Practical AI

Introduction

Universität Zürich[UZH]

Dynamic and Distributed Information Systems

What is AI?

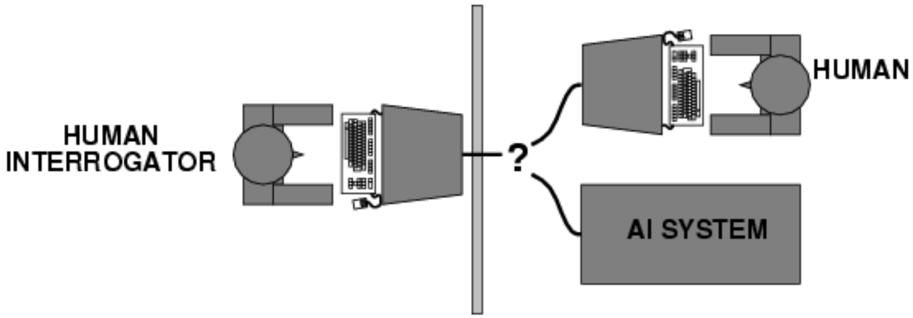
Views of AI fall into four categories:

Thinking humanly	Thinking rationally
Acting humanly	Acting Rationally

The textbook advocates "acting rationally"

Acting humanly: Turing Test

- Turing (1950) "Computing machinery and intelligence":
- "Can machines think?" → "Can machines behave intelligently?"
- Operational test for intelligent behavior: the Imitation Game
- Predicted that by 2000, a machine might have a 30% chance of fooling a lay person for 5 minutes
- Anticipated all major arguments against AI in following 50 years
- Suggested major components of AI: knowledge, reasoning, language understanding, learning



Thinking humanly: cognitive modeling

- 1960s "cognitive revolution": information-processing psychology
- Requires scientific theories of internal activities of the brain
- How to validate? Requires
  - 1) Predicting and testing behavior of human subjects (top-down)
  - or 2) Direct identification from neurological data (bottom-up)
- Both approaches (roughly, Cognitive Science and Cognitive Neuroscience) are now distinct from AI

Thinking rationally: "laws of thought"

- Aristotle: what are correct arguments/thought processes?
- Several Greek schools developed various forms of *logic*: *notation* and *rules of derivation* for thoughts; may or may not have proceeded to the idea of mechanization
- Direct line through mathematics and philosophy to modern AI
- Problems:
  - Not all intelligent behavior is mediated by logical deliberation
  - What is the purpose of thinking? What thoughts should I have?

Acting rationally: rational agent

- Rational behavior: doing the right thing
- The right thing: that which is expected to *maximize goal achievement, given the available information*
- Doesn't necessarily involve thinking – e.g., blinking reflex – but thinking should be in the service of rational action

Rational agents

- An agent is an entity that perceives and acts
- This course is about designing rational agents
- Abstractly, an agent is a function from percept histories to actions:

$$[f: \mathcal{P}^{\star} \rightarrow \mathcal{A}]$$

- For any given class of environments and tasks, we seek the agent (or class of agents) with the best performance
- Caveat: computational limitations make perfect rationality unachievable

  → design best program for given machine resources

Said differently: What this class is all about

- The study of the principles by which natural or artificial machines manipulate knowledge and thereby *act rationally*.
  - how knowledge is acquired
  - how goals are generated and achieved
  - how concepts are formed
  - how collaboration is achieved
- What this class is not about:
  - Understanding intelligence as a phenomenon

Or said in the words of Allan Newell:

… Exactly what the computer provides is the ability *not to be rigid and unthinking but, rather, to behave conditionally.* That is what it means to apply knowledge to action: It means to let the action taken reflect knowledge of the situation, to be sometimes this way, sometimes that, as appropriate …

And dynamic behavior is central to businesses:

"The ultimate goal is *flexibility*…"

Jorma Ollila, former CEO of Nokia

Success Story: Medical Expert Systems

- Mycin (1980)
  - Expert level performance in diagnosis of blood infections
- Today: 1,000’s of systems
  - Everything from diagnosing cancer to designing dentures
  - Often outperform doctors in clinical trials
  - Probably biggest impact in data exchange
- Major hurdle today:
  - non-expert part
  - doctor/machine interaction

