Intelligent Multi-Agent Systems

Uros Krcadinac
University of Belgrade, Serbia

Milan Stankovic
University of Belgrade, Serbia

Vitomir Kovanovic
University of Belgrade, Serbia

Jelena Jovanovic
University of Belgrade, Serbia

INTRODUCTION

Since the AAAI (http://www.aaai.org) Spring Symposium in 1994, intelligent software agents and agent-based systems became one of the most significant and exciting areas of research and development (R&D) that inspired many scientific and commercial projects. In a nutshell, an agent is a computer program that is capable of performing a flexible, autonomous action in typically dynamic and unpredictable domains (Luck, McBurney, Shehory, & Willmott, 2005). Agents emerged as a response of the IT research community to the new data-processing requirements that traditional computing models and paradigms were increasingly incapable to deal with (e.g., the huge and ever-increasing quantities of available data).

Agent-oriented R&D has its roots in different disciplines. Undoubtedly, the main contribution to the field of autonomous agents came from artificial intelligence (AI) which is focused on building intelligent artifacts; and if these artifacts sense and act in some environment, then they can be considered agents (Russell & Norvig, 1995). Also, object-oriented programming (Booch, 2004), concurrent object-based systems (Agha, Wegner, & Yonezawa, 1993), and human-computer interaction (Maes, 1994) are fields that have constantly driven forward the agent R&D in the last few decades.

In addition, the concept of an agent has become important in a diverse range of subdisciplines of IT, including software engineering, computer networks, mobile systems, decision support, information retrieval, electronic commerce, and many others. Agents are used in an increasingly wide variety of applications, ranging from comparatively small systems such as personalized e-mail filters to large, complex, mission critical systems such as air-traffic control.

BACKGROUND

Even though it is intuitively clear what an "agent" or "multi-agent system" is, we may say that there is no universal consensus over some key definitions among researchers in the field. One of the most broadly used definitions states that "*an agent is a computer system, situated in some environment, that is capable of flexible autonomous action in order to meet its design objectives*" (Jennings, Sycara, & Wooldridge, 1998, p. 8). There are three key concepts in this definition: *situatedness*, *autonomy*, and *flexibility*. *Situatedness* means that an agent receives sensory input from its environment and that it can perform actions which change the environment in some way. *Autonomy* is seen as the ability of an agent to act without the direct intervention of humans and that it has control over its own actions and internal state. In addition, the autonomy implies the capability of learning from experience. By *flexibility*, we mean the agent's ability to perceive its environment and respond to changes in a timely fashion; it should be able to exhibit opportunistic, goal-directed behavior and take the initiative whenever appropriate. Also, an agent should be able to interact with other agents and humans, thus be *social*. Some authors emphasize the importance of the concept of *rationality*, which will be discussed in the next section.

With an agent-oriented view of the world, it soon became clear that a single agent is insufficient. Most real-world problems require or involve multiple agents

to represent the decentralized nature of the problem, multiple perspectives, or competing interests. Systems composed of multiple autonomous components (agents) are considered *multi-agent systems* (MAS) and historically belong to distributed artificial intelligence (Bond & Gasser, 1998). MAS can be defined as a loosely coupled network of problem solvers that work together to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfee & Lesser, 1989). The characteristics of MAS are (Jennings et al., 1998):

- Each agent has incomplete information or capabilities for solving the problem, thus each agent has a limited viewpoint;
- There is no global system control;
- Data are decentralized; and
- Computation is asynchronous.

In addition to MAS, there is also the concept of a *multi-agent environment*, which can be seen as an environment that includes more than one agent. Thus, it can be cooperative, competitive, or both.

AGENT-BASED SOFTWARE AND MULTI-AGENT SYSTEMS

Agents and Environments

Dominant researchers in the field mostly refer to agents as software entities that perceive their *environment* through *sensors* and act upon that environment through *actuators* (Russell & Norvig, 1995). There is an obvious analogy with a human agent who has ears, eyes, and other organs as sensors, and arms, legs and other organs as actuators. Information about the state of the environment that the agent acquires through sensors is called *agent's percept.* An agent typically collects its percepts during the time, so its action in any moment generally depends on the whole sequence of percepts up to that moment. If we could generate a decision tree for every possible percept sequence of an agent, we could completely define the agent's behavior. Strictly speaking, we would say that we have defined the *agent function* that maps any sequence of percepts to the concrete action. The program that defines the agent function is called the *agent program*. These two concepts are different; the agent function is a formal description of the agent's behavior, whereas the agent program is a concrete implementation of that formalism.

As Russell and Norvig (1995) stipulate, one of the most desirable properties of an agent is its *rationality.* We say that an agent is rational if it always does the action that will cause the agent to be the most successful. The rationality of an agent depends on the performance measure that defines what is a good action and what is a bad action, the agent's knowledge about the environment, the agent's available actions, and also the agent's percept history. One of the most cited definitions of a rational agent is: "*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*" (Russell & Norvig, 1995, p. 36).

The main challenge in the field of intelligent software agents is to develop an agent program that implements the desired functionalities. Since it is a computer program, we need to have some computing device with appropriate sensors and actuators on which the agent program will run. We call this *agent architecture*. Thus, an agent is essentially made of two components: the agent architecture and the agent program.

Another important concept in the field is the so-called *belief-desire-intention* (BDI) model (Bratman, Israel, & Pollack, 1988). BDI agents are characterized by a "mental state" with three components: *beliefs*, *desires*, and *intentions*. *Beliefs* correspond to information that the agent has about its environment. *Desires* represent options available to the agent—different possible states of affairs that the agent may choose to commit to. *Intentions* represent states of affairs that the agent has chosen and has committed resources to. An agent's practical reasoning includes constantly updating beliefs from information in the environment, deciding what options are available (i.e., recognizing its desires), "filtering" these options to determine new intentions, and acting on the basis of these intentions (Jennings et al., 1998).

Multi-Agent Interactions

In MAS and multi-agent environments, the individual agents need to interact with one another, either to achieve their individual objectives or to manage the dependencies that ensue from being situated in a common environment. These interactions range from simple