

Chapter 7.14

How Can We Trust Agents in Multi-Agent Environments? Techniques and Challenges

Kostas Kolomvatsos

National and Kapodistrian University of Athens, Greece

Stathes Hadjiefthymiades

National and Kapodistrian University of Athens, Greece

ABSTRACT

The field of Multi-agent systems (MAS) has been an active area for many years due to the importance that agents have to many disciplines of research in computer science. MAS are open and dynamic systems where a number of autonomous software components, called agents, communicate and cooperate in order to achieve their goals. In such systems, trust plays an important role. There must be a way for an agent to make sure that it can trust another entity, which is a potential partner. Without trust, agents cannot cooperate effectively and without cooperation they cannot fulfill their goals. Many times, trust is based on reputation. It is an indication that we may trust someone. This important research area is investigated in this book chapter. We discuss main issues concerning reputation and trust in MAS. We present

research efforts and give formalizations useful for understanding the two concepts.

INTRODUCTION

The technology of Multi-agent systems (MAS) offers a lot of advantages in computer science and more specifically in the domain of cooperative problem solving. **MAS** are systems that host a number of autonomous software programs that are called agents. **Agents** act on behalf of their owners giving them access to information resources easily and efficiently. Users state their requirements and agents are responsible to fulfill them. Hence, MAS include many entities trying to solve their problems that are beyond of their capabilities. For this reason, in many cases, agents must cooperate with others in order to find the

appropriate information and services to achieve their goals.

It is obvious that **MAS** are dynamic and distributed environments where agents may cooperate and communicate with others in order to complete their tasks. A key challenge arises from this nature of MAS. In such open systems, entities change their behavior dynamically. Thus, there is a requirement for trust between agents when they must exchange information. Therefore, the basic question in such cases is: How and when can we trust an agent? Agents, in the majority of cases are selfish and their intentions and beliefs change continually.

We try to address this dilemma throughout this chapter. Specifically, we cover the fields of reputation and trust in MAS. This is an active research area, which is very important due to the fact that these two concepts are used in commercial applications. However, open issues exist in many cases, as it is difficult to characterize an agent as reliable or not.

In our work, we try to provide a detailed overview of reputation and trust models highlighting their importance to open environments. Due to the abundance of the relevant models, only the basic characteristics of models are discussed. We discuss basic concepts concerning MAS, reputation and trust. Accordingly, we present efforts, formalizations, and models related to the mentioned concepts. Finally, we discuss about trust engineering issues and we present future challenges and our conclusions.

BACKGROUND

Multi-Agent Systems (MAS)

Software agents and agency have been active research areas for many years due to their importance in various domains. The Web and the recently emerged Semantic Web are the most appropriate examples of such systems. In this section,

basic characteristics of MAS are described. Our goal is to provide necessary knowledge about these systems and their requirements for security.

With the rapid evolution of the Internet, Software agents are a very important research area in Computer Science. **Software agents** are components of software or hardware which are capable of acting on behalf of a user in order to accomplish tasks (Nwana, 1996). The owner of an agent may be a human or another computational entity. Tasks are requested by the owners of agents in order to fulfill their needs. There are different kinds of agents. One can meet information agents that search for information sources, mobile agents that move from an environment to another, intelligent agents that can learn from their owners and the environment and so forth. For an extensive discussion of the different types of agents one can refer to Nwana (1996).

In the most cases, agents must deal with complicated tasks that demand cooperation with others. A **Multi-agent system (MAS)** can be defined as a loosely coupled network of problem solvers that interact to solve problems that are beyond the individual capabilities or knowledge of each problem solver (Durfée & Lesser, 1989). In such systems agents can cooperate or compete with others to complete their tasks. We must note that such systems are open. An open system is one in which the structure of the system is capable of dynamically changing (Sycara, 1998). In open MAS, the basic components may change over time such as information sources or agents' behaviors. From this point of view, it can be assumed that in open MAS (Huynh, Jennings, & Shadbolt, 2006):

- Agents have different owners and for this reason they are selfish and may be unreliable;
- There is no knowledge about the environment in which agents must interact with each other; and

20 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/can-trust-agents-multi-agent/29539

Related Content

Class Patterns and Templates in Software Design

Julio Sanchez and Maria P. Canton (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 546-587).

www.irma-international.org/chapter/class-patterns-templates-software-design/29409

Agility in Software Development and Project Value: An Empirical Investigation

VenuGopal Balijepally, Gerald DeHondt, Vijayan Sugumaran and Sridhar Nerur (2022). *Research Anthology on Agile Software, Software Development, and Testing* (pp. 356-376).

www.irma-international.org/chapter/agility-in-software-development-and-project-value/294472

A Graph Transformation Approach for Modeling UML Diagrams

Hiba Hachichi (2022). *International Journal of Systems and Service-Oriented Engineering* (pp. 1-17).

www.irma-international.org/article/a-graph-transformation-approach-for-modeling-uml-diagrams/300782

A Hybrid Approach for Feature Selection Based on Correlation Feature Selection and Genetic Algorithm

Pooja Rani, Rajneesh Kumar and Anurag Jain (2022). *International Journal of Software Innovation* (pp. 1-17).

www.irma-international.org/article/a-hybrid-approach-for-feature-selection-based-on-correlation-feature-selection-and-genetic-algorithm/292028

Modeling Context-Aware Distributed Event-Based Systems

Eduardo S. Barrenechea, Rolando Blanco and Paulo Alencar (2012). *Handbook of Research on Mobile Software Engineering: Design, Implementation, and Emergent Applications* (pp. 82-94).

www.irma-international.org/chapter/modeling-context-aware-distributed-event/66461