Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Curriculum in Ingegneria Informatica, Gestionale e dell'Automazione

---

Smart Home Reasoning Systems: From a Systematic Analysis Towards a Hybrid Implementation for the Management of Uncertainty and Inconsistency

Ph.D. Dissertation of:
Dagmawi Neway Mekuria

Advisor:
Prof. Aldo Franco Dragoni

Curriculum Supervisor:
Prof. Francesco Piazza

XVIII edition - new series

Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Curriculum in Ingegneria Informatica, Gestionale e dell'Automazione

---

Smart Home Reasoning Systems: From a Systematic Analysis Towards a Hybrid Implementation for the Management of Uncertainty and Inconsistency

Ph.D. Dissertation of:
Dagmawi Neway Mekuria

Advisor:
Prof. Aldo Franco Dragoni

Curriculum Supervisor:
Prof. Francesco Piazza

XVIII edition - new series

Università Politecnica delle Marche
Scuola di Dottorato di Ricerca in Scienze dell'Ingegneria
Facoltà di Ingegneria
Via Brecce Bianche – 60131 Ancona (AN), Italy

Acknowledgments

First and foremost, I would like to express my special appreciation and thanks to my supervisor Professor Aldo Franco Dragoni, for allowing me to work on a very interesting topic and for his invaluable guidance throughout my PhD. I would also like to thank him for his support, encouragement, motivation, and for providing me ample opportunities to independently explore and develop my ideas.

I would also like to express my deepest gratitude for my colleagues Paolo Sernani and Nicola Falcionelli not only for the valuable discussions and collaborations over the years but also for welcoming me into AIRTLab and providing me assistance beyond their responsibilities, even in non-academic matters. I am extremely indebted to you, and I could not have imagined having better colleagues for my PhD studies.

I would like to thank MIND S.R.L and its employees, for co-financing and supporting this thesis.

I would especially like to thank Professor Paolo Giorgini, for the opportunities he provided me during my stay in Italy, and for encouraging me to start this PhD, as well as for his time spending as my thesis committee member. My sincere thanks also extend to Professor Paulo Leitao for serving as my committee member.

I would like to thank my colleagues and friends in DII, and I particularly want to thank Interview Lab and VRAI members, for the pleasure and nice time which makes my social life.

Many thanks for Dr. Solomon Tadesse and his family, Andrea and Mohamed for the great friendship in Ancona and for being there when needed. Many thanks also to all my friends in Tej-Bet, Menen, BDU, Addis Ababa, UNITN, Trento and all over the world.

My deepest gratitude to my family for their love, care and help, despite the long distances. Without their unconditional love and support, I could have neither started nor finished this journey.

Ancona, November 2019

Dagmawi Neway Mekuria

Abstract

A smart home is a residence equipped with technologies that facilitate monitoring of residents, promote independence and increase the quality of life. In general, smart homes control the operations of the home environment and automatically adapt it to its inhabitants' needs. The smart home reasoning system (SHRS) is in charge of determining the automatic control and adaptation operations of the home system. Recently, there has been extensive research concerning different aspects of the SHRS. However, there is a clear lack of systematic investigation targeted at these systems. To close the gap, in the first part of this thesis we explore the SHRS domain. For this reason, we applied the systematic literature review (SLR) method by conducting automatic and manual searches on six electronic databases, and in-depth analysis of 135 articles from the literature.

From the SLR, this thesis identifies that about 43% of smart homes are designed to provide general home automation services. It also presents twelve major requirements and features of the SHRS. In addition, the SLR finds out that 55.5% of the research contributions in SHRS domain are theoretical, and 51.5% of them are based on symbolic artificial intelligence techniques. Further, it characterizes the usage and application trends of different reasoning techniques in smart home domain, and evaluates the major assumptions, strengths, and limitations of the proposed systems in the literature. Additionally, it discusses the challenges of reasoning in smart home environments. Finally, it underlines the importance of utilizing hybrid reasoning approaches and the need to handle uncertainty and inconsistency issues of the SHRS, as well as overlapping, simultaneous and conflicting multiple inhabitants' activities and goals in the smart home environment.

The SLR identifies reasoning under uncertainty as one of the major challenges of SHRSs. Uncertainty is inevitable in smart home environments as sensors may read inaccurate data or due to the existence of unobserved variables for privacy reasons. Furthermore, the dynamic nature of the home environment and vague human communications may result in ambiguous, incomplete and inconsistent contextual information, which ultimately lead the smart home system into uncertainty.

With this in mind, the second part of this thesis tackle some of the challenges of uncertainty, in particular, uncertainty due to vague human communication

and missing information in ambient intelligence environments. For this, we proposed probabilistic multi-agent system architecture for reasoning under uncertainty in smart home environments. The proposed smart architecture is based on the notion of multi-agent systems (MAS) technologies and probabilistic logic programming techniques. Afterwards, we show how the probabilistic reasoning technique enables the agents to reason under uncertainty. Furthermore, we discuss how intelligent agents enhance their decision-making process by exchanging information about missing data or unobservable variables using agent interaction protocols. Besides, when an agent lacks the necessary computational resources to accomplish its reasoning tasks, we illustrate how it can take advantage of the interaction protocols and delegate the tasks for other agents in the system. In general, we demonstrate that the combination of MAS technologies and probabilistic logic programming can help in building a reasoning system, which is capable of performing well under vague inhabitant commands and missing information in a partially observable environment.

In the final part of the thesis, we tackled inconsistency issues in SHRSs, by identifying five major sources of inconsistencies in rule-based SHRSs. Specifically, we define, formalize and demonstrate how conflicting, duplicate, overlapping, self-looping and circular rules in SHRSs can be detected using satisfiability modulo theories. The proposed method was validated empirically using rules collected from a real-world SHRS as a model. The experimental results provide compelling evidence for the reliability and effectiveness of the proposed solution. The method presented in this part of the thesis can have multiple applications. First, it can be used to build a static (off-line) rule-based reasoning system verification tool. Second, it can be integrated as a rule validation component of the reasoning system. Besides, with some adaptation, the method can be directly used to verify the consistency properties of reasoning systems in other domains.

Contents

List of Figures

List of Tables

Abbreviations

AAL Ambient Assisted Living

ADL Activities of Daily Living

AI Artificial Intelligence

ANN Artificial Neural Network

BDI Belief-Desire-Intention

Bel degree of belief

BEMS Building Energy Management Systems

CNET Contract Net Protocol

CTL Computational Tree Logic

DA Device Agent

ECA Event-Condition-Action

EM-HC Elderly Monitoring and Healthcare

FIPA Foundation for Intelligent Physical Agents

HCI Human Computer Interaction

HEM Home Energy Management

HVAC Heating Ventilation and Air Conditioning

IEA International Energy Agency

IEL Infinite Execution Loop

IoT Internet of Things

IP Request Interaction Protocol

JADE Java Agent DEvelopment Environment

$\mathcal{LRA}$ Linear Arithmetic over Rationals

LTL Linear Temporal Logic

MAS multi-agent system

MLN Markov Logic Network

NA Negotiator Agent

PoC Proof-of-Concept

RA Reasoner Agent

RG research goals

RQ research questions

SA Service Agent

SAT Propositional Satisfiability

SHRS Smart Home Reasoning Systems

SLR systematic literature review

UCD User-centered Design

Chapter 1

Introduction

Over the past decade, the interest in home-based assistive and monitoring technologies is rapidly growing [1]. The rise of interest in these technologies can be justified for several reasons, such as due to the growing number of world elderly population, the need to reduce buildings' energy consumption, and the recent advancements in Artificial Intelligence (AI) and the Internet of Things (IoT). First, the growing number of world elderly population (aged 65 and older), which is estimated to be more than 1.6 billion by 2050, representing 16.7 percent of the total world population of 9.4 billion [2], can be considered as the main factor. Over time, aging damage leads to a gradual decrease in physiological reserves, an increased risk of chronic diseases, and a general decline in the capacity of the individual [3]. These aging-related complications cause many challenges in the quality of life of older people and their carers. Ambient Assisted Living (AAL) technologies are regarded by many as a primary solution to tackle these challenges. As a result, national governments, non-profit organizations and industries are promoting the use of home-based health care and elderly monitoring solutions to extend the time older people can live independently in their preferred environment, to promote better and healthier lifestyles for individuals at risk, to enhance security, prevent social isolation and create a network of support around older people and their caregivers. And, this kick in the increasing interest in the adoption of home-based assistive and monitoring technologies.

Second, the building sector is accounted for more than 40% of the total world annual energy consumption [4]. In addition, according to a report from the International Energy Agency (IEA), in 2013 more than 65% of the world's electricity was generated from fossil fuels, thus contributing to CO_2 emissions [5]. Moreover, due to the continued rise in the cost of energy, the operating costs of buildings with high electricity demand are rising. These factors make Building Energy Management Systems (BEMS) an area of interest for both public and private sectors. BEMS helps to monitor, control, and optimize the energy consumption needs of a building. As a result, it enables us to face the growing energy demands of the world population while mitigating climate change and reducing operational costs. These advantages of BEMS result in the recent boom of interest in home-based technologies, which allows to actively monitor the buildings' energy consumption and to reduce their power usage.

Third, the recent advancement and proliferation of the IoT and AI are playing a key role in the continuously growing interest in these technologies. The combination of AI and IoT is leading to the development of many fascinating home automation products. These products are known to have the connectivity and remote control features of IoT, with some learning, prediction, and recognition skills of AI. As a result, these smart home devices are being integrated into a large number of homes for security, entertainment and comfort purposes.

Taking into account the growing interest in these technologies, smart home systems aim to provide a universal solution, which comprises all the aforementioned needs of home-based assistive and monitoring services, confronts the challenges presented by them, and benefits from their presented opportunities. Specifically, the term "*smart home*" refers to a residence equipped with technologies that facilitate monitoring of residents, promote independence and increase the quality of life [6]. Recently, smart homes have been applied to provide home energy management [7], elderly monitoring and healthcare [8], comfort [9], and entertainment services [10]. To effectively deliver these services, the smart home system needs to perceive the state of the residence through sensors, and automatically adapt the home environment to its inhabitants' preferences through actuators. The automatic adaptation process of the living environment is mainly determined and controlled by the reasoning system, which is considered to be the brain of the smart home system. Precisely, the primary role of a Smart Home Reasoning Systems (SHRS) is to make appropriate decisions towards achieving the comfort and efficiency goals of the inhabitant and their environment. SHRS is an underexplored research area in the AAL domain. Hence, several challenges in the field need to be addressed before it brings true intelligent behaviors to our living environments. To this end, in this thesis we are interested to systematically investigate the SHRS domain and tackle some of the key challenges in the field.

1.1 Research Questions and Objectives

Within the above stated context, the overall aim of this thesis is twofold. First, it aims to explore, investigate, and characterize different aspects of SHRSs. Second, it aims to propose solutions that can help address major challenges of SHRSs. To this end, the following research goals (RG) and research questions (RQ) have been formulated:

- **Research Goal 1 (RG1):** The goal here is to systematically investigate the smart home reasoning system domain. Researchers and practitioners have been extensively applying different AI methods in AAL environments. However, there is a clear lack of systematic investigation targeted at exploring the reasoning systems integrated into these environments. Most existing studies either presented a comprehensive overview of smart home systems, or explored smart home technologies from a single application domain perspective, or examined the networking challenges introduced by the home IoT devices. Yet, no contribution devoted its study to examine, characterize, and evaluate SHRSs and their utilized reasoning techniques. With this in mind, this thesis sets as its first goal conducting a systematic literature review (SLR) in the SHRS domain. In this regard, it poses the following research questions:

  RQ1: What are the primary purposes of smart home systems?

  RQ2: What are the essential features and characteristics of a SHRS? And, what requirements it needs to fulfill to exhibit the often pledged services of smart homes?

  RQ3: What are the major reasoning approaches, methodologies, tools and technologies extensively used for the design, development, and integration of the SHRS? And, what is their application trend over the years?

  RQ4: What are the major assumptions made by the literature to effectively present the behaviour and operation of their proposed reasoning system?

  RQ5: Which of the proposed approaches are conceptual, proof-of-concept implementations, and tested in a real-world living environment?

  RQ6: What are the strengths and limitations of the proposed systems and their utilized reasoning approaches?

  RQ7: And, what are the main challenges of reasoning in a smart living environment?

  Chapter 2 of this thesis will present a detailed discussion about the formulation, purposes, and objectives of the above RQs.

- **Research Goal 2 (RG2):** Broadly, the second goal of this thesis is to tackle some key challenges of SHRSs. These challenges are identified from the results of the SLR carried out to achieve RG1. Specifically, the SLR answers RQ7 by revealing seven broad but distinct research challenges of SHRSs that need to be addressed by future research in the field. Accordingly, this thesis aims to propose solutions for two of the identified research problems. The choice of these research problems is partly motivated by the immediate research needs of this doctoral study funding organization, MIND S.R.L[1].

As part of this goal, the first challenge that this thesis aims to deal with is reasoning under uncertainty in smart home environments. Uncertainty is prevalent in smart homes for several reasons, such as due to the dynamic nature of the environment, vague inhabitant commands, sensor failures, or erroneous data reads. However, literature in the field has been giving little attention to this issue. In this regard, we aim to study the advantages of modeling the smart home environment as a multi-agent system (MAS). Further, we aim to examine if the combination of MAS technologies with hybrid AI techniques such as probabilistic logic programming, help to meet the challenges of reasoning under uncertainty in smart home settings. To this end, the following research question is posed

RQ8: How can we effectively model the complex, dynamic, and distributed nature of the smart home environments in terms of multi-agent systems? And, could the combination of MAS technologies and probabilistic logic based reasoning techniques help to tackle the major causes of uncertainty in SHRSs?

The second research challenge that this thesis aims to address is the consistency verification of SHRS. While answering RQ2, RQ6, and RQ7, the SRL highlights the need to integrate a standard conflict detection and resolution method into the reasoning system. Moreover, it insists that the SHRS automatic adaptation process must not lead the home into an incomprehensible and uncontrollable state, and it should not exhibit unpredictable behavior. However, the reasoning system is prone to forgo these requirements for several reasons, such as, due to conflicting inhabitant preferences in a multi-resident smart home environment, as a result of conflicting services, or owing to software and hardware components failure. Besides, violations of some internal properties of the system, such as consistency of the reasoning system, are key factors that could seriously affect the regular operation of the smart home. Therefore, it is essential to analyze and validate the consistency properties of the reasoning system. Accordingly, the thesis aims to identify, define, and formalize the primary causes of inconsistencies in SHRS. Further, it aims to propose a method to automatically analyze and detect these common sources of inconsistencies in the system. To this end, the following research question is posed:

---

[1] http://www.mind.cc/

RQ9: What are the primary causes of inconsistencies in SHRS? How can we formalize these sources of inconsistencies? And how formal methods can be used to verify the consistency of the system?

Towards achieving RG1, we conduct an SRL targeted at examining the literature in the SHRS domain. Whereas to answer RQ8, we model the smart home environment as a MAS and utilize a probabilistic logic programming technique to give the agents an ability to reason under uncertainty. Besides, the agents rely on agent interaction protocols to delegate their reasoning tasks with each other, and to exchange missing information about their operating environments. For RQ9, we identify the major sources of inconsistencies in rule-based SHRSs. Afterward, we define and formalize these sources of inconsistency as Satisfiability Modulo Theories (SMT) constraints. Finally, we demonstrate how the consistency of the SHRS can be verified using an SMT solver.

1.2 Contribution and Structure of the Thesis

This thesis is structured into the following five chapters, and the contributions presented in these chapters are summarized below.

- **Chapter 1: Introduction -** This chapter provides a description of the motives and reasons behind this thesis. It introduces the reader to the main research goals and objectives of this thesis. It explains in more detail the contributions of this thesis and how the work presented is structured.
- **Chapter 2 : Smart Home Reasoning Systems: A Systematic Literature Review -** This chapter systematically investigates the literature in smart home reasoning systems domain. Specifically, it thoroughly examines a variety of AI techniques and their application for ambient assisted environments. For this reason, it utilizes the SLR method by conducting automatic and manual searches on six electronic databases, and in-depth analysis of 135 articles from the SHRS literature.

  From the SLR, the chapter identifies about 43% of smart homes are designed to provide general home automation services. It also presents twelve major requirements and features of an SHRS. Besides, the study finds out that 55.5% of the research contributions in SHRS domain are theoretical, and 51.5% of them are based on symbolic AI techniques. Further, it characterizes the usage and application trends of different reasoning methods in the smart home domain and, evaluates the major assumptions, strengths, and limitations of the proposed systems in the literature. Subsequently, the chapter discusses the challenges of reasoning in AAL environments. Moreover, it underlines the importance of utilizing hybrid reasoning approaches, and the need to address challenges caused by uncertainty and inconsistency in the smart home reasoning systems. Finally, it highlights handling of overlapping, simultaneous and conflicting inhabitants' activities in multi-resident AAL environments as opportunities for future research. This work at its preliminary stage has been published in [11] and its complete version in [12].

- **Chapter 3: A Probabilistic Multi-Agent System Architecture for Reasoning Under Uncertainty in Smart Home Environments** The SLR presented in chapter 2 of this thesis reveals that the dynamic nature of AAL environments and vague human communications may result in ambiguous, incomplete and inconsistent contextual information. These ultimately lead to uncertainty, which is inevitable in smart home environments due to inaccurate sensor data or due to the existence of unobserved variables for privacy reasons. Aiming at tackling some of these challenges (i.e. achieving RG1, and specifically answering RG8), this chapter applies a probabilistic logic reasoning technique into multi-agent based smart home architecture. Accordingly, it shows how the probabilistic reasoning technique enables the agents to reason under uncertainty. Further, it discusses how intelligent agents enhance their decision-making process by exchanging information about missing data or unobservable variables using agent interaction protocols. In addition, it presents the proof-of-concept implementation and experimental evaluation of the proposed smart home system. In general, this chapter demonstrates that the combination of multi-agent system technologies and probabilistic logic programming can help in building a reasoning system, which is capable of performing well under vague inhabitant commands and missing information in partially observable environments. This work has been published in [13] and [14].

- **Chapter 4: Consistency Verification of a Rule Based Smart Home Reasoning System with Satisfiability Modulo Theories -** The SLR presented in chapter 2 of this thesis identified conflict detection and resolution as one of the major requirements and challenges of SHRSs. The same study highlighted the lack of contributions in the literature to verify the consistency of these systems. With that in mind, chapter 4 answers RG9 of this thesis by proposing a method for consistency verification of a rule-based smart home reasoning system. To this end, it defines, formalizes and presents a static (off-line) analysis method for five primary causes of inconsistencies in rule-based SHRSs, using satisfiability modulo theories as a tool. The primary causes of inconsistencies considered in this study are conflicting, duplicate, overlapping, self-looping, and circular rules. Subsequently, it presents an empirical validation of the proposed method using a real-world smart home reasoning system as a model. In general, this chapter presents a reliable and effective solution to analyze and verify the consistency of SHRSs. This work is submitted for publication in the 16th International Conference on Intelligent Environments (IE2020).

- **Chapter 5: Conclusions -** This chapter concludes the thesis and outlines future research directions in the smart home reasoning system domain.

1.3 Publications

1.3.1 Relevant publications

A significant part of this dissertation has been published in the following journal and conferences.

1.3.1.1 International journals

- *Dagmawi Neway Mekuria*, P. Sernani, N. Falcionelli, and A. F. Dragoni **Smart home reasoning systems: A systematic literature review**. In Journal of Ambient Intelligence and Humanized Computing. Springer, 2019, pp. 161-179.

1.3.1.2 International conferences, symposiums, and forums

- *Dagmawi Neway Mekuria*, Paolo Sernani, Nicola Falcionelli, Aldo Franco Dragoni **Probabilistic Logic Reasoning in Multi-Agent Based Smart Home Environment.** In proceeding of Italian Forum of Ambient Assisted Living, Springer, 2019, pp. 161-179.

- *Dagmawi Neway Mekuria*, Paolo Sernani, Nicola Falcionelli, Aldo Franco Dragoni **A Probabilistic Multi-Agent System Architecture for Reasoning in Smart Homes**. In proceding of IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA), IEEE, 2019, pp. 1-6.

- *Dagmawi Neway Mekuria*, P Sernani, N Falcionelli, AF Dragoni **Reasoning in Multi-Agent Based Smart Homes: A Systematic Literature Review**. In proceeding of Italian Forum of Ambient Assisted Living, Springer, 2018, pp. 161-179.

1.3.1.3 Under review

- *Dagmawi Neway Mekuria*, Paolo Sernani, Nicola Falcionelli, Aldo Franco Dragoni **Consistency Verification of a Rule Based Smart Home Reasoning System with Satisfiability Modulo Theories.**, In the 16th International Conference on Intelligent Environments (IE2020) June 22-25, 2020, Madrid, Spain.

1.3.2 Additional publications

In addition, during my PhD, I have collaborated with others in publishing the following articles which are not covered in this thesis.

