Robotics and Autonomous Systems 106 (2018) 192–206

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

# IoRT cloud survivability framework for robotic AALs using HARMS☆

Mauricio A. Gomez [a,b,*], Abelghani Chibani [b], Yacine Amirat [b], Eric T. Matson [a]

[a] Department of Computer and Information Technology, Purdue University, West Lafayette, IN, USA
[b] University Paris-Est Creteil (UPEC), LISSI, Vitry-Sur-Seine, France

## HIGHLIGHTS

- Internet of Robotic Things (IoRT) as an extension of Internet of Things (IoT).
- Survivability of the original final goals of multi-agent systems based on cloud computing, model checking, and HARMS.
- Mesh communication in HARMS to determine leader or head selection.
- Tests results of HARMS implementation for Ambient Assisted Living systems (AALs).

## ARTICLE INFO

*Article history:*
Received 24 November 2017
Received in revised form 5 May 2018
Accepted 6 May 2018

*Keywords:*
Model checking
Cloud computing
Multi-agent systems
Heterogeneous agents
HARMS

## ABSTRACT

The Internet of Robotic Things, which includes ambient assisted living systems has been pushed to be developed by the research community for reasons such as the population gap between elderly people and their caregivers. Due to the critical mission that is assigned to those systems; interruptions, failures, worse still, full malfunction should not be allowed to materialize. Such systems ought to keep running in a proper way notwithstanding problems caused either by internal and external system collapses or bad intentioned actions in their surroundings. Therefore, including survivability features must be insured to Ambient Assisted Living systems (AALs) using Humans, software Agents, Robots, Machines, and Sensors (HARMS). HARMS stands for the model that allows through the indistinguishability feature to any type of actor to communicate and interact. This work proposes a framework which takes advantage of the Cloud to overcome the state explosion problem encountered when using model checking. Model checking techniques are used to find a possible solution when a problem is already faced by the system — instead of its original purpose to detect errors on the systems during the design stage. This paper presents the implementation of the proposed framework and validates the functionality with experiments. The conducted experiments evaluate the advantages of using cloud tools to offload the model checking capability for applications such as multi-agent systems.



## 1. Introduction

The Internet of Robotic Things (IoRT) is a new concept which extends the Internet of Things (IoT) capabilities [1]. IoRT explores the possibilities where any kind and number of connected agents, including robots, can give a solution to societal problems. Additionally, it is also motivated by the three functions found in most robot systems: perception, actuation, and control [2]. Vermesan et al. [1] present six components: sensing, actuating, control, planning, perception, and cognition. Three different characteristics in the list imply devices and robots are gifted with sophisticated artificial intelligence (AI) techniques. In this sense, IoRT systems promote smart collaboration, between different types of connected actors allowing them to overcome single agent disadvantages [2]. In other words, in IoRT systems each device provides different capabilities either evaluating, or actuating, or both to accomplish a specific common goal that cannot be concluded by only one of the agents. However, developing autonomous IoRT services represents a challenge with a number of issues that appear when several smart objects and assorted agents collaborate in a common setting.

Zuo [3] summarizes the definition of a survivable system as one system that possesses the ability to support its mission; providing an acceptable level of essential services regardless of the hostility of the environment. Such systems should include the ability to recover when the situation has been improved either by internal actions or external situations. Recently, there has been more studies of approaches of autonomous systems. Ambient Assisted Living systems (AALs) are not the exception, although properties like survivability are still in early stages of research. Additionally,

☆ This work was not supported by any organization.

* Corresponding author at: Purdue University, West Lafayette, IN, USA.
*E-mail addresses:* mgomezmo@purdue.edu (M.A. Gomez), achibani@u-pec.fr (A. Chibani), amirat@u-pec.fr (Y. Amirat), ematson@purdue.edu (E.T. Matson).

https://doi.org/10.1016/j.robot.2018.05.001
0921-8890/

understanding that the concepts of survivability and resilience are applicable to many fields related to systems in general and computational ones. The research community has been focusing on different topics related to survivable systems primarily on areas such as network [4–6]. In addition, when the panorama of the system includes several heterogeneous agents the problem becomes more complex and hence less studied at the moment. Furthermore, when the problem includes humans as agents, not as normal end users, then, the scenario becomes more entangled due to their implicit unpredictable behavior. Nonetheless, IoRT systems should be reliable and able to accomplish the original set of goals in a survivability manner. The main priorities should be focused in these issues when systems like AALs are already being developed to interact with humans that will completely rely on the suggestions made by the system.

In a previous work [7], we introduced the idea of applying model checking during runtime over the cloud for AALs. In this work we focus on the possibility of having an automatic survivability mechanism which can provide an on-line solution to problems encountered by the system. In other words, in this work we focus on the accomplishment of the final goals in contrast to only finding errors during execution time. Moreover, we use the cloud to offload the model checking process and reduce the time spent due to the state explosion problem. This paper makes an analysis of how using the cloud and model checking tools can lead to a possible solution for the survivability of the system final goals.

