Artificial Life

3 – Multi-agent systems

Stefan Bornhofen
Vincent Gardeux

Course Outline

1) Complexity, Emergence & CA (sb)
2) Fractals and L-systems (sb)
3) Multi-agent systems (vg)
4) Swarm intelligence (vg)
5) Artificial evolution (vg)
6) Virtual Ecosystems & Perspectives (sb)

Social Systems

- In nature, all intelligent entities develop social behavior : humans, animals, …
- If you observe the human hierarchical system, you can see humans as entities that interact with other humans, other entities or with their environment.
- In order to generalize these systems, we call **agents** these kind of social entities.
- These generalized social systems are so called **multi-agent systems**

Intelligent Agent (Software Agent)

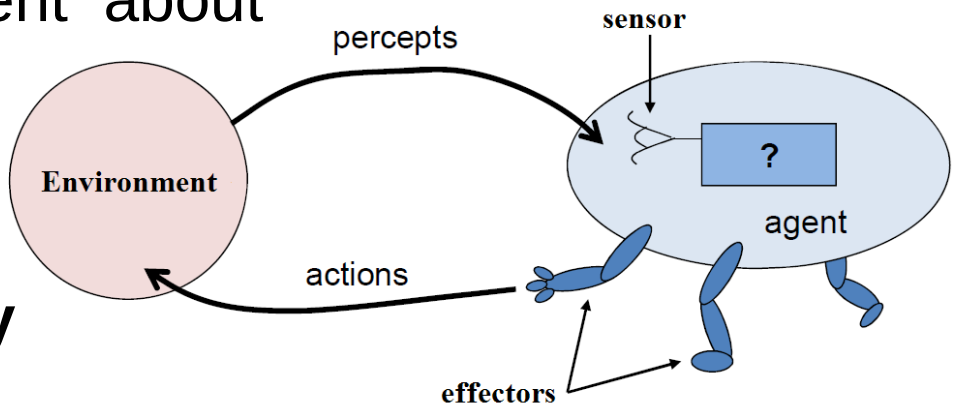
What's an agent?

- Several definitions have been made to define precisely what an agent is, and this concept has been develop in different ways in literature.

- **Agent** : **It's an entity that can sense and effect in an environment in order to reach its goals, depending on its abilities, knowledge and beliefs**.

Intelligent Agent Model

- Agent = entity that can percepts its environment with its **sensors** or effects in it with its **effectors**.
- Each agent has **autonomous** actions, it decides by itself what it will does.
- Its sensors inform the agent about what it **perceives** from its environment and its effectors can do several actions in order to **modify** that environment.



- These actions are limited and gathered in a **set of available actions** with **preconditions** on these actions.

Intelligent Agent Model

- A software agent can learn by its experience. So it has a set of **knowledge** acquired by experience or when he was born.
- We can also modelize a set of **beliefs** on its current environment, directing its future actions. A new experiment can update this set of beliefs.
- An agent may also have **personal ambitions** it tries to achieve in parallel with the global goal of all agents in a multi-agent system.

Intelligent Agent Model

- Another definition : « *An intelligent agent is a computer system capable of* ***flexible*** *autonomous action in some environment* » [Woodridge, 2002]
- Flexible means :
  - Reactive : It can percepts its environment and respond quickly if it changes.
  - Pro-active : Not only the environment motivate its response but its own goals or utility function makes it to take initiative.
  - Social : In a MAS, it has the ability of interact with other agents (and possibly humans) in order to achieve its own goals or even help others to complete their.

Agents can learn?

- Agents can learn from their experience or by other agents experience via the communication system.
- Agents performance may be measured by an **utility function** depending on the problem to solve.
- They are adaptive in the way that they **optimize** this utility function in order to perform their task better.
- The M-A model can include **rewards** for agents which perform well. It ‘s called **reinforcement learning**.
- Reinforcement learning algorithms attempt to find a **policy / strategy** that maps **states** of the world to the actions the agent ought to take in those states.

Different kinds of agents

Agents can be categorized in 2 main types:

- **Reactive Agents** :
  - Purely reactive agents : agents with simple reflexs.
  - Agents with state : agents that keep a state of the world.
- This kind of agents only react to changes in the environment. These agents have no plannification or deliberation process. They can act quickly.