1.3.2.1 International journals

- N. Falcionelli, P. Sernani, A. Bruguès, *Dagmawi Neway Mekuria*, D. Calvaresi, M. Schumacher, A. F. Dragoni, and S. Bromuri **Indexing the event calculus: Towards practical human-readable personal health systems**. In Artificial intelligence in medicine. Elsevier, 2019, pp. 154-166.

1.3.2.2 International conferences, workshops and demos

- Nicola Falcionelli, Paolo Sernani, *Dagmawi Neway Mekuria*, Aldo Franco Dragoni **An Agent-Swarm Simulator for Dynamic Vehicle Routing Problem Empirical Analysis.** In procedding of International Conference on Practical Applications of Agents and Multi-Agent Systems (PAMAS) , Springer, 2019, pp. 246-250

- Paolo Sernani, Matteo Biagiola, Nicola Falcionelli, *Dagmawi Neway Mekuria*, Stefano Cremonini, Aldo Franco Dragoni **Time Aware Task Delegation in Agent Interactions for Video-Surveillance.** In proceeding of First International Workshop on Real Time compliant Multi-Agent Systems (RTcMAS), 2018, pp. 16-30.

- Nicola Falcionelli, Paolo Sernani, *Dagmawi Neway Mekuria*, Aldo Franco Dragoni **An Event Calculus Formalization of Timed Automata.** In proceeding of First International Workshop on Real Time compliant Multi-Agent Systems (RTcMAS), 2018, pp. 60-76.

- Nicola Falcionelli, Paolo Sernani, Albert Bruguès, *Dagmawi Neway Mekuria*, Davide Calvaresi, Michael Schumacher, Aldo Franco Dragoni, Stefano Bromuri **Event calculus agent minds applied to diabetes monitoring**. In A2HC/A-HEALTH 2017, LNAI 10685, pp. 40-56, 2017.

Chapter 2

Smart Home Reasoning Systems: A Systematic Literature Review

2.1 Introduction

Several researchers have been extensively studying the application of artificial intelligence methods for the ambient assisted living domain. However, there is a clear lack of systematic investigation targeted at exploring the reasoning systems integrated into smart homes. Most existing reviews such as [15] and [1] presented a comprehensive overview of the smart home systems. Few others such as [16] and [6] explored smart home technologies from a single application domain perspective (e.g. health care) and, examined the networking challenges introduced by the home Internet of Things devices. A review by [17] studied smart home inhabitants and their uses of the underlying infrastructure from socio-technical viewpoint. On the other hand, a few other surveys such as the one by [18], tailored their work in the investigation of the smart homes system requirements, from a very broad standpoint. The work presented by [19] briefly explored SHRS. However, the center of the review was the requirements, system architecture and integration of home care decision support system, without a detail evaluation of the reasoning approaches integrated into the systems. To the best of our knowledge, no other contribution devoted its study to systematically examine, characterize and evaluate the reasoning systems of smart homes and their utilized reasoning techniques. To close the gap, this chapter reports the results of a systematic literature review conducted to examine the aforementioned domain. In this regard, the chapter aims to:

- Determine the primary purposes of smart home systems.
- List the key features, characteristics, and requirements of a smart home reasoning system.
- Identify and characterize the main reasoning approaches, methods, tools, and technologies utilized in the literature to propose a SHRS.

- Analyze the principal assumptions, strengths, and limitations of the SHRS presented in the literature.

- And, report the challenges of building a reasoning systems for smart living environments.

The rest of the chapter is organized as follows: Section 2.2 describes our research questions and the systematic literature review protocol adapted for the literature search and analysis. Section 2.3 presents and discusses the review results. And, section 2.4 presents a conclusion with a brief summary of the systematic literature review results.

2.2 The Review Process

A systematic literature review is a method that enables the evaluation and interpretation of all accessible research relevant to a particular research question, subject matter, or event of interest. It aims at presenting a fair evaluation of a research topic by using a trustworthy, rigorous, and auditable methodology [20]. Two common reasons for performing an SLR are:

(i) The need to evaluate and summarize existing evidence concerning a particular technology.

(ii) The need to identify gaps in the technology that will potentially lead to topics for future investigations.

In this study, we carried out an SLR by adapting the guidelines proposed by [20]. Our adapted review process, presented in Fig. 2.1, consists of three main phases: *Planning, Conducting, and Reporting.* The rest of this section discusses the principal components of the adapted approach and its applications for this study.

2.2.1 Planning the review

As stated by [20], researchers should examine existing literature on the topic of interest, and determine the need for a review, before conducting any SLR. Further, the study outlined two most important pre-review activities: *defining the research question(s)* and *producing a review protocol.* Accordingly, we decided to conduct this review, after realizing the fast-growing research interest in smart living environments and identifying the absence of any systematic investigation specifically targeted at SHRS.

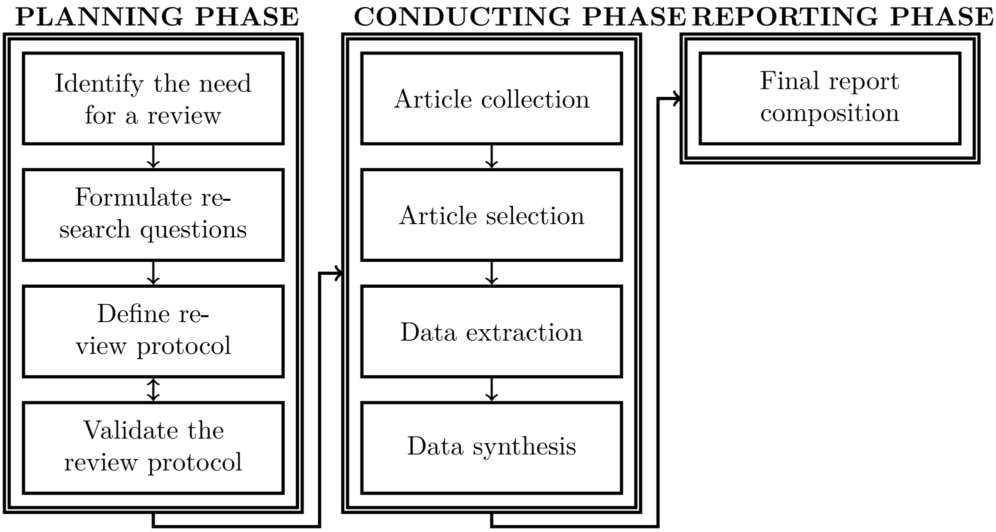

Figure 2.1: The systematic review process

I. **Formulating the research questions**

As discussed above, formulating the research questions is one of the most important activity of an SLR. Hence in order to define the most relevant research questions for our topic of interest, we utilized the Goal-Question-Metrics approach proposed by [21], and identified the three main coordinates (i.e *issue, object and view point*) of the goal in this SLR and its purposes as follow:

- **Purpose:** *Explore, Analyze, and Compare*
- **Issue:** *Reasoning Systems*
- **Object:** *Smart Home Systems / AAL*
- **Viewpoint:** *Researcher's Point of View*

Based on the above-defined goals and purposes of the SLR, RQ1 to RQ7 of this thesis has been formulated. For the completeness of this chapter, the research questions are listed again below:

RQ1: What are the primary purposes of smart home systems?

RQ2: What are the essential features and characteristics of a smart home reasoning system? And, what requirements it needs to fulfill to exhibit the often pledged services of smart homes?

RQ3: What are the major reasoning approaches, methodologies, tools and technologies extensively used for the design, development, and integration of the SHRS? And, what is their application trend over the years?

RQ4: What are the major assumptions made by the literature to effectively present the behaviour and operation of their proposed reasoning system?

RQ5: Which of the proposed approaches are conceptual, proof-of-concept implementations, and tested in a real-world living environment?

RQ6: What are the strengths and limitations of the proposed systems and their utilized reasoning approaches?

RQ7: And, what are the main challenges of reasoning in a smart living environment?

II. **Defining the review protocol**
The review protocol specifies the methods to be followed while conducting the systematic review. As discussed by [20], the protocol should define the article search and collection strategies, the primary studies selection criteria and data extraction, analysis and dissemination strategies. The rest of this section discusses the main procedures of our review protocol and how we practiced it to conduct this SLR.

2.2.2 Conducting the review

This phase is composed of the following main activities: *article collection, article selection, data extraction and data synthesis*. Further, each activity is composed of other sub-activities.

I. **Article collection**
One aim of an SLR is to find as many primary studies as possible related to the research questions, using a repeatable search strategy [20]. Apart from carrying out a comprehensive and exhaustive search for the primary studies, it is important to define and strictly implement a well-defined search strategy. Therefore, for our article collection process, we defined and applied the following two sub-activities.

(i) *Definition of the search query*: to build the search query shown in figure 2.2, we adopted the following steps recommended by [22]:
  - Derive major search terms from the research questions.
  - Collect keywords from known primary studies for additional main search terms[1].

---

[1] As part of the search query, along with Reasoning System, Decision Support System, and Expert System, the use of Artificial Intelligence was considered. However, after some preliminary article collection process, it gets eliminated for two reasons. First, the term AI is too broad, and the search query comprising AI resulted in downloading too many articles, often not in the smart home domain. Second, we noticed that most papers in the SHRS domain, which listed Artificial Intelligence in their keywords, also have at least one of the above three terms as part of their keywords, thus avoiding repetitions.

((“smart home” *OR* “smart building” *OR* “smart house” *OR* “smart living environment” *OR* “connected home” *OR* “context aware home” *OR* “context aware building” *OR* “context aware living environment” *OR* “building automation” *OR* “home automation” *OR* “domotic” *OR* “ambient assisted living” *OR* “active assisted living” *OR* “ambient intelligence”) *AND* (“reasoning system” *OR* “decision support system” *OR* “expert system”))

Figure 2.2: Basic search strings

  - Identify synonyms of the main search terms.
  - Construct search strings using Boolean *“AND”* to join the main terms and *“OR”* to include synonyms.

(ii) *Conduct the search*: with the aim of performing an exhaustive search, we identified the following six electronic databases: *ScienceDirect, SpringerLink, IEEExplore, ACM Digital Library, Google scholar and Semantic scholar*. Then, we performed an automated search on each of the sources, except ACM Digital Library[2]. When required, we adapt the basic search string to the search engine of each source.

The automated article collection was done through an in-house web crawler that is able to detect and ignore article repetitions. It stops the gathering process after detecting a sequence of *ten (10)* articles that are unrelated to the query of interest. The crawler determines the *“relatedness”* of an article by matching the title of the paper with a predefined set of keywords[3]. As shown in figure 2.3, the literature search in the aforementioned six electronic databases returned 2175 articles, contributed from 56 countries during the years 1996 through 2017.

II. **Article Selection**

To select appropriate studies for inclusion in the review, the following two sets of inclusion and exclusion criteria were specified:

(i) *Initial Selection:* as the number of collected papers were too large to consider for full-text analysis, in this phase, we evaluated the papers based on their abstract, title and list of keywords. The inclusion and exclusion criteria applied at this phase of selection are listed below. Accordingly, a paper was admitted into the next phase of selection, if it met all the inclusion criteria and excluded if it met at least one of the exclusion criteria.

Inclusion criteria:

---

[2] ACM End-user policy specifically prohibits automatic downloading of articles.

[3] The set of related keywords was defined according to the reviewers’ subjective knowledge and experience on the field.

- Articles whose title indicates that it deals with reasoning systems in smart living environments.
- Articles whose keywords match with some of the search terms defined.
- Studies that present very general decision support techniques for pervasive or ubiquitous computing domain, and that can be directly applied to smart living environments.
- Articles that focused on smart home inhabitants activity recognition, learning and monitoring were included as potentially relevant contributions, to determine if their proposed techniques can be applied to the problem domain.
- Studies on home appliance optimization, planning and scheduling are included, only if the results of these studies are used for decision making/reasoning processes in the home environment.

Exclusion criteria:

- Duplicate reports of the same study, in this case only the most recent and complete version is included.
- Papers that are not published in English language.
- Studies that focused on smart grids, smart cities, non-residential buildings, and outdoor intelligence services.
- Studies tailored for specific smart home appliances (e.g. smart fridge, smartphone, smart mirror, domestic robots...), but do not generalize their solution for smart living environments.
- Grey literature

In addition, as abstracts might be insufficient to rely on when selecting, if there was any doubt whether a study should be included, it was added to the list of potentially relevant studies. The outcome of this selection stage was 472 papers, i.e., we excluded 1703 papers.

(ii) *Secondary selection:* in this phase, the potential of an article to be included as a primary study was assessed by skimming over the full-text of the contribution. An article must satisfy both of the following criteria to be part of the primary study.

A *Contribution:* The primary contribution of the study should be aimed towards the definition of theoretical foundation and/or the provision of empirical evidence (through implementation, tests, critical analysis, or critical evaluations) for constructing a reasoning system.

B *Context:* the study should be defined in the context of domotics and/or other closely related domains. In addition, a study proposed to address a broader concept but can be mapped into a smart home scenario, was considered to meet this criterion.

After the execution of this selection stage, we remained with 135 papers, which made the primary studies of this SLR.

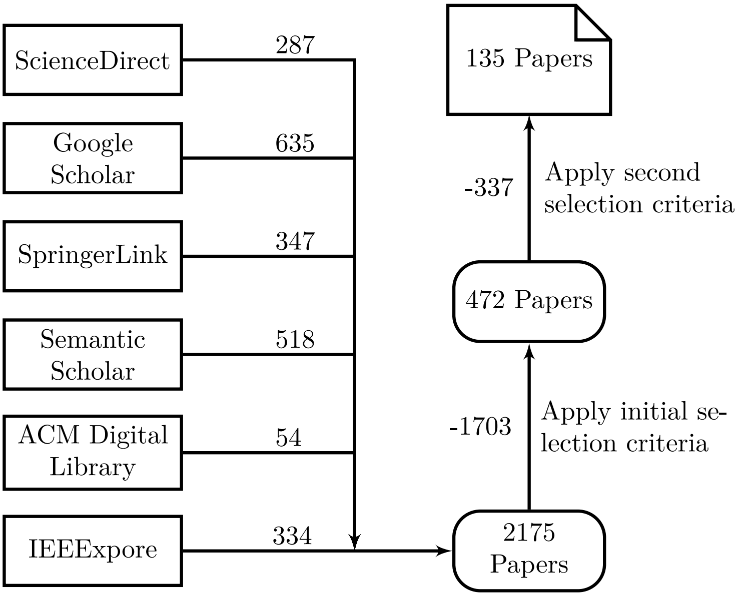

Figure 2.3: Article collection and selection process

III. **Data extraction**

The data needed to answer our research questions was extracted through extensive examination and review of the primary studies. Thereafter, the extracted data was recorded in a spreadsheet for subsequent analysis. Some of the fields of our data extraction form were: *publication year, country, targeted purpose, article abstraction, reasoning technologies (i.e. approach, methodology, tools and technologies), strengths, assumptions, limitations, and challenges of the primary studies.*[4]

IV. **Data synthesis**

During an SLR, the extracted data should be synthesized in a manner suitable for answering the questions that the SLR seeks to answer [20]. To this end:

- In section 2.3.1, the identified purposes of our primary studies are categorized into four smart home application areas and presented in a dia-

---

[4] The assumptions, limitations, strengths and challenges of an article were mostly determined based on the knowledge of the reviewers.

grammatic form. Further, a descriptive synthesis of their distribution is performed (addressing RQ1).

- In section 2.3.2, twelve (12) essential features and requirements of SHRS are identified, and a descriptive synthesis of them is presented (addressing RQ2).
- In section 2.3.3, a detail quantitative and qualitative analysis of the identified reasoning approaches, tools and methodologies is performed (addressing RQ3). Further, the extracted data is presented both in diagrammatic and tabular form.
- In section 2.3.4, the three groups of identified assumptions are discussed in detail (addressing RQ4).
- In section 2.3.5, the primary studies level of practicality is discussed in a quantitative way, and the result is presented in a diagrammatic form (addressing RQ5).
- And, in section 2.3.6 and 2.3.7 a descriptive synthesis of the strength and limitation of the primary studies is performed (addressing RQ6) and challenges of reasoning in smart living environments are discussed (addressing RQ7).

2.3 Result and Discussion

This section presents and discusses the result of our SLR while answering the above defined research questions.

2.3.1 Primary purposes of smart home systems

Our first RQ aims to identify the primary purposes of smart home systems presented in the literature. To this end, we assessed application areas of each primary study, and then based on their shared features and intended services, we arranged them into the following four groups.

(A) *General purpose:* an article was assigned to this group:

  - If it presented a very general or holistic smart home reasoning system, without limiting its solution to any specific service or scenario.
  - Or, if it dealt with a specific requirement or challenges of a smart home reasoning system, nonetheless we considered its contribution to be adaptable into a more general solution with no or little effort.

(B) *Comfort-centered services:* an article was assigned to this group if it proposed to provide thermal, visual, audiovisual, air quality or other comfort related services.

(C) *Home Energy Management (HEM):* an article was assigned to this group if it proposed a solution for energy efficient use of home appliances or the reduction of home energy consumption.

(D) *Elderly Monitoring and Healthcare (EM-HC):* an article was assigned to this group if it proposed a solution towards inhabitants health status monitoring or assistance for ageing people to perform their everyday tasks.

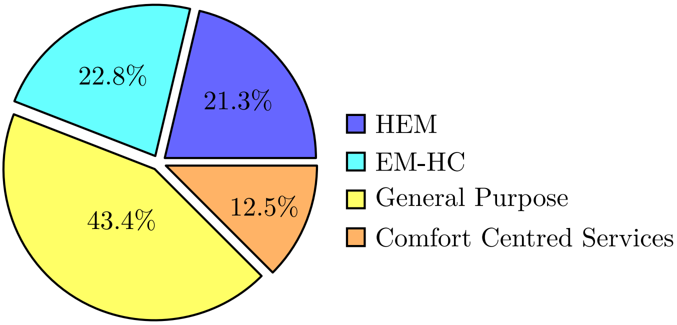

Figure 2.4: Primary purposes of smart homes

And, as shown in figure 2.4, most of the examined primary studies presented a general purpose smart home system. Whereas, this SLR identifies HEM and healthcare related services as the other widely presented smart home applications in the scientific contributions. On the other hand, systems which proposed their solution for Heating Ventilation and Air Conditioning (HVAC), light control, CO and CO_2 monitoring and entertainment, which we collectively categorized as comfort centered services above, covers eighths of the examined studies.

The major reason for the large research interest on health care and elderly monitoring related smart home systems can be the fast-growing number of people living alone in developed countries and their increasing need for home care services. Whereas the rapid increase in energy demands, the shift towards dynamic electricity pricing and demand response applications, together with the fact that the building sector represents more than 40% of the total world annual energy consumption [4], can be considered as the major factor for the substantial number of research contributions on automation systems for home energy monitoring and consumption reduction applications.

2.3.2 Features, characteristics and requirements of SHRSs

Aimed at answering RQ2, we thoroughly analyzed the data extracted from the primary studies of this review, and identified the following 12 closely interdependent and sometimes overlapping fundamental features, characteristics and requirements of a smart home reasoning system:

- *Activity and situation awareness:* most of the examined articles stated that a smart home reasoning system must be capable of understanding the status of people, things and various devices in the home environment. They also claimed that it should be aware of the ongoing activities and the present situation in the home. In addition, [23] stated that to automatically control the home environment, the system needs to be aware of the structure of the entire building, including the list of available device control commands, their locations and other constraints.

- *Reactive and proactive decision-making:* [24, 25] stated that the reasoning system should be reactive to vocal and other user commands, it should have the ability to detect status changes and proactively determine the adaptation needed to execute the relevant command and adapt the home environment to its inhabitant's needs. Likewise, [26] strongly recommended for the system to have two control modes: autonomous control and event-based control. The autonomous control will have a total authority on the environment by anticipating users' needs based on the learned information. Whereas, the event-based mode will react to inhabitants activities and enable the system to trigger control commands based on immediate user actions.

- *Context awareness:* in AAL environments the same sensor data, inhabitant activity or user interaction may have different meanings according to the changes in time and location. Therefore, in order to make sure that the command is executed smoothly on the target smart home device, and renders high-quality services, the reasoning system should consider these contextual changes while executing user command or during the self-adaptation process. For example, if the command *'turn on the light'* is pronounced in a bedroom with many lamps, the user may not specify which lamp(s) to turn on because he expects the system to guess it. Moreover, it would not be natural to ask the user to specify any details of a command, as it is up to the system to check the context and make the most relevant decision [24].

- *Ability to learn:* most of the examined articles declared self-learning as an essential component which needs to be integrated into every smart home reasoning system. More specifically, [27] described learning inhabitants' patterns of daily activities as a crucial component of smart environments.

- *Ability to predict:* some of the examined articles declared an ability to predict the future states of the inhabitants and their environment as another key feature of the SHRS. For example, [28] highlighted the importance of this feature to provide high-end services by computing users' situation with their context information.

- *Ability to plan:* some of our primary studies emphasized the need to integrate planning and scheduling components into the reasoning system. To this end, the works demonstrated the advantages of these features by employing them for successful implementation of routine smart home operations.