SNOMED CT
The global language of healthcare

Success Story: Data Mining & ML

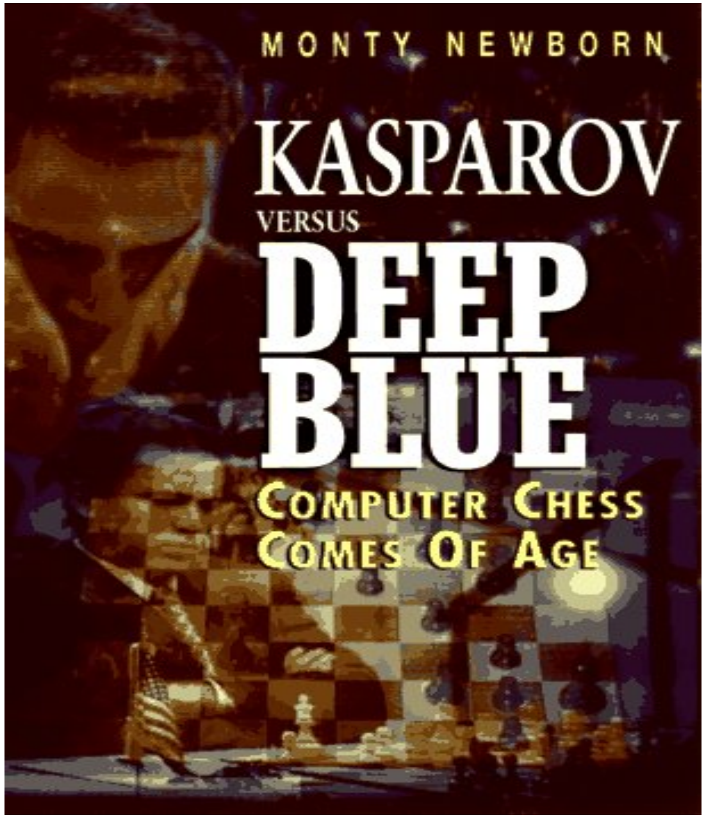
- Bases:
  - Statistics, Machine Learning, Data Bases
- Applications:
  - House price estimations
  - Credit scoring
  - Fraud detection (Credit cards, Cell phones)
  - Marketing (Cumulus)
  - Information Filtering (Spam)
  - Human Computer Interaction (Clippy & Co.)

courtesy JPL

Success Story: Chess

- I could feel – I could smell – a new kind of intelligence across the table
  - Kasparov

- Examines 5 billion positions/second
- “Intelligent” behavior emerges from brute-force search



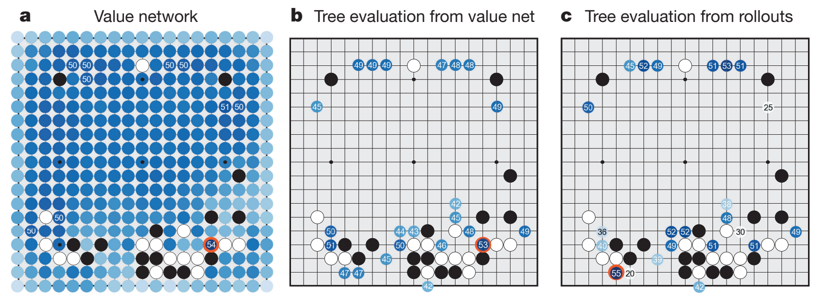
Success Story: Go & Poker (Liberatus)

