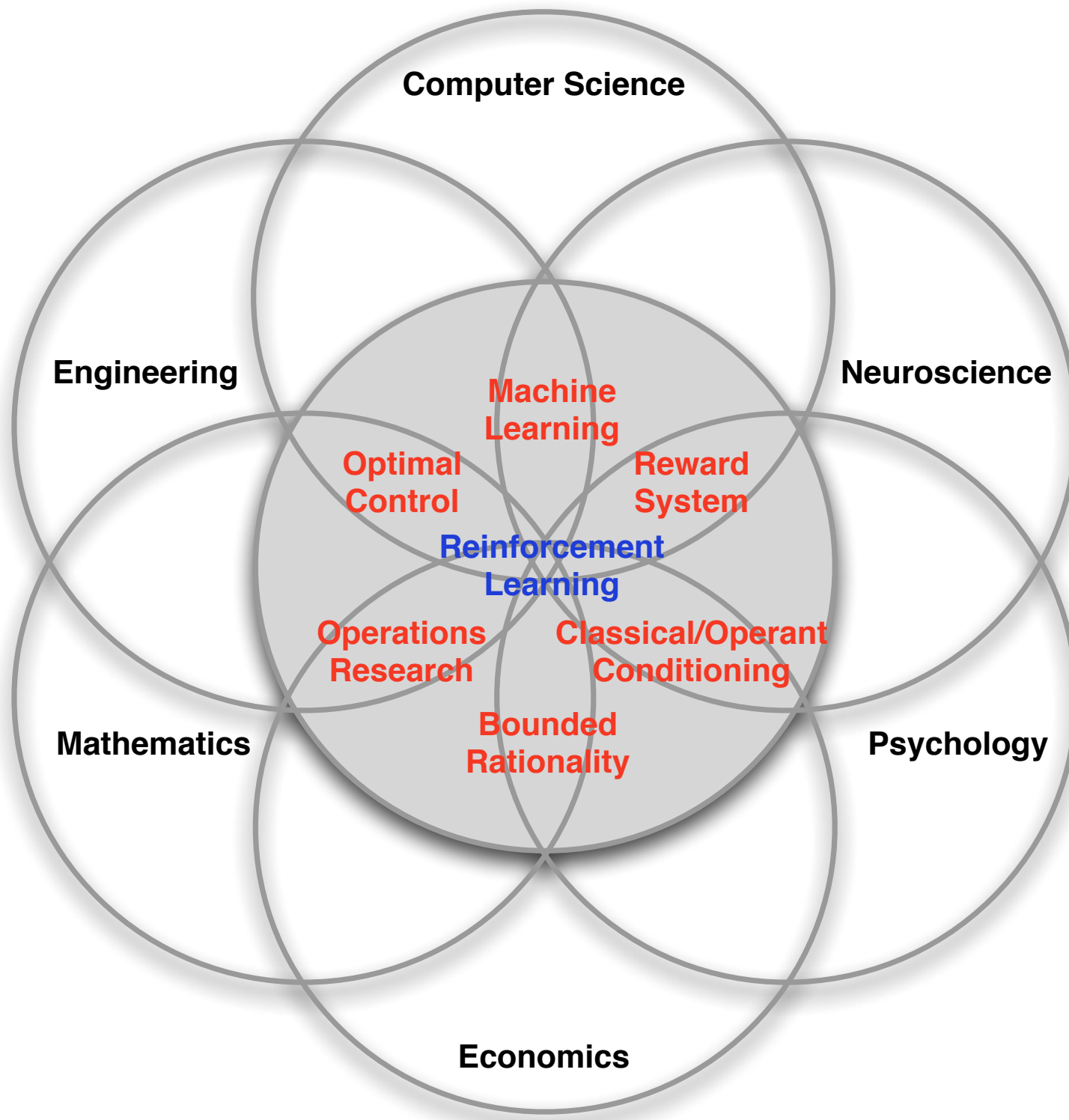
The RL Interface



- Environment may be unknown, nonlinear, stochastic and complex
- Agent learns a policy mapping states to actions
 - Seeking to maximize its cumulative reward in the long run