- *Reasoning under uncertainty and incomplete knowledge:* uncertainty is inevitable in AAL environments, as sensors may read inaccurate data or due to the existence of unobserved variables for privacy reasons, further vague human communication (such as *'turn on the light'*), and the dynamic nature of the system may all lead to incomplete, inconsistent and ambiguous contextual information. It is therefore important to build the reasoning system by incorporating different decision-making strategies so that it demonstrates robust and reliable reasoning performance under a variety of circumstances.

- *Conflict detection and resolution:* the primary studies underlined also on the need to integrate a standard conflict detection resolution strategies into the reasoning system. More specifically, [29] stated that to meet the quality and comfort goals of building automation system while optimizing towards energy, carbon footprint, and cost-efficiency, the system requires an ability to handle large amounts of information and resolves conflicting behavior. Furthermore, along with system level conflicts, smart home inhabitants can also have conflicting interests and goals at the same time and in the same place. Hence the system should also be able to deal with these situations and work in a way that maximizes the satisfaction of its inhabitants as much as possible.

- *Reason about time:* as noted by [27], most activities in a smart home environment are time-dependent. Thus being able to represent and reason about the order in which the activities occurred and their duration is necessary for the diagnosis of the situation. Further, to capture these changes either in the form of activities or events some form of temporal reasoning is required [30]. Therefore, reasoning about time can be regarded as a cornerstone for all other components of the SHRS.

- *Explain, prompt and notify:* even though it was discussed only in a few of the primary studies, an ability to explain its decision is another key feature for any automated decision support system. For instance, [31] characterized an efficient smart home system as a system which should not only carry out automated actions on behalf of the user but also be asked to interpret (e.g. *explain why the lights are dimmed*) its decision. In general, a smart home reasoning system does not only need to understand the ongoing activities of inhabitants and automatically control the home environment, but it is also essential for the system to convey an understanding of how it makes its decisions. Thus, human users will have at least a basic understanding and appropriate trust in the automated decision.

In addition, the examined studies extensively discussed the need for the reasoning system to have a prompt and notify component. For example, [32] stated that for an application targeted to help Alzheimer patients, it is important to identify when an activity step has been missed or performed erroneously and deliver an appropriate prompt when one is required. Further, during detection of security breaches or unexpected electricity consumption by a particular device, the home automation system should notify the homeowner in the most convenient way possible.

- *Easy to configure and setup:* along with the continuous learning of user preferences and the subsequent adaptation of the environment to the learned behavior, it is also essential for the SHRS to get feedback and new configurations directly from its users and operators. Accordingly, many of the examined articles argued that the reasoning system should be designed to be easily extensible, customizable and configurable to meet the different demands of the homeowner. Specifically, [33] claimed that the time needed to set-up the system into a new environment should be reasonably low. Likewise, [34] stated that users should be allowed to add a new device or rule with little effort, without affecting the running of the application. Further, [26] added that the inhabitants should have the chance to give feedback on the automatic decisions of the system or to override it all using different types of user interfacing such as web interface, touch screen unit, microphone and text messages.

- *Predictable behaviour:* as discussed earlier, it is vital for a smart home system to actively learn from its inhabitant's daily activities, and automatically adapt to their changing behaviors. However, the automatic decision making and self-adaptation operations must not lead the home into an incomprehensible, unpredicted and uncontrollable state. Most of the examined articles did not discuss this requirement, yet few contributions such as [35] presented the need to counterbalance the autonomy and reasoning power of the system by the ability of the users to fully anticipate and control the system's decisions.

- *Natural interaction:* another interesting requirement of the SHRS identified by this SLR is the ease of use and the level of interaction the system requires to operate, from User-centered Design (UCD) and Human Computer Interaction (HCI) principles perspective. [24] briefly discussed this feature and claimed that AAL decision support systems should provide a very natural and easy to use interaction mechanism for its human users. Further, [36] declared that the system should need minimum effort to operate from the inhabitant and they added that the less the inhabitant has to know about this system is the better. In general, a smart home system should be designed by strictly adhering to the fundamental UCD and HCI principles.

Besides the aforementioned twelve features and requirements of a smart home reasoning system, most of the articles that we examined for this SLR, strongly suggested a smart home system, especially its decision support system must be designed by putting in consideration the reliability, safety, and security requirements of the inhabitants, and must also guarantee the privacy of the data generated at home. Furthermore, they argued that it should be designed to be interoperable, as it needs the ability to interface with any electrical and electronic devices in the home [37]. Computational speed of the decision process and its response time must also be considered as the system needs to provide a real-time response to users' behavior [38]. Moreover, some of the studies reported that the reasoning system must be designed in a modular way and needs to be tested carefully before shipped out for an actual use. On the other hand, [39] shared an interesting point of view on assessing psychological and social science aspects of a reasoning system while designing AAL environments. The study also pointed out the importance of some additional scientific backgrounds, which are necessary to model the emotional and psychological state of the user that interacts with an immersive computing framework.

2.3.3 Reasoning approaches

The results of RQ3, which explores the reasoning approaches utilized to bring intelligence into the living environments, is discussed by categorizing the primary studies into the following four groups.

- *Symbolic approaches:* contains contributions which are mainly based on symbolic AI techniques. Logic-based systems, semantic web and ontologies, search, optimization and planning, and other symbolic models are part of this group.

- *Statistical approaches:* contains all examined contributions which are based on statistical and probabilistic artificial intelligence techniques.

Machine learning, Data mining, Markov model and Bayesian inferences are part of this group.

- *Hybrid approaches:* contains contributions which are based on the combination of the above two approaches.
- *General:* contains contributions which proposed general architectural or algorithmic solutions that can be implemented by either of the aforementioned reasoning approaches.

As shown in figure 2.5, more than half of the primary studies are based on symbolic approaches. Whereas statistical and probabilistic AI techniques cover nearly a quarter of the examined studies, and articles based on hybrid approaches are 18.4% of the total examined contributions. The wide usage of symbolic AI techniques can be explained by listing several reasons. First, these methods are long established, mature and generally considered a competent approach when one thinks about building a decision support system. In addition, most symbolic methods are relatively easy to develop and understand. Moreover, the facility provided by semantic web technologies to model complex context-aware systems like AAL, to share and reuse this contextual knowledge and to infer new knowledge from it, also played a key role in the popularity of symbolic approaches. Furthermore, contributions which mainly targeted to provide home energy management applications widely utilized search and optimization techniques to control the energy consumptions of home appliances. Whereas, factors such as limited availability of datasets, computational intensity, and non-reusability of models hinders the number of statistical and probabilistic contributions.

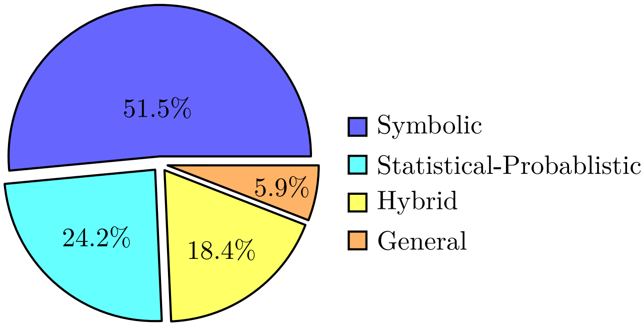

Figure 2.5: Reasoning approaches

A more detailed distribution of the reasoning techniques applied by the examined articles is depicted in figure 2.6. As shown in the figure, knowledge-based approaches are the most popular, followed by machine learning and data mining methods, which covered 32 of the reviewed studies. On the other hand,

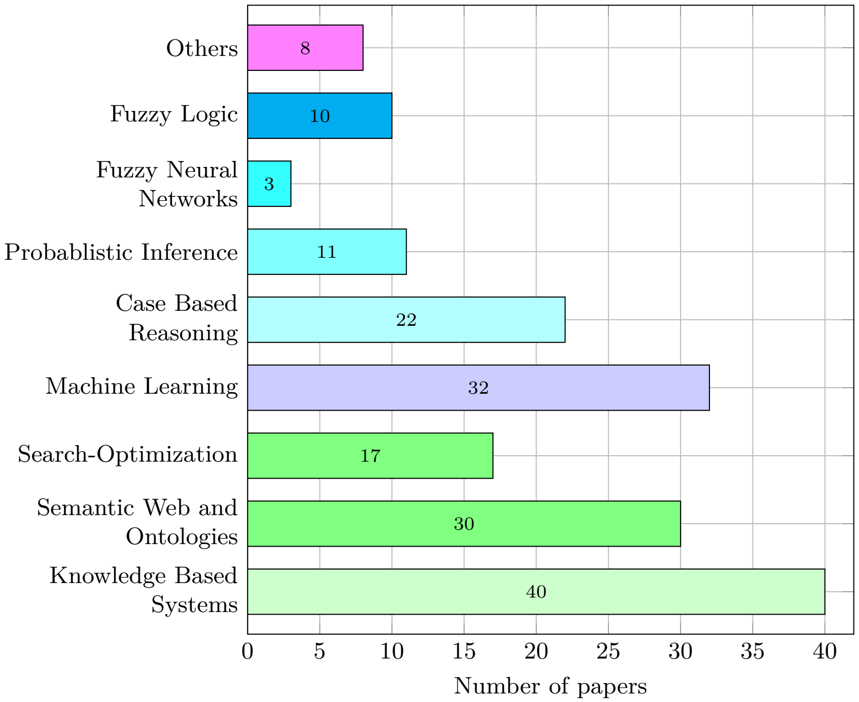

Figure 2.6: Detail distributions of the reasoning approaches

22 of our primary studies applied probabilistic inferences techniques such as Markov and Bayesian networks, mainly to handle the uncertain nature of living environments. Additional 30 research contributions utilized semantic web and ontology technologies. Besides 17, 22 and 10 articles were presented based on Search and optimization, Case-based reasoning and Fuzzy logic techniques respectively.

Figure 2.7 shows the reasoning approaches application trend over the past two decades. As can be noted on the figure, starting from the mid-2000's the research contribution towards AAL reasoning systems starts to grow. As also discussed by [1] the establishment of the European AAL Joint Association program during that period can be considered as one reason. Further, the same figure shows that the interest in applying symbolic approaches for SHRS, which based on our primary studies started in 1996, is decreasing since the year 2012. In contrast, the number of statistical and hybrid contributions, which again based on our primary studies started around the year 2001 and 2006 respectively, are relatively growing since 2011. A reason for this can be the groundbreaking results that are currently being achieved by machine learning and data mining methods in wide application domains, and the growth in processing capacity of modern computers. In addition, as the research on automated decision support systems is long established and well studied, the rising interest in hybrid approaches can be due to researcher's growing awareness of the strengths and limitations of both symbolic and statistical approaches, and the subsequent need to combine the advantages of these approaches.

Table 2.1 shows the complete list of decision-making methods, techniques, tools, and technologies used by the examined studies. And, table 2.2 shows the list of smart home sensor and actuators extracted from the primary studies of this review. In most of the statistical AI-based systems, the data generated from the sensors coupled with the control parameters of the actuator devices, are used as a training set to learn the model of the reasoning system. Whereas, in knowledge-based systems, the input-output parameters of these devices are directly encoded into the rules of the decision-making unit.

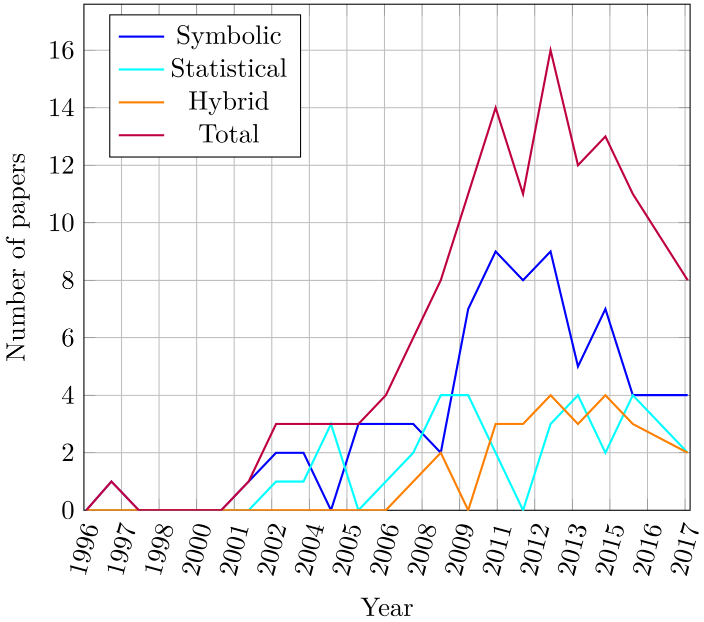

Figure 2.7: Application trends of the reasoning approaches

Table 2.1: Complete list of decision-making methods, techniques, tools, and technologies

Methodology and Techniques				Tools and Technologies			
Name	# of Papers	Name	# of Papers	Name	# of Papers	Name	# of Papers
Description Logic	3	Support Vector Machine	4	Golog	1	jCOLIBRI:CBR Framework	1
Time Ontology	1	Artificial Neural Network	9	SWI-Prolog	2	ConcurTaskTrees (CTT)	1
Defeasible Logic	2	Case Based Reasoning	9	Eurler Inference Engine	3	Mamdani- Fuzzy Inference	1
Allen Temporal Logic	1	Reinforcement Learning	3	Apache Jena	5	Netica:Bayesian Network	1
Situation Calculus	1	Clustering Algorithms	7	JESS Rule Engine	7	Choco Solver	1
Predicate Logic	2	Decision Tree	4	SPARQL and RDF	8	Clingo ASP Solver	1
Event Calculus	1	k-nearest Neighbors	3	Web Ontology Language	14	CPN Tools	1
Rule Based System	15	K-means clustering	3	SWRL	8	Java	8
Rule Nets	1	Expectation-Maximization	1	Protege	8	C#	6
Formal Concept Analysis	1	Fuzzy Neural Networks	3	Pellet	6	Python	1
Markov Model	8	Fuzzy Logic	10	Spindle	1	c++	1
Bayesian Networks	9	Belief-Desire-Intention paradigm	6	SunSPOT	1	Visual Basic 6	1
Dynamic Evidential Network	2	Fuzzy Cognitive Maps	1	Logtalk	1	PHP	1
Dezert-Smarandache Theory	1	Critical Path Method	1	JADE Agent Platform	6	Scala	1
Dempster-Shafer Theory	2	Thayer's Emotional Model	1	Microsoft Agent	1	.NET	1
Constraint satisfaction	2	Multiple-Criteria Decision-Making	1	SPADE Agent Platform	1	R Software	1
Search-Optimization	11	Stochastic Dynamic Programming	1	NetLogo	1		
Petri Nets	2	Prometheus Methodology	1	WEKA(Machine Learning)	2		
Finite State Machine	1	Gorgias Argumentation Framework	1				
Timed Automata	1	ASEME Methodology	1				

Table 2.2: List of smart home sensor devices that generate data about the inhabitant and its environment in order to perform reasoning. And, the list of smart home actuator devices, which the reasoning system controls as a result of the decision-making process.

Actuators (Output devices)		Sensors (Input devices)	
Type	Name	Type	Name
Home appliances	TV, Refrigerator, Microwave oven, Smart plugs Washing machine, Dishwasher, Rice cooker Electric water boiler, Steam cooker, Range hood Cloth dryer, Hair dryer, Solar panel	Medical sensors	Blood pressure, Body temperature Heart rate sensor, Respiratory rate monitor Fall detector, Pulse oximeter, Stress level detector Electrocardiogram (ECG), Pill weight measurement
Environmental comfort control devices	HVAC system, Air cleaner Fan, Light bulbs Blinds, Motorized curtains	Environmental sensors	Temperature, Humidity, Luminance, Air speed Thermal, Smoke, Indoor air quality, Noise Solar radiation, Gas leak detector
Communication devices	Multimedia devices, Intercom, Telephone Video streaming device, Speakers	Energy and water consumption	Water consumption, real-time electricity price Smart power measure
Electrical and electronics devices	Energy storage, Heat pump, Photovoltaic (PV) PHEVs, Other electronics	Others	Biometric sensor, Smart watches Magnetic door/window, Window break sensor Pressure sensitive bed, Floor sensors, Motion sensors, Positioning system, IButtons Location tracking system, Reed switch, TV status Water tap open/close, Bathhub sensor, Accelerate Soil moisture sensor, Proximity sensor, Camera Microphone, Google Calendar API, Weather station Seat pressure sensor, RFID sensors Contact sensors, Magnetic sensor
Safety and security devices	Door looker, Fire alarm system Alarm system, Fire extinguisher		

2.3.4 Assumptions

Towards answering our fourth RQ, we analyzed the extracted data and spotted three levels of assumptions made by the primary studies while presenting their respective works. First, we located assumptions about the operating environment of the reasoning system (i.e. the home), its devices and the data generated from these devices. The second group of observed assumptions was regarding the smart home inhabitants and their interaction with the environment. Finally, we observed sets of assumptions concerning the reasoning system itself.

I **Assumptions about the home environment**
In some of the examined articles, a room is defined as an intelligently controllable area of the home and used for a specific activity of the dwellers (i.e. cooking, sleeping, personal hygiene, or relaxing). Thereupon, they assumed each room to have a dominant user, user group and a predefined list of priority order between the smart home services (e.g. inhabitant comfort over energy saving). As it is very hard to define the priority constraints in rooms which tend to have interpersonal interactions, the existence of a user interface to define the priority order and dominant user of that particular space for a given amount of time is assumed. In general, most of the articles which made these assumptions mainly targeted to address other challenges of a smart home reasoning system while taking for granted the non-existence of conflicting inhabitants needs in the environment. Besides, most of the primary studies considered the home environment as fully observable. To further simplify the reasoning process, some contributions such as [40] assumed each sensor data to have a constant/fixed weight over time (e.g. temperature 22 in winter and summer has the same value, however, this may not be true all the time) and to produce contextual information. Likewise, [27] assumed the possibility of associating all generated events with a linear, discrete and totally ordered representation of time. On the other hand, studies targeted on HEM services assumed the availability of price signal, electricity usage history, and consumption profile in real-time. Further, [41] assumed constant power consumption of a smart home device while performing distinct operations (e.g. a dishwasher performing washing, drying and disinfection).

II **Assumption about the inhabitants**
In order to ease the challenges that could raise by overlapping and simultaneous inhabitant activities, the majority of examined articles presented their contributions by assuming the existence of single inhabitant in the

environment. In addition, a significant number of the contributions assumed the inhabitant to be rational all the time. However, as we discussed in section 2.3.1, around a quarter of our primary studies are proposed to provide elderly monitoring and health care services. Besides, most of the residents in these environments are known to be elderly who may suffer from cognitive impairments and who do not have a consistent intention or persistent goal [42]. As a result, we found this assumption to be weak, since these group of people cannot be assumed to be regularly rational. Moreover, other investigated articles made various assumptions regarding what constitutes 'normal' or 'abnormal' behaviors and 'if', 'when' and 'how' a specific operation should be performed [43]. For instance, [44] assumed the elderly to be unsafe if s/he remains at a specific space beyond some threshold. In general, some of the AAL systems which targeted to provide EM-HC services assumed the availability of information regarding the health status and ongoing activities of the residents in real-time.

III **Assumption about the reasoning system**

Most decision-making approaches are known to operate under different assumptions. Therefore to keep our discussion concise, this section avoids presenting specific assumptions of every reasoning methods utilized by the examined articles. Instead, this section focuses on the fundamental assumptions of the primary studies, which are assumed for the proper functioning of the reasoning system in the AAL environment. Accordingly, this review noted that most examined contributions proposed their systems by assuming the presence of complete, consistent and non-conflicting knowledge. The decision support system proposed by [41] assumed to have the prediction ability of electric power consumption and production. Whereas, the system proposed by [45] assumed to have inhabitants identification and recognition capability, so that it monitors and secure the home environment. On the other hand, the contribution made by [46] presented a human-assisted decision support system to control a partially-observable environment. For this, they assumed the system to interact with its inhabitants during the decision-making process, specifically asking the residents to provide missing information or to perform some tasks that the system cannot perform due to lack of corresponding actuators.

In general, most articles presented based on symbolic approaches assumed to have complete knowledge about the environment. And, as discussed by [47], in supervised learning algorithms the influence of cultural effects and habits are assumed to be negligible since their inclusion as a feature is nearly impossible. Further, to build a fuzzy architecture that would support learning from user actions and proactively anticipate users' needs based on the learned data, the work presented by [36] assumed the non-existence of a predefined rule base. Whereas [42] assumed the presence of a plan base enumerating all possible action paths in order to infer the possible goals pursued by the inhabitant and predict their next action. Likewise, [48] assumed the existence of a statistical pattern database to maintain a comfortable living environment.

On the other hand, studies which proposed a distributed decision support system mainly utilized multi-agent system architecture to present their solutions. In these systems, agents are assumed to be rational, collaborative and willing to disclose and share part of their local knowledge with other agents in the system. Further, the existence of a reliable communication channel for the agents to exchange unlimited messages with other ambient agents is assumed.

2.3.5 Article abstraction

To help us analyze and understand the research contribution's level of practicability, we categorized the primary studies of this SLR into the following three groups:

- *Conceptual:* an article was assigned to this group if it presented only a theoretical research work.
- *Simulated:* an article was assigned to this group if it demonstrated or tested the feasibility of its theoretical contributions within a simulated environment.
- *Tested in a real-world environment:* an article was assigned to this group if it tested or integrated its proposed solutions in real-world smart living environments.

