氏名	Platon Eric Louis Marcel
学位（専攻分野）	博士（情報学）
学位記番号	総研大甲第1055号
学位授与の日付	平成１９年３月２３日
学位授与の要件	複合科学研究科　情報学専攻 学位規則第６条第１項該当
学位論文題目	Model of Exception Management in Multi-Agent Systems
論文審査委員	主　査　教授　佐藤　健 教授　武田　英明 教授　米田　友洋 教授　本位田　真一（国立情報学研究所） 助教授　Nicolas Sabouret（パリ第６大学）

論文内容の要旨

Multi-agent systems are often presented as the next major generation of software model and technology to cope with the increasing complexity in modern applications. MAS are distributed systems of autonomous and interacting entities named agents. They are possibly large-scale systems and the agent research community aims at having agents collaborate or compete with one another to achieve their functions in a highly modular and flexible way. A variety of applications of agent technologies can be observed in state-of-the-art software developed from autonomous robots in manufacturing to software agents that assist users over the Internet. Multi-agent systems are therefore promising models and technologies in the future advance in Software engineering and Artificial intelligence.

Multi-agent systems are software in the first place, and constructing dependable systems requires dedicated endeavors and practices. Dependability refers to qualities of a system, in terms of availability to the user of the system, reliability to provide the functions it is designed for, and safety and security of execution. Fault tolerance techniques were developed in traditional Software engineering to increase the degree of dependability of software, and current achievements allow guaranteeing several of the aforementioned qualities in many cases of close and homogeneous systems. Multi-agent systems challenge the current achievements and target more complex systems, as required in the current demand from software users and the infrastructure of our society. Multi-agent systems target open and heterogeneous systems of autonomous agents.

Among the techniques to increase the dependability of software systems, exception handling is notably famous for its strength and simplicity. Programming languages have for long exception handling capabilities to process conveniently and systematically exceptional conditions encountered during a program execution. Distributed computing has however shown that exception handling required specific extensions in the case of distributed applications, and work on software architectures and component-based development have shown the need for other models as well. Multi-agent systems set forth challenging properties that also need to reconsider the question of exception.

The aim of this thesis is to study the notion of exception in Multi-agent systems and to propose a framework adapted to the challenges of openness, heterogeneity, and especially the autonomy of agents. Related work in the agent community has achieved in the past a number of results that showed the need for system-level exception management in Multi-agent systems. The management encompasses handling and the required mechanisms around the handling. The achievements to date set limitations on the type of MAS they can apply to. Agents are often not autonomous and the system-level approaches require agents to perfectly collaborate in the exception management procedure. In this thesis, the ability of agents to deal with exceptions by themselves in the first place is seen as a prerequisite to guarantee autonomy. Exception management then relies on agent-level mechanisms to cope with the shortcomings of current achievements and complement them. Agents keep the capability to freely choose when to initiate exception handling, and when to accept system-level support or rely on individual skills.

The approach developed in this thesis ensures the autonomy of agents by a novel execution model that guarantees the agent preserves control of itself all along its execution and despite the occurrence of exceptions. The model lets the agent decide whether an event is an exception as an individual decision, thus enforcing further the autonomy. The model is formally described and corresponding software architecture is proposed to implement it. The architecture is subsequently applied to a case study to validate the approach, compare it to

existing work, and evaluate its computational cost. The perspectives of this work lie in a number of challenges that can be further elaborated in the framework proposed in this thesis. In particular, the automatic generation of handling strategies by agents in a range of situations is a promising capability that can expand the autonomy of agents in dealing with various exceptional situations. Another notable research direction is the evaluation by an agent of handling strategies received from other agents in the system. The interest in this topic is particularly relevant in future endeavors to bridge previous work, that essentially provide agents with strategies, with the present approach for autonomous agents that are able to estimate when such an external support is acceptable.

The body of the thesis starts with an introduction of key concepts in multi-agent systems, including autonomy, openness and heterogeneity. It is followed in Chapter 2 by a survey of exception management in various disciplines, such as programming language, distributed systems, and existing multi-agent systems. The survey shows that no attempt has been made to give a clear definition of the concept of exception in multi-agent systems, so that the concept of exception remains an intuition. The survey also shows the lack of achievements to engineer agents with exception handling capabilities: Tasks are either incompatible with Multi-agent systems properties or they are ad hoc and complicated. In Chapter 3, the thesis provides a formal definition of "agent exceptions". An agent exception is defined as the interpretation by the agent of a perceived event as unexpected. The definition is clearly distinguished from an exception in programming language, where we usually assume a global control of the system. In distributed software like Multi-agent systems, we need a local and autonomous exception handling since we cannot assume such a global control. In Chapter 4, the thesis proposes an agent execution model and architecture for exception handling, based on the aforementioned definition of exception. The execution model consists of three layers, namely expectation matching, known exception handling, and unknown exception handling. In the expectation matching layer, an agent sorts events to detect exceptional ones. If the agent considers an event as an exception (according to its private state), it is sent to the known exception handling layer. In this layer, if the agent owns an appropriate handler, it processes the exception by exploiting the handler. Otherwise the event is sent to the unknown exception handling layer, where external handlers are searched or generated. In Chapter 5, the thesis presents experimental results conducted on a case study of the above mechanism. It shows qualitative and quantitative evaluations of the mechanism. The case study is a market of rational agents where agents act and compete in the market. The experiments show that although the above mechanisms introduce overhead computation for handling exceptions, it would be acceptable extra cost if exceptions rarely occur. The case study also serves to illustrate the benefit of the approach to engineer exception handling behaviors in agents, where the task of the designer is focused on writing appropriate handlers only. The thesis concludes with a summary of the contributions of the thesis in Chapter 6.

論文の審査結果の要旨

The PhD defense of Eric Platon consisted of one-hour presentation in English which was open to public and a closed session with the PhD committee for further evaluation. In his presentation, Eric Platon showed main contributions in his thesis including (1) a new definition of exceptions named “agent exception” which is distinguished from usual programming exceptions to realize autonomy (2) a proposal of an agent architecture that encompasses a new agent exception management mechanism which provides a foundation to handle exceptions and (3) a design framework based on the architecture which makes a robust multi-agent systems against exceptions. The committee agreed that Eric Platon's answers to the questions in the closed session were satisfactory and the proposals related with exception handling mechanism in the doctoral thesis make original and significant contributions to multi-agent systems research. Moreover the committee recognized Eric Platon's excellent research skills from his publications; 6 post-proceedings papers in Lecture Note in Computer Science and 5 refereed papers in international conferences and 1 survey journal paper. The committee members unanimously concluded that Eric Platon successfully passed the thesis examination.