Autonomous Agents and Multiagent Systems

Introduction

Francesco Amigoni

Course introduction

A few words about myself

Francesco Amigoni
- francesco.amigoni@polimi.it
- 02 2399-3475
- Room 105, first floor, DEIB building 20
- Wednesday, 14:00-16:00 (formally), by appointment

Courses
- Artificial Intelligence (MSc)
- Autonomous Agents and Multiagent Systems (MSc)
- Intelligent Multiagent Systems (PhD)

Research on agents from mid-1990s
- In the past: architectures for multiagent systems, modelling complex systems, …
- Currently: autonomous decision making in physical environments (agents are robots, appliances, satellites, …)

Course introduction
General information

MSc in Computer Science and Engineering, II year, 5 credits
Open also to students from other MSc programs

Web page:
https://amigoni.faculty.polimi.it/AutonomousAgentsAndMultiagentSystems.html
Also on Beep: news and announcements, links to recordings

Course introduction
Class schedule

Weekly schedule

- Tuesday, 8:15-11:15 (actually 8:30-11:00 with a break), online
- Wednesday, 14:15-15:15 (actually 14:30-15:15), room 25.S.2
- Friday, 14:15-15:15 (actually 14:30-15:15), room 2.1.4

Classes in presence are repeated two times (same content)

1. odd “codice persona” number
2. even “codice persona” number

→ see web page of the course for the schedule (it could change!)

Classes in presence can be attended from remote

All classes will be recorded and links to recordings will be made available on Beep

Course introduction
Teaching assistant

Teaching assistant: prof. Nicola Gatti

To remotely attend a class, you have to join the room of the lecturer teaching that specific class:

http://politecnicomilano.webex.com/meet/francesco.amigoni

http://politecnicomilano.webex.com/meet/nicola.gatti

→ see web page of the course for the schedule

Course introduction
Exam

Exam is written with questions and numerical exercises

→ see examples from past years on the web page of the course

At the moment, I don't know if exams in January and February will be in presence or from remote

Grades are from "rimandato" (failed) to "30 e lode" (31 points in the exam)

You can refuse your grade if you are not satisfied and take another exam

There will be 5 exams this year (2 in January/February, 2 in June/July, and 1 in September)

You can choose which exam(s) you take

No other way to pass the exam or increase your grade: no additional projects, no oral exam, …

Course introduction
Textbook

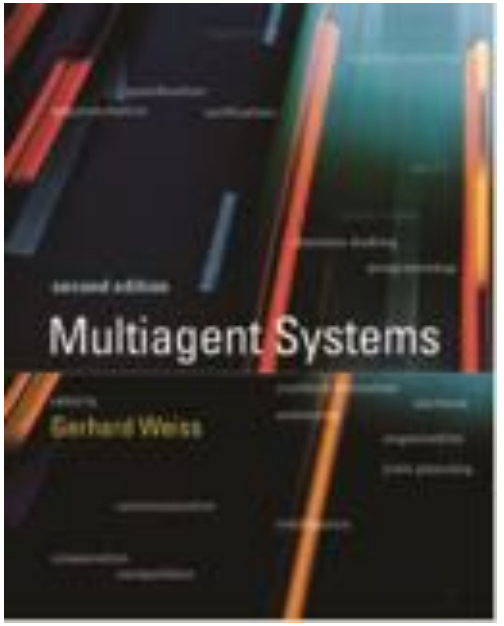
Gerhard Weiss (editor), *Multiagent Systems* (second edition), The MIT Press, 2013, ISBN 978-0-262-01889-0

Available as ebook in Politecnico's library:
https://ebookcentral.proquest.com/lib/polimi/detail.action?docID=3339590

Web site of the book:
http://www.the-mas-book.info

Other material will be made available

→ see web page of the course



Course introduction
Topics

1. Introduction to the concepts of autonomous agents and of multiagent systems
2. Autonomous agents as rational decision makers: architecture for intelligent agents, Markov decision processes
3. Interactions between self-interested agents: short introduction to game theory, coalition formation
   – negotiation, voting mechanisms, auctions
4. Interactions between cooperative agents: decision-theoretic multiagent planning, distributed constraint optimization
   + topics and algorithms
5. Multiagent learning: multiagent Markov decision processes and stochastic games, evolutionary game theory
   + topics and algorithms
6. Examples of real-world applications of agent-based systems

Course introduction
Relationships with (some) other courses

- Artificial Intelligence
- Machine Learning
- Game Theory
- Economics and Computation
- Distributed Systems
- Videogame Design and Programming
- Recommender Systems
- …

Autonomous agents

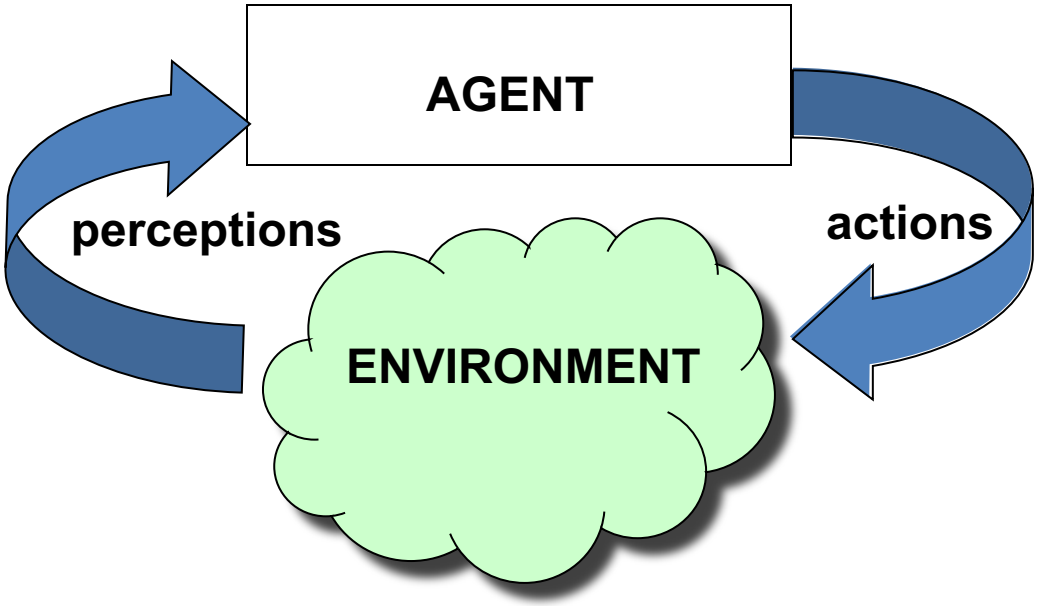
Some systems

1997 - Deep Blue 2009 - Google autonomous car

1998 - Deep Space 1 2011 - Watson 2016 - AlphaGo

2050 - RoboCup (?)

Unifying paradigm: agent

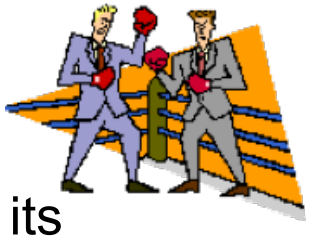


Agent: robot, software program

Environment: real or virtual (software environment)

- Partially observable, non deterministic, dynamic, with other agents, …

Defining an agent



There is not “the” definition of agent

Stuart Russell and Peter Norvig

- “An agent is anything that can be viewed as *perceiving* its environment through sensors and *acting* upon that environment through effectors.”