As shown in figure 2.8, more than half of the examined studies tested the feasibility of their contributions in a simulated environment. However, most of these contributions did not test their proposed systems using an established AAL system simulator such as the one proposed by [49], [50] and [51], instead, the contributions were tested as an ad-hoc implementation. The studies which are evaluated in an uncontrolled real-world environment were only 18.5%. From these studies, we noted that the real-world tests allowed them to thoroughly examine the strengths and limitations of their contributions and to clearly illustrate the feasibility of their theoretical contributions. The rest 26% of our primary studies presented entire theoretical methods, without any evaluation of their proposed approaches. As a result, we conclude that with some effort, most of the investigated scientific contributions can be used to build part of a

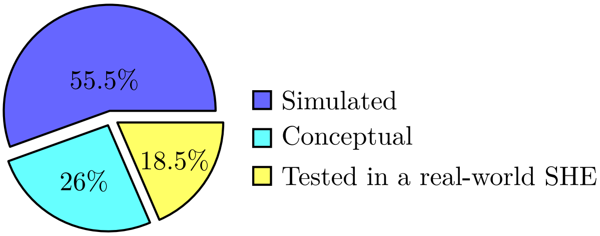

Figure 2.8: Article abstraction

smart home reasoning system. Further, we noted that besides the theoretical contributions and the simulated experimental studies an increased number of research contributions in the AAL domain need to be tested in a real-world environment. However, the complexity of the system and the cost of experimentation can be stumbling block for the evaluation of the contributions in a realistic setting. As a result, the practical aspects of the proposed methods can be left unknown [24].

2.3.6 Strengths and Limitations

Towards identifying the strengths and limitations of the primary studies, first, we analyzed the number of contributions which are not strictly attached to a specific AAL scenario or application. Subsequently, we determined the contributions that can possibly be used for a more general smart home reasoning system. As shown in figure 2.9, 94 of the examined articles discussed their solutions by indicating the potential of their research works in multiple AAL scenarios.

In addition, 46 of the primary studies proposed different reasoning modules which are able to detect, learn and adapt to user's preference changes by actively building upon their initial knowledge. Whereas a few other articles such as [52] presented a system which works by directly accepting user's explicit and implicit feedback. On the other hand, the algorithm proposed by [53] adapts to a change in an activity pattern after the third occurrence of a specific change in that specific pattern. In general, we noted that most other contributions either did not address the need for learning and adaptability or presented with a set of predefined static rules without the capabilities of self-learning. As a result, an adaptation in these systems is only achieved if the user defines a new set of rules each time his/her behavior changes [54]. Based on our discussion in section 2.3.2, these home automation systems cannot be regarded as complete, because they lack the ability to automatically adapt to their inhabitant's needs.

Again, based on our discussion in section 2.3.2, the reasoning system needs to be easily controllable and configurable, and its communication with its users should be as natural as possible. To this end, 55 of the reviewed articles provide some form of communication interfaces to allow users to express their preferences, define inhabitants expected behavior, add or remove rules and control devices. Moreover, some of these systems provide local and remote access to control the home environment through web interfaces and mobile applications. They are also able to stream information about the current condition of the home, and few of them are voice controllable. However, as can be noted in figure 2.9, only six of our primary studies discussed explanation mechanisms that enable experts and inhabitants to check how and why each solution has been obtained.

Conflict detection and resolution is another key feature of a smart home reasoning system that we identified previously. And as shown in figure 2.9, 26 of the examined studies directly or indirectly presented this feature as part of their proposed system. To this end, most of these studies widely applied defeasible logic techniques or defined priority orders between production rules. Further, as part of their proposed systems 33 of the examined studies tried to improve the confidence level of contextual information, reduce conflicting sensory data, and derive high-level contexts from low-level raw sensor data by using ontology-based models and data fusion techniques. As discussed by [55], the use of ontologies for data modeling and representation serves two purposes for these systems. First, it provides a formal way to model and represent interrelations between the data from multiple sources, thus facilitating data fusion and the construction of situations. Second, it gives the generated data rich metadata and well-defined meaning, thus enabling automated comprehension of the situational information significance.

To address a similar challenge, 29 of the primary studies presented a reasoning framework for decision making under uncertain circumstances. Most of these studies adopted fuzzy logic and variety of probabilistic machine learning approaches to work with the vague, ambiguous, imprecise, nosy or missing information generated from smart home devices. However, we observed that most other contributions simply neglected to handle the problems caused by the imperfect nature of contexts in the smart home environment. Furthermore, 26 of the investigated articles considered temporal knowledge representation, temporal constraints, time-dependent rules and some other time-dependent behaviors in their proposed decision support system.

In addition, 30 of the examined articles presented their solution as a distributed decision support system mainly based on multi-agent systems. From the presented results of these articles, we noted that modeling the system in a distributed and modular architecture allowed them to effectively tackle the challenges raised by the largely open and pervasive nature of the smart living environments. Moreover, some of these studies demonstrated the flexibility, modularity and fault tolerance advantages of distributed expert systems in their contributions. In contrast, the rest 105 examined contributions are presented either as a centralized solution or as a single component of the reasoning system. And, even though homes are places where usually more than one inhabitants live in, surprisingly all except three of the examined contributions neglect to deal with simultaneous, overlapping and sometimes conflicting activities of multiple inhabitants. Further, 17 of the examined articles explicitly declared their contributions as a system that can only handle scenarios of a single inhabitant.

Besides, a smart home system needs to be secured, and it should preserve the privacy of the data generated in the home environment. To meet these requirements, a variety of security and privacy mechanisms can be integrated into different components of the smart home system. For instance, firewalls, authentication and authorization methods can be used on the residential gateway, and security and data encryption technologies of the network protocols can be utilized to guarantee the secure data transfer between different components of the system. However, as the primary focus of this review is the smart home reasoning systems, we limit our investigation only in examining how the primary studies tackled security issues in AAL environments at the reasoning system level. And, as shown in figure 2.9, only four of the primary studies deal with security issues, and all the other contributions completely ignored the problem. In this regard, [37] discussed the security and privacy of user data in the design and specification of a smart home system and proposed a security monitoring module as a single component of the entire system. Likewise, [56] presented a security mechanism that integrates context-awareness with automated reasoning, to perform authentication and access control in ubiquitous computing environments. However, both of these contributions are only focused on controlling unauthorized access (such as burglary) into the home environment and do not deal with other smart home security issues. On the other hand, [48] and [57] introduced an activity-aware approach to home security monitoring and threat detection. These studies mainly utilized activity learning techniques to detect possible threats and anomalous events in the smart home sensor data and take appropriate actions. Nonetheless, none of these four articles deals with major smart home security threats, which are common in AI-based systems (such as adversarial attacks) and internet-of-things based environments (such as Denial-of-service attacks). In general, this result shows addressing the major security issues of a smart home system at the reasoning system level is under-explored and need to be studied more in the future.

In general, knowledge-based decision support systems offer a human-readable, easily understandable and end-user modifiable representation of rules. In addition, the behavior of these systems can be validated by human inspection and verified automatically. However, the disadvantages of most examined rule-based expert systems are their inability to learn and adapt, and their relatively slow response rate if a large set of rules is considered [53]. In addition, hard rules do not provide any means of probabilistic reasoning and can infer only if complete information is provided [58]. Moreover, designing a complete rule-based system is considered to be almost impossible, as the system designer cannot envision all the situations of the smart home environment.

On the other hand, due to the recent breakthrough in statistical and probabilistic AI techniques, an increasing number of research studies extensively utilized a variety of machine learning methods to effectively tackle some of the major challenges of the smart home system. For instance, [59] combined Bayes Belief Networks, Multilayer-Perceptron, and Sequential Minimal Optimization to recognize activities of daily living (ADL) in smart environments. And, [48] applied Support Vector Machines to detect anomalous situations and actions in smart home data. Likewise, [60] utilized Self-Organizing Maps to successfully predict energy demands in the home. In the MavHome project [61] different machine learning techniques are employed for activity pattern detection, activity prediction, and anomaly detection. Similarly, [62] utilized Convolutional Neural Networks and Deep Q learning techniques to learn the pattern and behaviour of the inhabitants and to determine the subsequent actions that need to be triggered automatically. In general, most examined articles, which are based on statistical AI approaches exhibit good prediction and adaptation capabilities. They are also characterized by their ability to learn user preferences and improve over time. However, a large part of these contributions are focused on a specific problem domain, and their proposed solution cannot be considered as an all-round smart home reasoning system. In addition, building a complete decision support system based on statistical approaches is very challenging, as it needs a suitably large dataset of the entire activities and operations in the AAL environment. Often, the reusability of these systems is limited to the environment and scenarios that have produced the dataset. And, the decision process and outputs of these black-box-like systems are hard to understand and unreadable by end-users. On top of these, most statistical approaches such as Artificial Neural Network (ANN), Decision Tree, Support Vector Machine, and others have a limitation in modeling unambiguous and incomplete data.

Whereas, most examined hybrid decision support systems combined the advantages of the aforementioned two approaches. These systems utilized rule-based inference to reason on pure sensor data, when a predefined condition is met and, applied statistical inference approaches to predict and learn inhabitant's behavior. Further, some of these systems also adopted probabilistic reasoning techniques and considered spatiotemporal related issues to reason under uncertainty.

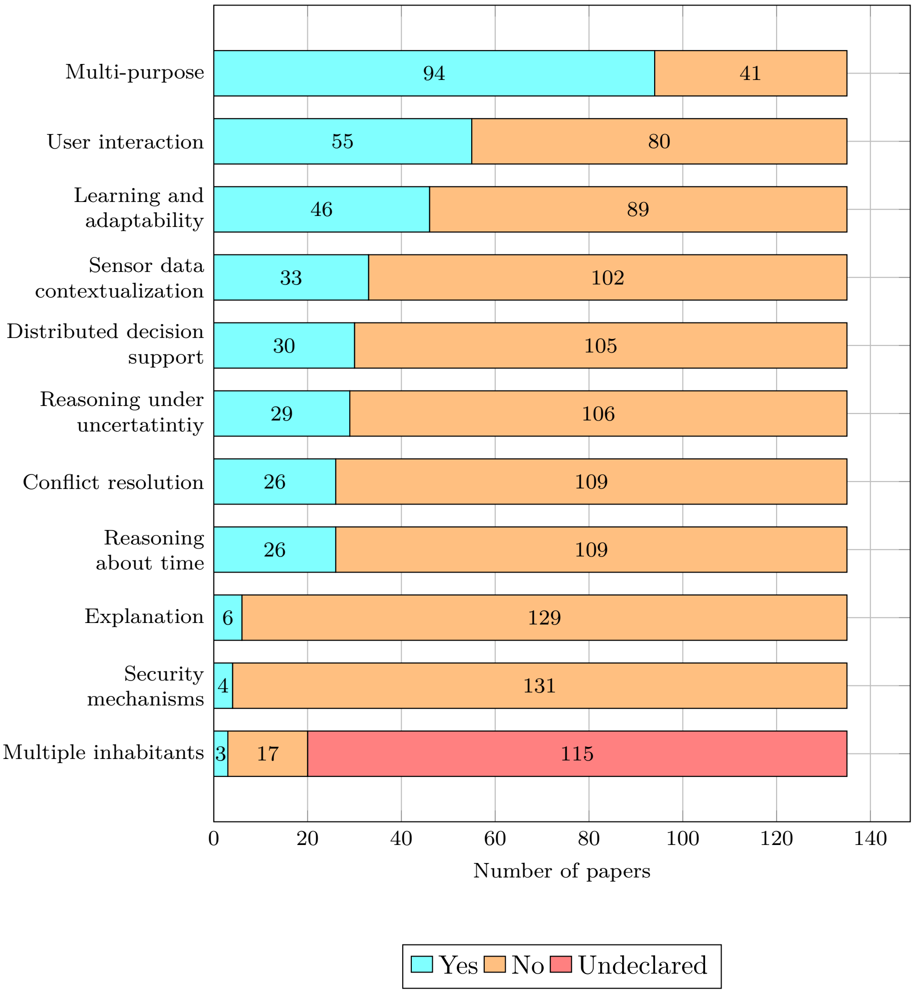

Figure 2.9: Result from the evaluation of the primary studies

2.3.7 Challenges of reasoning in smart homes

Towards answering our final research question, we carefully analyzed the challenges that the primary studies of this SLR encountered while building a reasoning system for smart living environments. As a result, we identified the following six main categories of challenges that most of the examined studies discussed in their research contributions.

- *Complex, dynamic and partially observable environment:* even though most smart home services and devices are embedded in the living environment, some others (such as smartphones) dynamically leave and join the home network. This dynamics and mobility of devices present a significant challenge in planning, scheduling and controlling of AAL services. [63] highlighted this challenge by labeling a smart home environment as complicated and dynamic, because the whole infrastructure for all devices, network and communication interface cannot be defined in a fixed manner, and consequently makes the controlling of the environment challenging. In addition, [64] outlined the fact that the number of inputs for a smart home reasoning system cannot be fixed, as new sensors and actuators can be added to the system, the number of inputs will dynamically change. Accordingly, it is essential for the system to adapt and efficiently function in such a constantly changing environment. Similarly, the article presented by [29], described homes as relatively less organized environments, since the number of possible occurring objects, events and scenarios and the way how to react to them is almost infinite. On the other hand, [46] discussed the challenges of partially observable environments and the hurdles that they present on the AAL reasoning and decision making systems. Towards addressing this problem, the same article suggested a decision support system that can collaborate with human users, by asking users to provide the missing information or to perform some actuation of which the system is incapable of, and pinpointed recognition of the inhabitants state and their environment as a key challenge in building a smart home decision support system.

- *Heterogeneous data sources and unreliable data:* the reasoning process in smart homes involves the collection and aggregation of sensor data and requires the generation of high confidence level contexts from heterogeneous data sources. Further, due to faulty hardware, delays between production and consumption of the information, or even networking problems, unreliable, inconsistent and incomplete data are very common in smart environments [33]. As discussed by [65], this challenge becomes even more demanding when the AAL environment comes to serve the requirements of elderly people who need continuous monitoring and care. These all can lead to one of the most common decision-making problems in dynamic environments, that is how to make the best possible decision based on uncertain or incomplete information [66]. As we noted in the some of the examined articles such as [57], part of these challenges is tackled through preprocessing of sensor data, dividing it into manageable sizes, and then extracting the required feature. As we discussed in section 2.3.3, data fusion and probabilistic reasoning techniques have been widely applied to cope with uncertainty in AAL environments. However, the challenges of reasoning under incomplete information is not yet fully addressed by most of the examined articles, and it is identified as one of the core challenges of smart home reasoning systems, that hiders the design of a more practical smart home system for complex applications.

- *Activity modeling and recognition:* most examined studies identified modeling and recognizing simultaneous and overlapping Activities of Daily Living (ADL) in multi-resident environments as another key challenge of decision support systems in smart living environments. For instance, [67] pointed on the need to include several factors in an activity model, such as location, time and the physical objects related to that particular activity. Moreover, it discussed the fact that these factors may change over time, and how the change makes ADL modeling even more challenging. Inhabitant activity modeling and recognition can be even more complicated, as the same activity can be performed differently by different individuals, and the same individual could carry out multiple activities concurrently. Therefore, the activity model should be designed in a way that can capture the different varieties of the activities and determine which observed events belongs to that activity. Accordingly, [68] declared user activity modeling as a complex, challenging and difficult task. Further, even if recognition of inhabitants states and activities in multi-resident living environments is still a challenge, detection of discontinuous and interleaved activity patterns of the user is discussed as even more complex by some of our primary studies. To this end, the study by [68] listed four main challenges of activity recognition task in smart living environments: *concurrent activity recognition, interleaved activity recognition, multiple inhabitants and interpretation of ambiguity.*

- *Conflicting needs and changing user habits:* dealing with conflicting inhabitant preferences, particularly in a multi-resident environment is another significant challenge identified in this review. In accord with this, [43] stated that understanding and coordinating the different needs of multiple individuals living in the same environment remain major challenges for the ambient intelligent research community. For instance, efforts to reduce energy consumption while ensuring a comfortable living environment can lead to conflicts. In addition, simultaneous execution of pre-planned and disruptive operations can cause inconsistent system behavior. On top of this, inhabitants' habits may slowly change over time, accordingly, the algorithms of the system should have also the adapting ability to match it.

- *Context representation and contextual reasoning:* one of the examined studies [43] relates challenges of ambient intelligence technologies with points such as how to extract high-level contexts from the raw sensor data, how to integrate the extracted heterogeneous contexts, how to infer the highest level information regarding the current human needs, and how to control a context-aware application that automatically provide services according to human needs. Likewise, while presenting an ontology-based context-aware system for smart home, [69] discussed the collection of contextual information from heterogeneous sources, ontology management, ontology querying and issues related to database explosion as the key challenges of realizing the system. Similarly, [70] defined the computing environment of context-aware applications as ubiquitous and distributed. Subsequently, they identify the heterogeneity of context representation, the limitation of devices capacity and network bandwidth as the main challenges of context communication in these systems. And to tackle these issues, the authors indicate the need to design a context management system, which puts into consideration the resource-constrained devices and the restricted network bandwidth. In general, efficient and accurate representation of contexts can be seen as one challenge in these systems, generation of high-level contexts from raw sensor data, and control the proper functioning of the smart home system as another. In addition, as the generated contexts can be inconsistent and conflicting with each other, standard conflict detection and resolution methodologies need to be studied.

- *Availability of datasets:* unavailability of sufficient training dataset for realistic AAL scenarios, is another challenge highlighted mainly in some of statistical artificial intelligence based primary studies of this SLR.

In addition, the recent advancement and proliferation of the IoT present a multitude of challenges and opportunities for the development of smart home systems. First, the masses of data generated from the IoT devices about the inhabitants and their environment helps to build more accurate prediction and classification models. Second, these arrays of devices enable to effectively monitor and control the overall operation of the home environment and to provide a variety of smart home services with less cost and minimal human interaction. However, the large amounts of data generated from these devices can also impose a significant burden on the decision-making unit of the home, if not filtered and pre-processed starting from the data sources. In addition, some of these resource-constrained devices are unreliable, and hardware failures are known to be common in IoT based systems, these could result in a considerable data loss, and ultimately lead the system into uncertainty. On top of these, the heterogeneity of the data generated from these devices, and the distinct network protocols they use to communicate with, could make the utilization of these devices complex. And, most importantly, the capability of most smart home IoT devices to connect to external networks (such as the Internet) make them vulnerable to different types of security attacks. This could allow attackers to acquire confidential data, alter the command being transmitted to actuator devices, or compromise the entire operation of the smart home. And ultimately, it will lead to major safety and privacy violation. Hence, to benefit from the true potential of IoT technology in the smart home environment, future research should also focus on addressing research challenges in IoT interoperability, sensor fusion, contextual and uncertainty reasoning, smart home security and privacy issues.

2.4 Chapter Conclusions

This chapter presented the results of a systematic literature review targeted at exploring smart home reasoning system. The literature review was focused on answering seven research questions, and our main findings are the following:

- Most smart homes are designed to provide general home automation services such as light and HVAC control. Nonetheless, health care and home energy management are the most popular specific AAL services presented in the literature.

- The most essential features and requirements of an SHRS are *activity and situation awareness, context awareness, ability to learn and adapt, ability to plan, ability to predict, ability to explain its decision, ability to reason about time, and ability to reason under uncertainty.* Further, the SHRS needs to be equipped with a standard conflict detection and resolution strategy and should provide a natural interaction mechanism for its users.

- Most research contributions in SHRS domain are based on symbolic artificial intelligence approaches. However, over the past few years, the interest in statistical and hybrid approaches is relatively growing.

- Most research contributions in SHRS domain assumed the existence of a single inhabitant in a fully observable environment. Further, many studies in the same domain assumed the presence of complete, consistent and non-conflicting information about the home environment and an always rational inhabitant.

- Most research works in the SHRS domain are presented as theoretical contributions. Thus, to examine the proposed methods practical aspects, an increased number of contributions needs to be studied in realistic settings.

- Most contributions in the SHRS domain demonstrated the potential applicability of their proposed methods in multiple AAL scenarios. Further, a good number them proposed solutions for learning inhabitants' changing behavior, self-adaptation, reasoning under uncertainty and conflict resolution. However, very few of them integrates decision explanation mechanisms and, above all, only a very limited number of them handles overlapping and simultaneous multiple inhabitants' activities.

- The dynamic and partial observability of the AAL environment, the heterogeneity and unreliability of smart home data, the conflicting and frequently changing inhabitants needs, and context representation and contextual reasoning are the major challenges of building a reliable and practical SHRS.

In conclusion, to give a true intelligent behavior for the SHRS, at least most of the aforementioned features and requirements of the system need to be satisified, and the challenges presented by them need to be addressed. A starting point in overcoming some of these challenges can be combing the best of symbolic and statistical AI techniques, going forward towards a more hybrid and distributed reasoning system. Further, the integration of standard conflict detection and resolution strategies and contextualization of sensor data can improve the performances of SHRS. In addition, more research on the handling of multiple inhabitants ADLs need to be performed. Moreover, relatively simple and cost-effective ways to test SHRS in real-world settings need to be studied.