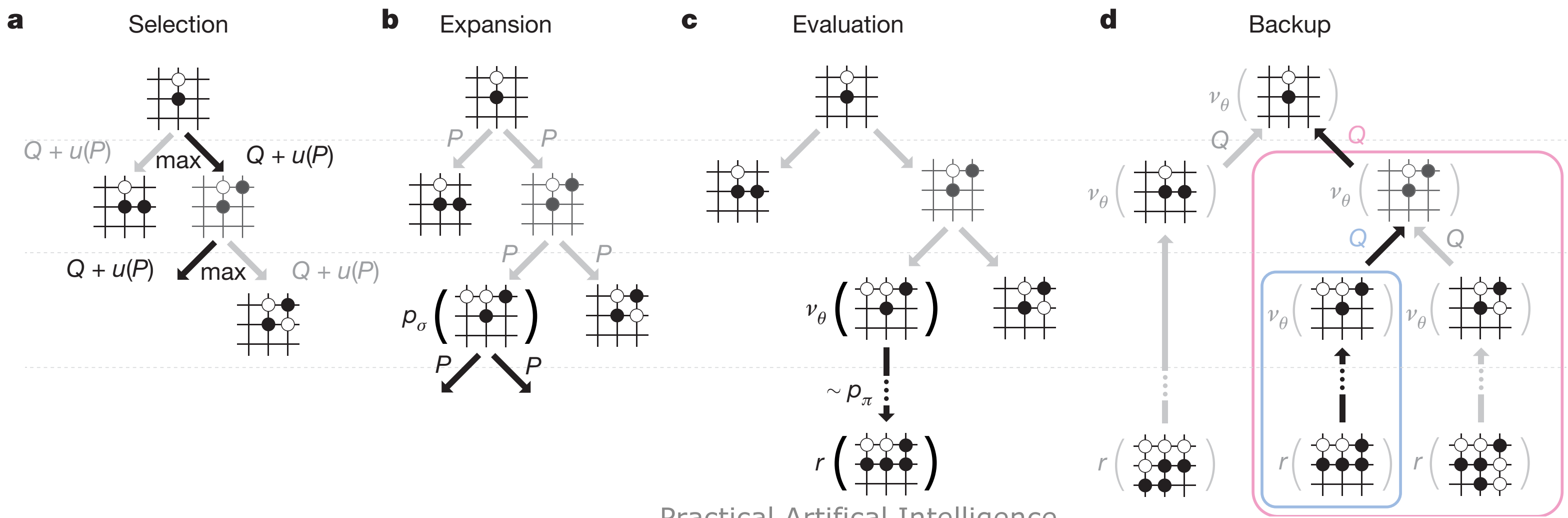
ARTICLE

doi:10.1038/nature16961

Mastering the game of Go with deep neural networks and tree search

David Silver[1*], Aja Huang[1*], Chris J. Maddison[1], Arthur Guez[1], Laurent Sifre[1], George van den Driessche[1], Julian Schrittwieser[1], Ioannis Antonoglou[1], Veda Panneershelvam[1], Marc Lanctot[1], Sander Dieleman[1], Dominik Grewe[1], John Nham[2], Nal Kalchbrenner[1], Ilya Sutskever[2], Timothy Lillicrap[1], Madeleine Leach[1], Koray Kavukcuoglu[1], Thore Graepel[1] & Demis Hassabis[1]





DIS

Success Story Watson

Historic Perspective

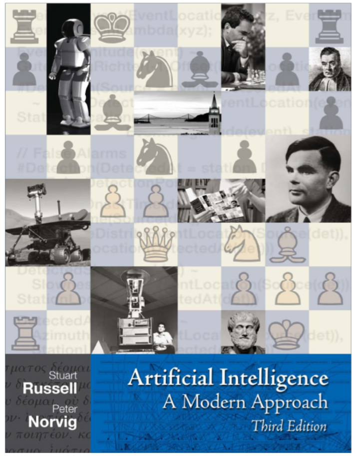
- 1940's - 1960's: Artificial neural networks
  - McCulloch & Pitts 1943
- 1950's - 1960's: Symbolic information processing
  - General Problem Solver – Simon & Newell
  - "Weak methods“ for search and learning
  - 1969: Minsky's Perceptrons
- 1940’s – 1970’s: Control theory for adaptive (learning) systems
  - USSR – Cybernetics – Norbert Wiener
  - Japan – Fuzzy logic
- 1970's – 1980’s: Expert systems
  - “Knowledge is power" – Ed Feigenbaum
  - Logical knowledge representation
  - AI Boom
- 1985 – 2000: A million flowers bloom
  - Resurgence of neural nets – backpropagation
  - Control theory + OR + Pavlovian conditioning = reinforcement learning
  - Probabilistic knowledge representation – Bayesian Nets – Judea Pearl
  - Statistical machine learning
- 2000’s: Towards a grand unification
  - Unification of neural, statistical, and symbolic machine learning
  - Unification of logic and probabilistic KR
  - Autonomous systems
- 2010’s: The emergence of ML
  - Neural Networks revisted by GPUs
  - Deep Learning
  - Big Data

This course focuses on

- AI techniques that have been employed in practice:
  - Smart Searching, Knowledge representation/reasoning
  - Probabilistic reasoning, Learning
  - Other
- Practical Case studies of their use
  - Games, … (we will see)

Course Mechanics

- Time/space: Tue 14–15:45 pm & 16–17:45, BIN 2.A.01
- Lecturer: Prof. A. Bernstein, Ph.D.
  Daniele Dell'Aglio, Matthias Baumgarter, Hoda Heidari, Michele Loi
- Assistants: Suzanne Tolmeijer, Pengcheng Duan, Narges Ashena
- Requirements: 3 Assignments, Final Exam
- Assignments: eLearning, paper & pencil, programming
- Book: Russel, S., & Norvig, P.
  Artificial Intelligence: A Modern Approach ***3e***
- Web–site: OLAT!