Pattie Maes

- “Autonomous agents are computational systems that inhabit some complex dynamic environment, sense and act autonomously in this environment, and by doing so *realize a set of goals* or tasks for which they are designed.”

What is an agent?

An agent is characterized by listing some of its *properties*, according to Michael Wooldridge and Nicholas Jennings

- Autonomy
- Reactivity
- Pro-activeness
- Sociality

The examples show some of these properties

Other properties could be added: benevolence, rationality, …

Rational agent

It would be useful to design agent functions that make agents rational, namely that make them do the “right thing”

How can rationality of an agent be defined?

Rational agent

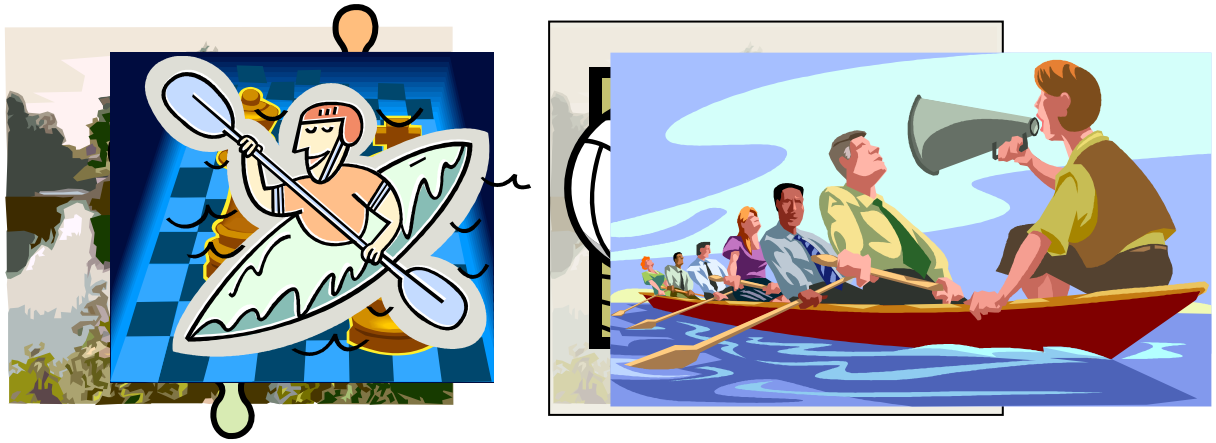
For every possible sequence of perceptions, a rational agent chooses the action that maximizes the expected value of its performance measure, given its knowledge up to that moment

Perception sequence and available knowledge

The perception sequence represents the available knowledge of an agent about its environment

From the standpoint of an agent, its environment can be:

- Completely/partially observable
- Static/dynamic
- Discrete/continuous
- Single agent/multiagent

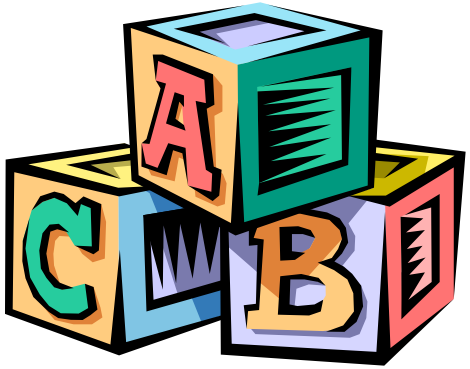


Actions

Through actions an agent can change the state of its environment

From the standpoint of an agent, its environment can be:

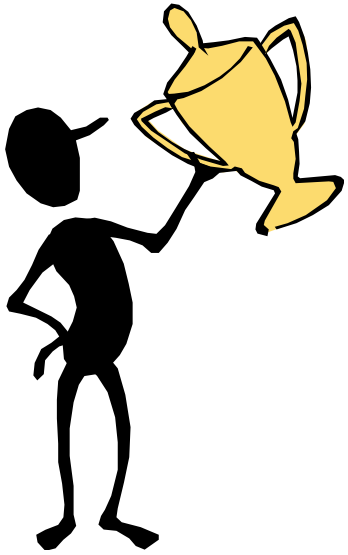
- Deterministic/stochastic



Performance measure

The performance measure is the criterion for evaluating the success of the behavior of an agent

The performance measure is defined by the designer



What a rational agent does?

Rational agent

For every possible sequence of perceptions, a rational agent chooses the action that maximizes the expected value of its performance measure, given its knowledge up to that moment