Chapter 3

A Probabilistic Multi-Agent System Architecture for Reasoning Under Uncertainty in Smart Home Environments

3.1 Introduction

Our investigation of SHRSs, which was presented in chapter 2 of this dissertation, identified reasoning under uncertainty and incomplete knowledge in smart home environments, as one of the major challenges of these systems. Specifically, it underlined that the existence of unobserved variables for privacy reasons, together with sensor failures and inaccurate data read, makes uncertainty prevalent in AAL environments. On top of these, the dynamic nature of the home environment and vague human commands (such as "*turn on the light*", without specifically telling the system which light to turn on) may result in ambiguous, incomplete and inconsistent contextual information, which ultimately lead the system into uncertainty. However, existing literature in the field has been paying little attention to address uncertainty issues in smart home environments. As a result, in this chapter we aim to tackle some of these challenges, in particular, *uncertainty due to vague inhabitant commands* and *missing information.* For this, we utilized the notion of multi-agent system technologies and a probabilistic logic programming technique and proposed a probabilistic multi-agent system architecture for reasoning under uncertainty in smart home environments.

Multi-agent systems are systems which are composed of multiple interacting autonomous intelligent agents. These systems can be used to solve problems that are difficult or impossible for an individual agent or a monolithic system to solve. Typically multi-agent systems research refers to software agents. However, the agents in MAS could equally well be robots, humans or human teams [71]. In general, a multi-agent system consists of intelligent agents and their operating environment. And, a multi-agent system architecture is considered as an ideal candidate to cope with the dynamic and distributed nature of a smart home environment [72]. As stated in [73], intelligent agents possess the following characteristics:

- Autonomy: agents operate without the direct intervention of humans or others, and have some kind of control over their actions and internal state.
- Social ability: agents interact with other agents (and possibly humans) via some kind of agent communication language.
- Reactivity: agents perceive their environment and respond in a timely fashion to changes that occur in it.
- Proactiveness: agents do not simply act in response to their environment, they are able to exhibit goal-directed behaviour by taking the initiative.

On the other hand, probabilistic logic is a semantic generalization of logic, in which the truth values of sentences are probability values, between 0 and 1 [74]. In general, this technique combines probability theory with logical reasoning, mainly to provide artificial intelligence applications with an ability to reason under uncertain information.

Based on these two technologies, in this chapter, we define multiple intelligent agents and show how the probabilistic reasoning technique enables the agents to reason under uncertainty using an ambiguous inhabitant command as a scenario. Further, we discuss how the intelligent agents enhance their decision-making process by exchanging information about missing data or unobservable variables using agent interaction protocols. In addition, we describe how the agents take advantage of agent negotiation protocols to delegate their reasoning tasks for other agents in the system. The feasibility of the proposed MAS architecture is studied using a Proof-of-Concept (PoC) implementation and experimental analysis of the system. In general, this chapter demonstrates that the combination of MAS technologies and probabilistic logic programming can help in building a reasoning system, which is capable of performing well under vague inhabitant commands and missing information in a partially observable environment.

The remainder of the chapter is structured as follows: Section 3.2 gives an overview of the related works. Section 3.3 briefly introduces the ProbLog language and agent interaction protocols. In section 3.4, we present our multi-agent system architecture and discusses the probabilistic reasoning process using a simple scenario. Section 3.5, evaluates the proof-of-concept implementation of the proposed system presenting the experimental settings and the test results. And, section 3.6 ends the chapter by presenting some conclusions and suggestion for further work.

3.2 Related Research

Over the years, several studies have exploited the basic advantages of intelligent agents to model smart home environments and to automatically control their overall operations. For instance, [75] introduced a multi-agent system architecture to provide health care services in the AAL environment. The core of the system is a Belief-Desire-Intention (BDI) agent, which represents the reasoning module of the overall architecture. Similarly, [76] proposed ThinkHome, a smart home architecture composed of a knowledge base and a multi-agent system. ThinkHome is populated by various specialized BDI agents, which are responsible for solving different problems by utilizing ontological reasoning methods. Further, [77] modeled the home environment as a MAS, and utilized contract based negotiation protocol for power management in home automation domain. Each of these contributions showed the advantages of modeling an AAL environment in terms of MAS, however, none of them consider issues related to uncertainty while presenting their reasoning modules. On the other hand, the MavHome architecture [78], which is a hierarchy of collaborative rational agents, was designed to meet the overall goals of a smart home environment. In MavHome each agent is composed of four cooperating layers: Decision, Information, Communication, and Physical. The decision layer is built by combining different machine learning algorithms, including Markov decision process. However, the contribution barely discusses issues related to uncertainty and methods to handle them in AAL environments.

Some other studies attempted to tackle uncertainty related challenges in the AAL environment using probabilistic graphical models. For instance, [79] utilized Multi-Entity Bayesian Networks to present a reference model for the AAL system that deals with uncertainty. However, the contribution is more tailored to the detection and prediction of unwanted situations than decision-making under uncertain conditions. Likewise, [38] discussed a smart home reasoning scenario which incorporates uncertainty reasoning using a rule-based system and Bayesian networks. And, [24] presented a framework to build a home automation system reactive to voice commands. In which, Markov Logic Network (MLN) is used to build the decision-making module of the system, and the uncertainty of the decision model was learned from data. Further, the authors integrated and tested their proposed system in a real-world environment, and presented some interesting results. However, both of the aforementioned works did not structure their solutions as a MAS, thus did not benefit from the advantage of autonomous agents. Further, MLNs are known to require a non-trivial effort by experts to properly model uncertainties in terms of weights [80]. Taking this into account, [81] utilized ProbLog for complex activity recognition in the smart home domain but again did not discuss a reasoning system as a whole.

Having these in mind, in this chapter we utilized the combination of MAS and probabilistic logic programming techniques to tackle uncertainty related issues in AAL environments. Specifically, the smart home system is modeled in terms of collaborative intelligent agents, and probabilistic reasoning is utilized to give the agents an ability to make a decision under an uncertain situation.

3.3 Background

This section briefly introduces the probabilistic logic programming technique and the agent interaction protocols utilized in this chapter.

3.3.1 ProbLog

ProbLog is a probabilistic extension of Prolog, and like Prolog, its program consists of a set of definite clauses. However, in ProbLog every clause c_i is annotated with the probability p_i that it is true, and these probabilities are mutually independent with each other [82]. That is $P(A \cap B) = P(A)P(B)$. Listing 1 shows a simple ProbLog program[1], in which the first clause indicates the fact *burglary* is true with probability 0.7 and false with probability 0.3. And the third clause indicates, if *burglary* and *earthquake* are true, then *alarm* will be true as well with 0.9 probability.

```
0.7::burglary.
0.2::earthquake.

0.9::alarm :- burglary, earthquake.
0.8::alarm :- burglary, \+earthquake.

evidence(\+earthquake).
query(alarm).
```

Listing 1: Simple ProbLog program

Unlike Prolog, where one is interested in determining whether a query succeeds or fails, in ProbLog we are interested in computing the probability that it succeeds. To realize this, ProbLog introduces an additional operator :: and two predicates *evidence* and *query*. The operator :: allows defining probability values over facts and rules. Whereas, the *evidence* clause allows to condition part of the program to be true or false, and the *query* clause allows to indicate the entity for which we want to compute the probability. Listing 1 computes

---

[1] This program is a modified form of an example provided in the official website of ProbLog.

the probability of *alarm* when we know that there is no *earthquake*. In addition, ProbLog supports modeling of non-binary choices. For example, listing 2 shows a simple ProbLog model for the throwing of a die in which each side of the die has equal probability value 1/6. In probabilistic logic, this kind of program is called *annotated disjunction*. And, it expresses that at most one of the clauses are true.

```
1/6::die(D, 1); 1/6::die(D, 2); 1/6::die(D, 3);
 ↪  1/6::die(D, 4); 1/6::die(D, 5); 1/6::die(D, 6).
```

Listing 2: Annotated disjunction in ProbLog

Below, the equations for computing the success probability of a ProbLog query are cited and summarized from [82] and [81].

Given a ProbLog program $T = \{p_1 : c_1, ..., p_n : c_n\}$. The probability of a world ω, that is a certain instance of the program, is defined as follows:

$$P(\omega|T) = \prod_{c_i \in \omega} p_i \prod_{c_i \in \omega_T \setminus \omega} (1 - p_i) \tag{3.1}$$

Where $\omega_T \setminus \omega$ describes the set of clauses that were not instanced in ω but are part of T, i.e. the set of false ground probabilistic atoms. Then, the success probability of a query q in the ProbLog program T is computed as follow:

$$P(q|\omega) = \begin{cases} 1, & \exists\theta : \omega \models q\theta \\ 0, & \text{otherwise} \end{cases} \tag{3.2}$$

$$P(q, \omega|T) = P(q|\omega).P(\omega|T) \tag{3.3}$$

$$P(q|T) = \sum_{\omega \subseteq W} P(q, \omega|T) \tag{3.4}$$

In short, the probability that a ProbLog query q succeeds is the sum of the probabilities of those worlds where q can succeed.

3.3.2 Agent interaction protocols

In a MAS environment, it is essential to have a standard agent interaction mechanism, that allows agents to collaborate and coordinate with each other in order to achieve their goals. Hence in our proposed MAS architecture, we utilized the Foundation for Intelligent Physical Agents (FIPA) Request Interaction Protocol (IP) [83] and Contract Net Protocol (CNET) [84]. IP allows one agent, the *Initiator*, to request another agent, the *Participant*, to perform an action. The *Participant* processes the request and makes a decision whether to *accept* or *refuse* the request. Once the request has been agreed upon, the *Participant* must communicate a *failure*, an *inform-done* or *inform-result* messages for the *Initiator* agent [85]. The *Participant* will send a *failure* message if it fails in its attempt to fulfill the request. It sends an *inform-done* message if it successfully completes the request and only wishes to indicate that it is done. Whereas, the *Participant* agent sends an *inform-result* message if it wishes to indicate both that the request is done and notify the *Initiator* of the results.

Likewise, CNET is a market-based agent interaction protocol which allows the *Manager* agent to have some task performed by one or more other *Contractor* agents. For this, the *Manager* and the *Contractor* agents engage in continuous negotiation process, before the *Manager* agent awards the contract for one or more *Contractor* agents, and the *Contractor* agents perform the task and return the result for the *Manager* agent. In both IP and CNET, at any time an agent can be an *Initiator/Manager*, a *Participant/Contractor* or both.

For the proposed MAS, IP is chosen instead of other similar protocols such as FIPA-Query or FIPA-REQUEST-When because it allows the agents to exchange sensor and actuator data in a relatively simple and effective way. Whereas, CNET is picked over other market-based negotiation protocols for several reasons. First, negotiation protocols such as the English and Dutch auctions are not part of the standard FIPA specification[2], but for the MAS part of our contribution, we require to follow standard FIPA specifications. Second, CNET allows agents to play different roles at different times (that is Manager and Contractor), thus provides flexibility. Third, none of the above auction-based protocols are implemented in the MAS development framework we utilized to evaluate the proposed architecture.

3.4 The Multi-Agent System Architecture

Home is a place where one lives. Commonly, it is composed of several rooms, dedicated for the specific activities of daily living. As shown in figure 3.1, our proposed multi-agent system architecture follows this logical structure of the home environment. Specifically, a group of collaborative intelligent agents manages each room of the smart home. These agents specialize in the specific tasks of the home automation system (such as monitoring the brightness or the air conditioning of the room) and collectively control the comfort and efficient operation of that specific room. In addition, these agents collaborate

---

[2] http://www.fipa.org/repository/standardspecs.html

with other agents, which resides in another room of the home, to achieve the overall goals of the smart home system. In order to realize this architecture, we propose four kinds of room-level agents: *Device, Service, Reasoner, and Negotiator* agents. In general, this kind of architecture allows to benefit from the basic advantages of intelligent agents such as autonomy, social ability, reactivity, and pro-activeness. It enables to modularize the smart home system into autonomous, collaborative and distributed components, which provide well-tailored services based on their location in the home. Further, the architecture allows to design a highly customizable and fault tolerant smart home system. Below the purposes and functioning of the aforementioned agents are discussed.

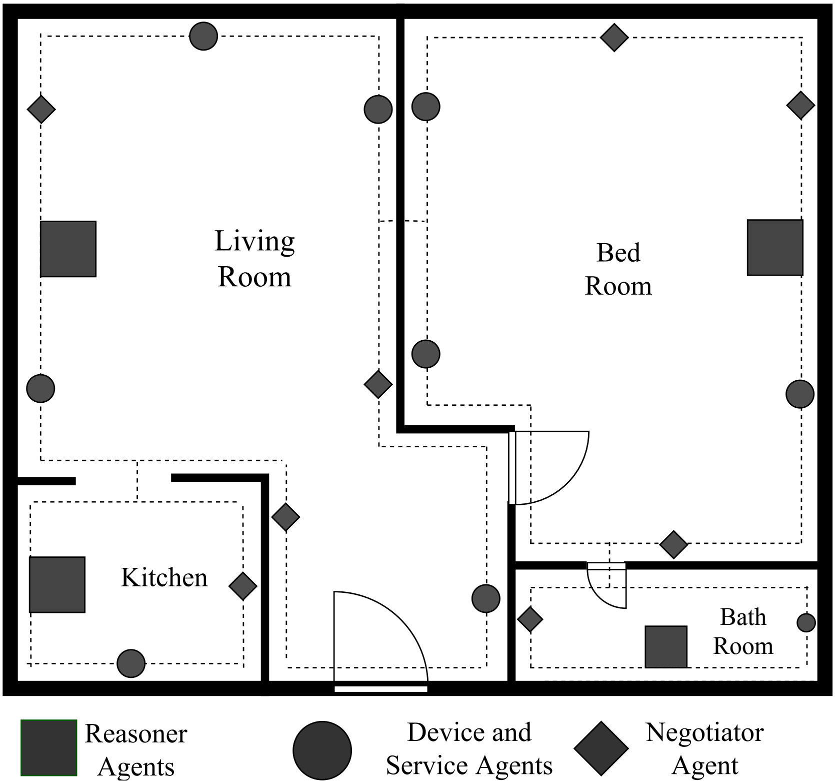

Figure 3.1: The smart home floor plan

3.4.1 Device agents

In the proposed architecture, a Device Agent (DA) controls the overall operation of a smart home device (i.e a sensor or an actuator) or group of devices. A DA which is responsible for controlling a specific sensing device(s), continuously monitors the changes in the environment and combines low-level sensor readings with other data in the home to generate high-level contextual information about the state of the home or its inhabitant. For instance, a device agent which controls room temperature, monitors a group of temperature sensors in that room and generates contextually meaningful information (e.g. *warm, cold* ...) by processing the sensor data with the season of the year, time of the day, type of the room etc. In addition, device agents are also able to determine their degree of belief (Bel) for the generated contextual information (e.g. *Bel(warm) = 0.9*). Here, a degree of belief can be understood as a number in the range of $[0, 1]$, that represents the measure of the agent's confidence in the generated context based on all available evidence. Figure 3.2 depicts a device agent which controls a group of sensors. In order to combine separate pieces of information and determine its *Bel* about the newly derived context, a DA may use evidential theories such as Dempster-Shafer theorem [86] or other probabilistic approaches.

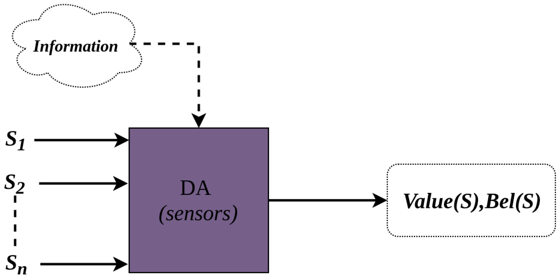

Figure 3.2: A device agent which controls a group of sensors

On the other hand, a device agent which controls the operation of an actuator(s) put into consideration the current situation of the home and its inhabitants, while executing a user command or during a self-adaptation process. For instance, when operating an electrical device the agent should check the real-time electric prices, and schedule the operation for off-peak hours if the outcome of the operation is not urgently required. This ensures the efficient execution of the command on the targeted smart home device and renders high-quality services in a cost effective way possible. Figure 3.3 depicts a device agent which control an actuator(s).

3.4.2 Service agents

Service Agent (SA) are general purpose agents which provide house level information, that is not specific to a single room or space in the home environment. Global information, which can be acquired from external data sources (such as weather data, and real-time electricity price) and information from other

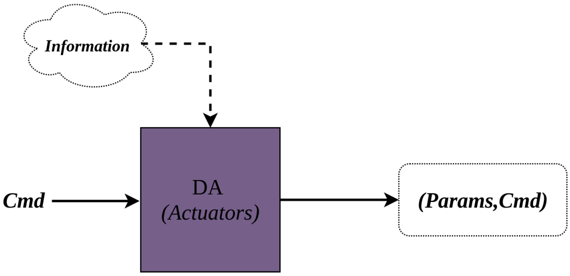

Figure 3.3: A device agent which controls an actuator(s)

smart home software components (such as activity recognition and person localization systems) can be coupled into services agent and integrated into the proposed multi-agent system architecture. Like device agents, SAs are also able to determine and share their degree of belief about their produced information.

3.4.3 Reasoner agents

In the proposed MAS architecture, each room is equipped with a Reasoner Agent (RA), which is responsible for the automatic control of the room environment and its adaptation to the inhabitant's needs. As the systematic literature review presented in chapter 2 of this thesis revealed, a starting point to overcoming some of the major challenges of reasoning in smart living environments is combining the best of symbolic and statistical AI techniques. As a result, the decision-making unit of this agent is designed based on a probabilistic logic programming technique called ProbLog. This technique allows the design of a hybrid reasoning system, which benefits from the advantages of both symbolic and statistical artificial intelligence methods. As a result, along with other assets of hybrid systems, RA possesses the ability to act under uncertainty and perform well with erroneous sensor data and ambiguous user commands. In addition, ProbLog enables to learn the structure and parameters of the probabilistic rules from data and to model the system using human-readable and easily modifiable rules. Besides, these agents are designed to cope with unavailable, conflicting and inconsistent sensor data by actively collaborating with other agents in the system.

Algorithm 1 details the overall operation of an RA when it receives a user command or detects a change in the home. Here note that the smart home user is considered to be an external entity. When an RA receives a user command, first it determines the smart home service associated with it (e.g. the user command "*turn on the light*" can be associated with the smart home service "*light*"). Following this, it discovers a list of device and service agents which provide relevant information to the requested service, and then it identifies the device agent which controls the behavior of that specific smart home service (lines 2-3). Subsequently, it will send an information request about the current state of the inhabitant and its environment to the relevant service and device agents (line 4). Whereas, when a DA receives an information request, it determines the value of the requested information and its degree of belief about it (e.g. *useractivity="cooking", Bel=0.99*), and send back a reply for the reasoner agent. However, some of the data provider agents may fail to provide the requested information on time or may fail to reply at all due to sensor failure or other problems. Moreover, RA may discover some inconsistencies or conflicts in the collected data (lines 5-6). In this kind of situations, RA will request its negotiator agent to interact with other negotiator agents in the system and gather the missing data from similar data provider agents in other rooms of the smart home (line 7). When the interaction process ends, the negotiator agent will transfer the result to the reasoner agent, which will use the data to improve its level of uncertainty or resolve conflicts, so that it makes the most appropriate decision (lines 8-9). When RA receives all information, it will check again if some data are still missing (lines 10-11), if so, it will use default values for these data (lines 12-13). Here, the default values of a sensor can be determined from the values of other sensors in the environment, and the current contexts of the home. After resolving the missing information, RA will build the ProbLog model and checks if it has the ability to solve it locally (lines 16-17). RA's ability to solve a ProbLog model is determined by the local availability of a ProbLog engine and computational resources required to solve the model. If RA has reasoning ability, it solves the ProbLog model locally (line 19), that is determining the success probabilities of all the associated commands related to the user request and the situation of the home. Afterwards, it picks the command with the highest probability, that is the command considered to give maximum user satisfaction in the current state of the home. Whereas, if RA does not have reasoning ability, it will request the negotiator agent in the same room to delegate the reasoning task for other RAs in the system (line 21). If the negotiator agent fails to find another RA willing to solve the model, RA will take the default command for the user request (line 26). Otherwise, it will use the result of the interaction, which is a command with the highest success probability to serve the user request (line 24). Finally, RA will send the control command for the devices controller agent (line 29). Section 3.4.3.1 briefly describes the underlying decision-making process of this agent, using a simple scenario, and figure 3.4 depicts the sequence diagram of the agents' interaction in the scenario. The function and operation of the negotiator agent are discussed in the next section.