Provisional Time Plan

Date	Subject	Readings (Ch.)	Instructor	Tutorial session	Assignment
19.2	Introduction	1	AB		
	Part 1: Intelligent Search & Logic				
26.2	Informed Search	3, 4	AB	Informed search (ST)	
5.3	Adversarial search	5	ST	Adversarial search (ST)	**A1** out
12.3	Constraint satisfaction	6	AB		
19.3	Logic	7, 8, 12	DD		
	Part 2: Learning				
26.3	Probability revisited	13	AB		
2.4	Naïve Bayes and other algorithms	18	AB		**A1** in / **A2** out
9.4	Bayesian Belief Networks and Reasoning	14	AB	Conditional Independence (NA)	
16.4	Reinforcement Learning	21	HH	Reinforcement Learning (ST)	
23.4	NO CLASS (Spring Break)	-			
30.4	Neural Networks	18,21	MB	Neural Networks (MB)	**A2** in / **A3** out
	Part 3: Other				
7.5	Recommender Systems	Slides	AB		
14.5	Ethical issues with AI	Slides	ML		
21.5	TBA	Slides	TBA		
28.5	Q&A	-	ST/MB		**A3** in
11.6	**EXAM** (according to VVZ; see also dean's office regulations)				

AB = Prof. Abraham Bernstein
ST = Suzanne Tolmeijer
DD = Daniele Dell'Aglio
MB = Matthias Baumgartner
NA = Narges Ashena
HH = Hoda Heidari
ML = Michele Loi

Practical AI

Agents

Universität Zürich[UZH]

Dynamic and Distributed Information Systems

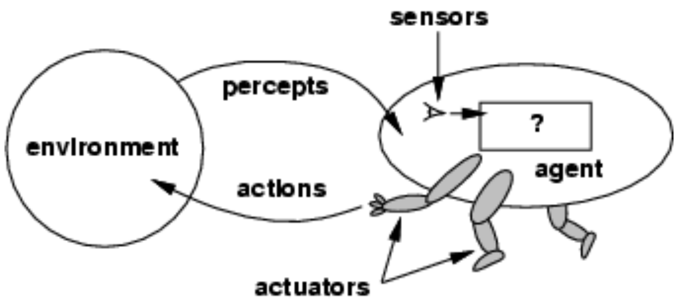
Outline

- Agents and environments
- Rationality
- PEAS (Performance measure, Environment, Actuators, Sensors)
- Environment types
- Agent types

Agents

- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators
- Human agent:
  - eyes, ears, and other organs for sensors;
  - hands, legs, mouth, and other body parts for actuators
- Robotic agent:
  - cameras and infrared range finders for sensors;
  - various motors for actuators

Agents and environments

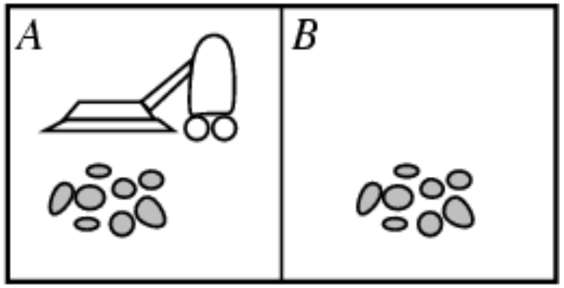


- The agent function maps from percept histories to actions:

$$[f\colon \mathcal{P}^\star \rightarrow \mathcal{A}]$$

- The agent program runs on the physical architecture to produce *f*
- agent = architecture + program

Vacuum-cleaner world



- Percepts: location and contents, e.g., [A, Dirty]
- Actions: *Left*, *Right*, *Suck*, *NoOp*

Rational agents

- An agent should strive to "do the right thing", based on what it can perceive and the actions it can perform. The right action is the one that will cause the agent to be most successful
- Performance measure: An objective criterion for success of an agent's behavior
- E.g., performance measure of a vacuum-cleaner agent could be amount of dirt cleaned up, amount of time taken, amount of electricity consumed, amount of noise generated, etc.

Rational agents

- Rational Agent: For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.

Rational agents

- Rationality is distinct from omniscience (all-knowing with infinite knowledge)
- Agents can perform actions in order to modify future percepts so as to obtain useful information (information gathering, exploration)
- An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)

PEAS

- PEAS: Performance measure, Environment, Actuators, Sensors
- Must first specify the setting for intelligent agent design

- Consider, e.g., the task of designing an automated taxi driver:
  - Performance measure
  - Environment
  - Actuators
  - Sensors

PEAS

- Must first specify the setting for intelligent agent design
- Consider, e.g., the task of designing an automated taxi driver:
  - Performance measure: Safe, fast, legal, comfortable trip, maximize profits
  - Environment: Roads, other traffic, pedestrians, customers
  - Actuators: Steering wheel, accelerator, brake, signal, horn
  - Sensors: Cameras, sonar, speedometer, GPS, odometer, engine sensors, keyboard