A rational agent is not omniscent

A rational agent is not clairvoyant

A rational agent can explore to acquire new information, can learn, …

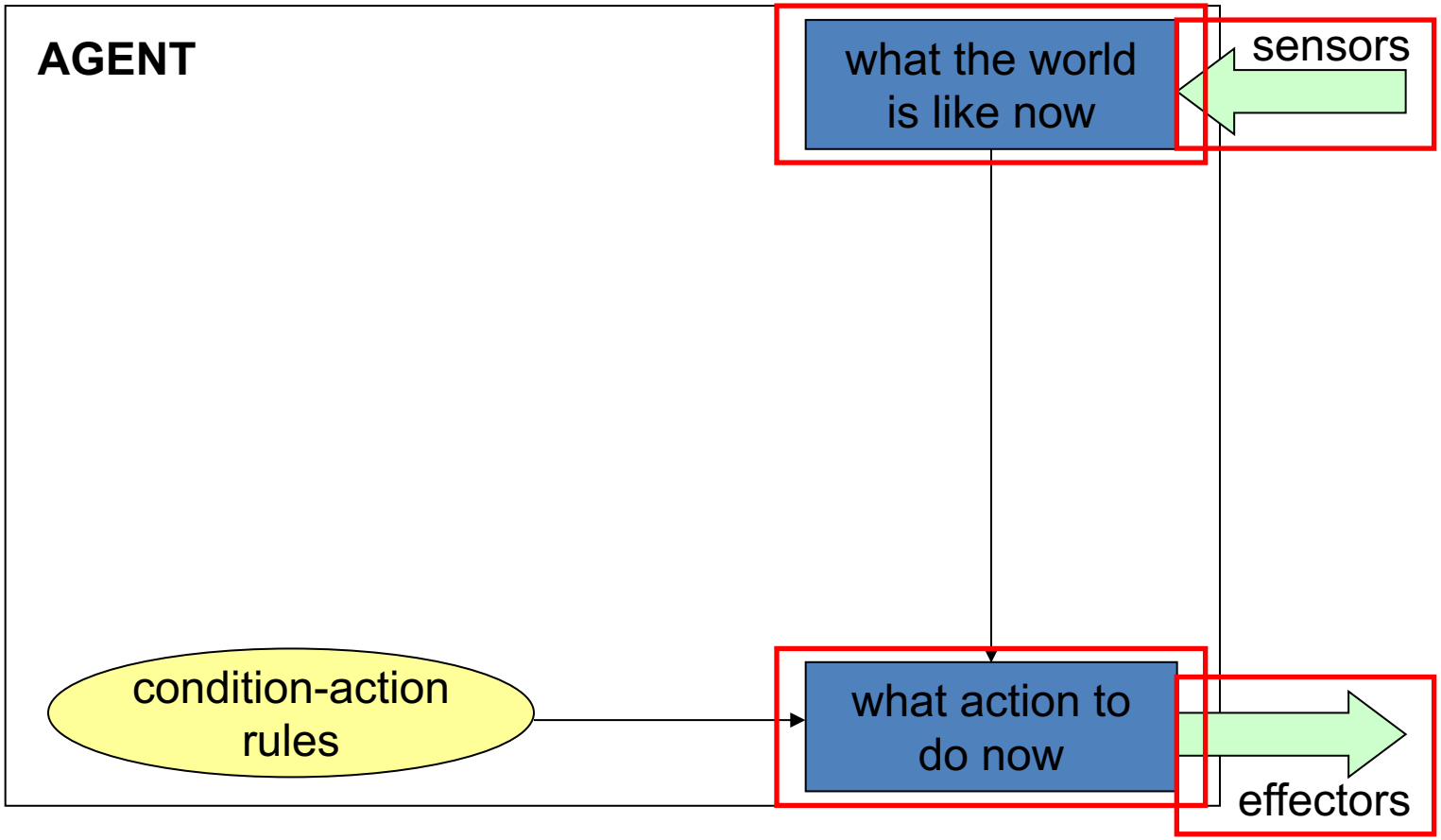
Architectures for agents

Agent architectures can be classified in four basic types:

- Simple reflex agents
- Reflex agents with state
- Goal-based agents
- Utility-based agents