Algorithm 1: RA's reasoning algorithm

```
input : userRequest
$service \longleftarrow determineRequestedService(userRequest)$;
$dataProviderAgents \longleftarrow discoverDataProviders(service)$;
$devicesControllerAgent \longleftarrow discoverDevicesController(service)$;
$sensorData \longleftarrow collectSensorData(dataProvidersList)$;
$missingData \longleftarrow getMissingData(service, sensorData)$ ;
if not empty missingData then
    $sendInformationRequest(negotiator, missingData)$;
    $missingData \longleftarrow receiveData(negotiator)$;
    $sensorData.merge(missingData)$;
    $missingData \longleftarrow getMissingData(service, sensorData)$ ;
    if not empty missingData then
        $missingData \longleftarrow getDefaultValues(missingData)$;
        $sensorData.merge(missingData)$;
    end
end
$probLogModel \longleftarrow buildProbLogModel(sensorData, service)$;
$ableToReason \longleftarrow ableToReasonLocally()$;
if ableToReason then
    $controlCMD \longleftarrow solveProbLogModel(probLogModel)$;
else
    $sendReasoningTaskRequest(negotiator, probLogModel)$;
    $result \longleftarrow receiveReasoningResult(negotiator)$;
    if result.status = SUCCESS then
        $controlCMD \longleftarrow result.data$ ;
    else
        $controlCMD \longleftarrow getDefaultCommand(service, sensorData)$;
    end
end
$sendCommand(devicesControllerAgent, controlCommand)$;
```

3.4.3.1 Scenario: Dealing with ambiguous user command

Suppose the kitchen has four kinds of lights (*i.e. ceiling, sink, dining and cooking lights*) and the inhabitant stands near the kitchen-sink (*loc="near-sink", Bel=0.95*) while preparing her dinner (*act="cooking", Bel=0.95*). Meanwhile, she issues a voice command *"turn on the light"* to the system, without specifying the exact light she wants the system to turn on. This kind of ambiguous user commands are one of the major sources of uncertainty in smart home environments, and hereinbelow we will use this scenario to illustrate how the reasoner agent function under such circumstances.

To simplify the discussion let us again assume that, RA controls the lighting of the room only based on the inhabitant's current location and activity. That is, the system turns on the *cooking, sink, dining or ceiling light*, when the inhabitant is around the kitchen cabinet and preparing food, around the sink and washing plates, on the dining table and eating, or anywhere in the room and tidying up respectively. The current location and activity of the inhabitant are also separately utilized to control the lighting of the room, with distinct probability values (*Pr*). That is, activity-based light control (*Pr=0.7*) is assumed to give better user satisfaction compared with location-based control (*Pr=0.5*). Further, for any *"turn on the light"* command, it is also assumed to be fair, if the RA turns on the ceiling light with *Pr = 0.5*. Listing 3, shows the ProbLog representation of the aforementioned scenario. The parameters of the probabilistic clauses can be mined from a dataset using a probabilistic rule learner tool such as ProbFOIL[87], or manually designed by a knowledge engineer. The parameters of the rules and facts in listing 3 are from the assumptions made in this scenario.

```
0.95::loc(near_sink).
0.95::act(cooking).
0.90::cookingLight:- act(cooking), loc(cooking_area).
0.70::cookingLight:- act(cooking).
0.50::cookingLight:- loc(cooking_area).
0.90::sinkLight:- act(washing), loc(near_sink).
0.70::sinkLight:- act(washing).
0.50::sinkLight:- loc(near_sink).
0.90::dinningLight:- act(dinning), loc(dinning_table).
0.70::dinningLight:- act(dinning).
0.50::dinningLight:- loc(dinning_table).
0.90::ceilingLight:- act(tidyingup).
0.50::ceilingLight.
```

Listing 3: RA's Light Control Rules

In light of the above scenario, when RA receives the voice command, it first determines the set of possible worlds based on the current activity and location of the inhabitant. The possible worlds are the list of commands, whose logical formulas evaluated as true. And, as shown in figure 3.5, *cooking light, sink light and dining light* commands are all each in one possible world, which is marked as light green rectangles in the figure. Subsequently, as shown in figure 3.6 the agent determines the success probabilities of each possible worlds using equation 3.1, and sum up the probability values of each command using equation 3.4. Finally, it executes the command with a larger total sum probability value. In this scenario, *"cookingLight"*, with *0.665* success probability has the largest total probability (shown in 3.7) . For the process of determining the possible worlds and their success probabilities, the reasoner agent fully relies on the ProbLog solver.

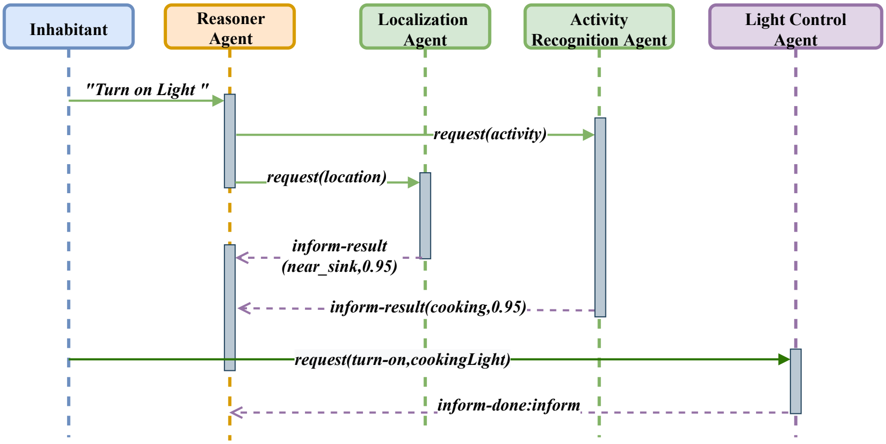

Figure 3.4: Sequence diagram representing the interactions between the agents

Figure 3.5: All worlds for the turn on the light command

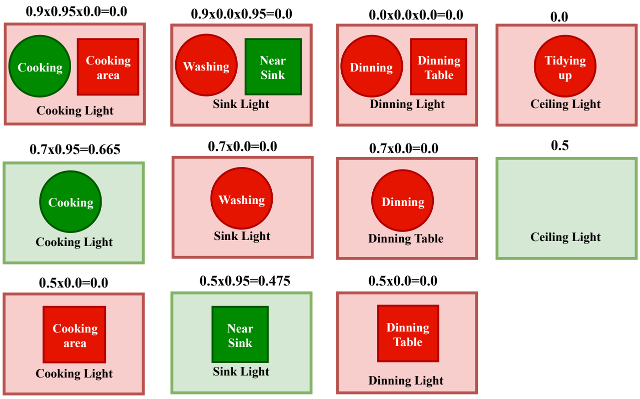

Figure 3.6: The agent determining the success probabilities of each possible worlds

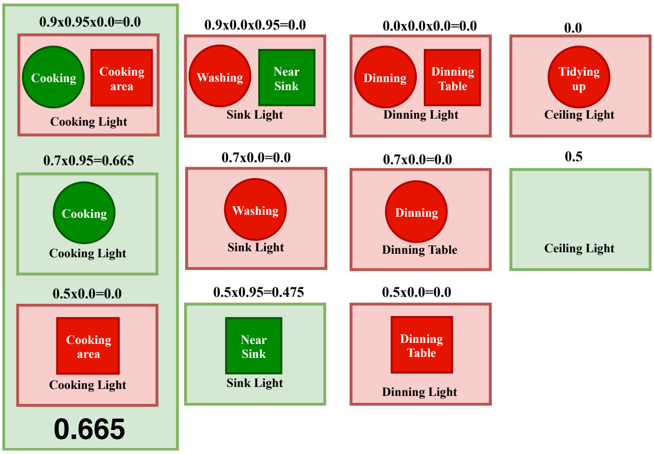

Figure 3.7: The reasoner agent sum up the probability values of lights, and picks the largest

3.4.4 Negotiator agents

In the proposed MAS based smart home architecture, each room has a Negotiator Agent (NA), which is in charge of handling the collaboration process between agents of different rooms. Specifically, NA enables other agents in the system to exchange information about the state of the environment they are operating in. Further, when reasoner agents are not able to solve their decision-making problem due to computational resource limitation, the negotiator agent enables them to delegate their reasoning tasks (i.e. determining the success probability of a query) for other reasoner agents in the system. In practice, this agent is designed based on the FIPA (The Foundation for Intelligent Physical Agents) Request Interaction and Contract Net Protocols, thus at different times, it acts as the *Initiator* or *Participant* of an information exchange process, or it acts as the *Manager* or *Contractor* of a negotiation process over reasoning tasks.

Here one may question the need to have a separate NA instead of integrating the aforementioned protocols in the RAs of each room. The motive behind this is to divide and distribute the tasks and responsibilities of RA and NA agents. That is letting the RA and its ProbLog based implementation only center on the decision-making process of the system, while NA handles the collaboration activities between agents of different rooms. In general, this kind of modular design improves the performance and maintainability of the system and reduces its complexity.

NA plays the *Initiator* role when it receives an information request message from the reasoner agent in the same room. As the Request Interaction Protocol initiator, the negotiator agent first looks for other negotiator agents in the house, who are registered to provide the information that the RA requested. Afterward, it will send a *request* message with the submission deadline, for all the information provider agents. Upon receiving the information from each of the participants (i.e. *inform-result* message), it will determine the best reply based on some criteria or combines separate replies to derive one, and communicates the final result for the reasoner agent. For instance, if RA is interested in the total number of people in the house after each *Participant* NAs communicated the number of people in their respective rooms, the *Initiator* NA will sum the values, or if the requested information is room temperature, NA will determine the statistical mean value for the collected temperature values.

Likewise, NA plays the *Participant* role, whenever it receives an information request message from another agent of the same type. Accordingly, based on the availability of a device agent in the same room, that is able to generate the requested information, the negotiator agent will accept or reject the request by sending *agree* or *refuse* messages respectively. If it accepts, it will send a *request* message for the information provider device agents. At last, when it receives the data from all device agents, it will process and communicate the result for the initiator NA. This process should help the reasoner agents to cope with sensor failures and perform well in partially observable environments.

On the other hand, when the negotiator agent receives a request from the RA in the same room, to delegate a reasoning task for a reasoner agent in another room, it plays the *Manager* role of the CNET protocol, by sending *call-for-proposals* with the task description, to other negotiator agents (i.e. *Contractors*) in the system. Whereas, when a *Contractor* agent receives a *call-for-proposal* message, it determines the time the reasoner agent in its room needs to solve the ProbLog model, and submits its proposal by sending *propose* message for the *Manager* agent. The time that the reasoner agent requires to solve the model can be determined by adapting the techniques proposed in [88]. Upon receiving all replies from the *Contractor* agents, the *Manager* agent will award the contract for the *Contractor* agent which proposes to complete the task at the earliest time possible.

3.5 Experimental Evaluation

A Proof-of-concept implementation of the proposed multi-agent system architecture is developed using the Java Agent DEvelopment Environment (JADE) framework [85] and ProbLog Python library. JADE is an open-source FIPA compliant middleware for developing multi-agent systems. It provides a ready-to-use and easy-to-customize MAS development platform, efficient agent communication mechanisms, effective agent life-cycle management, support for agent mobility, yellow and white page services, and a GUI for debugging and monitoring MAS application [85]. ProbLog is available as an online tool on the web and for download. The offline version offers both command line access to inference and learning and a Python library for building statistical relational learning applications [89]. For the purposes of giving the probabilistic reasoning ability for the reasoner agents in our system, we integrated the ProbLog Python library into the JADE MAS platform.

3.5.1 Experimental setup

For the PoC implementation, we simulated the four-room smart home architecture presented in figure 3.1. In which, agents of each room run inside a JADE container hosted on a single board computer and connected with each other through a local area network. The MAS hosted in each node is composed of one Reasoner, one Negotiator, one JADE Directory Facilitator, and four Device and Service agents, that are *Inhabitant Activity Recognition, Inhabitant Localization, Luminosity sensor, and Light service controller agents.*

The device and service agents are designed to randomly generates synthetic sensor data from their respective predefined set of values. For the purposes of heterogeneity, we host two of the rooms in Raspberry PI 3 Model B+ nodes (quad-core A53 (ARMv8) 64-bit @ 1.4GHz, 1GB SDRAM, Gigabit Ethernet, Raspbian OS), and the other two in Intel Galileo boards (single core i586 CPU @ 400 MHz, 256 MB DRAM, 100 Mb Ethernet, Yokto Linux OS). The Galileo boards are ideal to simulate nodes with less computation capability. In the proposed MAS architecture, this represents a node that cannot perform reasoning by itself. Plus, as the latest version of Yokto does not support Python 3, which is a prerequisite to run a ProbLog engine, the two rooms which are hosted in the Galileo boards were without the ability to solve ProbLog model, thus need to delegate their reasoning tasks to other reasoner agents in the system. This setup allows us to measure the CPU time needed by the RA to reason locally on the Raspberry PI nodes, and the CPU time needed by the RA to negotiate and solve the ProbLog model on another node in the system. Further, to make a comparison of the reasoning time in the two hardware configurations, the latter case was tested both on the Raspberry Pi (with no local reasoning ability) and Galileo boards. In this evaluation, scenario one and its ProbLog model (presented in listing 3) is utilized. Figure 3.8 depicts the experimental setup used to run the tests and evaluate the proposed multi-agent system.

Figure 3.8: The experimental setup.

3.5.2 Tests and Results

To measure the CPU time that an RA requires to reason, we run two groups of experiments. First, we measure the CPU time that an RA requires to perform the entire reasoning process locally on a Raspberry PI node. Second, we measure the CPU time that an RA requires to delegate and solve the ProbLog model via the negotiator agent, both on the Raspberry PI and Galileo nodes. Further, we conducted four distinct tests for each of the aforementioned two experiments, by incrementally changing the number of locally unavailable (missing) information required to reason from zero to three. All the tests were run 100 times, and the mean and standard deviation of their reasoning time was recorded. Table 3.1 and 3.2 summarizes the results of the experiments, and figure 3.9, 3.10 and 3.11 present comparable results of each run.

Table 3.1: CPU time(ms) of reasoning locally vs. over negotiation with varying Missing Information, on Raspberry PI 3 nodes.

Raspberry PI	No. of missing information			
	None	One	Two	Three
Reasoning locally	597.2±17	617.9±17	618.2±23	650.2±17
Reasoning over negotiation	674.9±28	716.6±34	727.13±32	729±28

Table 3.2: CPU time(ms) of reasoning via negotiation with varying missing information(MI), on Raspberry PI 3 vs. Galileo nodes.

Board	No. of missing information			
	None	One	Two	Three
Raspberry	674.9±28	716.6±34	727.13±32	729±28
Galileo	828±60	920±71	925.3±42	900.5±41

As can be seen in table 3.1 and figure 3.10, the reasoner agent which run on the Raspberry PI node required a relatively small amount of time to reason locally compared with the one which needs to delegate the reasoning task to another reasoner agent. This is due to the negotiation process and the message exchange between the agents involved in it. Whereas, as shown in table 3.2 and figure 3.11, the difference in reasoning time between the Galileo and Raspberry PI nodes is relatively big (i.e. $\approx +181.5$ ms grand/pooled mean). Intuitively, this is due to the difference in the computational capacity of the two boards. In addition, as can be seen in figure 3.9 the reasoning time increases with the number of missing information, and from the experiment we also noticed that the time required to collect missing information(MI) from other device agents in the system is nearly constant.

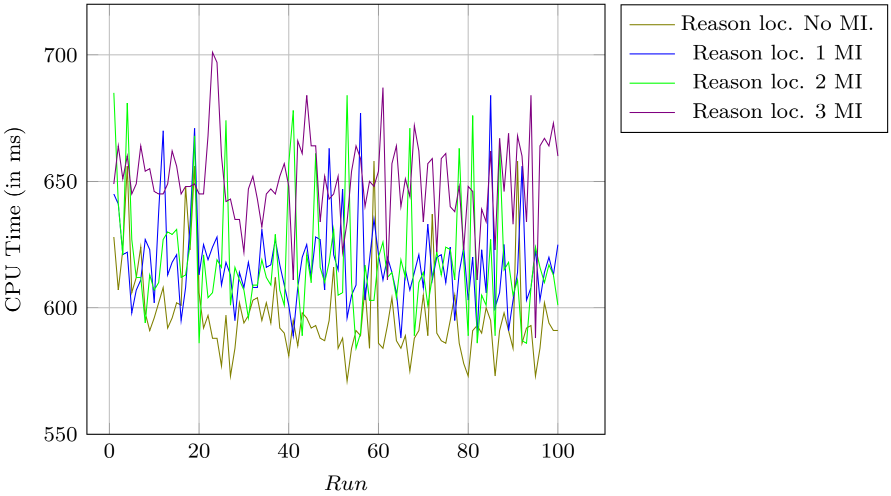

Figure 3.9: Reasoning locally on the Raspberry PI nodes

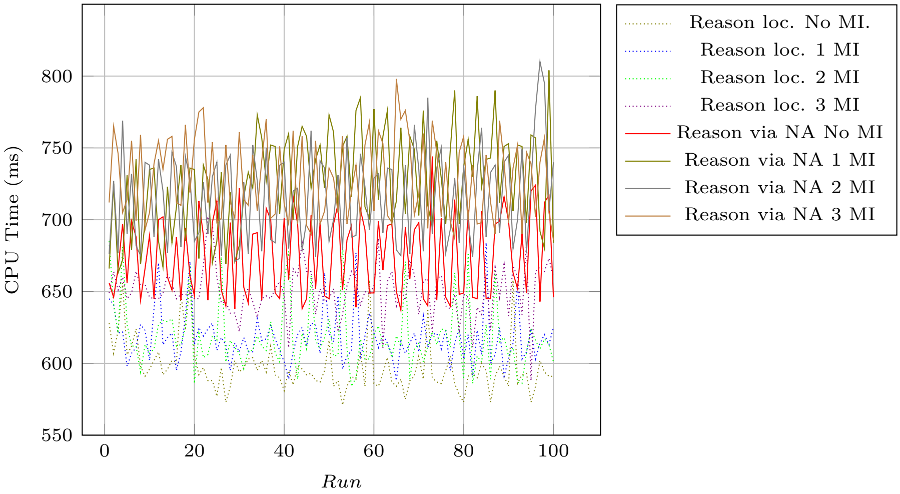

Figure 3.10: Graphical comparison of reasoning locally and over negotiation on Raspberry PI nodes

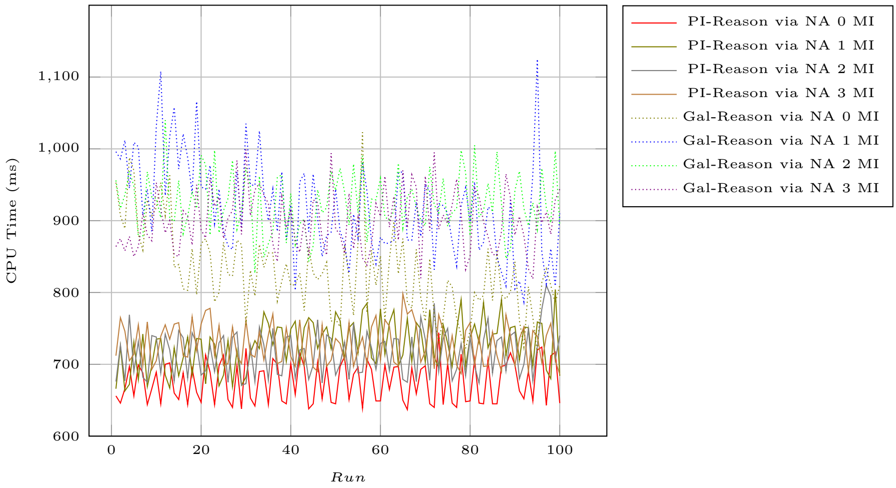

Figure 3.11: Graphical comparison of reasoning over negotiation on Raspberry PI and Galileo nodes

In general, from the proof-of-concept implementation and experimental tests, we observed the practicality and feasibility of the proposed MAS based smart home architecture. First, it was observed that the probabilistic inference technique allows the reasoner agent to reason with ambiguous user commands and partial information. Second, collaborative MAS architecture enables the agents to cope well in partially observable environments. In addition, the MAS architecture makes it possible to design heterogeneous, highly customizable, robust and modular system.

3.5.3 Threats to the validity of the experiments

Being a preliminary study on the application of probabilistic logic reasoning in multi-agent based smart home system, the proposed experiments do inevitably suffer from threats to validity. Therefore, future works will address the identified limitations.

First, the core ProbLog model used in the experiments is based on the scenario discussed in section 3, so a model with few probabilistic logic clauses is not a computational challenge both for the proposed system and the integrated ProbLog solver. Second, all the device and service agents used in our PoC implementation are simulated, consequently, the data generation time of these agents was almost negligible. Therefore, to draw a strong conclusion about the reasoning time of the agents a more complex ProbLog model and real sensors data need to be used.