Signature features/challenges of RL

- Evaluative feedback (reward) to construct the goal to maximize
- Operate over sequentiality, delayed consequences
- Learn from trial and error, to **explore** as well as **exploit**
- Work with non-stationarity
- Consider the whole problem as a close-loop system control problem
- Naturally interacts with other disciplines




Technical elements of RL

- **A Policy**
 - Defining agent's way of behaving at each time t
 - State \rightarrow action mapping
- **A Reward**
 - Defining the goal and what is good at this moment; cannot be manipulated by the agent
 - Stochastic function of the state and action
- **A Value Function**
 - Defining what is good in the long run
 - We seek actions to reach states of highest values
- **A Model of the environment (optional)**
 - Mimics the behavior of the environment
 - Used for planning

Our main approaches

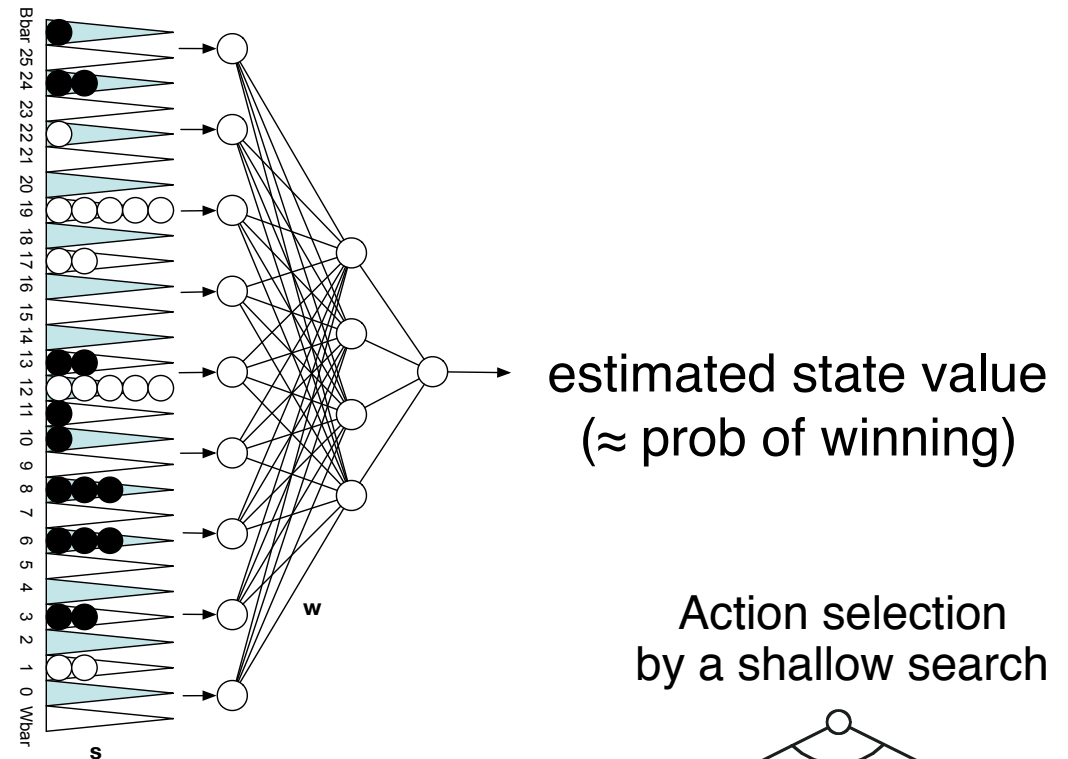
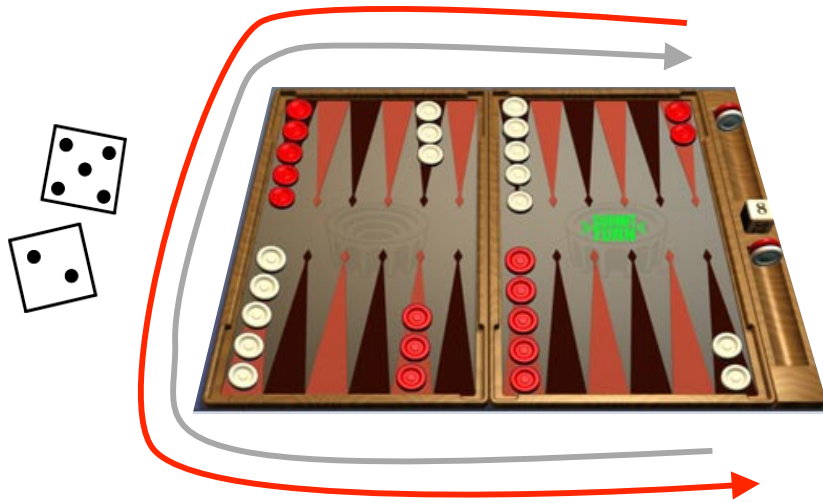
- Mainly value functions based
- Or policy gradient methods
- Cf. Evolutionary methods
 - These methods evaluate the “lifetime” behavior of many non-learning agents, each using a different policy for interacting with its environment, and select those that are able to obtain the most reward (directly search in the policy space)
 - Like the genetic algorithms (in biological evolution)
 - Do not learn during the individual lifetime (may be useful when the agent cannot accurately sense the state of environment)
 - We try to maximize the reward (as the goal); but most of times we live with suboptimal solutions