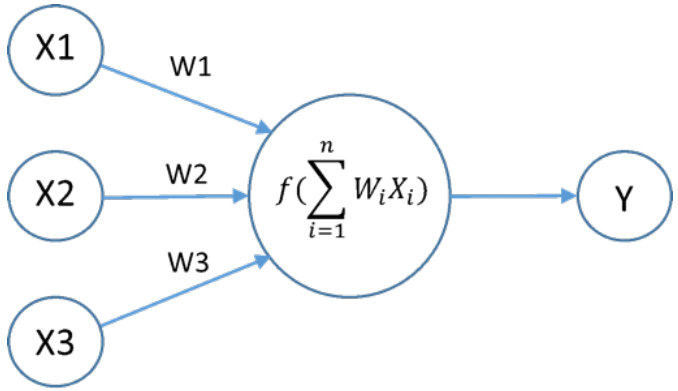
All these types of agent can also learn

Simple reflex agents



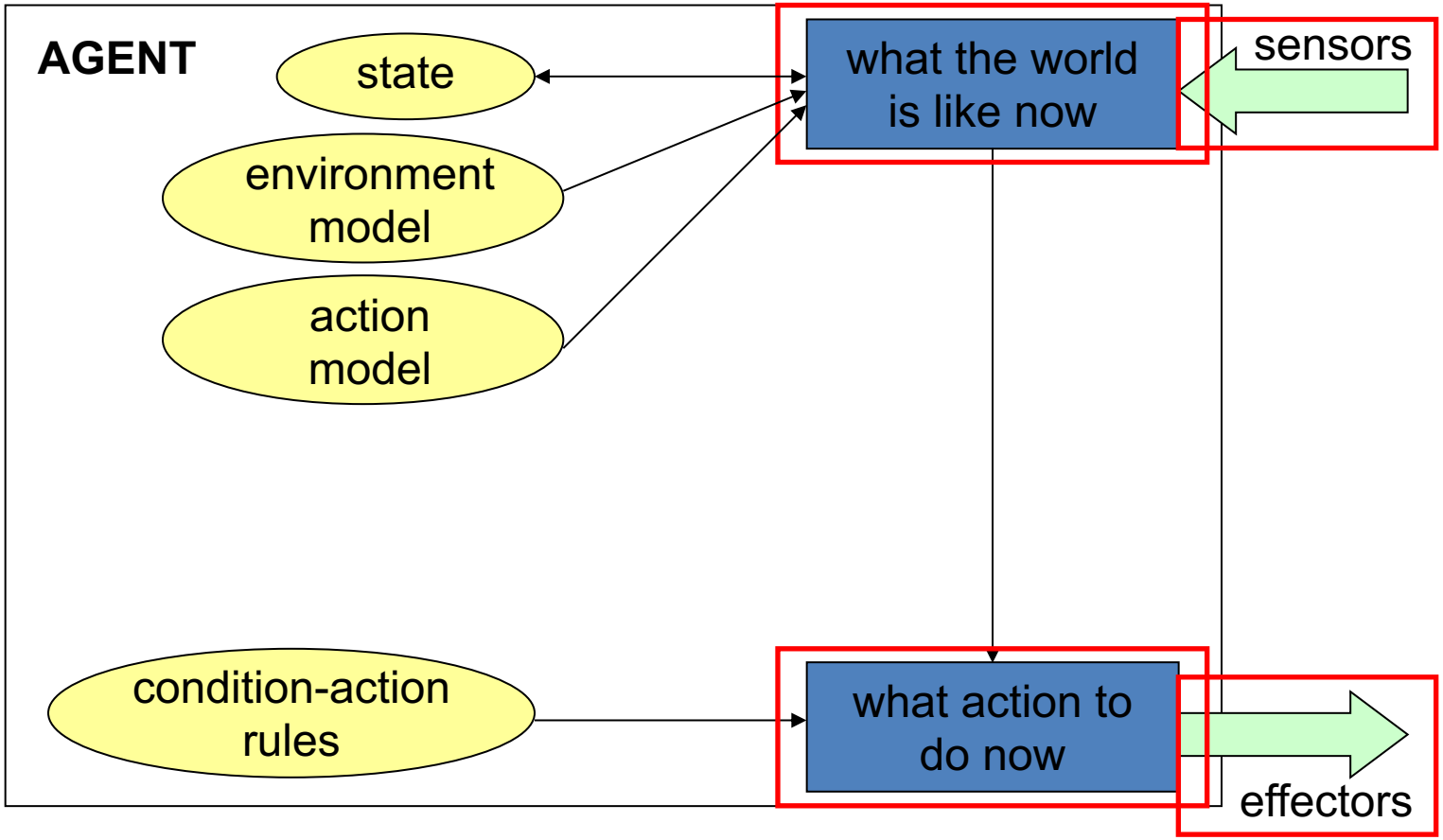
Examples of simple reflex agents

First versions of Roomba



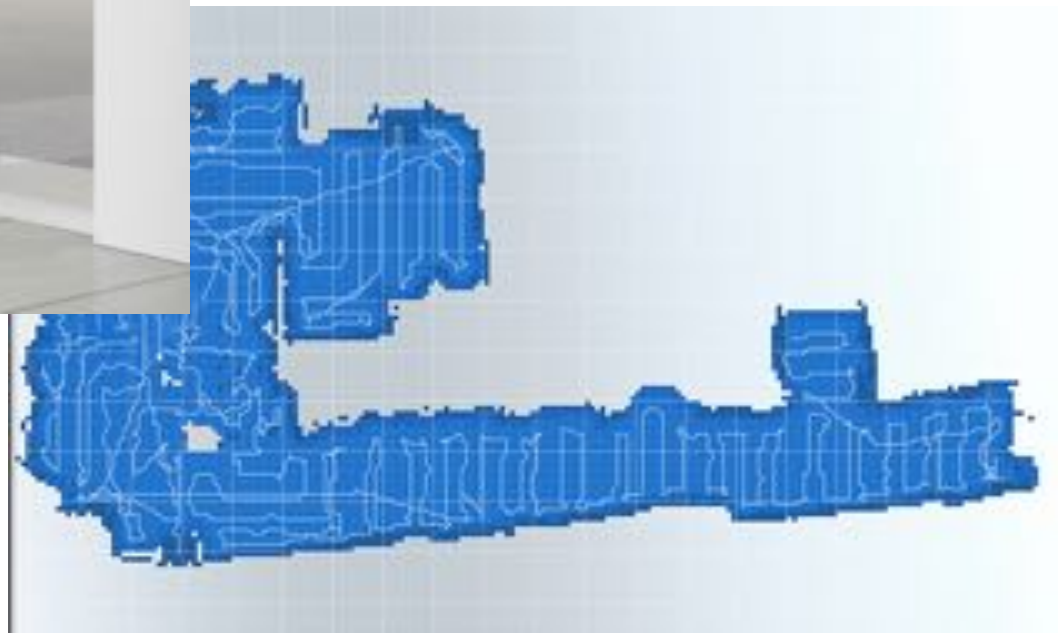
Neural networks

Reflex agents with state

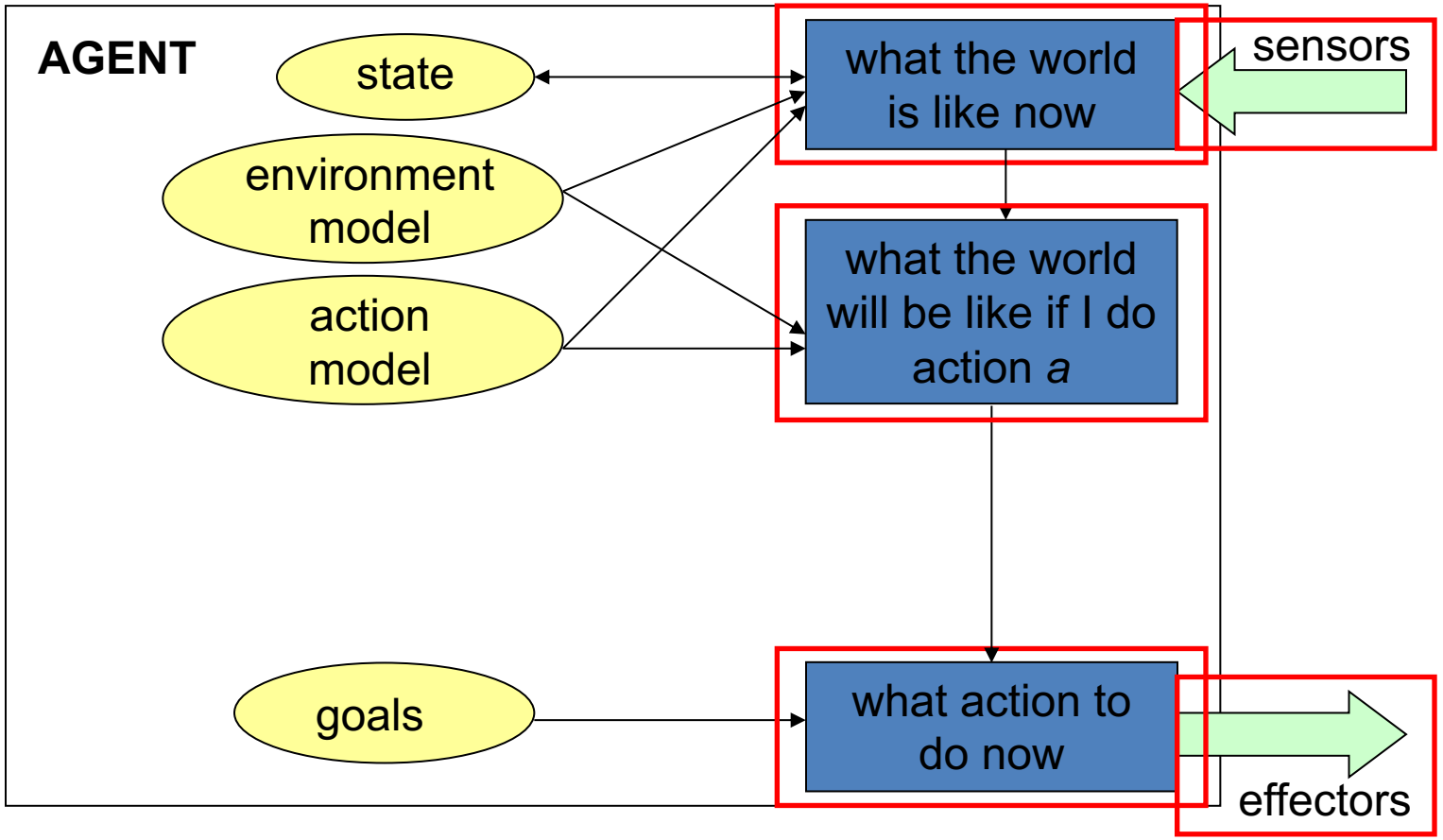


Examples of reflex agents with state

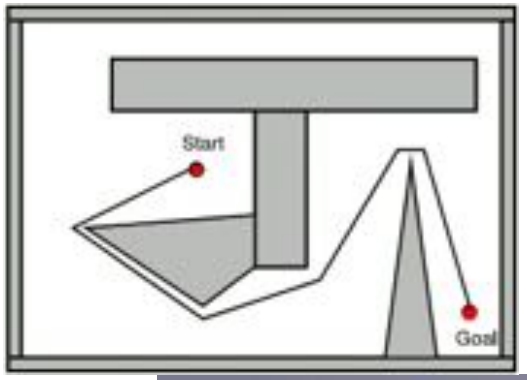
Dyson 360 Eye

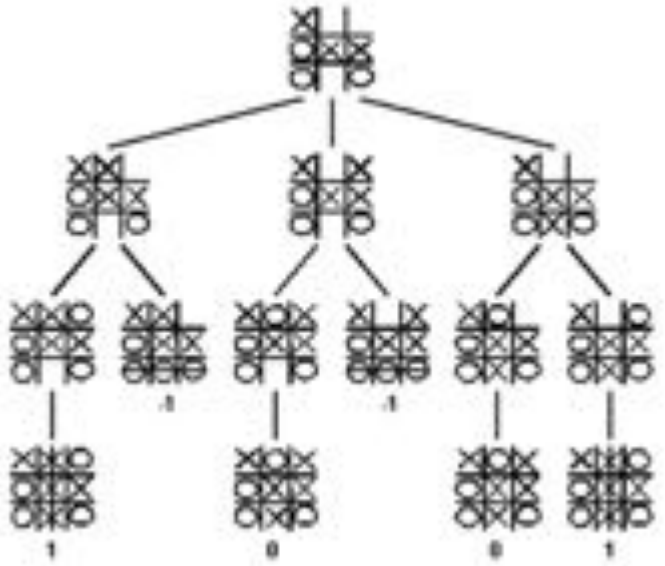


Goal-based agents



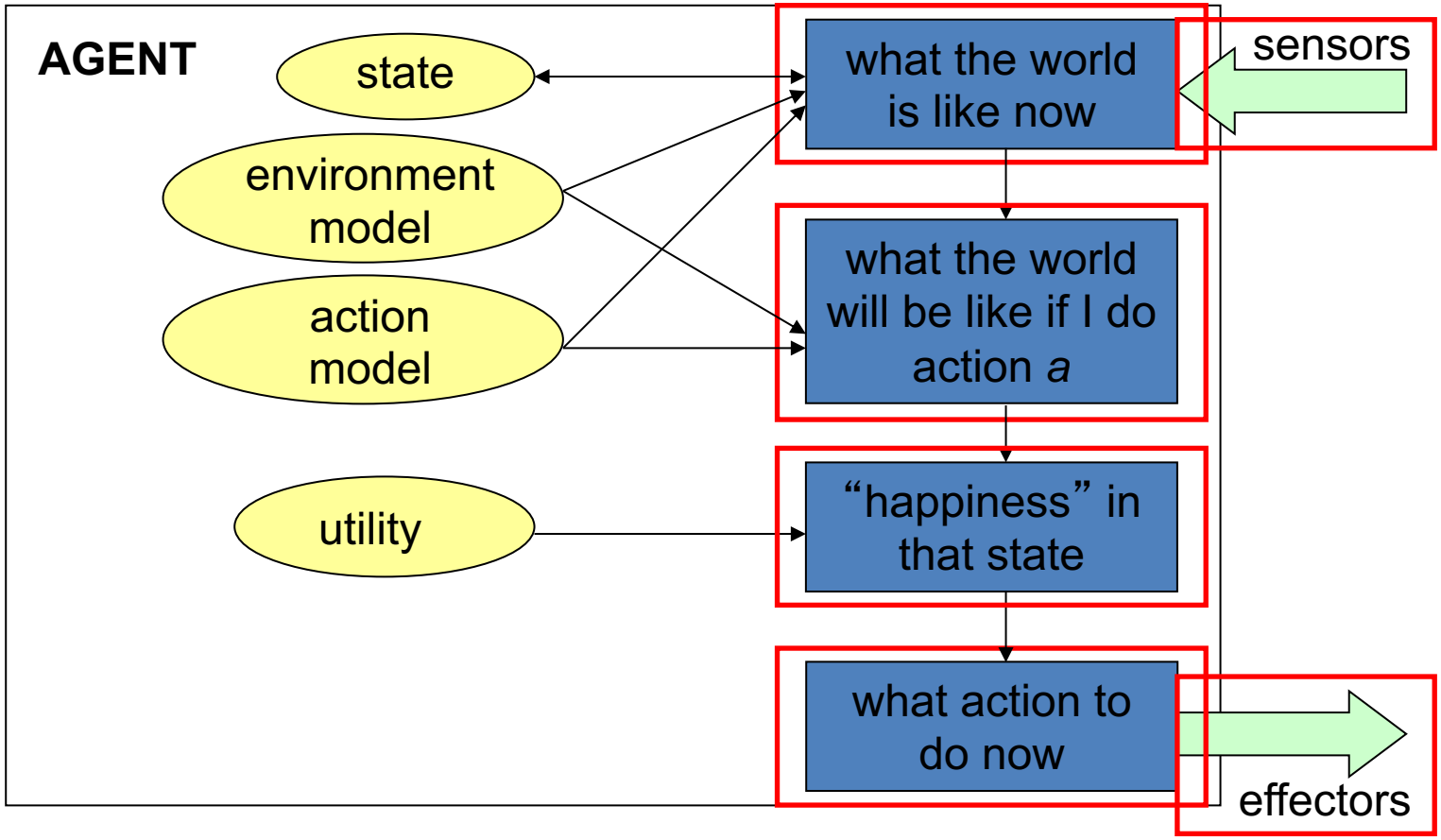