- **Cognitive Agents (deductive reasoning agents) :**
  - Agents with goals : plan their actions thinking about their goals
  - Agents using utility function : using detailled goals
  - BDI Agents : (Beliefs, Desires, Intentions) Architecture proposed by Wooldrige in 1999.
- This kind of agents deliberate to choose their actions. It can be a schedule set up by the agent in order to complete its task.

Different kinds of agents

Purely reactive agent :

- Agent which decide what to do without reference to their history
- It bases its decision making entirely on the present, with no reference at all to the past.
- It can be modelized by a set of *if…then* rules.

Example : A thermostat is a purely reactive agent

Agent is { *off* if temperature OK
on otherwise

Different kinds of agents

Agents with state:

- Agent's sensors don't give it a global view of the world
- Agent can maintain information about the world in order to distinguish 2 situations which are different but have similar perception for the agent.

Example : A control agent in a military boat

If the agent's sensor (sonar) detect a missile and after that the signal disappeared due to an obstacle.

Agent must save :

- position,speed and direction of the missible at t
- State of the military boat after its own actions (speed, position, direction, …)

Different kinds of agents

Agents with goals:

- In several situations, agents can be motivated by general or particular goals to achieve.
- Agent will combine information on its goals with information on its current action in order to select actions that will lead to reach its goals.
- In complex situations : several actions have to be followed in order to reach a goal, the agent should be able to schedule its actions.

Example : A control agent in a military boat

If the agent goal is to reach La Rochelle, it has to make the boat dodge the local obstacles or missile to reach its goal.

Agent must process all the local actions but keeping in mind its main objective

Different kinds of agents

Agents using utility function :

- Sometimes, goals definition is not detailled enough to compute a good behavior.
- A goal is a state that an agent can satisfy or not (boolean)
- With an utility function, we have a numeric value to compare decisions. The agent can prefere an option to another.

Example : A control agent in a military boat

If certain paths are more dangerous and others are shorter, which one must be chosen? Its unique goal is to reach the destination.

Agent with goals can't choose from several objective to achieve if they are conflicting.

Agent vs Object

- Agent : **autonomous** entity that can interact with its environment.
- Object : **passive** entity which has a particular state and on which we can do several operations.
- Agent is at a higher abstraction level than an Object => It may be composed of several Objects.
- Unlike objects, which are defined in terms of *methods* and *attributes*, an agent is defined in terms of its **behavior**

Multi-Agent Model

- « *An agent that cannot interact with other agents is an isolated entity, an information processing system without adaptive characteristics* » [Ferber 95]
- In a multi-agent system, several agents try to reach a given goal.
- That implies **interactions** between agents : cooperation, coordination, negotiation, competition,…
- In order to manage these interactions with other agents, two different models can be used :
  - An agent can observe other agents and create beliefs on their behavior or even mimic them.
  - An agent is able to **communicate** with others agents via a given **communication system**.

Multi-Agent Model

Communication between agents can be made in different ways :

- Simple : using a synchrone or asynchrone message transmitted from the **speaker** to the **hearer**. The protocol of interpretation of the message must be set previously (distance of hearing, encoding/decoding, …)

- Complex : using an Agent Communication Language (ACL).

Already existing frameworks

- Agent-Oriented Programming is currently a well-known field, so lots of protocols and generalized systems have already been tested and implemented :
  - Agent Communication Languages : KIF, KQML, FIPA-ACL, …
  - Agent Coordination : CDPS, MACE, …
  - Development Platform : CORMAS, JADE, …

Applications

- There are a huge number of application areas, we can extract 3 major categories :

  - Simulation of complex phenomenon : particle physics (agent = elementary particle), chemistry (agent = molecule), cell biology (agent = cell), ehtnology (agent = human), ...

  => The aim is principally to validate a given model.

  - Problem solving : distributed artificial intelligence, autonomy of the agents allows the solving system to be adaptive.

  - Program conception : agent-oriented programmation, contrary to an object an agent can take initiative and refuse to obey a request.