Some RL Successes

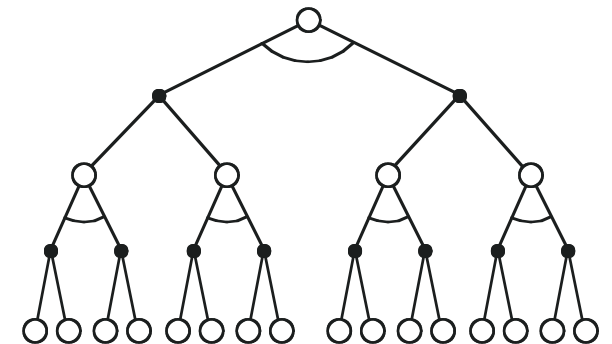
- Learned the world's best player of Backgammon (Tesauro 1995) 
- Learned acrobatic helicopter autopilots (Ng, Abbeel, Coates et al 2006+)
- Widely used in the placement and selection of advertisements and pages on the web
- Used to make strategic decisions in *Jeopardy!* (IBM's Watson 2011)
- Achieved human-level performance on Atari games from pixel-level visual input, in conjunction with deep learning (Google Deepmind 2015)
- In all these cases, performance was better than could be obtained by any other method, and was obtained without human instruction

Example: TD-Gammon

Tesauro, 1992-1995



Action selection
by a shallow search



Start with a random Network

Play millions of games against itself

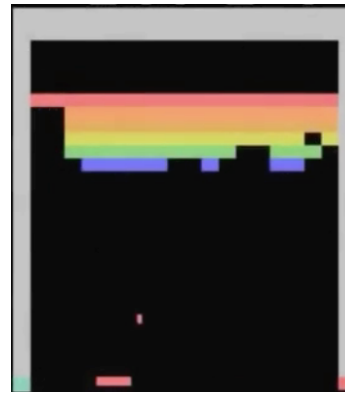
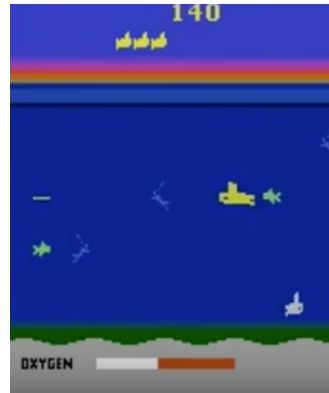
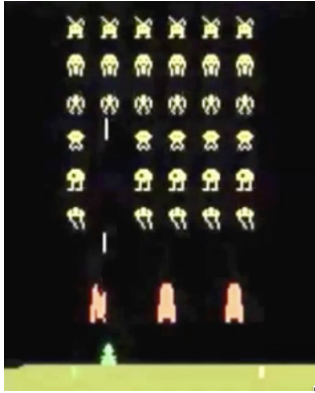
Learn a value function from this simulated experience