Examples of goal-based agents



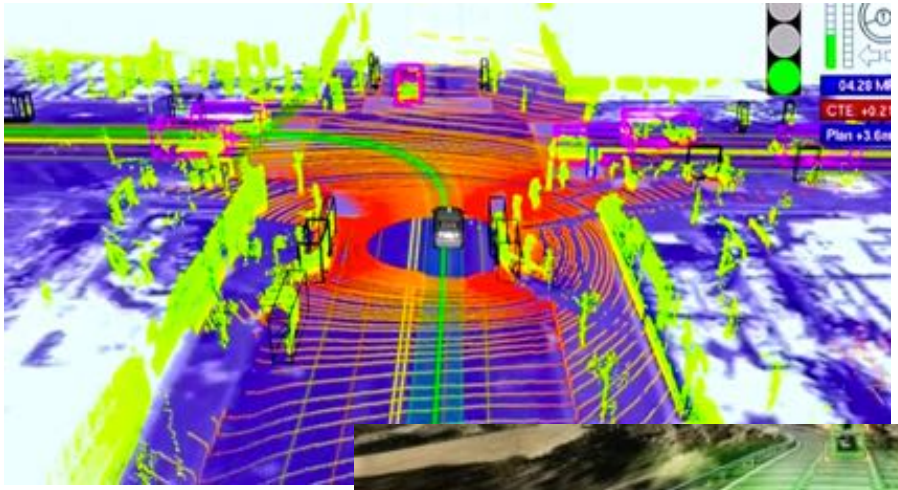


Action (movement) planning

Utility-based agents



Examples of utility-based agents



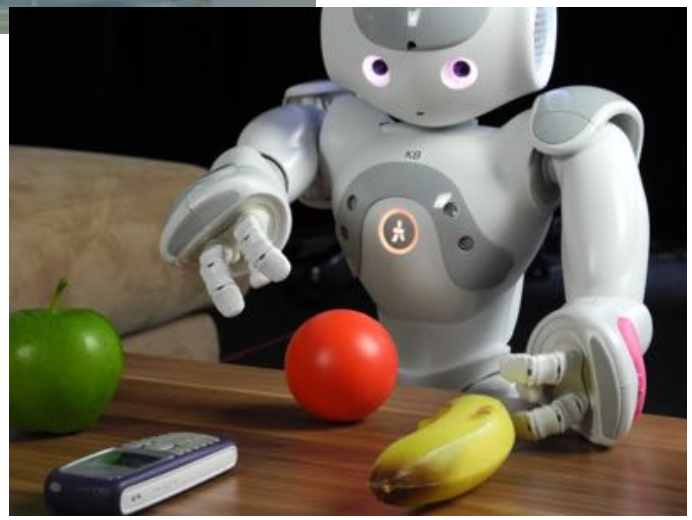
Decision on the best action (movement)

Learning agents

All the agents presented before can improve their performance with learning

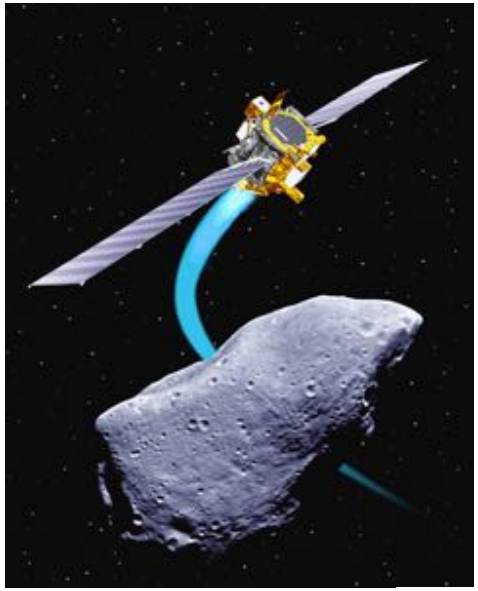
Every component of the decisional process of an agent can be modified in order to perform better

Examples of learning agents



Multiagent systems

Interaction with the environment...