PEAS: More examples

Agent Type	Performance Measure	Environment	Actuators	Sensors
Robot Soccer Player	Winning game, goal for/against	Field, ball, own team, other team, own body	Devices (e.g., legs) for locomotion and kicking	Camera, touch sensors, accelerometers, orientation sensors, wheel/joint encoders
Internet Book shopping agent	Obtain requested/interestin g books, minimize expenditure	Internet	Follow link, enter/submit data in fields, display to user	Web pages, user requests
Autonomous Mars rover	Terrain explored and reported, samples gathered and analyzed	Lauch vehicle, lander, Mars	Wheels/legs, sample collection devive, analysis devices, radio transmitter	Camera, touch sensors, accelerometers, orientation sensors, wheel/joint encoders, radio receiver

Environment types

- Fully observable (vs. partially observable): An agent's sensors give it access to the complete state of the environment at each point in time.
- Deterministic (vs. stochastic): The next state of the environment is completely determined by the current state and the action executed by the agent. (If the environment is deterministic except for the actions of other agents, then the environment is strategic)
- Episodic (vs. sequential): The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.

Environment types

- Static (vs. dynamic): The environment is unchanged while an agent is deliberating. (The environment is semidynamic if the environment itself does not change with the passage of time but the agent's performance score does)

- Discrete (vs. continuous): A limited number of distinct, clearly defined percepts and actions.

- Single agent (vs. multiagent): An agent operating by itself in an environment.

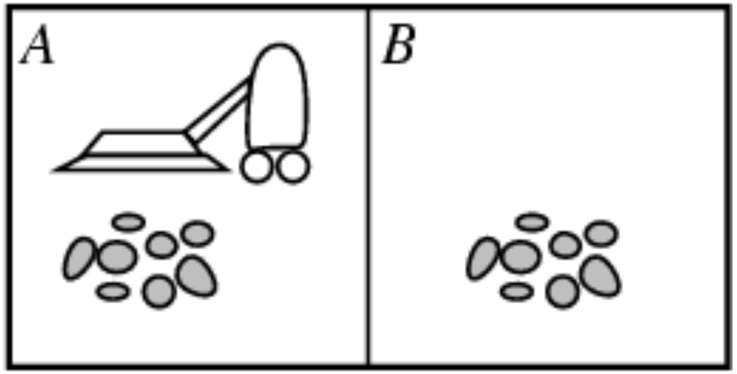
Environment types: Examples

Task Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Robot soccer	Partially	Stochastic	Sequential	Dynamic	Continuous	Multi
Internet book-shopping	Partially	Deterministic*	Sequential	Static*	Discrete	Single
Autonomous Mars rover	Partially	Stochastic	Sequential	Dynamic	Continuous	Single
Mathematician's assistant	Fully	Deterministic	Sequential	Semi	Discrete	Multi

Agent functions and programs

- An agent is completely specified by the agent function mapping percept sequences to actions
- One agent function (or a small equivalence class) is rational
- Aim: find a way to implement the rational agent function concisely