Six weeks later it's the best player of backgammon in the world

Originally used expert handcrafted features, later repeated with raw board positions

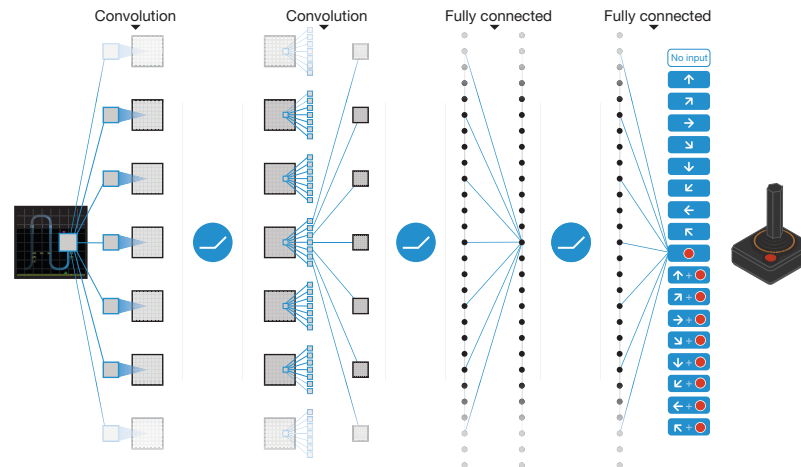
RL + Deep Learning, applied to Classic Atari Games

Google Deepmind 2015, Bowling et al. 2012



- Learned to play 49 games for the Atari 2600 game console, without labels or human input, from self-play and the score alone

mapping raw
screen pixels



to predictions
of final score
for each of 18
joystick actions

- Learned to play better than all previous algorithms and at human level for more than half the games

Same learning
algorithm applied
to all 49 games!
w/o human tuning

The coming of artificial intelligence

- *Intelligence is the ability to achieve goals*
- When people finally come to understand the principles of intelligence—what it is and how it works—**well enough to design and create beings as intelligent as ourselves**
- A fundamental goal for science, engineering, the humanities, ...for all mankind
- It will change the way we work and play, our sense of self, life, and death, the goals we set for ourselves and for our societies
- But it is also of significance beyond our species, beyond history
- It will lead to new beings and new ways of being, things inevitably *much more powerful than our current selves*

Milestones in the development of life on Earth

	year	Milestone	
The Age of Replicators	14Bya	Big bang	
	4.5Bya	formation of the earth and solar system	
	3.7Bya	origin of life on earth (formation of first replicators) DNA and RNA	
	1.1Bya	sexual reproduction multi-cellular organisms nervous systems	
	1Mya	humans culture	Self-replicated things most prominent
	100Kya	language	
	10Kya	agriculture, metal tools	
	5Kya	written language	
	200ya	industrial revolution technology	
	70ya	computers nanotechnology	Designed things most prominent
The Age of Design	?	artificial intelligence super-intelligence	
		...	

AI is a great scientific prize

- cf. the discovery of DNA, the digital code of life, by Watson and Crick (1953)
- cf. Darwin's discovery of evolution, how people are descendants of earlier forms of life (1860)
- cf. the splitting of the atom, by Hahn (1938)
 - leading to both atomic power and atomic bombs

Is human-level AI *possible*?

- If people are biological machines, then eventually we will reverse engineer them, and understand their workings
- Then, surely we can make improvements
 - with new materials and technology now available
 - anything can be eventually improved
 - design can overcome local minima, make great strides, try things much faster than biological evolution

Yes

If AI is possible, then will it *eventually*, inevitably happen?