Applications

Example Fields :

- Social Science
  - Insect societies, group dynamics in fights, growth and decline of ancient societies, group learning, spread of epidemics, civil disobedience
- Economics
  - Stock market, self organising markets, trade networks, consumer behaviour, deregulated electric power markets
- Ecology
  - Population dynamics of salmon and trout, land use dynamics, flocking behavior in fish and birds, rain forest growth
- Political Sciences
  - Water rights in developing countries, party competition, origins and patterns of political violence, power sharing in multicultural states

Applications

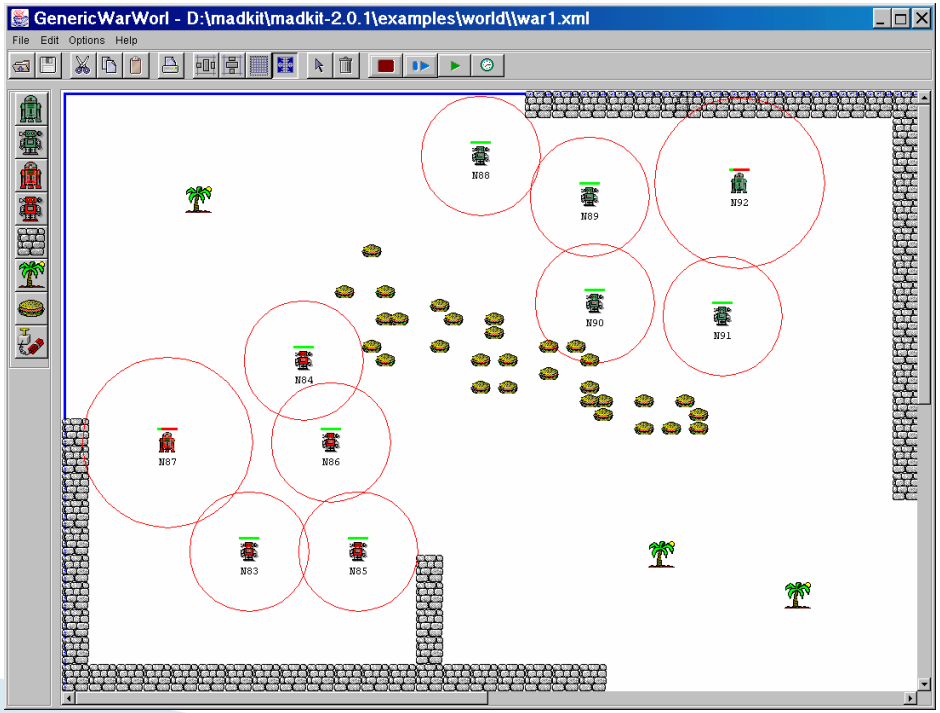
On the Web :

- Documentary agents
  - Used to search the Web for new informations. They can collect information automatically. => Increase the database of potential query results.
- E-Commerce : Buyer and Seller Agents
  - Compare the prices of differents offers
  - Analyse the maket in order to fix best prices

Applications

Warbot

- Developped by J. Ferber
- Simulation environment of robots behavior, moving and acting in a simulated space
- 2 kinds of robots (agent) : missile launcher and explorer.
- Goal : Define the better strategy in order to kill the opposite team.
- Observing cooperation and competition strategies.



Applications

ClearPath

- A local collision avoidance algorithm between multiple agents for real-time simulations
- formulates the conditions for collision free navigation as a quadratic optimization problem
- use a discrete optimization method to efficiently compute the motion of each agent
- This resulting algorithm can be parallelized by exploiting data-parallelism and thread-level parallelism

Applications

BioDyn

- Software made to model and simulate complex biological processes
- The model can simulate :
  - E. Coli and Lambda phage interaction
  - Enzymatic reactions
  - Immune responses
- Here an example of bacteria (white) Vs Phage (red)

Applications

RenderBots

- Multi-Agent Systems for Direct Image Generation
- rendering artistic styles such as stippling and hatching
- stroke-based rendering : each stroke is a particle of the drawing = an agent
- The final image is created when the simulation is finished by having each RenderBot execute its painting function

References

- « An Introduction to Multiagent Systems », Michael Woodridge, John Wiley & Sons, 2002.
- « Multiagent Systems : A Modern Approach to Distributed Artificial Intelligence », Gerhard Weiss, MIT Press, 1999.
- « Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence », Ferber. J, Addison Wesley, 1999.