3.6 Chapter Conclusions

Prior works widely applied multi-agent systems to model AAL environments. However, most of them have not considered issues related to uncertainty reasoning, while presenting their decision-making components. Contrarily, few others tackled these challenges using probabilistic graph models, but either they have not discussed SHRSs as a whole, or they have not utilized the advantages of MAS to support the decision-making process in the AAL environments. With this in mind, in this study, we proposed a probabilistic multi-agent system architecture for reasoning in smart homes, based on a probabilistic logic programming technique called ProbLog and multi-agent system technologies. Accordingly, we illustrated how the probabilistic reasoning technique enables the intelligent agents to reason under uncertain situations. Further, we discussed how the agent interaction protocols enhance the decision-making process by allowing the agents to exchange information about missing data or unobservable variables between the agents. Therefore, this study showed that the integration of MAS and probabilistic logic programming can help in building a reasoning system for AAL environment, which is capable of performing well under vague inhabitant commands, missing information, and in partially observable environment.

Most notably, this is the first study to our knowledge which integrates the ProbLog reasoning engine into a multi-agent system framework and, to utilize it for tackling uncertainty issues in smart home environments. In general, our PoC implementation and experimental analysis suggest that this approach appears to be effective in counteracting uncertainties in the reasoning systems of these complex and dynamic environments. However, some limitations are worth noting. First, some operations of the proposed system, especially the service and device agents are presented based on several assumptions (such as their ability to generate contextually meaningful information and determine their degree of belief about it). Second, the structure and parameters of the probabilistic logic rules are designed based on the subjective knowledge of the authors, yet can be learned from data. Therefore, future work will address these limitations and evaluate the proposed system thoroughly.

Chapter 4

Consistency Verification of a Rule Based Smart Home Reasoning System with Satisfiability Modulo Theories

4.1 Introduction

The smart home reasoning system determines the automatic adaptation process of the home environment to its inhabitants' needs. As part of the requirements of this system, the automatic adaptation process must not lead the home into an incomprehensible and uncontrollable state, and should not exhibit unpredictable behaviour. However, the reasoning system is prone to forgo these requirements due to several reasons, such as a result of conflicting inhabitant preferences in a multi-resident smart home environment, due to conflicting services (e.g. energy efficiency versus comfort), or owing to software and hardware components failure. In addition, violations of some internal properties of the system, such as consistency of the reasoning system, are key factors that could seriously affect the regular operation of the smart home. Therefore, it is essential to analyze and validate some consistency properties of the reasoning system before its initial deployment and use. However, as discussed in chapter 2 of this thesis, in SHRS literature, only a few contributions have been made towards this direction. Mainly, these studies focused on rule conflict resolutions, and for this, most of them utilized defensible logic techniques or defined priority orders between the rules. Nonetheless, none of these studies presented a method to analyze and detect the primary causes of inconsistencies in smart home reasoning systems.

Accordingly, this chapter proposes a method for consistency verification of a rule-based smart home reasoning system. To this end, it defines, formalizes and presents a static (off-line) analysis method for five primary causes of inconsistencies in rule-based smart home reasoning systems, using satisfiability modulo theories as a tool. The primary causes of inconsistencies considered in this study are conflicting, duplicate, overlapping, self-looping, and circular rules. These consistency properties are identified as part of the systematic literature review presented in chapter two of this thesis. The motivation behind focusing solely on rule-based reasoning systems is because most of the contributions in the field are presented as knowledge-based systems, thus addressing larger domain. The proposed method is validated empirically using a real-world smart home reasoning system as a model. Besides, the CPU time that a state-of-the-art SMT solver requires to analyze the model is examined. In general, this chapter presents a reliable and effective solution to analyze and verify the consistency of smart home reasoning systems.

The remainder of the chapter is structured as follows: Section 4.2 gives an overview of the related researches. Section 4.3 introduces satisfiability modulo theories. Section 4.4, presents the definition and formalization of the major causes of inconsistencies in rule-based smart home reasoning systems and discusses the proposed inconsistency verification method. Section 4.5, validates the proposed method by presenting the experimental settings and the test results. And, section 4.6 ends the chapter by drawing some conclusions and suggestions for further improvements.

4.2 Related Research

Knowledge base verification is a long-existing research field, to which several researchers contributed over the years. For instance, in 1982, [90] presented an approach to verify the completeness and consistency of a rule-based expert system. However, the ever-changing techniques and technologies introduced to build reasoning systems, and the variety and complexity of the application domains these systems are getting applied to, make consistency verification of a reasoning system still a challenge. Accordingly, to address some of these challenges, several studies proposed different rule models and utilised a variety of techniques. For instance, in [91] the authors extended the Event-Condition-Action (ECA) rule model with new concepts such as *subject, object and post-condition*. Afterwards, they propose an algorithm to detect and resolve rule conflicts using Vague set theory. The EVA method [92] propose a declarative meta-language to validate the consistency of a knowledge-based system. Whereas, the PREPARE method [93] uses a special class of Petri nets, the *predicate/transition nets*, to represent knowledge base and to verify its consistency properties. However, as the aim of this chapter is to present a consistency verification technique for smart environments reasoning systems using formal methods, the subsequent discussion of this section will mainly focus on research contributions presented in the same domain and using similar techniques.

In this regard, [94] models the rule-based system as finite-state transition and expresses rule conflict and circularity as Linear Temporal Logic (LTL) formula, after that, it uses model checking techniques to detect inconsistencies in the system. However, this contribution is presented using a simplistic example, and its applicability for larger models is not studied, nor the performance is evaluated. In contrast, the work presented in this chapter is experimentally validated using a real-world smart home reasoning system as a model. Besides, the work did not address other sources of inconsistencies in a rule-based reasoning system, such as duplicate and overlapping rules. Likewise, [95] proposed an approach to analyze the behaviour of ECA rules. For this, it first translates the rules into an extended Petri net model and then it utilizes Computational Tree Logic (CTL) based model checking techniques to study two correctness properties: *termination and confluence.* The study defines termination as a property that the system does not remain internally busy forever without responding to external events, and confluence ensures that any possible interleaving of a set of triggered rules yields the same final result. [96] presented formalization to translate an ECA rule-based system into Heptagon/BZR program, which is a synchronous dataflow programming language with support for equation and automata. Afterwards, this study discussed how redundant, conflicting, and circular rules can be detected using the Heptagon/BZR model. The authors defined rule conflicts as two rules which have the same condition but opposite actions. Whereas, this chapter has an extended view of rule conflicts, that is two rules are conflicting if they can be satisfied with the same situation, and results in the execution of at least one contradictory action, without the need to have exactly the same conditions and actions parts. [97] proposed a rule conflict detection scheme tailored for smart building systems. For this, it defines a new formal rule model called UTEA based on *User, Triggers, Environment entities and Actuators.* The definition of a new rule model allows to analyze conflicts between multiple smart home users with different user authority levels. In the article, the authors classified rule conflicts into five categories: *Shallow, Execution, Environment Mutual, Direct dependency, and Indirect Dependency conflicts.* The first two categories are the same with duplicate and overlapping rules definition of the present work, whereas the third and fourth, are equivalent with the definition of rule conflicts in this chapter, and the last one is identical with circular rules. The rules in their proposed system are represented in XML format and stored in a rule database, and for the analysis of rule conflicts, they presented a conflict detection algorithm, whereas in our work we rely on satisfiability modulo theories for the consistency verification problem.

The closest proposal to our approach is presented in [98], in which the authors utilized SMT based techniques for the verification of ECA based rules of intelligent environments. The study defined a rule-based reasoning system as consistent if its rules are neither unused nor redundant nor incorrect. An incorrect rule described as a rule whose values are not admissible in the system. For example, suppose the luminosity of the light actuator can be set in the range of [0, 100], a rule which sets a value 110 for this actuator is an incorrect. In general, the article presented the formalization of the aforementioned properties and a prototype tool for statically analyze them. However, it did not address other sources of inconsistencies such as conflicting rules, self-looping rules and infinite-execution loops (or circular rules). Instead, in this chapter, we formalize rule conflicts, duplicate rules, overlapping rules, self-loops and circular rules, and discuss how they can be detected using SMT.

4.3 Background

This section briefly introduces satisfiability modulo theories.

4.3.1 Satisfiability Modulo Theories

Propositional Satisfiability (SAT) is the most well-known constraint satisfaction problem, which aims to decide whether a formula over Boolean variables, formed using logical connectives, can be made true by choosing true/false values for its variables [99]. Representing some of the constraint satisfaction problems into SAT requires an added expressiveness of equality, arithmetic, arrays, datatype operations, quantifiers and some other theories. The research field concerned in handling these problems is called *Satisfiability Modulo Theories*. SMT is the problem of deciding the satisfiability of quantifier-free first-order formula ϕ with respect to some decidable theory T [100]. Linear Arithmetic over Rationals ($\boldsymbol{\mathcal{LRA}}$) is a theory in *SMT*, which checks the satisfiability of a formula ϕ consisting in atomic propositions and linear-arithmetic constraints over rational variables (e.g. $2.1x_1 - 3.4x_2 + 3.2x_3 \leq 4.2$), combined using Boolean operators (i.e. $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$) [101]. To effectively represent smart home rules, which mainly reason with Boolean and Real valued sensor/actuator data, as SMT constraints, in this work we utilized $\boldsymbol{\mathcal{LRA}}$. In general, an $\boldsymbol{\mathcal{LRA}}$ interpretation μ is a function which assigns truth values to Boolean atoms and rational values to numerical variables. And, we say μ satisfies ϕ in $\boldsymbol{\mathcal{LRA}}$ or μ is a solution for ϕ in $\boldsymbol{\mathcal{LRA}}$ - iff μ makes the formula ϕ evaluate to true and ϕ is $\boldsymbol{\mathcal{LRA}}$ satisfiable iff it has at least one $\boldsymbol{\mathcal{LRA}}$ interpretation μ such that μ satisfies ϕ. Otherwise, ϕ is unsatisfiable, in other words $\neg\phi$ is valid.

Listing 4 shows a simple SMT program, in which two Boolean and one Real variables are declared, and an SMT constraint is formulated. That is:

$((temperature < 18) \land \neg window\_open \rightarrow heater\_on)$. The third and second to last rows of the code ask for the satisfiability of the problem and return the values of *temperature*, *window_open* and *heater_on* variables, in the case of SAT. And, the last line returns the unsatisfiable core (UNSAT core), which is the subset of clauses that are unsatisfiable, in case of UNSAT. In other words, our SMT problem is to find out if the *heater* starts, when the *temperature* is under 18, and the *window* is not open.

```
(set-option :produce-models true)
(set-option :produce-unsat-cores true)

(declare-fun temperature () Real)
(declare-fun window_open () Bool)
(declare-fun heater_on () Bool)

(assert (implies (and (< temperature 18)
        (not window_open)) heater_on))

(check-sat)
(get-model)
(get-unsat-core)
```

Listing 4: SMT Example

4.4 Inconsistencies in Rule-based SHRS: Definition and Formalization

As introduced in section 4.1 of this thesis, most smart home reasoning systems are designed based on symbolic AI techniques and they are presented mainly as rule-based systems. With that in mind, this section first briefly introduces *Event-Condition-Action* rules and rule bases, and afterwards, it defines, formalizes, and shows how the five most common sources of inconsistencies in rule-based smart home reasoning system can be detected using SMT.

4.4.1 Rules and Rulebase

In this study, we consider Event-Condition-Action rules, which consists of three parts: an *event*, a *condition* (or “IF” statement) and an *action* (or “THEN”) parts. An ECA rule triggers, when an *event* occurs, if the *condition* matches the current state of the environment, then the *action* performs changes in the state of the system or its operating environment. For the convenience of modeling

ECA rules as SMT constraints, in this study, we simplified ECA rules into *Condition-Action* rules, by incorporating the event part of the rule with the condition part. In this study, a rule base RB is considered as a system, which consists of a set of Condition-Action rules that governs the overall operation of the home environment. And, a Condition-Action rule R_i is defined as follow:

$$R_i : C_i \rightarrow A_i \tag{4.1}$$

Where C_i and A_i are propositional formula (Boolean and/or Linear arithmetic) and respectively represent, the condition and the action part of the rule R_i. For example, the rule: "***if*** *temperature is under 18* ***and*** *window is not open,* ***then*** *turn on the heater*, can be represented as follow:

$$((temperature < 18) \wedge \neg window\_open) \rightarrow heater\_on$$

And, as discussed in section 4.3 this example can be effectively represented as an SMT problem shown in listing 4. In addition, for definition 4.1, we assumed that C_i and A_i are in disjunctive normal form, so that the rule definition can be rewritten as follows:

$$R_i : \vee_{k=1}^{m} c_{ik} \rightarrow \vee_{k=1}^{n} a_{ik} \tag{4.2}$$

where $\vee_{k=1}^{m} c_{ik} = C_i$ and $\vee_{k=1}^{n} a_{ik} = A_i$. And, m and n are the number of disjunctive clauses in the condition and action parts of the rule respectively. Listing 5 shows a running example for the set of rules in a smart home rule base RB_1.

4.4.2 Major sources of inconsistencies in SHRSs

Based on the above definition of a rule and a rule base, this section discusses the five major sources of inconsistencies in rule-based smart home reasoning systems, which are identified as part of the systematic literature review presented in chapter 2 of this thesis, and shows how each of these inconsistencies can be detected by formalizing them as SMT constraints.

4.4.2.1 Rule conflicts

Given two rules R_i and R_j, we say R_j *threatens* the operation of R_i, if the condition part of R_j becomes satisfied whenever the condition part of R_i is satisfied, but if the action of these two rules are conflicting in nature. Besides, these two rules are defined to be *conflicting*, if they threaten each other. Simply, two rules are in structural conflict if their condition parts are satisfied in the same situation, but if the execution of their action parts have opposite effect in

$$
\begin{aligned}
RB_1 = \{ & \\
& r_1 : (no\_of\_people > 0) \rightarrow light\_on \\
& r_2 : (no\_of\_people > 0) \rightarrow \neg(light\_on) \\
& r_3 : (no\_of\_people > 0) \wedge day \rightarrow \neg(light\_on) \\
& r_4 : (no\_of\_people > 0) \rightarrow \neg(light\_on) \vee shutter\_open \\
& r_5 : (no\_of\_people > 0) \rightarrow \neg(light\_on) \wedge shutter\_open \\
& r_6 : (no\_of\_people > 0) \vee day \rightarrow \neg(light\_on) \\
& r_7 : (no\_of\_people > 0) \vee day \rightarrow \neg(light\_on) \vee shutter\_open \\
& r_8 : (no\_of\_people > 0) \vee (temp < 18) \rightarrow light\_on \vee heater\_on \\
& r_9 : (no\_of\_people > 0) \rightarrow light\_on \\
& r_{10} : light\_on \rightarrow light\_on \\
& r_{11} : (no\_of\_people > 0) \vee day \rightarrow (no\_of\_people > 0) \\
& r_{12} : (no\_of\_people > 0) \rightarrow (no\_of\_people > 0) \wedge day \\
& r_{13} : (no\_of\_people > 0) \rightarrow (no\_of\_people > 0) \vee day \\
& r_{14} : (no\_of\_people > 0) \wedge day \rightarrow (no\_of\_people > 0) \\
& \}
\end{aligned}
$$

Listing 5: A running example for smart home rules and rule base

the home environment. Formally, the threaten relationship between two rules is defined as follow:

Definition 1 *Rule R_j threaten Rule R_i iff:*

$$((C_i \rightarrow C_j) \wedge \neg(A_i \wedge A_j)) \textit{ is } \textbf{\textit{valid}}.$$

This definition captures basic rule conflicts such as the conflict between r_1 and r_2, and threatening relationship between rules such as r_3 and r_1, r_1 and r_5, and r_1 and r_6 in listing 5, but do not capture more complex and potentially conflicting rules like the potential conflict between r_1 and r_4, r_1 and r_7, r_2 and r_8, and r_5 and r_8. These rules are potentially conflicting, because, their conditions can be satisfied with the same situation (i.e. $(no\_of\_people > 0) = \top$)), and this can result in conflicting actions (i.e. $light\_on = \top$ and $light\_on = \bot$). For this, by checking the existence of any contrary disjunctive clauses between the action parts of the two rules, we further extended definition 1 as follow.

Definition 2 *R_j threatens R_i, if the condition part of R_j becomes satisfied whenever condition part of R_i is satisfied. And, if there exists any contrary* *disjunctive clauses in the action of the two rules. Formally, Rule R_j threaten Rule R_i if:*

$$((C_i \rightarrow C_j) \wedge \neg(\wedge_{k=1}^{n} a_{ik} \wedge \wedge_{k=1}^{m} a_{jk})) \text{ is } \textbf{\textit{valid.}}$$

This new definition allows to capture the potential conflicting rules that we mentioned above. However, it fails again to capture more sophisticated potentially conflicting rules, such as the potential conflict between r_6 and r_8 and, r_7 and r_8 in listing 5. For this, we further extended the above definition by adding a new constraint, that is by asserting every disjunctive clause in C_i to be included in the model (i.e. $\wedge_{k=1}^{n} c_{ik} = \top$).

Definition 3 *Given every disjunctive clause in C_i is satisfiable, that is $\wedge_{k=1}^{n} c_{ik} = \top$, where n is the number of disjunctive clauses in C_i, R_j threatens R_i, if the condition part of R_j becomes satisfied whenever the condition part of R_i is satisfied. And, if there exists contrary disjunctive clauses between R_i and R_j. Formally, Rule R_j threaten Rule R_i iff:*

$$((\wedge_{k=1}^{n} c_{ik} \rightarrow C_j) \wedge \neg(\wedge_{k=1}^{n} a_{ik} \wedge \wedge_{k=1}^{m} a_{jk})) \text{ is } \textbf{\textit{valid.}}$$

This new definition effectively captures both the basic and the more complex potentially conflicting and threatening rule relationships. Listing 6 shows the SMT encoding for the application of Definition 3, to check if r_7 is in conflict with r_8:

```
(assert (not (and
(implies (and (> no_of_people 0) day)
(or (> no_of_people 0) (< temp 18)))
(not (and (and (not light_on) shutter_open)
(and light_on heater_on))))))
(check-sat)
```

Listing 6: Conflicting rule checks SMT example

4.4.2.2 Duplicate rules

Two rules are defined to be duplicate if they have syntactically the same condition and action parts. Formally duplicate rules can be defined as follow:

Definition 4 *R_i and R_j are duplicate iff:*

$$((C_i \leftrightarrow C_j) \wedge (A_i \leftrightarrow A_j)) \text{ is } \textbf{\textit{valid}}$$

In listing 5, r_1 and r_9 are duplicate rules. In general, duplicate rules may not cause logical problems, but affects system performance and maintainability. Listing 7 shows the SMT constraint defined based on definition 4, to check if r_1 and r_9 of listing 5 are duplicate rules.

```
(assert (not (and
(= (> no_of_people 0) (> no_of_people 0))
(= light_on light_on))))
```

Listing 7: Duplicate rules check SMT example

4.4.2.3 Overlapping rules

Two rules are overlapping, if they have the same action parts, but one of them has a more restrictive condition part than the other. Formally, overlapping rules can be defined as follow:

Definition 5 *R_j overlaps with R_i iff:*

$$((C_j \rightarrow C_i) \wedge (A_i \leftrightarrow A_j)) \textit{ is } \boldsymbol{valid}.$$

In listing 5, r_2 and r_6, r_3 and r_2, r_3 and r_6, r_4 and r_7, r_{10} and r_{11}, r_{14} and r_{10}, and r_{14} and r_{11} are all overlapping rules. Like duplicate rules, this kind of rules may affects the performance and maintainability of the reasoning system, but may not cause logical problems. Basically, overlapping rules are very common in rule-based smart home reasoning systems, as the system often requires to make decisions based on numerical sensor values, which have more risk of being overlapping when compared with some constant values or thresholds. For example, definition 5, perfectly detects $Rule_1$ and $Rule_2$, of listing 8, as overlapping. Since, both of these rules can be satisfied when the temperature goes beyond 25°C. Moreover, the UNSAT Core extraction feature of SMT solvers provides a valuable example on when these two rules can overlap (e.g. when temperature $= 26$). This allows the knowledge engineer to understand not only there are overlaps between the two rules, but also get an accurate example of when the rules could overlap. Listing 9 shows the SMT constraint defined based on definition 5, to check if r_3 and r_2 of listing 4 are overlapping rules.

4.4.2.4 Infinite execution loops

An Infinite Execution Loop (IEL) is caused in the reasoning system, when a rule causes itself to fire over and over again (i.e. *self-loop*), or when two or

$$Rule_1 : (temperature > 23.5) \rightarrow heater\_on$$
$$Rule_2 : (temperature > 25) \rightarrow heater\_on$$

Listing 8: Example of overlapping rules based on numerical sensor data

```
(assert (not (and
(implies (and (> no_of_people 0) day)(> no_of_people 0))
(= (not light_on) (not light_on)))))
```

Listing 9: Overlapping rules check SMT example

more rules causes to fire each other in a circular fashion (i.e. *circular rules*). Simply, a rule is self-looping if the execution of its action part leads back to the satisfaction its condition part. Below, these two main causes of IELs are defined and formalized.

Definition 6 *Rule R_i is self-looping iff*

$$(A_i \rightarrow C_i) \text{ is } \textbf{\textit{valid}}.$$

This definition perfectly captures basic self-looping rules such as r_{10}, r_{11} and, r_{12} in listing 5, but do not capture other potentially self-looping rules in the rule base, such as r_{13}. For this reason, we extended our definition by checking weather the satisfaction of each disjunctive prepositional formula a_i in the action part of the rule, that is A_i, will lead back to the satisfaction of the condition part.

