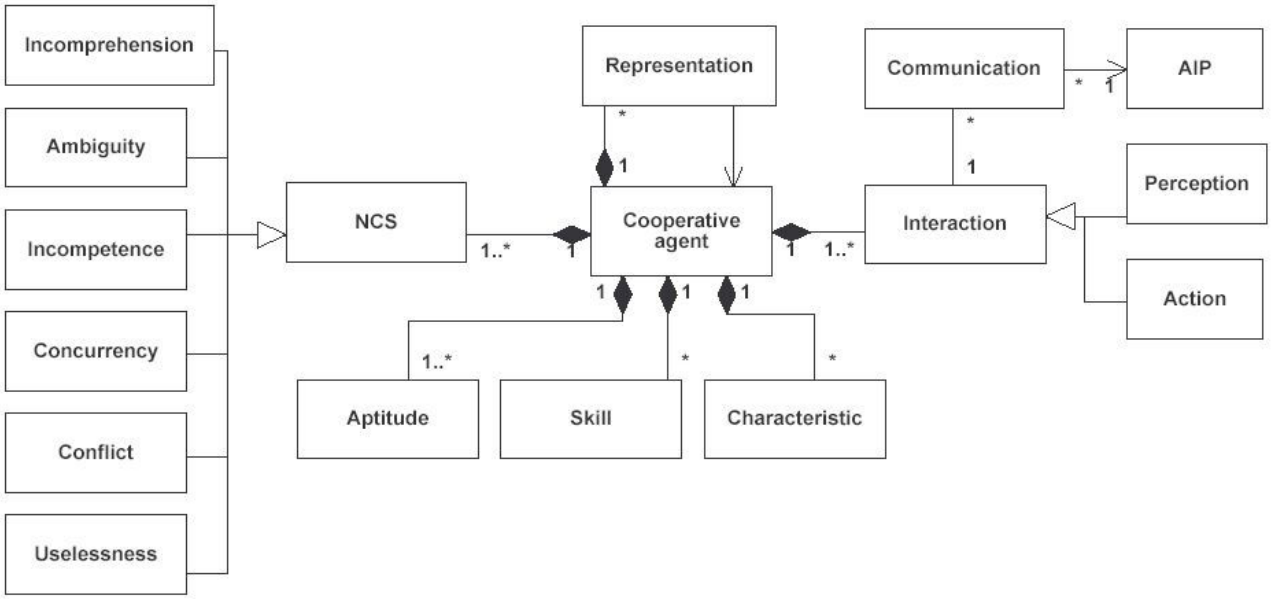
MAS Model in ADELFE

1. Model

ADELFE is a methodology that aims at helping a designer to build Adaptive Multi-Agent Systems (AMAS) i.e. MAS able to adapt themselves when their environment is changing.



A MAS is composed of agents, each agent has a lifecycle which consists in having perceptions, taking decisions and then doing actions (perceive-decide-act).

Agents are "**cooperative agents**" because they possess a cooperative social attitude. This social attitude is implemented using rules allowing Non Cooperative Situations (**NCS**) detection and solving. These NCS can be related to exceptions in classical programming. Different kinds of NCS can be detected according to the context of the concerned application, such as: **incomprehension**, **ambiguity**, **incompetence**, **concurrency**, **conflict**, **uselessness**.

An agent possesses world **representations** that are beliefs concerning other agents, the physical environment or the agent itself. These representations are used by the agent to determine its behaviour. If an agent has representations that may evolve (e.g. a semantic network), these representations can be expressed using a multi-agent system. A representation can be shared by different agents.

An agent is able to communicate with other agents or its environment. This **communication** can be done in a direct manner (by exchanging messages) or an indirect one (through the environment). Tools that enable an agent to communicate are **interaction** languages. When an agent uses a direct communication through messages exchanges, **AIP**s may also be used to express the communication pattern between agents.

For an agent, an **action** is a means to act on the environment during its action phase. Data (for example, a move length or the maximum size for a message) or something the agent can perform (e.g., to move or to send a message, FIPA's ACL) may represent an action.

An agent can have **perceptions** which are means to receive information from its physical or social (other agents) environment.

Perceptions and actions are two kinds of **interactions**.