Agent program for a vacuum-cleaner agent



Reflex-Vacuum-Agent (location, status) → an action

if *status* = *Dirty* then return Suck

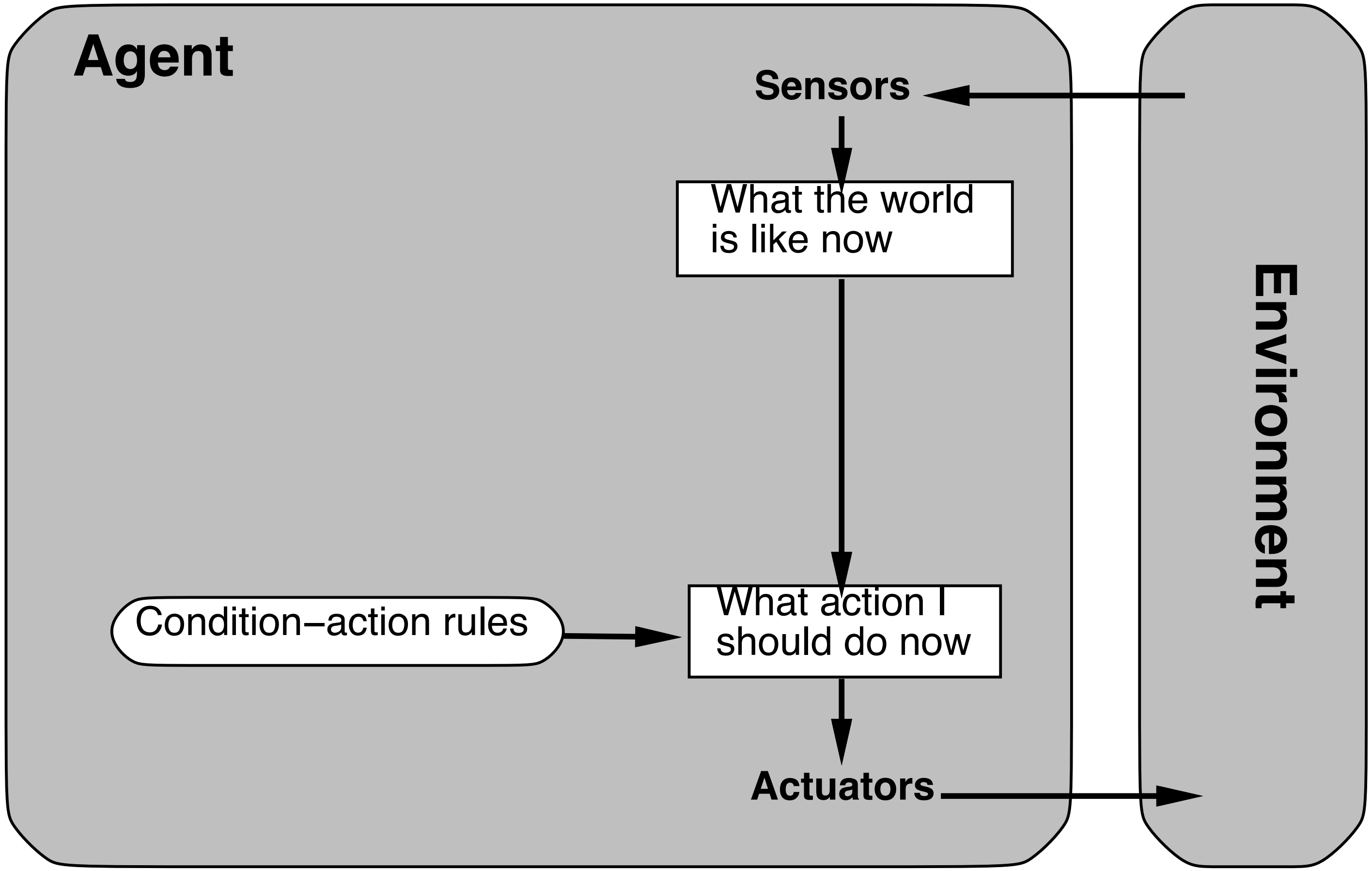
elseif *location* = *A* then return *Right*