- No. Not if we destroy ourselves first
- If that doesn't happen, then there will be strong, multi-incremental economic incentives pushing inexorably towards human and super-human AI
- It seems unlikely that they could be resisted
 - or successfully forbidden or controlled
 - there is too much value, too many independent actors

Very probably, say 90%

When will human-level AI first be created?

- No one knows of course; we can make an educated guess about the probability distribution:
 - 25% chance by 2030
 - 50% chance by 2040
 - 10% chance never
- Certainly a significant chance within all of our expected lifetimes
 - We should take the possibility into account in our career plans

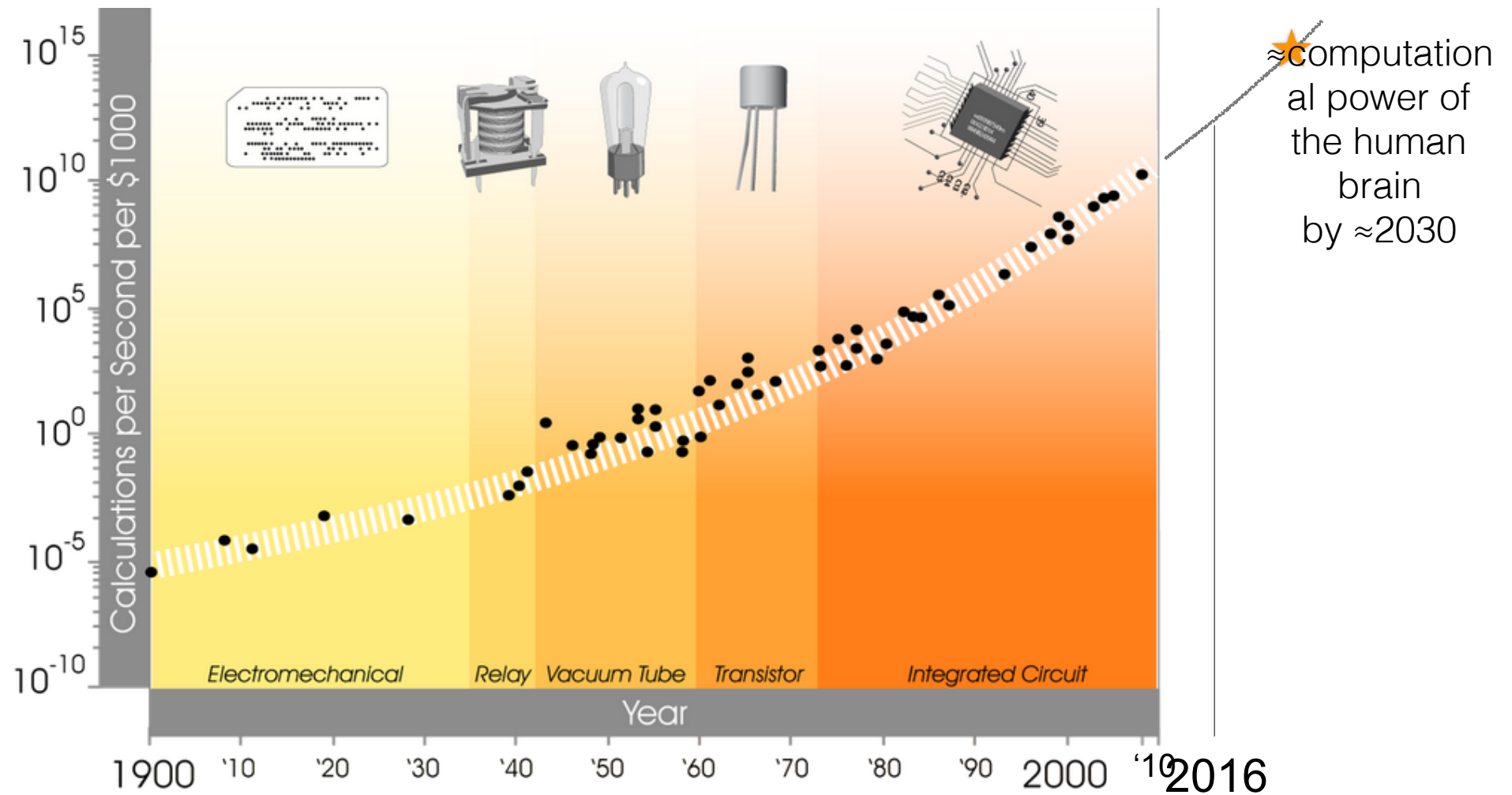
Corporate investment in AI is way up

- Google's prescient AI buying spree: Boston Dynamics, Nest, Deepmind Technologies, ...
- New AI research labs at Facebook (Yann LeCun), Baidu (previously Andrew Ng), Allen Institute (Oren Etzioni), Vicarious, Maluuba...
- Also enlarged corporate AI labs: Microsoft, Amazon, Adobe...
- Yahoo makes major investment in CMU machine learning department
- Many new AI startups getting venture capital

The 2nd industrial revolution

- The 1st industrial revolution was the *physical power* of machines substituting for that of people
- The 2nd industrial revolution is the *computational power* of machines substituting for that of people
 - Computation for perception, motor control, prediction, decision making, optimization, search
 - Until now, people have been our cheapest source of computation
 - But now our machines are starting to provide greater, cheaper computation

The computational revolution



Advances in AI abilities are coming faster; in the last 5 years:

- IBM's Watson beats the best human players of *Jeopardy!* (2011)
- Deep neural networks greatly improve the state of the art in speech recognition and computer vision (2012–)
- Google's self-driving car becomes a plausible reality (\approx 2013)
- Deepmind's DQN learns to play Atari games at the human level, from pixels, with no game-specific knowledge (\approx 2014, *Nature*)
- University of Alberta's Cepheus solves Poker (2015, *Science*)
- Google Deepmind's AlphaGo defeats the world Go champion, vastly improving over all previous programs (2016)

Cheap computation power drives progress in AI

- Deep learning algorithms are essentially the same as what was used in '80s
 - only now with larger computers (GPUs) and larger data sets
 - enabling today's vastly improved speech recognition
- Similar impacts of computer power can be seen in recent years, and throughout AI's history, in natural language processing, computer vision, and computer chess, Go, and other games

