Computer Science, Graduate Center, CUNY

# Modeling Adaptive Autonomous Agents

## by Pattie Maes

Jinzhong Niu

September 22, 2005

# Outline

- Definition
- Guiding Principles
- Examples
- Overview of State of the Art

# What is an autonomous adaptive agent?

**agent** a system that tries to fulfill a set of goals in a complex, dynamic environment

**properties**

- autonomous — making decisions itself
- adaptive — able to improve over time with experience
- effective — successful at eventually achieving its goals

# Forms of agents

In terms of the types of environments it inhabits, an agent can be:

- physical robots, e.g. robot soccer players
- software agents, e.g. trading agents
- virtual physical robots, e.g. Lipson's robots in the simulated environment

# Traditional AI vs. autonomous agent

- isolated and often advanced competences vs. lower-level competences
  - top-down AI vs. bottom-up AI
  - e.g. a medical diagnosis system and a garbage collecting robot
- closed (typically through a human operator) vs. open (directedly situated in the environment)
- no time-constraints and one problem at a time vs. acting in a timely fashion and multiple goals simulteneously
- static knowledge structure vs. dynamic behavior-producing modules
  - knowlege-based AI vs. behavior-based AI
- once for all vs. developmental

# Architecture for modeling autonomous agents

- autonomous agent research
  - principles and organizations
  - tools, techniques, and algorithms
- a table specifying which architecture are the most simple solution for a given class of agent problems?

# Outline

- Definition
- Guiding Principles
- Examples
- Overview of State of the Art

# Modelling agents in the context

- Build various functions in an integrated way rather than independently
- Take advantage of the world where the agents are situated
- Use the ability of learning to avoid the requirement of a perfect solution at the beginning (take time for incremental improvement)
- Seek help from the peers

# Interaction dynamics can build complexity from simple components

- Simple internal modules that work together can lead to emergent functionality.
  - e.g. a wall-following robot
- Simple atomic capabilities together with feedback mechanisms can produce complex behaviors.
- Agents with simple behaviors can compose a social system that can exhibit advanced structures or functionality.
  - e.g. markets involving primitive trading agents

Benefits: more robust, flexible, and fault-tolerant than programmed, top-down organized complexity

# Outline

- Definition
- Guiding Principles
- Examples
- Overview of State of the Art

# Examples

- a mobile robot
  - traditional AI approach
  - autonomous agent approach
- an interface robot
- a scheduling system

## Shakey: the ‘first electronic person’

Developed in 1969 by the Stanford Research Institute, Shakey was the first fully mobile robot with artificial intelligence. Shakey was named after its rather unstable movements.

- perception module
- environment model
- planning module
- execution module

# The agent-based approach

- competence modules
  - recognizing and going through doors
  - wall following
  - obstacle avoidance
  - ...
- a simple arbitration scheme

# Outline

- Definition
- Guiding Principles
- Examples
- Overview of State of the Art

# Subproblems of modeling autonomous agents

- action selection
  - what actions should an agent take next so as to optimize the achievement of its goals
- learning from experience
  - how to improve the performance of action selection

# Action selection — Difficulties

- resource limitations
- possibly incomplete and inconsistant information
- dynamic, unpredictable environment
- time-varying goals
- ...

# Action selection — Criteria

- goal guided
- real-time
- robust
- developmental
- adequately good
- ...

# Action selection — Progress made

- Hand-built, flat networks requires the designer of the agent to solve the action selection problem from scratch by designing a set of reflex modules and arbitration mechanism.
    - hard to apply for new agents
    - hard to scale up
    - unable to deal with time-varying goals
- Compiled, flat networks automates the design of the arbitration mechanism.
    - requiring a specification of the goals and desired behaviors
    - limited capability
- Hand-built, hierarchical networks organize different competence modules in a more hierarchical way.

# Action selection — open problems

- nature of goals (what kinds, how they change over time)
- scaling up by evolution or learning
- reusability
- understanding interactions' contribution to emergent behaviors
- command fusion
- deadlock in decentralized architecture
- relationship between perception and action

# Learning from experience — The problem

- aims to improve the action selection over time
- why learning needed?
  - hard to program
  - not realistic to reprogram due to break-down or environment change
- meaning of improvement
  - time or number of actions needed to reach the end goal decreases
  - average or discounted expected reinforcement received over time increases

# Learning from experience — The problem (cont.)

- what action selection mechanism to be adopted
- how to learn (what to explore)
  - e.g. a fire-escaping mobile agent
- how to balance between exploration and exploitation (how to explore)

# Learning from experience — Progress made

- what to learn
  - arbitration network among different actions
  - composite actions
- architectures
  - Reinforcement learning systems learn how to map situations to actions maximizing the accumulated reward.
  - Classifier systems are a special case of RL, which learn to evaluate classifiers that is used to choose actions.
  - Model learners learn how actions map situations into other situations.

# Learning from experience — Open problems

- scaling up
- more reasonable exploration strategies
- learning the set of primitive actions
- learning to perceive
- comparing individual learning with evolution
- ...

# Thank you!