Aptitudes show the ability of an agent to reason both about knowledge and beliefs it owns. For instance, an aptitude of a software agent can be expressed by an inference en-

gine on a base of rules or any other processing on perceptions and world representations. Aptitudes can also be expressed using data, e.g. an integer value which represents the exploration depth of a planning tree.

An agent owns some **skills** that are specific knowledge that enable it to realise its own partial function. For instance, a skill may be a simple datum which is useful to act on the world (e.g., an integer distance which represents the minimal distance a robot has to respect to avoid obstacles) or may be more complex when expressing a reasoning that the agent makes during its decision phase (e.g., a reasoning to avoid obstacles). If they are complex and able to evolve, skills may also be implemented by MAS.

An agent may possess some **characteristics** which are its intrinsic or physical properties. It may be, for instance, the size of an agent or the number of legs of a robot-like or ant-like agent. A characteristic may also be something the agent can perform to modify or update one of its properties; for example, if the agent is an ant, enabling it to modify its number of legs.

2. Glossary

Adaptive Multi-Agent System (AMAS)
An adaptive multi-agent system is a multi-agent system which is able to change its behaviour while running. It does this to adjust its behaviour to its dynamic environment in order to achieve the task it is designed for or to improve its function or its performances. Such a system is characterised by the following points: (1) it is plunged into an environment, (2) it has a function to achieve and (3) it is composed of interacting agents. To design a system that realises a desired function (which is functionally adequate), a theory has proved that having a system formed by co-operative agents is sufficient; this co-operation directs the social attitude of these agents.
Agent
In ADELFE, an agent follows a classical definition: it is a virtual or physical autonomous entity, capable of actions in an environment, which possesses skills and can offer services, which can communicate directly or not, which is capable of perceiving its environment and has a partial representation of it, whose behaviour tends toward satisfying its objective taking into account its skills, perceptions, representations and communications. An agent has a lifecycle: perceive-decide-act. But agents in ADELFE are more specific, they have a special social attitude: they must be cooperative. So, an agent must detect and process Non Cooperative Situations to always act to come back in a state it judges being cooperative from its own point of view. For instance, an agent that does not possess an information requested by another agent will do all it can to find another agent able to answer this request. Agents, in ADELFE, are not a priori known. They must be identified among the different entities listed in the first step of the activity #6. Then, during activity #12, these entities are studied and potentially cooperative ones identified. These cooperative entities are then identified as “cooperative agents” in the last step of this activity. Their relationships are studied in the activity #13. During the activity #16, all their components (skills, aptitudes, …) are designed. And, in the activity #17, their behaviour can be verified.
Aptitude
An agent possesses some aptitudes to reason both about its knowledge and beliefs. More precisely this knowledge concerns operating knowledge like, for instance, ability to interpret a signal coming from another agent or from the environment. Identified and listed in activity #16.

Characteristic
A characteristic is an intrinsic or physical property of an agent. Identified and listed in activity #16.
Non Cooperative Situations
When the environment is unpredictable, or when the system is open, classical algorithms fail because the designer is unable to find an algorithm which is able to list all the existing possibilities. The aim of the AMAS technology is to design systems that do their best when a difficulty is encountered. In classical programs, these unexpected events can be processed as exceptions. In the AMAS theory context, these "exceptions" - expressing unusual situations that an agent may be faced with - are called "Non Cooperative Situations" (NCS). Different kinds of NCS exist, such as: incomprehension, ambiguity, incompetence, concurrency, conflict, uselessness. The second step of the activity #7 is a help to identify these NCS. The designer must identify the “cooperation failures” (a failure can be viewed as a cooperation protocol which is not obeyed or "bad" interactions that may occur between the system and its environment) that may lead to further NCS. The final NCS are identified and listed in the last step of the activity #16.
Perception
Perception is a means to receive information from the physical or social (other agents) environment. Thus, the designer has to give some perceptive capabilities to the agent. Identified and listed in activity #16.
Skill
Skills of an agent refer to its knowledge about the domain. Identified and listed in activity #16.
(World) Representations
World representations of an agent - or beliefs - concern other agents (its social environment), the physical environment or the agent itself. The agent must always be able to access these representations to decide of its behaviour and, possibly, it must be able to modify them. Identified and listed in activity #16.

Carole BERNON, Valérie CAMPS, Marie-Pierre GLEIZES, Gauthier PICARD
IRIT, Université Paul Sabatier, Toulouse, France