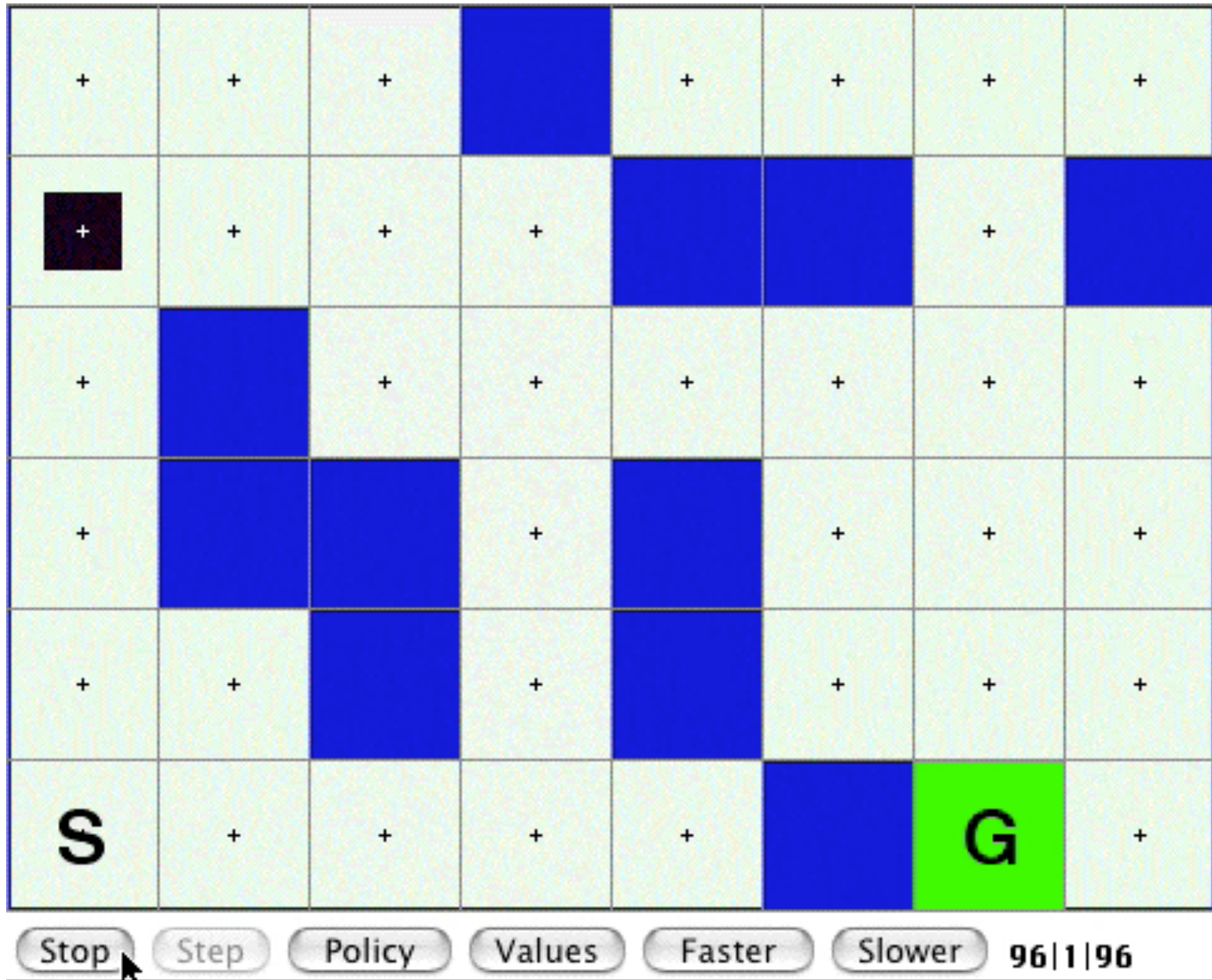
Algorithmic advances are also essential

- Algorithmic advances such as backpropagation, MCTS, policy-gradient reinforcement learning, and LSTM were *necessary but not sufficient*
- They were invented early, then waited for the computational power needed for them to shine
 - other algorithms are still waiting for more cheaper computation
- Algorithmic advances are slower, less reliable
- But they will accelerate with more computation, more focused effort

AI is not like other sciences

- AI has Moore's law, an enabling technology racing alongside it, making the present special
- Moore's law is a slow fuse, leading to the greatest scientific and economic prize of all time
- So slow, so inevitable, yet so uncertain in timing
- The present is a special time for humanity, as we prepare for, wait for, and strive to create strong AI

Demo: GridWorld Example



Textbooks

- *Reinforcement Learning: An Introduction*, by R Sutton and A Barto, MIT Press.
 - In-progress, online 2nd edition
 - <http://incompleteideas.net/sutton/book/the-book-2nd.html>
- Supplementary reading: *The Quest for AI*, by N Nilsson, Cambridge, 2010 (<http://ai.stanford.edu/%7Enilsson/QAI/qai.pdf>)

Prerequisites

- Some comfort or interest in thinking abstractly and with mathematics
- Elementary statistics, probability theory
 - conditional expectations of random variables
- Basic linear algebra: vectors, vector equations, gradients
- Basic programming skills (Matlab and Python)
 - If Python is a problem, choose a partner who is already comfortable with Python

Course Grading

- One assignment per class, due within a week
 - About 8 written / programming assignments – (8)
- Final course project (8), due within a month
- Total: 16

Main Contents

- Tabular Solution Methods (seeking optimal policy) for small-dimension problems
 - Multi-arm Bandits
 - Finite MDP
 - Dynamic Programming
 - Monte Carlo methods
 - TD learning
 - Multi-step bootstrapping
 - Planning and Learning
- Approximate methods (seeking effective but suboptimal policy) for large problems