...and with other agents

Multiagent systems

A *multiagent system* is a system in which a number of agents interact with each other (and with the environment)

From the above examples, to successfully interact agents must cooperate, negotiate, coordinate, …

In general, different agents are pursuing different goals; but sometimes a global goal for the system can be identified

- Cooperative systems (“benevolent agents”)
- Competitive systems (“selfish agents”)

Types of interactions with other agents

Coordination
Cooperation
Negotiation
Elections
Task allocation
Distributed optimization
Emerging behavior
...



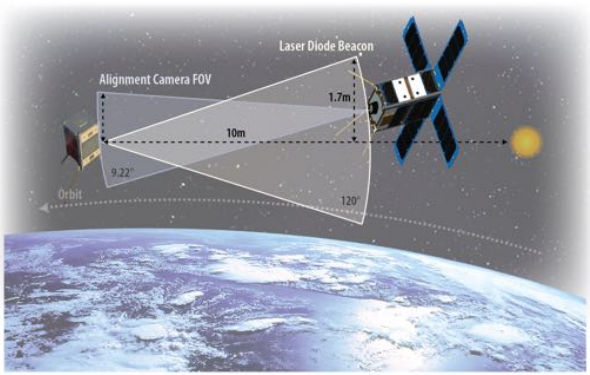




Coordination

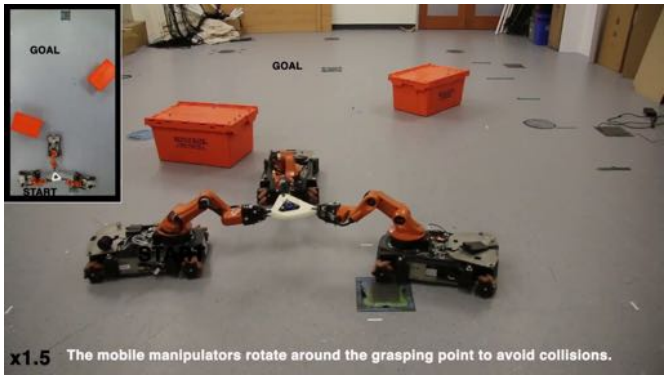
Selecting actions that do not interact with negative effects

Pre-programmed

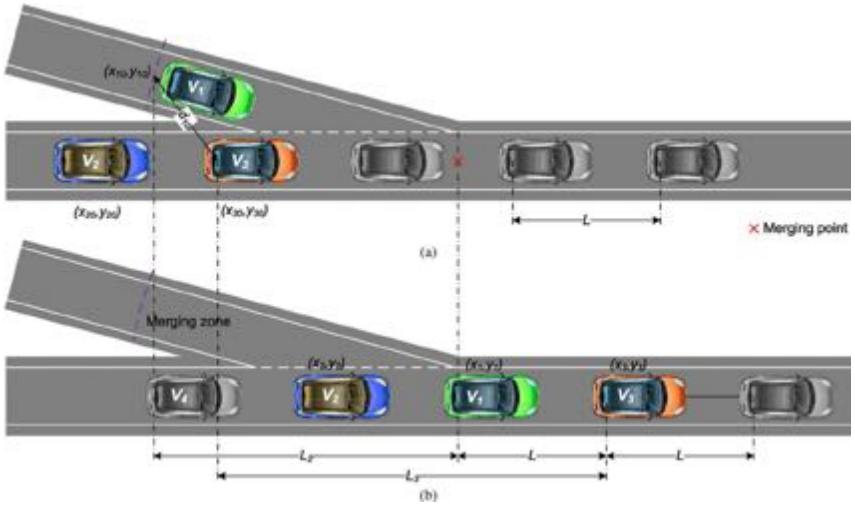


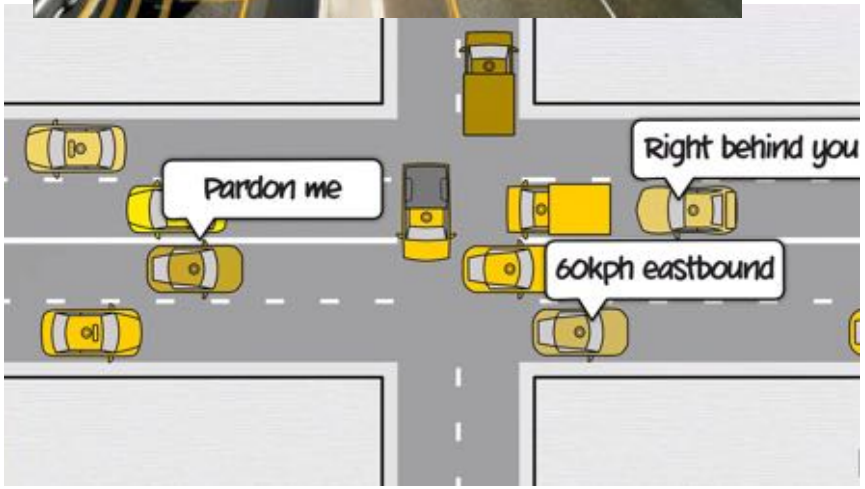
Convention-based

Communication-based



Coordination: Modeling other agents





Coordination: Centralized vs. distributed

Centralized

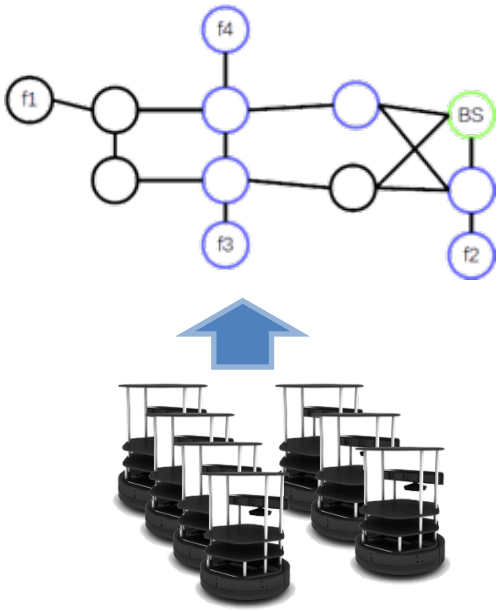
- global communication
- optimal solutions
- tight coordination
- computationally intensive
- bottleneck

Distributed

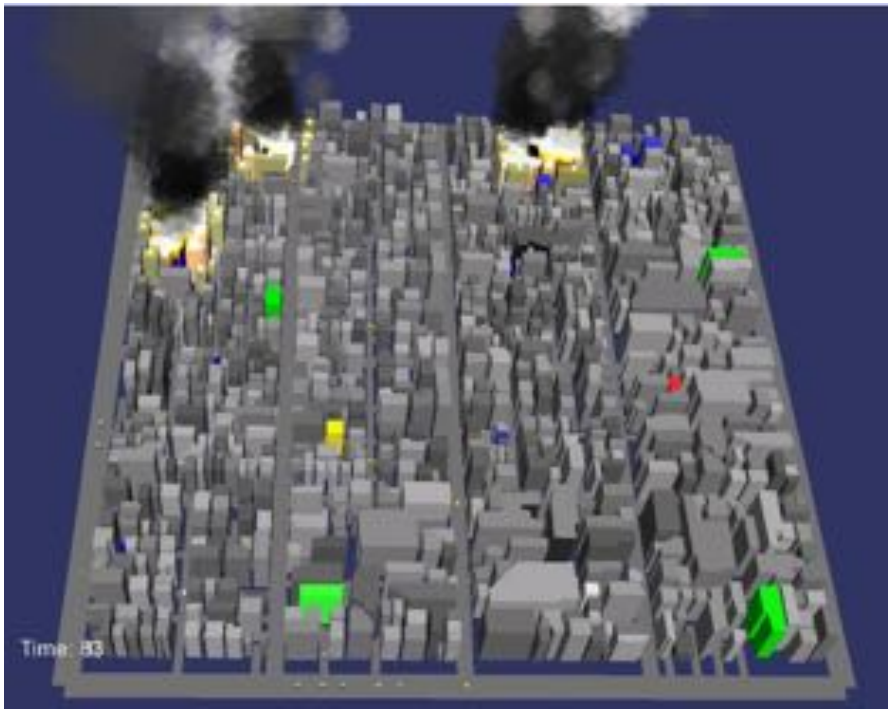
- local communication
- sub-optimal solutions
- loose coordination
- computationally cheap
- robust

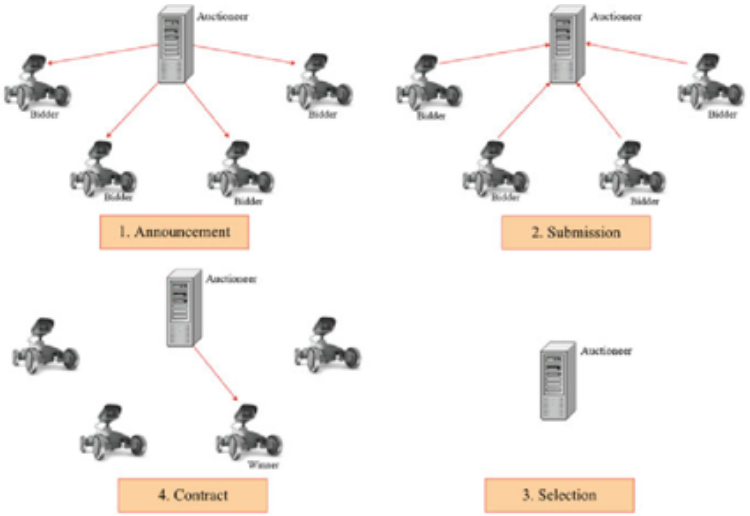
Coordination: A centralized method

1. Find the optimal positions to occupy to perform a task while preserving connection
   - Exact algorithms
   - Approximate algorithms (e.g., Steiner trees)
2. Assign robots to positions
   - Exact algorithms (e.g., Hungarian algorithm)



Coordination: A decentralized market-based method

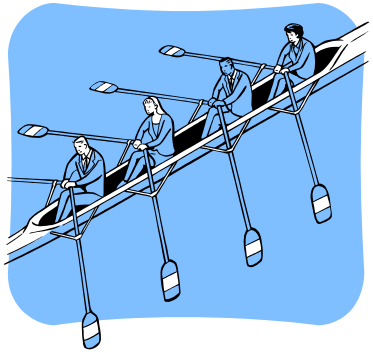




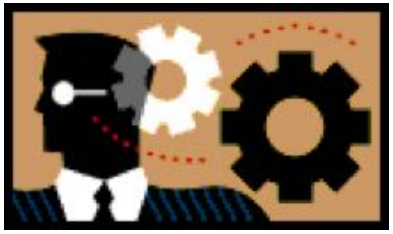
Two key problems

How to design individual agents able to act autonomously in order to reach a goal (*micro problem*, *agent problem*)

How to design systems in which more agents interact in a “useful” way (*macro problem*, *society problem*)



Multiagent systems as a design paradigm



Modeling and design of distributed complex systems

Agent architectures

- Reactive, deliberative, hybrid

Interaction mechanisms

- Planning, coordination, matchmaking, auctions, negotiation strategies, …

Agent-oriented software engineering and agent-oriented programming

- New abstractions: autonomous agent, cognitive agent, agent society, …

Multiagent systems as a programming technology

Agent communication languages

- FIPA ACL, KQML

Ontologies

Agent development frameworks

- JADE, ZUES, …

My personal view on agents: More paradigm than technology

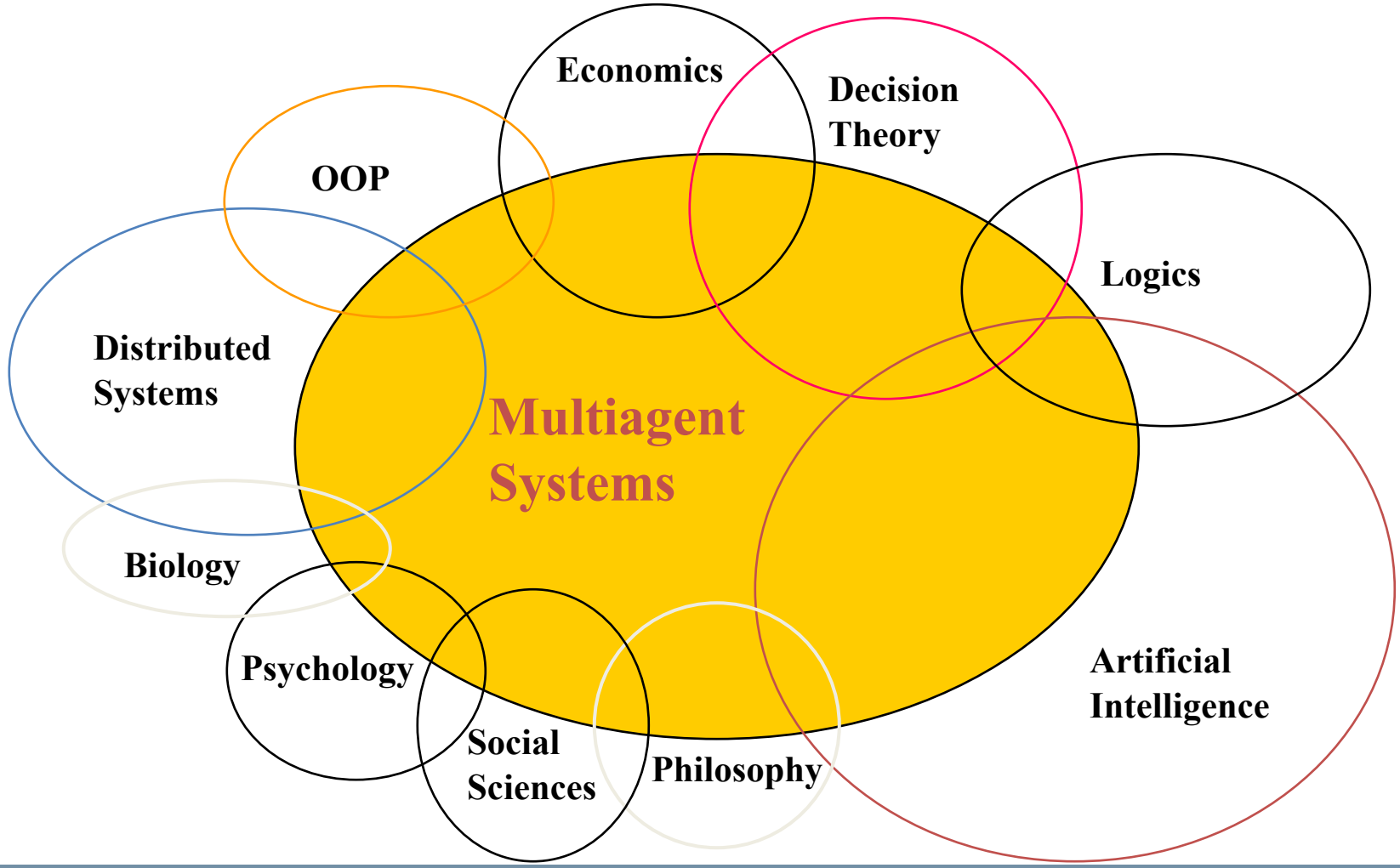
Agents are a very useful abstraction to model a number of entities

- Web services
- Nodes in a grid
- Robots
- People
- …

Multiagent systems provide “general” techniques, which are rather independent from the possible implementation

- Distributed optimization
- Task allocation
- Coordination
- …

Relations with other disciplines



Some problems...

Agents vs. objects
- An object's method is *invoked*, an agent is *requested* to execute an action

Agents vs. distributed systems
- Agents can be *self-interested*, their interactions being of the type studied in economics

Agents vs. artificial intelligence
- An agent includes different aspects of "intelligence": planning, learning, ...
- Artificial intelligence usually does not consider social interactions

Agents vs. game theory
- Game theory is usually only *descriptive*

Publication venues for agents

Specific and transversal venues

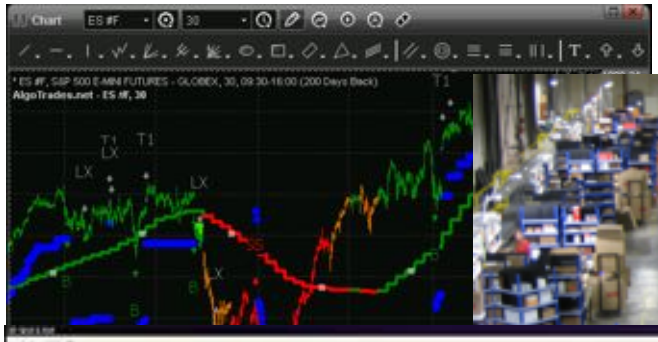
Journals

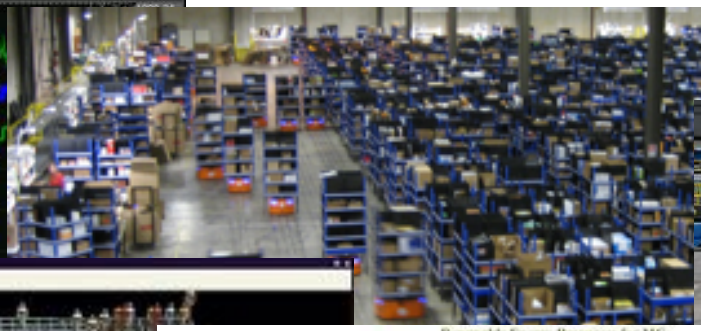
- Journal of Autonomous Agents and Multi-Agent Systems (JAAMAS)
- Artificial Intelligence Journal (AIJ), Journal of AI Research (JAIR), Journal of Machine Learning Research, IEEE Transactions on Robotics, Autonomous Robots, Robotics and Autonomous Systems, …

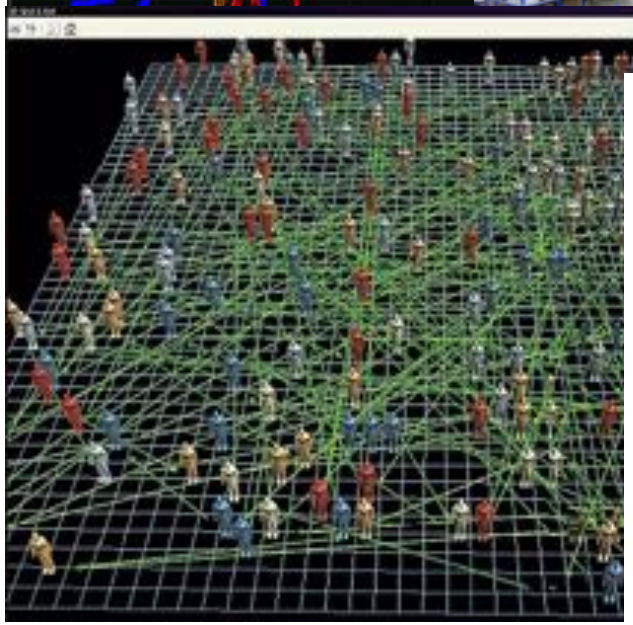
Conferences

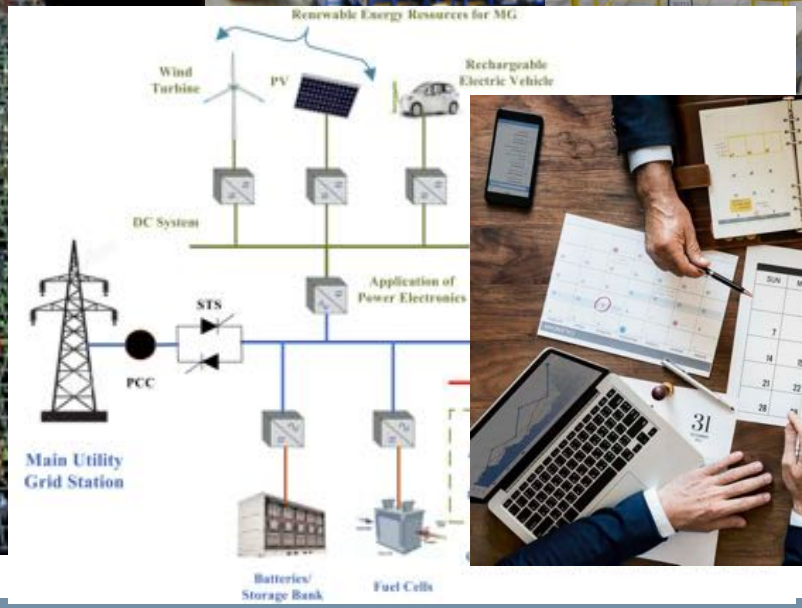
- Autonomous Agents and Multi-Agent Systems (AAMAS)
- AAAI, IJCAI, …
- ICRA, IROS, RSS, DARS, MRS, IAS, …

Some real-world applications
Stand-alone systems









Topics of this course

In this course we will focus on some (not all!) aspects of *interaction* in multiagent systems

- Concepts
- Problems
- Algorithms
- Programming (only pointers, it is up to you to code if you like)

What we leave out

Logics

Argumentation

Trust and reputation

Software engineering aspects: specification, validation, …

Agent-based simulation

Some self-interested interactions

…