AALs appear as a solution proposed to solve the problem of having many elderly people with not enough human caregivers in a near future. The literature related to the growth of percentage increase of aged population can be found in [8,9]. For instance, the European Union takes safety and security of elderly citizens at home as a high priority [10].

One of the points of view that AALs could be analyzed is through Multi-Agent Systems (MAS) paradigm. On one hand, Sycara in [11] defines MAS as the systems that (1) are conformed by agents with limited capabilities and information (2) system global control should not exist (3) control distributed information and data (4) work with not synchronous computation. On the other hand, Carley in [12] gives the meaning of an organization as the group of interactions and agents that possess characteristics such as goal adaptable, able and willing to work in social context, technology-driven, and ready for changes. Both definitions share common aspects. They are systems integrated by different agents lacking in complete information, and capabilities to solve the problem; global control is not possible, data fluctuates in a non-centralized manner and computation are asynchronous.

Within MAS strategies different autonomous agents perform an activity in a coordinated way which can have several different approaches, such as centralized, not centralized, autonomous, not autonomous, and such. Furthermore, many agents working in an autonomously collaborative way may lead to a vast gamma of possible failures which some of them are not detected in the design stage.

HARMS is based on the machine to machine (M2M) infrastructure to integrate humans, agents, and robots into collectives [13]. Later on, HARMS was defined as a model which enables five different layers: the network, communication, interaction, organization, and collective intelligence to any number of actors such as humans, software agents, robots, machines, and sensors. The most important feature of HARMS is the indistinguishability, given that the goal or objective is executed no matter which agent accomplishes it [14].

In this paper, we propose a framework which takes advantage of cloud services to execute model checking processing in order to accomplish survivability feature in AALs using HARMS. Specifically, finding solutions to problems the system could find during runtime.

The remainder of this work is organized as follows: Section 2 introduces the related work Section 3 discusses the AAL systems generic problems. A solution framework is proposed in detail in Section 4. In Section 5 an evaluation of the models is detailed. Then in Section 6 a complexity analysis of the algorithms is described. Section 7, explains the experiments performed to validate ideas previously mentioned in this paragraph. In Section 8 the results of the experiments are analyzed. Finally, conclusions and future work are presented at the end of this document in Section 9.

## 2. Related work

The spectrum of the problem addressed in this work comprehends several aspects as a whole proposed framework. For instance, survivability, as it is a topic, studied mostly in fields not related to ambient assisted living systems. Ayara and Najjar [15] present one example of a formal specification model for survivability in pervasive systems using an example of healthcare. Their method presents adaptation based on an evaluation of the degree of survivability to ensure the acceptable execution of the offered services. This method differs from the one presented in this work as it does not apply model checking. Even though the example was related to the healthcare system, it is not exactly an AALs. One aspect that has more interest for the research community is adaptability [16] that bases the solution on fault tolerant approaches to propose a reconfigurable model framework in home automation. On the other hand, we present an approach from the point of view of survivability during the system's mission. One trend of research for AALs is monitoring, as complete as possible, the information generated by all the devices related to the specifically targeted patient. Forkan, Khalil, and Tari [17] propose a cloud-based, real-time, context-aware platform to analyze the enormous amount of data generated by the different connected devices. The ultimate goal of this solution is to provide tailored services to meet environmental and user needs. The latter kind of solutions are grouped in context-aware solutions which differ from the solution presented in our work. In our solution, monitoring it is part of an assumption that we use it to only check if an error has occurred in the current status. Applying model checking and temporal logic in a cloud-based environment has been explored in the automated monitoring and to support automated real-time recognition of activities of daily living within a smart environment evaluated in a smart kitchen setting [18]. This work is focused on applying model checking on activity recognition rather than finding a solution to the problems encountered during runtime. In the case of verification during runtime for AAL systems, it has not been studied deeply. Efforts on dependability analysis in the AAL domain was explored in a case study converting UML patterns of behavior models of AALs architecture to a formal executable specification which is later verified in a model checking tool denominated PRISM [19]. Although it is interesting to transform the UML diagrams to code that can be executed in a formal verification tool. It differs to our approach in the point that we use HARMS with its feature of indistinguishability which makes the analysis required to be possible changed over time. Another example is presented as a specification and verification approach using MEDISTAM-RT and timed traces to assure the AALs specification execution in [31]. However, in our approach we use model checking to obtain a possible solution to a problem encountered during runtime. Schneider and Becker in [32] elaborated on components and their respective models to support typical adaptation scenarios in the AAL domain and have raised some issues with regard to the evolution of these models. Our work, on the other side, implements adaptations during runtime based in survivability principles. HARMS implementations over AALs is also a topic where research community has not been greatly focused yet. This may be caused by the novelty of the model,