elseif *location* = *B* then return *Left*

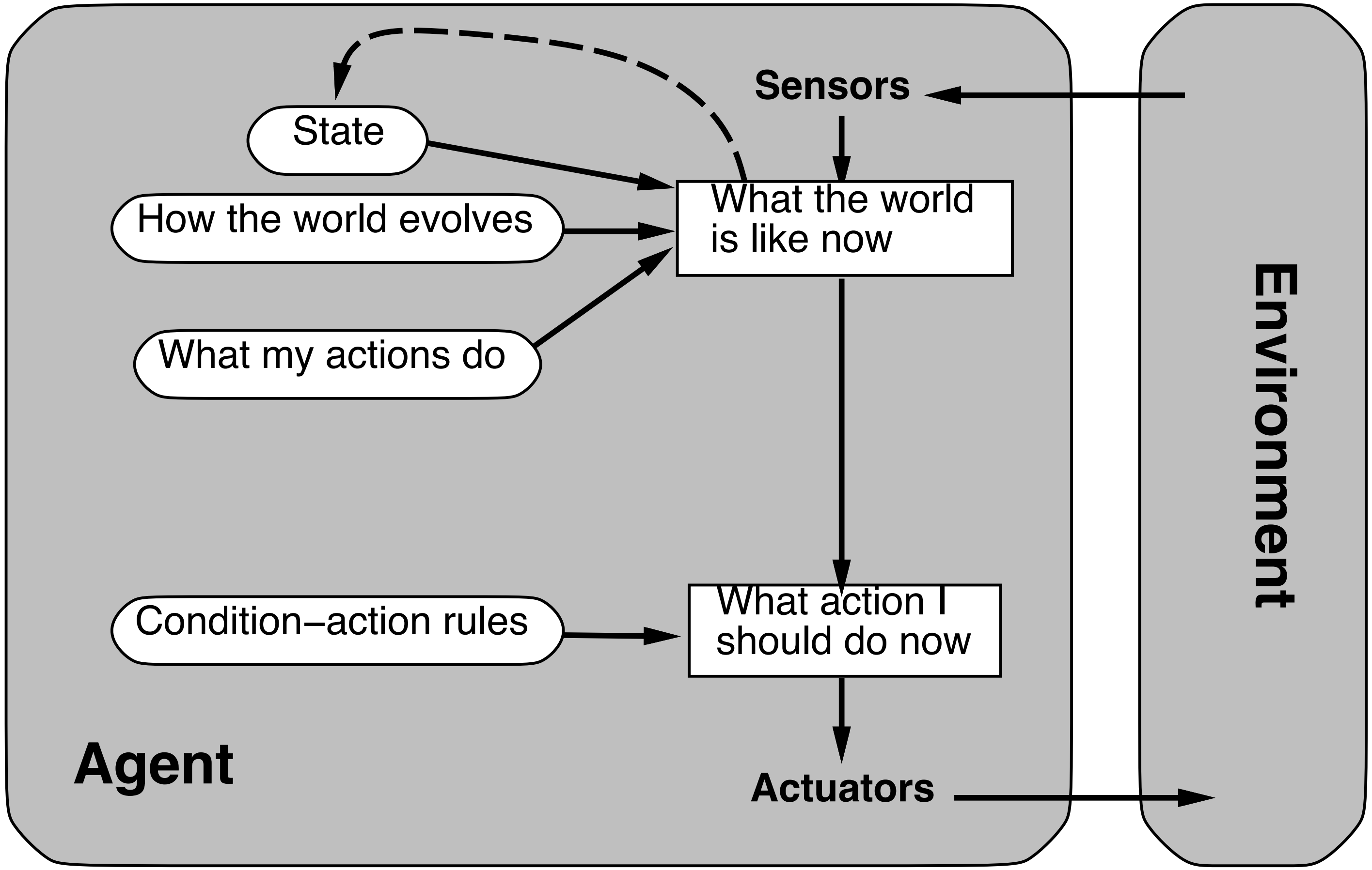
Agent types

- Four basic types in order of increasing generality:
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents

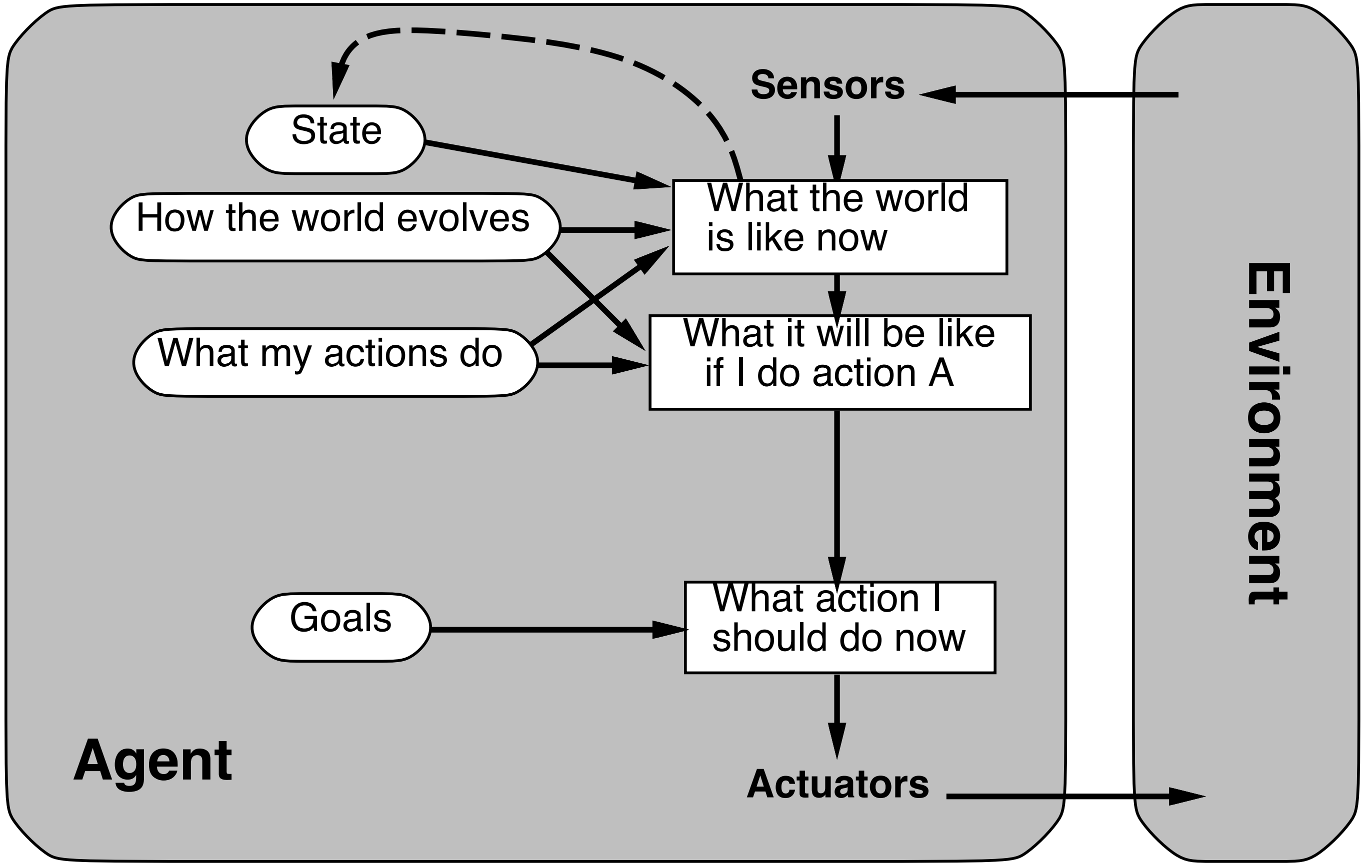
Simple reflex agents



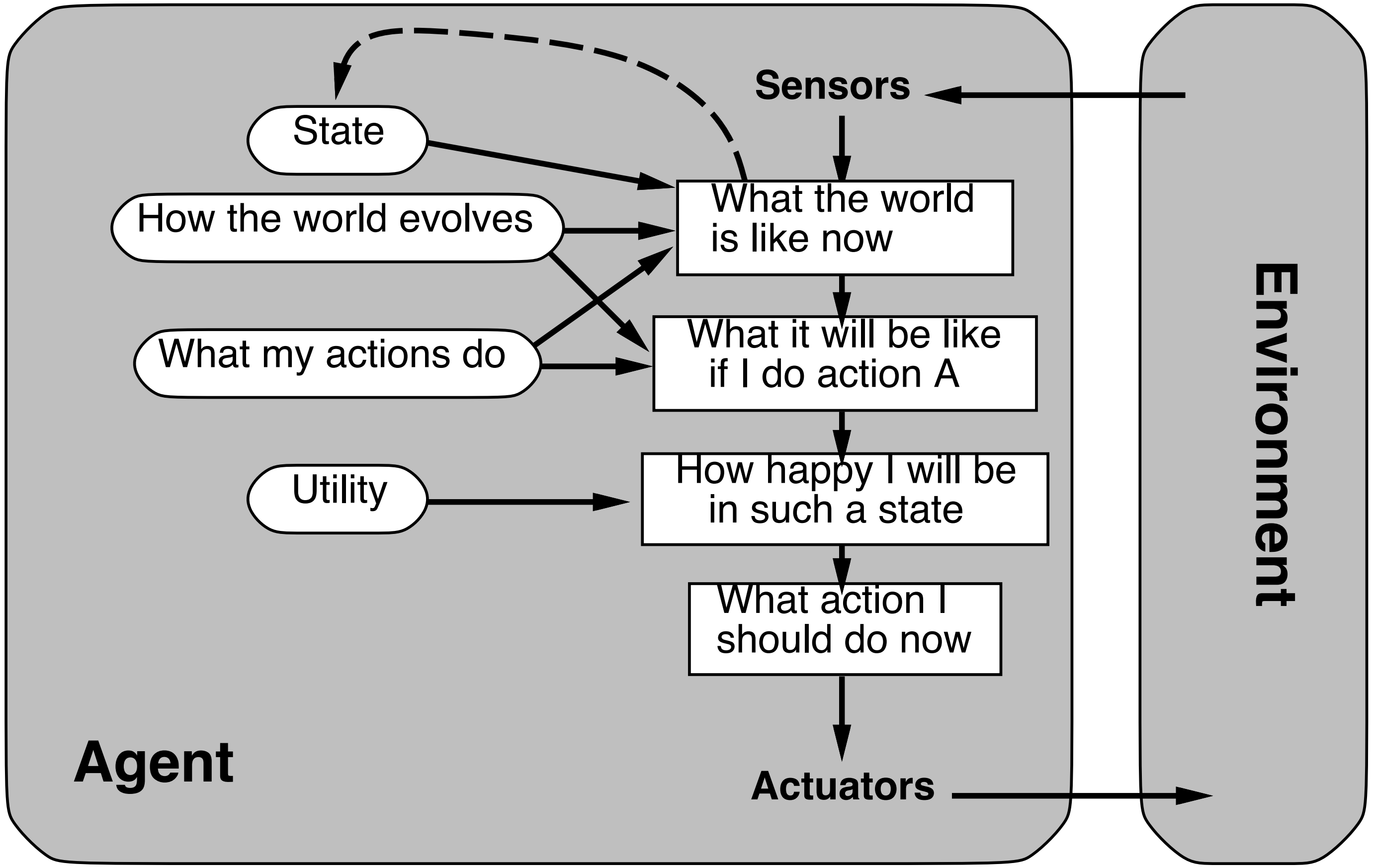
Model-based reflex agents



Goal-based agents



Utility-based agents



Learning agents