Definition 7 *R_i is self-looping iff*

$$(\wedge_{k=1}^{n} a_{ik} \rightarrow C_i) \text{ is } \textbf{\textit{valid}}.$$

Listing 10 shows the SMT constraint defined based on definition 7, to check if R_{13} of listing 4 is self-looping.

```
(assert (not (implies
(and (> no_of_people 0) day)
(> no_of_people 0))))
```

Listing 10: Self-looping rule check SMT example

On the other hand, we call a rule R_i causes a circular IEL, if its satisfaction trigger a sequence of rules in the system, which at some point will trigger back the rule itself in circular fashion. For example, in lisitng 11 below, the satisfaction of r_1 or r_2 will trigger r_3, which in turn triggers r_4, r_5 and r_6, which causes r_1 to trigger back. Therefore, r_1 can be identified as a rule, which causes circular IEL in the system. This kind of rules leads the system into inconsistent and unpredictable states. Thus, to effectively detect them, we extended the above stated definition of self-looping rules as follow.

$$
\begin{aligned}
RB_2 = \{ & \\
& r_1 : (no\_of\_people > 0) \rightarrow light\_on \\
& r_2 : (no\_of\_people > 0) \vee dark \rightarrow light\_on \\
& r_3 : light\_on \rightarrow shutter\_close \\
& r_4 : shutter\_close \rightarrow (no\_of\_people > 0) \\
& r_5 : shutter\_close \rightarrow (no\_of\_people > 0) \wedge dark \\
& r_6 : shutter\_close \rightarrow (no\_of\_people > 0) \vee winter \\
& \}
\end{aligned}
$$

Listing 11: Simple rule base and rule examples

Definition 8 *Given the set of rules in the rule-base are* ***satisfiable****, that is* $\wedge_{i=0}^{m} r_i = \top$*, where* m *is the number of rules in the rule base* RB *and* r_i *is a rule in* RB*. A rule* r_i *causes a circular IEL in the system, if the satisfaction of its action part directly or indirectly (i.e. in a circular fashion) leads back to the satisfaction of the condition part. This can be verified by checking the* ***validity*** *of the constraint* $(A_i \rightarrow C_i)$ *in the given model. Specifically,* $(A_i \rightarrow C_i)$ *is* ***valid****, if* $\neg(A_i \rightarrow C_i)$ *is* ***unsatisfiable****.*

This definition effectively captures IELs caused by all rules except r_6 in listing 11. It is because $\neg(\mathbf{(no\_of\_people} \vee \mathbf{winter}) \rightarrow \mathbf{shutter\_close})$ can be satisfied in the model, by setting $(no\_of\_people > 0) = \bot$ and $winter = \top$. Thus, no rule that leads to r_6 triggers back. To capture this kind of rules, which can potentially cause complex circular IELs in the reasoning system, again we extended our definition as follows. Precisely, by examining the impact of every disjunctive clauses in the action part of the rule.

Definition 9 *Given the set of rules in the rule base are* ***satisfiable****, that is* $\wedge_{i=0}^{m} r_i = \top$*, where* m *is the number of rules in the rule base* RB *and* r_i *is a rule in* RB*. A rule* r_i *causes a circular IELs in the system, if the satisfaction of* ***any disjunctive clause*** *in the action part of the rule, directly or indirectly* *leads back to the satisfaction of the condition part. This can be verified by checking the* ***validity*** *of the constraint* $(\wedge_{k=1}^{n} a_{ik} \rightarrow C_i)$ *in the given model.* $(\wedge_{k=1}^{n} a_{ik} \rightarrow C_i)$ *is* ***valid****, if* $\neg(\wedge_{k=1}^{n} a_{ik} \rightarrow C_i)$ *is* ***unsatisfiable****.*

Listing 12 shows the SMT constraint defined based on definition 9, to check if r_6 of listing 11 causes circular an IEL in the system.

```
;rules assertions
(assert (implies (> number_of_people 0) light_on))
(assert (implies (or (> number_of_people 0) dark)
         light_on))
(assert (implies light_on shutter_close))
(assert (implies shutter_close (> number_of_people 0)))
(assert (implies shutter_close
          (and (> number_of_people 0) dark)))
(assert (implies shutter_close
        (or (> number_of_people 0) winter)))

;IEL check assertion
(assert  (not (implies (and (> number_of_people 0)
         winter) shutter_close)))
(check-sat)
```

Listing 12: SMT Example

4.5 Experimental Evaluation

The proposed SMT formalization of major sources of inconsistencies in rule-based smart home reasoning systems was validated empirically. Besides, the CPU time that a state-of-the-art SMT solver requires to analyze a real-world smart home reasoning system was examined. For the experiment, Z3 SMT Solver[102] was utilized. Z3 is an efficient SMT solver freely available from Microsoft Research. Mainly, it is used for software verification and analysis purposes. Z3 supports theories such as *linear and nonlinear arithmetic, bit-vectors, arrays, quantifiers, and strings*. And, for many years, it achieved high results in annual SMT-competitions[1]. Z3 is available both as a command-line tool and binary API, which provides bindings for various programming languages. For this experiment, the command line Z3 tool and its SMT-LIB based textual input format were utilized.

---

[1] https://smt-comp.github.io/previous.html

4.5.1 Experimental setup

As a seed model for the experimental evaluations, 133 ECA based real smart home rules were collected from an openHAB[2] based open-source smart home project[3], and these rules were manually encoded as an SMT program. The encoded SMT model contains 334 variables (i.e. 218 Boolean, 83 String, and 22 Real), 25 conflicting, four overlapping, seven self-looping, and ten circular rules, which causes circular IELs in the system. Among the 133 rules, the most complex one was formed from the combination of 46 variables and the simplest one from five variables. Afterwards, more larger SMT models were produced by automatically replicating the seed model up to ten times. As discussed in the previous section, the proposed model was evaluated using Z3 SMT solver, version 4.8.6, and all the experiments have been performed on an Intel Core i7-7700HQ CPU @ 2.80GHz with 8 GB of RAM running Ubuntu 18.04 LTS. Table 1 summarizes the experimental data.

Table 4.1: Summary of the experimental data

Experiment	No. of Rules	No. of Conflicting Rules	No. of Duplicate Rules	No. of Overlapping Rules	No. of Self-looping Rules	No. of Circular Rules	No. of Boolean Variables	No. of String Variables	No. of Real Variables	Total Number of Variables
1	133	25	3	4	7	10	218	83	22	323
2	266	50	6	8	14	20	436	166	44	646
3	399	75	9	12	21	30	654	249	66	969
4	532	100	12	16	28	40	872	332	88	1292
5	665	125	15	20	35	50	1090	415	110	1615
6	798	150	18	24	42	60	1308	498	132	1938
7	931	175	21	28	49	70	1526	581	154	2261
8	1064	200	24	32	56	80	1744	664	176	2584
9	1197	225	27	36	63	90	1962	747	198	2907
10	1330	250	30	40	70	100	2180	830	220	3230

---

[2] https://www.openhab.org/

[3] https://github.com/ThomDietrich/openhab-config.git

4.5.2 Tests and Results

To validate the proposed formalization first we analyzed the encoded (i.e. seed) SMT model using Z3. Second, to measure the CPU time that Z3 requires to analyze and detect inconsistencies in real-world smart home reasoning system, we run each of the ten replicated models five times, and recorded the mean and standard deviation of the analysis time. In all the experiments, the proposed SMT constraints successfully detected all the inconsistencies (i.e. duplicate, overlapping, conflicting, self-looping and circular rules) present in the model. However, from the experimental results, the analysis process, especially for duplicate, overlapping and conflicting rules, is noted to be computationally very expensive compared with self-looping and circular rules. For instance, in order to analyze and detect conflicting rules in a rule-base which contains 1330 smart-home rules, the SMT solver required slightly more than ten minutes. On the other hand, to check the existence of self-looping rules in a rule-base which contains the same amount of rules, the SMT solver required on average only 0.33 seconds. This is due to the fact that, checking the existence of duplicate, overlapping, and conflicting rules requires sequential one-by-one comparison of each rule against another in the system. Specifically, given n rules, the SMT solver needs to perform $n(n-1)$ rule comparisons for conflicting and overlapping rules, and $\frac{n(n-1)}{2}$ rule comparisons for duplicate rules. As depicted in figure 4.1, the fact that the SMT solver needs to check only $\frac{n(n-1)}{2}$ constraints for duplicate rules reduces the analysis time by half with respect to conflicting and duplicate rules.

Whereas, to detect if a rule cause a circular IEL in the system, the SMT solver needs only to sequentially check each rule against all other rules in the system, at a time, until it finds the first circular IEL that the rule under investigation causes in the system. On the other hand, detecting self-looping rule is much less computationally expensive than the others. This is because the SMT solver requires only to check whether the condition part of the rule is a logical consequence of its action part, or not. Generally, for the analysis of circular and self-looping rules, the SMT solver needs to check the validity of the constraints defined in section 4.4, n times, for n number of rules. And, as can be noted in figure 4.2, the fact that the process of detecting cycles in the rule base goes until the first cycle is detected make the process of detecting rules, which causes a circular IEL, longer than detecting the self-looping ones. Table 4.2 presents the complete results of the experiments.

In general, the experimental results suggest that, the approach followed in this work is computationally expensive. Nonetheless, it provides a reliable and practical solution for analysing major causes of inconsistencies in rule-based smart home reasoning systems. Most importantly, smart homes are known to be safety critical systems, thus their behaviour and operation needs to be fully

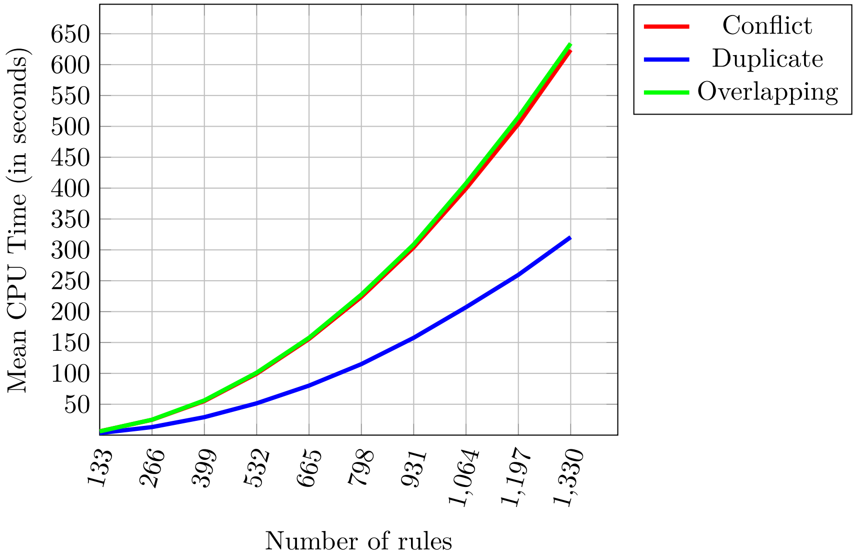

Figure 4.1: Mean CPU time comparison of conflicting, overlapping and duplicate rules analysis.

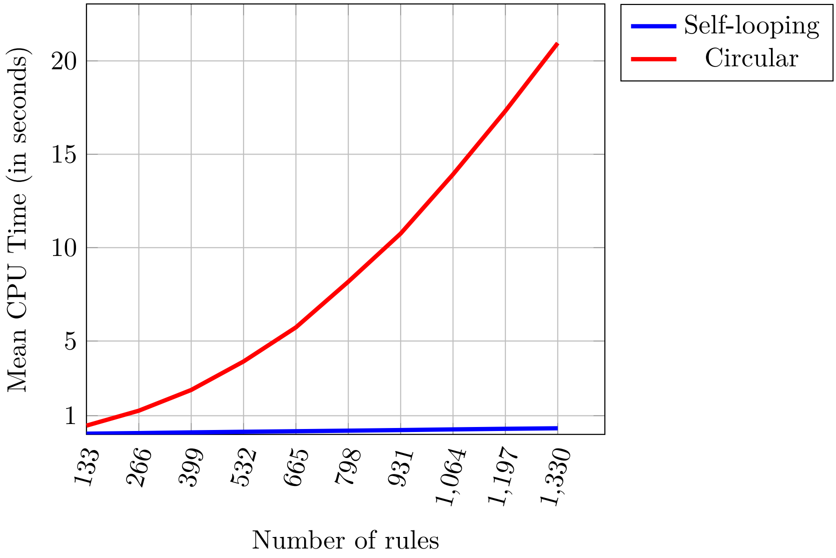

Figure 4.2: Mean CPU time comparison of self-looping and circular rule analysis.

Table 4.2: Mean CPU time (sec.) of inconsistency analysis of rule-based smart home reasoning system using SMT

Experiment	*No. of rules*	Mean analysis time(sec)				
		Duplicates	Overlaps	Conflicts	Self-loops	Circular
1	133	3.28±0.11	6.1±0.17	6.16±0.04	0.04±0	0.47±0
2	266	13.03±0.02	25±0.07	24.79±0.12	0.07±0	1.27±007
3	399	29.09±0.46	56.30±0.4	55.17±0.33	0.11±0.004	2.38±0.28
4	532	51.54±0.68	101.16±0.74	99.53±0.72	0.14±0	3.91±0.02
5	665	80.18±0.45	157.66±2	155.76±0.69	0.17±0	5.73±0.04
6	798	115.01±2.17	227.64±3.4	223.73±0.95	0.2±0	8.18±0.05
7	931	157.59±1.89	308.78±0.93	304.01±2	0.23±0.004	10.75±0.07
8	1064	207.15±1.31	407.71±1.47	399.28±1.86	0.27±0.004	13.92±0.01
9	1197	259.56±3.11	515.03±2.18	503.93±2.77	0.3±0	17.32±0.01
10	1330	320.63±2.03	634.31±5.96	623.89±1.1	0.33±0.004	20.95±0.08

studied and evaluated before their initial deployment and use. For this, the SMT based approach presented in this chapter can be considered as an effective solution.

4.5.3 Threats to the validity of the experiments

The results presented in the experimental analysis are based on a sample real-world smart home reasoning system, which we believe is complex enough to test our proposed method. However, first, the complexity of the system is highly dependent on the number of Boolean, String and Real variables in the system, and in the way, they are combined using the Boolean operators to form the rules. Second, the existence and nonexistence of each of the inconsistencies in the system may also affect the result. For example, for circular rules, the presented analysis technique looks for the first example that the rule under investigation could cause IEL in the system. But, it will identify a rule as a non-circular IELs causing, only after it analyzes the entire system. That means, analyzing a circular IEL free model could require more time compared with a model which contains rules that cause IELs in the system. Further, as the experimental results are based on a real-world reasoning system, the number of conflicting, duplicate, overlapping, self-looping and circular rules are not evenly distributed in the model. As a result, these factors may affect the validity of the presented experimental results.

4.6 Chapter Conclusions

The systematic literature review presented in chapter 2 of this thesis identified conflict detection and resolution as one of the major requirements and challenges of SHRS. The same study highlighted the lack of contributions in the literature to verify the consistency of the system. With that in mind, this chapter defines, formalizes and demonstrates how conflicting, duplicate, overlapping, self-looping and circular rules in smart home reasoning system can be detected using satisfiability modulo theories.

In general, the chapter presents a reliable and effective solution to analyze and verify the consistency of rule-based smart home reasoning systems. The proposed method is validated empirically using rules collected from a real-world smart home reasoning system as a model. Besides, the CPU time that a state-of-the-art SMT solver requires to analyze the model is examined. The experimental results provide compelling evidence for the effectiveness of the proposed solution, mainly by detecting all the inconsistencies present in the model. The proposed method has multiple applications. First, it can be used to build a static rule-based reasoning system verification tool. Second, it can be integrated as a rule validation component of the reasoning system. The rule validation component allows to verify the consistency of the system after the homeowner inserted a new rule or updated an existing one.

Some limitations are worth noting. First, the proposed method can only be used to detect structural but not semantic inconsistencies. However, in smart homes, the execution of some non-structurally conflicting rules may still result in an unwanted effect in the environment. For instance, having noticed that the inhabitant is watching movie, to dim the brightness of the room, the reasoning system may trigger a rule which closes all the shutters, followed by another rule which perceives the darkness of the room and turns on the lights. These two rules may not be structurally conflicting, but have opposite effects in the environment. Thus, these kinds of semantically conflicting rules need to be detected and resolved. Second, as can be noted from the experimental results, the approach followed in this study is computationally expensive. Therefore, future work will address these limitations. Specifically, as part of our future work, we aim to extend the basic ECA based rule model by introducing new attributes. The new attributes should allow specifying properties, such as *when* (e.g. weekend, summer, night ...) and *where* the rule will be executed (e.g. bedroom, living room, garden ...), and *which* environmental parameter their execution affects (e.g. brightness, temperature ...). It is believed that this kind of rule structure enables analyzing and detecting semantically conflicting rules. Further, it will make the inconsistency verification more modular, hence it improves the analysis time. Precisely, the extended rule structure will allow grouping the rules based on their spatiotemporal properties, and to verify their consistency within these groups, unlike the present study which evaluates a rule against all other rules in the system.

Chapter 5

Conclusions

This thesis dealt with three key challenges of smart home reasoning systems. These are: *the lack of systematic investigation of the domain, reasoning under uncertainty in the smart home environments, and consistency verification of the reasoning system.* Accordingly, its contributions aimed at the following two main goals:

RG1: A systematic investigation of the literature in the SHRS domain.

RG2: And, address some of the principal challenges of SHRSs' that are identified by the systematic investigation. Specifically:

  I. Propose a probabilistic multi-agent system architecture to model the dynamic and distributed nature of the home environment, and to deal with uncertainty issues in this environment.

  II. Propose a consistency verification method for rule-based smart home reasoning systems.

In this regard, this chapter is organized as follows: Section 5.1 summarizes the contributions of the thesis, and section 5.2 gives directions for future research inspired by the work done in the thesis.

5.1 Concluding Remarks

Over the past decade, the interest in smart home technologies is growing rapidly. For this several reasons can be given, such as the growing number of world elderly population and their need for home-based healthcare and assistance services, the rising world energy demand and the need to combat global climate changes, and the recent advancements in AI and IoT. As a result, smart homes promise to provide a universal solution that fulfills all these needs. Smart homes are already being in use for home energy management, elderly monitoring and healthcare, comfort and entertainment services. However, before we realize a truly intelligent smart home system, several challenges need to be addressed. Some of these challenges lie in the SHRS. The SHRS is a component of the smart home system, which is responsible for the automatic adaptation and control of the home environment. This system is expected to make appropriate decisions towards achieving the comfort and efficiency goals of the inhabitant and its environment. Within this context, this thesis investigated the SHRS domain and pointed out solutions that can help in addressing major research problems in the field. Below we would like to re-emphasize the major contributions of this thesis.

In chapter 2, we presented the results of a systematic literature review targeted at exploring SHRSs. The literature review was focused on answering seven research questions, and its main findings are the following:

- Most smart homes are designed to provide general home automation services such as light and HVAC control. Nonetheless, health care and home energy management are the most popular specific AAL services presented in the literature.

- The most essential features and requirements of the SHRS are: *activity and situation awareness, context awareness, ability to learn and adapt, ability to plan, ability to predict, ability to explain its decision, ability to reason about time, and ability to reason under uncertainty.* Further, the SHRS needs to be equipped with a standard conflict detection and resolution strategy and should provide a natural interaction mechanism for its users.

- Most research contributions in SHRS domain are based on symbolic AI approaches. However, over the past few years, the interest in statistical and hybrid approaches is relatively growing.

- Most research contributions in the SHRS domain assumed the existence of a single inhabitant in a fully observable environment. Further, many studies in the same domain assumed the presence of complete, consistent and non-conflicting information about the home environment and an always rational inhabitant.

- Most research works in the SHRS domain are presented as theoretical contributions. Thus, to examine the proposed methods practical aspects, an increased number of contributions needs to be studied in realistic settings.

- Most contributions in the SHRS domain demonstrated the potential applicability of their proposed methods in multiple AAL scenarios. Further, a good number of them proposed solutions for learning inhabitants' changing behavior, self-adaptation, reasoning under uncertainty and conflict resolution. However, very few of them integrates decision explanation mechanisms and, above all, only a very limited number of them handles overlapping and simultaneous multiple inhabitants' activities.

- The dynamic and partial observability of the AAL environment, the heterogeneity and unreliability of smart home data, the conflicting and frequently changing inhabitants needs, and context representation and contextual reasoning are the major challenges of building a reliable and practical SHRS.

Besides, the SLR discussed the opportunities and challenges presented by the recent advancement of AI and IoT for the development of smart home systems. Further, it highlighted the rarity of studies in tackling smart home security issues at the reasoning system level. As a conclusion, it insisted that to give a genuine intelligent behavior for the SHRS, at least most of the aforementioned features and requirements of the system need to be satisfied, and their presented challenges need to be addressed. Finally, the study provides the following two suggestions as a starting point in overcoming some of the challenges of SHRSs:

I. