# Introduction in
# Artificial Intelligence

## CSC545/645

## Ubbo Visser

# Short introduction

- Ubbo Visser

  

  - Office: Ungar Building, Room No 330A
  - Phone: 305-284-2254
  - Email: visser@cs.miami.edu (preferred especially if mail contains any attachment)
  - Office Hours: by appointment

- Stations
  - Münster, Brisbane, Bremen
  - Miami
    - Associate Professor

- Research interests
  - Artificial Intelligence with the focus on knowledge representation and reasoning.
  - Application areas: the Semantic Web and Multi-Agent Systems (Games, Robots, RoboCup)

# More introduction…

- Teaching Assistant
  - Chris Duarte

- Contact Hours
  - Each week there are two 75 minute sessions (TuTh 2:00 pm - 3:15 pm)
  - Classroom: Whitten LC 180

- Recommended Text Book
  - Artificial Intelligence - A modern approach -  4/3/E, Stuart Russell & Peter Norvig, Prentice Hall, ISBN-10: 0-13-604259-7, 0-13-461099-7

- Course Content
  - Chapters 1 to 6, 18 to 21, plus additional content from other textbooks/courses. If time is short, some of the concepts will be omitted. Course material will be uploaded before the lecture as .pdf files.
  Check http://www.cs.miami.edu/~visser regularly.

# Grading & general issues

| Item | Points |
|------|--------|
| Homework | 70 |
| Final | 30 |

– Grading

  – will be based on 100 points

– Scoring of Homework Assignments

  – The score of each homework will be mentioned in it.

  – The total score of all homework assignments will be scaled down to 70 points at the end of the semester for the purpose of final grading. For example, if all homework assignments collectively carry 100 points and a student gets 90 out of 100, he/she gets 90*70/100 or 63 out of 70 in Homework Assignment component for final grading.

– Class attendance and participation

  – Class attendance is not mandatory, although my exams will depend heavily of my lectures. Not all of the material will come from the text. Class participation is also important. Active interest in lectures is the easiest way to learn.

# General issues (2)

- Plagiarism
  - The penalty for copied homework of any kind can be immediate failure in the course. My policy on programs is as follows: There is no reason for two (or more) people handing in identical or nearly identical programs. I will regard such programs as either group-written or simply copied. If I have no hard evidence of copying, such programs will receive NO credit. More serious actions will be taken in cases where there is evidence of cheating.

- Late programs
  - Unless otherwise stated, programs will lose 20% of their value for each weekday (Monday through Friday) that they are late, down to a minimum value of 20%. The due date of a program is the latest date on which it can be run to get full credit.

# General issues (3)

- Dropping the course
  - Unless there are extreme extenuating circumstances, I will not allow anyone to drop a course after the drop date. Poor academic performance will never be an acceptable reason for a late drop.

- Incompletes
  - Unless there has been a documentable illness that caused you to miss substantial amounts of class and computer time, I will not give an incomplete grade in this course. Therefore, please do NOT waste my time asking about an incomplete grade unless you have a remarkably good reason.

- Make-up exams
  - I do not give make-up exams. You simply must show up and take them at the specified times.
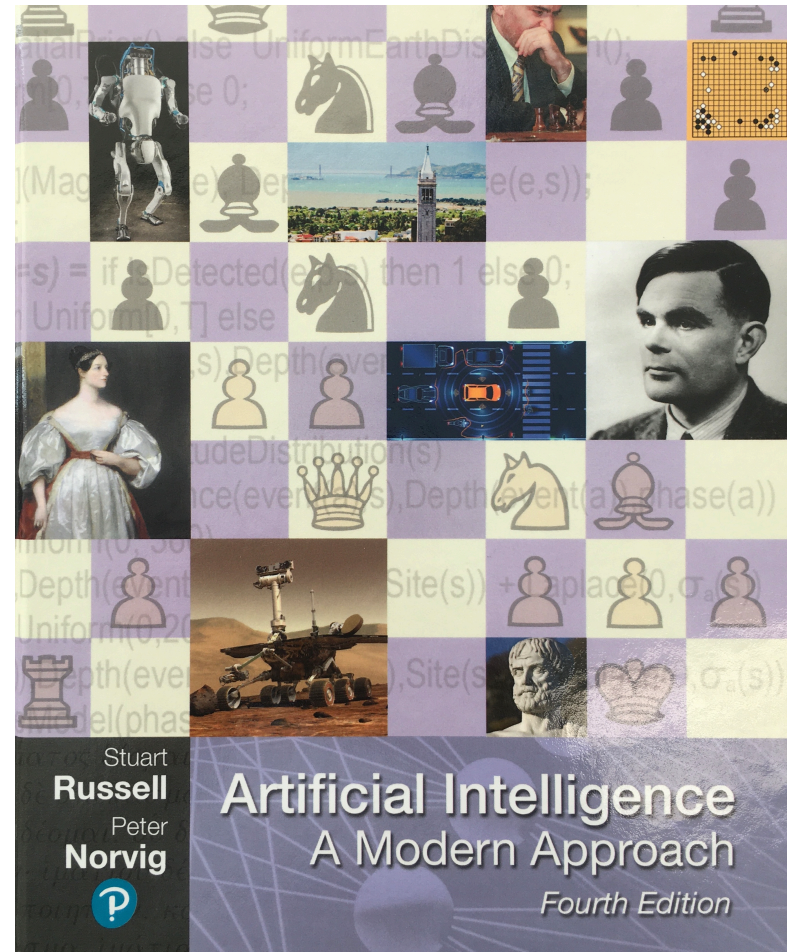
# Textbook

Stuart Russell & Peter Norvig

Artificial Intelligence: A Modern Approach, 4th edition.
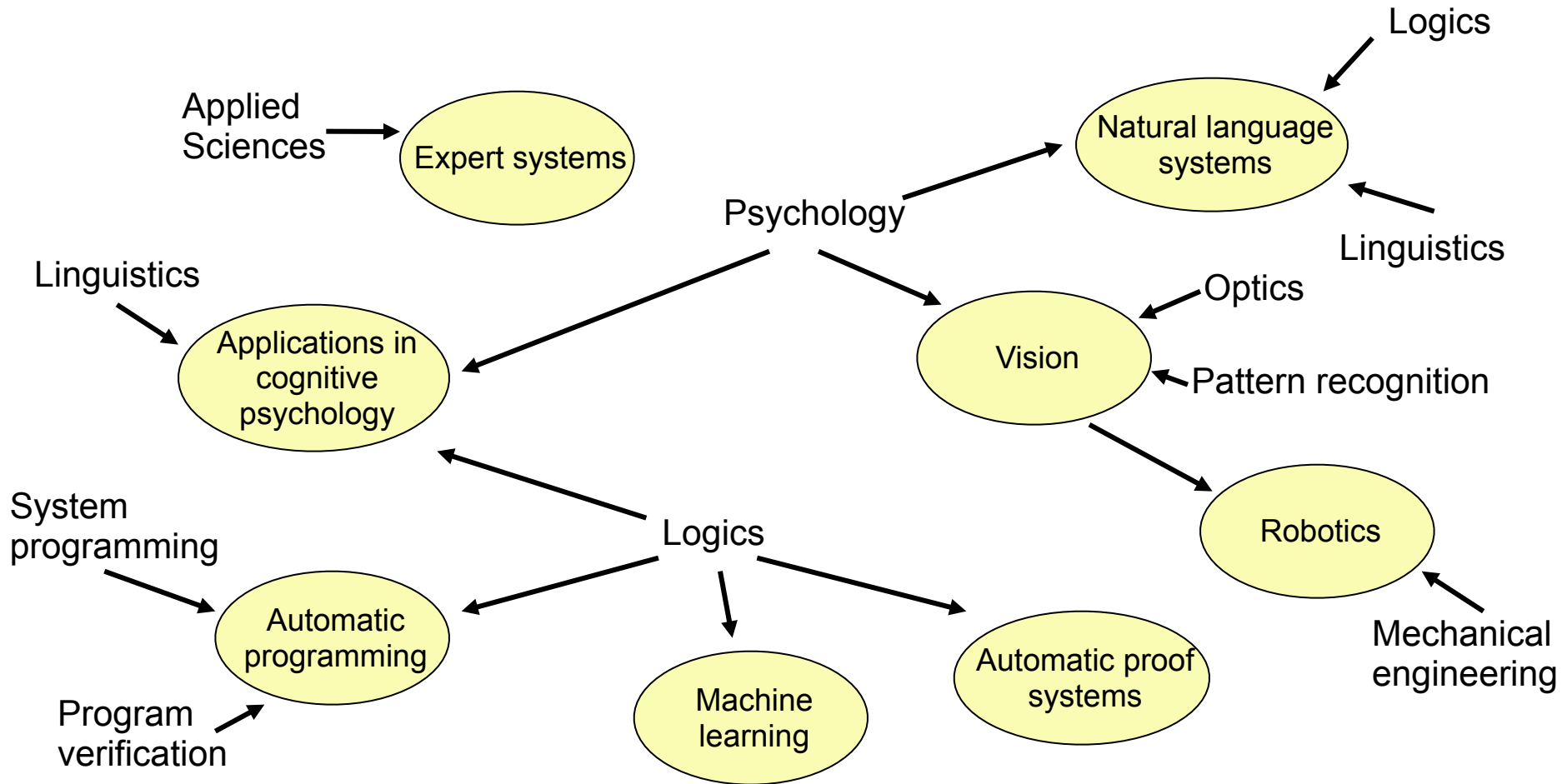
Prentice-Hall, 2020

ISBN 0-13-461099-7

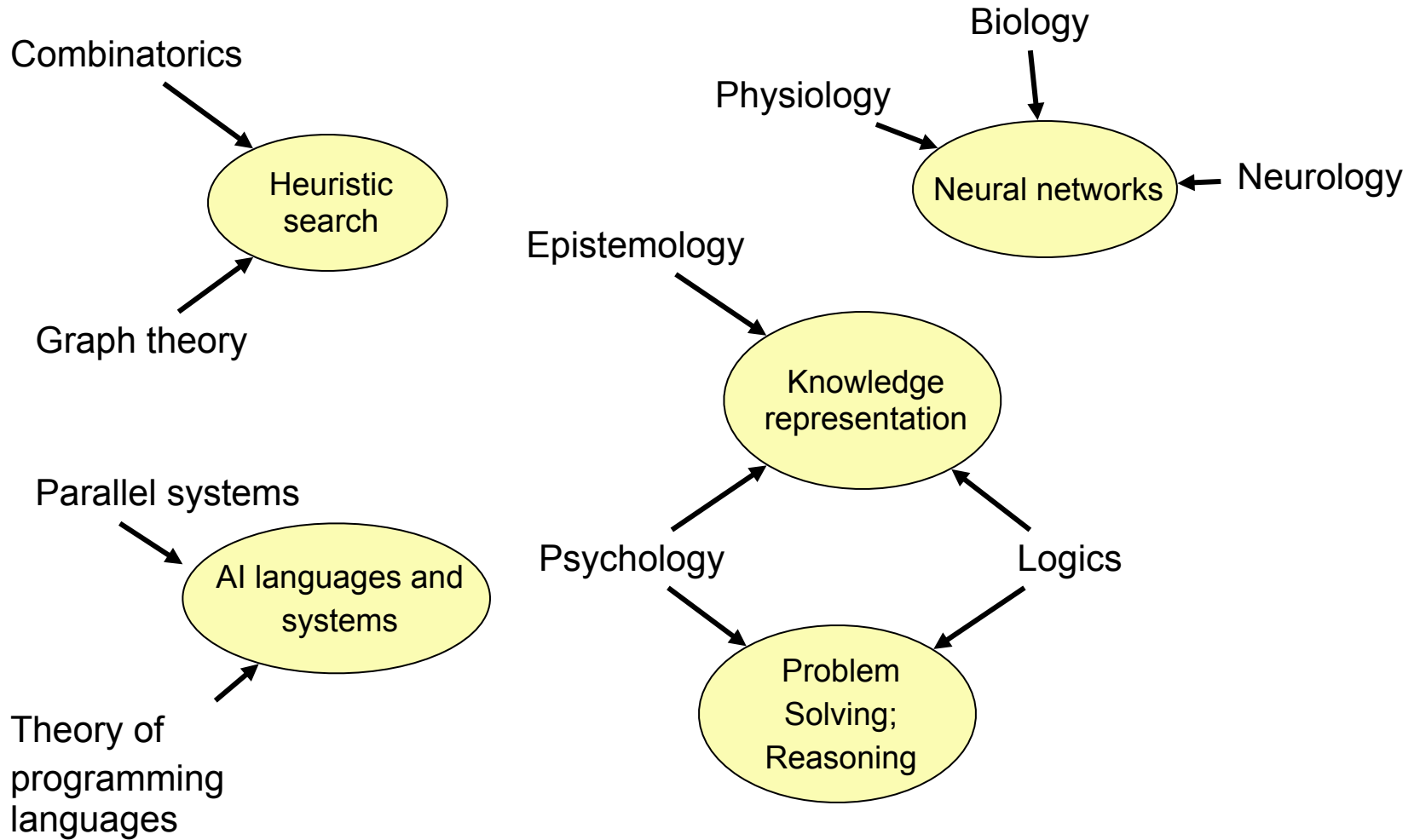# Artificial Intelligence

1. What is AI?
2. Intelligent agents

# 1. What is AI?

## Some areas of AI

# Some methods of AI

# Some known definitions

| | |
|---|---|
| "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning…" (Bellman, 1978) | "The study of mental faculties through the use of computational models." (Charniak+McDermott, 1985) |
| "The study of how to make computers do things at which, at the moment, people are better." (Rich+Knight, 1991) | "The branch of computer science that is concerned with the automation of intelligent behavior." (Luger+Stubblefield, 1993) |

*Source: Russell/Norvig, 1995*

- Systems that think like humans
- Systems that act like humans

- Systems that think rationally
- Systems that act rationally

„The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992)

# Thinking like humans

- **Cognitive approach: how do we think?**
  - Introspection
    - i.e. self observation while thinking
  - Psychological experiments
  - If theory of mind is correct, programs can be written
  - GPS, Newell & Simon 1961
  - Research area: Cognitive science

- **Separation between AI and cognitive science**
  - together in early stages
  - today: fruitful discussion
  - e.g. vision, natural language, learning

# Acting like humans



- Turing Test
  - Alan Turing 1950
  - Definition of intelligence

- Computer need for this:
  - Processing of natural language
  - Knowledge representation
  - Inference mechanisms
  - Learning methods

- Total Turing Test
  - so far: no physical interaction necessary
  - here: video signal to test perception ability

- Thus:
  - Understanding images
  - Robotics

# Rational thinking

- "Law of thoughts"
  - Aristotle's Syllogism as first "Logic"
  - Given: correct preconditions
  - Result: correct conclusion

- Example
  - All men are mortal
  - Socrates is a man
  - ...?

- Two problems with this approach
  - Transformation of non-formal knowledge in formal knowledge is difficult (e.g. accuracy)
  - Big difference between problem solving in principal and in practice
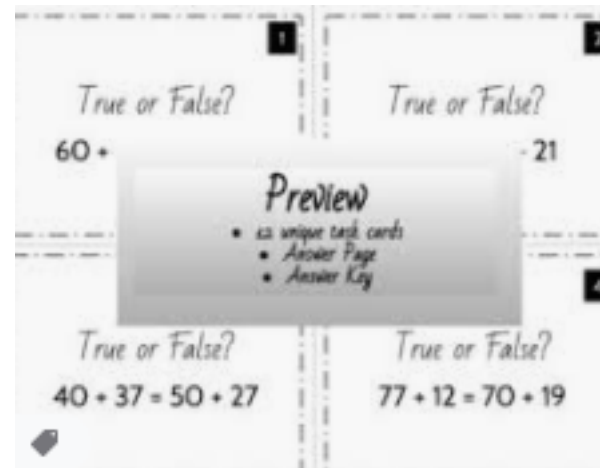
# Acting rational

- **Rational agent**
  - Agents act autonomous, perceive etc.
  - Rational agents act using the 'best outcome' principle
  - If uncertainty is given: best expected outcome
  - Law of thoughts – approach is based on correct inferences

  - This is sometimes part of a rational agent but not vice versa
  - Skills for Turing Test exists

- **AI and rational agents have two advantages**
  - More general as "LoT"
  - Better as e.g. behavioral approaches because rationality is defined

# Contribution of sciences

- Philosophy (428 BC – today)
  - Aristotle, induction, logics
- Mathematics (~800 – today)
  - Algorithms: "Entscheidungsproblem", NP-completeness, Probabilities
- Economy (1776 – today)
  - Utility functions, decision theory, OR, game theory, MDPs
- Neurosciences (1861 - today)
  - Parallel processing, neurons

- Psychology (1879 – today)
  - Behavior, cognitive psychology
- Computer (1940 – today)
  - Zuse's $Zn$, OS
- Control theory and Cybernetics (1948 – present)
  - stable systems, max. goal functions
- Linguistics (1957 – today)
  - NLP, knowledge representation

# Short History of AI

- Pre-AI (1943-1955)
  - McCulloch & Pitts
- Birth of AI (1956)
  - Dartmouth conference
- Enthusiasms (1952-1969)
  - Physical systems based on symbols, LISP
- Reality (1966-1973)
  - Only syntactic manipulation, intractability
- Knowledge based systems (1969-1979)
  - weak AI, GPS, expert systems

- AI as industry product (1980-today)
  - R1 at DEC, many firms XEROX, Boeing, TI,…
- Neuronal networks are back (1986 – today)
  - Backpropagation
- AI as science
  - HMM, Data Mining, Bayes
- Intelligent agents (since 1995)
  - SOAR, agent based view on AI
- Self-driving cars (since 2009)
  - SOAR, agent based view on AI

# Short History of AI
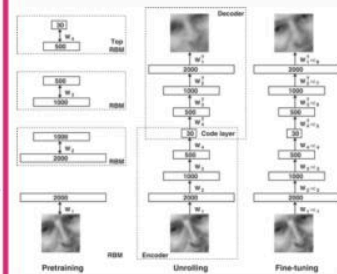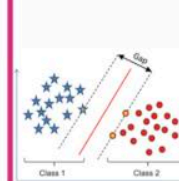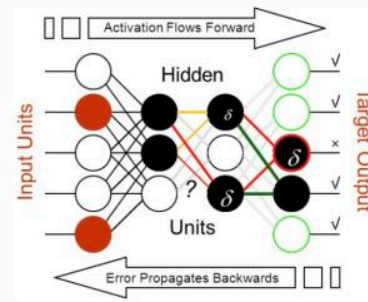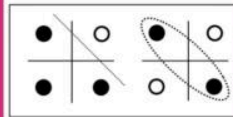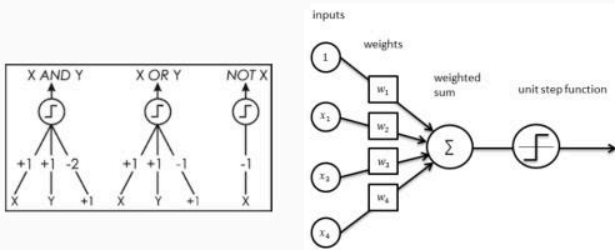
# 2. Intelligent agents

- In which we discuss the nature of agents, perfect or otherwise, the diversity of environments, and the resulting menagerie of agent types.

# Agents

- Russell und Norvig
  - "An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors."

- Wooldridge und Jennings
  - A weak notion: Essential properties of agents:
    - autonomy: agents operate without direct intervention of humans, and have control over their actions and internal state;
    - social ability: agents interact with other agents (and possibly humans) via an agent communication language;
    - reactivity: agents perceive their environment and respond in a timely and rational fashion to changes that occur in it;
    - pro-activeness: agents do not simply act in response to their environment, they are capable of taking the initiative (generate their own goals and act to achieve them).

# Agents

- Wooldridge und Jennings
  - A stronger notion: An agent has mental properties, such as knowledge, belief, intention, obligation. In addition, and agent has other properties such as:
    - mobility: agents can move around from one machine to another and across different system architectures and platforms;
    - veracity: agents do not knowingly communicate false information;
    - benevolence: agents always try to do what they are asked of;
    - rationality: agents will try to achieve their goals and not act in such a way to prevent their goals from being achieved.

# Agents

- ## Gheorghe Tecuci
  - An intelligent agent is a knowledge-based system that perceives its environment, reasons to interpret perceptions, draw inferences, solve problems, and determine actions; and acts upon that environment to realize a set of goals or tasks for which it was designed...

- ## IBM
  - One last definition: Intelligent agents are software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires.
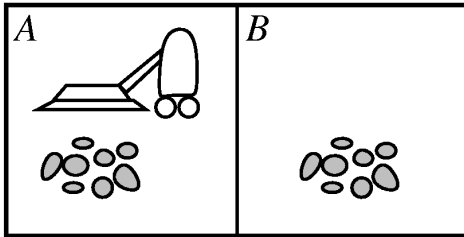
# Agents and environments
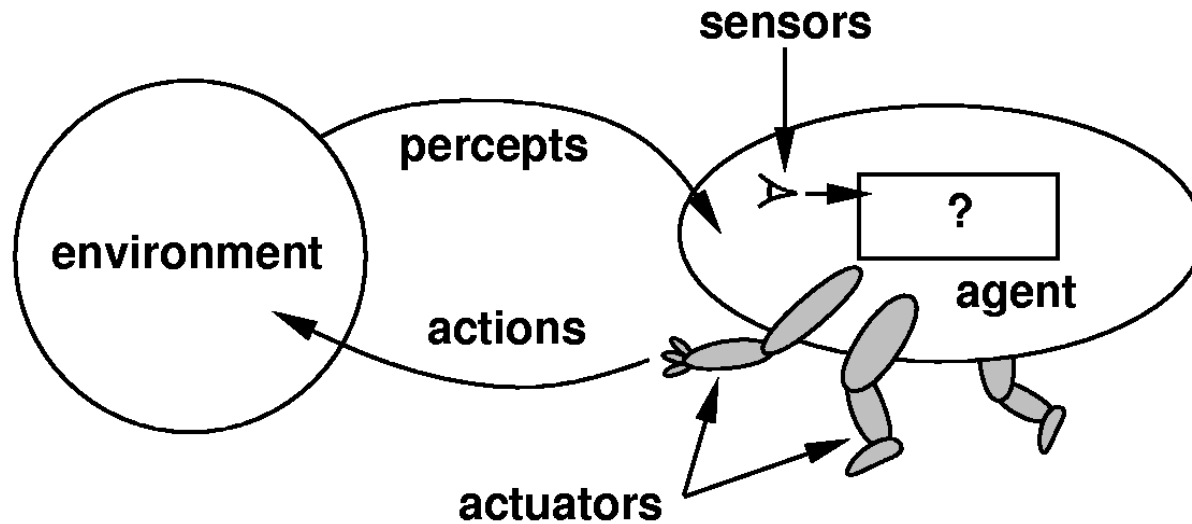


Agents interact with environments through sensors and actuators.

- Perception, perception sequence
- Agent function (abstract)
- Agent program (concrete)

23

# Example



A vacuum-cleaner world with just two locations

- Perception: A v B, dust yes/no
- Actions: move left, move right, suck, do nothing
- Agent functions: if dust exists, then suck, otherwise do nothing

| Percept sequence | Action |
|---|---|
| [A, Clean] | Right |
| [A, Dirty] | Suck |
| [B, Clean] | Left |
| [B, Dirty] | Suck |
| [A, Clean], [A, Clean] | Right |
| [A, Clean], [A, Dirty] | Suck |
| ⋮ | ⋮ |
| [A, Clean], [A, Clean], [A, Clean] | Right |
| [A, Clean], [A, Clean], [A, Dirty] | Suck |
| ⋮ | ⋮ |

**Figure 2.3** Partial tabulation of a simple agent function for the vacuum-cleaner world shown in Figure 2.2.

- Solution is simple: fill right column, various variants possible
- Question: How do I fill this column?
- Is there a good or bad, intelligent or dumb?

# Good behavior: rationality

- Rational agent
  - A rational agent is one that does the right thing…
  - First approximation, we will say that the right action is the one that will cause the agent to be most successful

  - Problem: How and when do we decide whether or not the agent was successful?

- Performance measures
  - Subjective
    - Agent evaluates himself
  - Objective
    - Evaluation done by observer: he defines standard for being successful in the environment
  - Example: soccer agent

# Good behavior

- Omniscience and rationality
  - An omniscient agent knows the effects of its actions and can act accordingly.

    But: who knows it all? → theoretical.

  - Rationality: expected success based on things that can be perceived.

- Rationality based on
  - The performance measure that defines the criterion of success.
  - The agent's prior knowledge of the environment.
  - The actions that the agent can perform.
  - The agent's percept sequence to date.

# Ideal rational agent

- An ideal 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.

- Autonomy
  - Inherent knowledge
  - A system is autonomous, if its behavior is determined by its own experience.

# Environments

| Agent Type | Performance Measure | Environment | Actuators | Sensors |
|---|---|---|---|---|
| Taxi driver | Safe, fast, legal, comfortable trip, maximize profits | Roads, other traffic, pedestrians, customers | Steering, accelerator, brake, signal, horn, display | Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard |

- PEAS for an automated taxi
  - P: performance measure
  - E: environment
  - A: actuators/effectors
  - S: Sensors

28

# Environment characteristics

- Fully observable vs. partially observable
- Deterministic vs. stochastic
- Episodic vs. Sequential
- Static vs. Dynamic
- Discrete vs. Continuous
- single-agent vs. multi-agents

| Task Environment | Observable | Deterministic | Episodic | Static | Discrete | Agents |
|---|---|---|---|---|---|---|
| Crossword puzzle | Fully | Deterministic | Sequential | Static | Discrete | Single |
| Chess with a clock | Fully | Strategic | Sequential | Semi | Discrete | Multi |
| Poker | Partially | Strategic | Sequential | Static | Discrete | Multi |
| Backgammon | Fully | Stochastic | Sequential | Static | Discrete | Multi |
| Taxi driving | Partially | Stochastic | Sequential | Dynamic | Continuous | Multi |
| Medical diagnosis | Partially | Stochastic | Sequential | Dynamic | Continuous | Single |
| Image-analysis | Fully | Deterministic | Episodic | Semi | Continuous | Single |
| Part-picking robot | Partially | Stochastic | Episodic | Dynamic | Continuous | Single |
| Refinery controller | Partially | Stochastic | Sequential | Dynamic | Continuous | Single |
| Interactive English tutor | Partially | Stochastic | Sequential | Dynamic | Discrete | Multi |

# Example: standard problems: Chess vs. RoboCup

|  | Chess | RoboCup |
|---|---|---|
| Environment | static | dynamic |
| State change | with each move | always (real time) |
| Information access | given | incomplete |
| Sensors | symbolic | not symbolic |
| Control | central | distributed |

# Structure of intelligent agents

- Architecture
  - Architecture can be a computer but also special hardware (e.g. image processing device). The architecture makes sure, that the agent gets its perceptions and also runs the program.
  - Agent = Architecture + Program

- Simple program
  - Skeleton, perception sequences are accepted, internal data structures will be updated.

```
function TABLE-DRIVEN-AGENT(percept) returns an action
    static: percepts, a sequence, initially empty
            table, a table of actions, indexed by percept sequences, initially fully specified

    append percept to the end of percepts
    action ← LOOKUP(percepts, table)
    return action
```

  - But:
    - Only one sequence at a time
    - No goal and no performance measure

# Table-driven agent

- **Large lookup table, will fail**

- Let $P$ be the set of possible percepts and let $T$ be the lifetime of the agent. The lookup table will contain $\sum_{t=1}^{T} |\mathcal{P}|^t$ entries. Consider the automated taxi: the visual input from a single camera comes in at the rate of roughly 27 megabytes per second (30 frames per second, 640x480 pixels with 24 bits of color information). This gives a lookup table with over $10^{250,000,000,000}$ entries for an hour's driving. Even the lookup table for chess—a tiny, well-behaved fragment of the real world—would have at least $10^{150}$ entries.

# Types of agent programs

We outline four basic kinds of agent program that embody the principles underlying almost all intelligent systems:

- **Simple reflex agents**
  - condition-action rules


- **Model-based reflex agents**
  - internal states

- **Goal-based agents**
  - explicit goals, more flexible


- **Utility-based agents**
  - explicit utility functions, degree of happiness
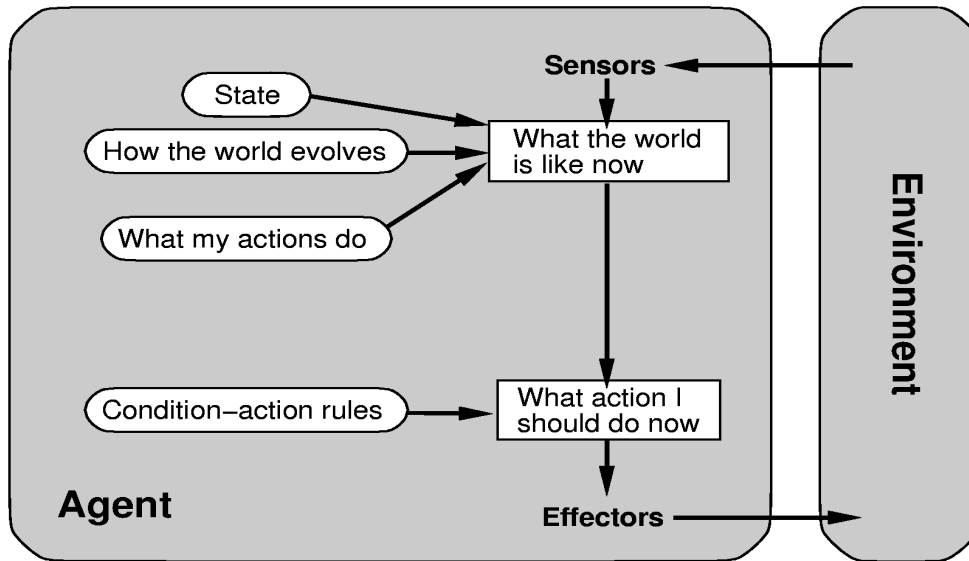
# Simple reflex agents
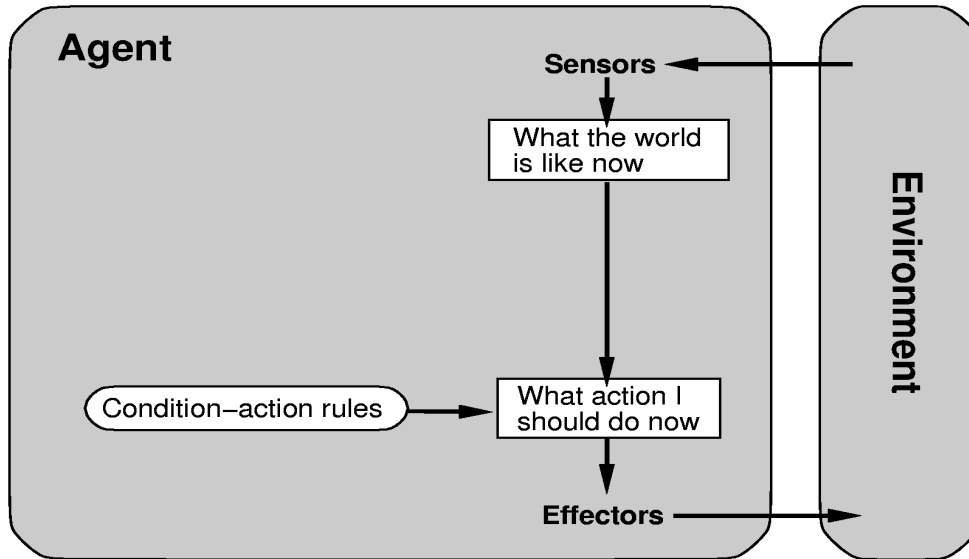


- actions based only on the current percept
- No history
- Only works if environment is fully observable

**function** SIMPLE-REFLEX-AGENT(*percept*) **returns** an action
  **static**: *rules*, a set of condition-action rules

  *state* ← INTERPRET-INPUT(*percept*)
  *rule* ← RULE-MATCH(*state*, *rules*)
  *action* ← RULE-ACTION[*rule*]
  **return** *action*

The agent program for a simple reflex agent in the two-state vacuum environment.
This program implements the agent function tabulated in Figure 2.3.

34

# Model-based reflex agents



- History for partial access of environment
- Internal states
- Update needs two kinds of knowledge:
  - How does the world function without agent
  - What kind of effects does agent have on environment
  - → Model of the world
- Only works if environment is fully observable

```
function REFLEX-AGENT-WITH-STATE( percept) returns an action
    static: state, a description of the current world state
            rules, a set of condition-action rules
            action, the most recent action, initially none

    state ← UPDATE-STATE(state, action, percept)
    rule ← RULE-MATCH(state, rules)
    action ← RULE-ACTION[rule]
    return action
```
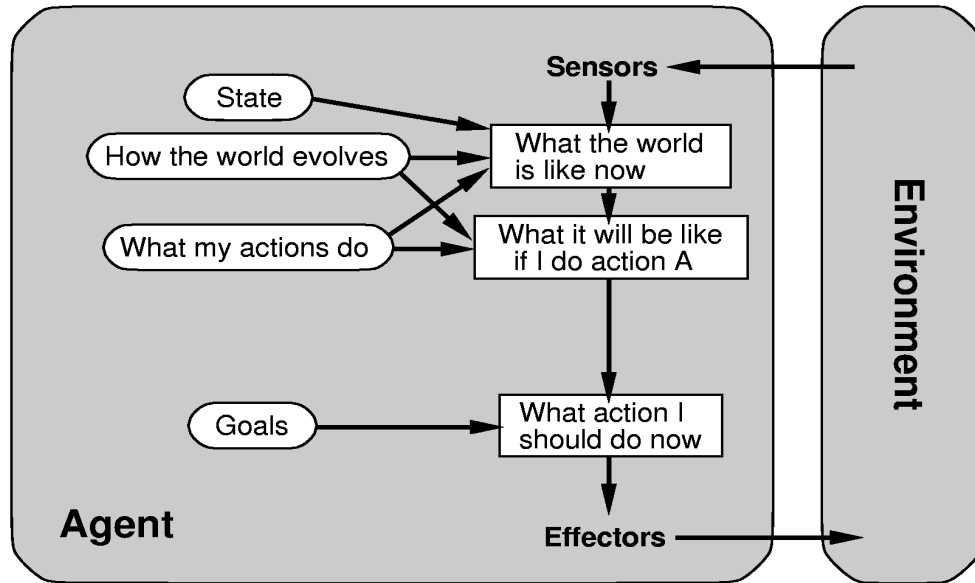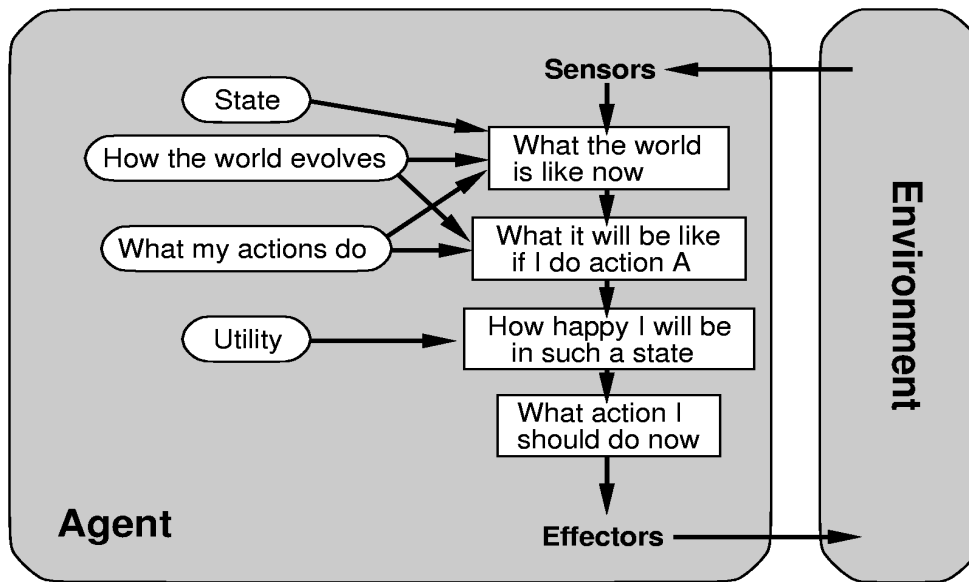
# Goal-based agents



- Model-based and goal-oriented agent
- Goal helps select actions
- Combination of goal and feasible actions
- Selection sometimes easy, most of the time difficult →Search, Planning

# Utility-based agents



- Goal-orientated sometimes not enough
  → e.g. various paths to Rome
- Priority with utility value
- Utility function as mapping between state and a real number
- Advantages with goal conflicts and uncertainty

# Learning agents
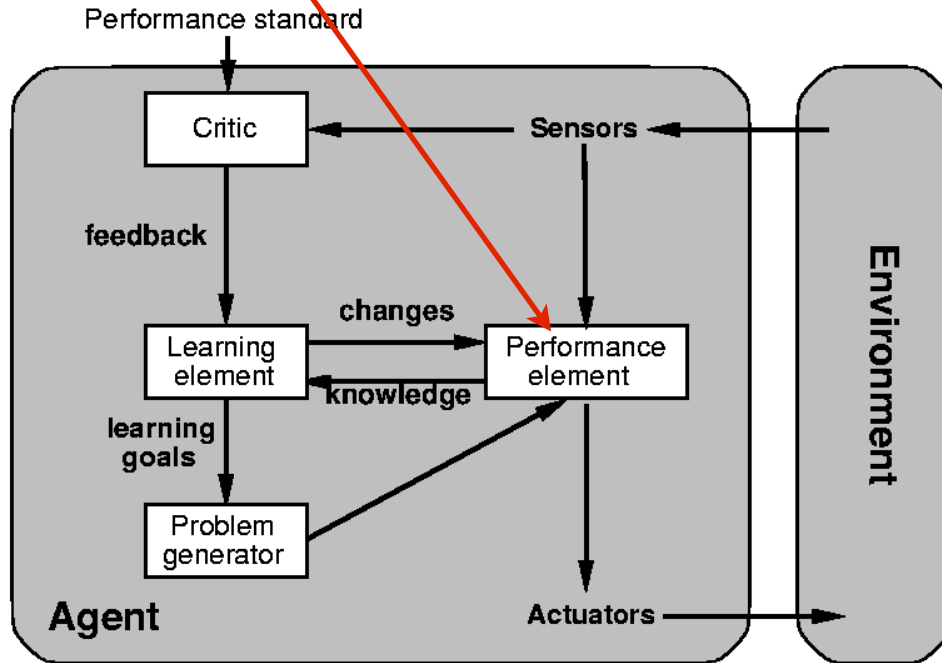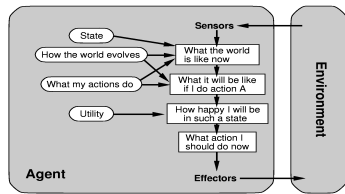


- Learning element for improvement

- Performance element for selection of external actions

- Critique: performance of agent?

- Problem generator for exploration

# Summary

- An **agent** is something that perceives and acts in an environment. The **agent function** for an agent specifies the action taken by the agent in response to any possible percept sequence.

- The **performance measure** evaluates the behavior of the agent in an environment. A **rational agent** acts so as to maximize the expected value of the performance measure, given the percept sequence it has seen so far.

- A **task environment** specification includes the performance measure, the external environment, the actuators, and the sensors. When designing an agent, the first step must always be to specify the task environment as fully as possible.

# Summary (2)

- Task environments vary along several significant dimensions. They may be fully or partially observable, deterministic or stochastic, episodic or sequential, static or dynamic, discrete or continuous, and single-agent or multi-agent.

- The **agent program** implements the agent function. There exists a variety of basic agent program designs, depending on the kind of information made explicit and used in the decision process. The designs vary in efficiency, compactness, and flexibility. The appropriate design of the agent program depends strongly on the nature of the task environment.

# Summary (3)

- **Simple reflex agents** respond directly to percepts, whereas **model-based reflex agents** maintain internal state to track aspects of the world that are not evident in the current percept. **Goal-based agents** act to achieve their goals, and **utility-based agents** try to maximize their own expected "happiness."

- All agents can be turned into **learning agents** that improve their performance through experience.

- Decision-making process through inferences is one of the main ideas of AI and important for successful design of agents, i.e. knowledge representation is important.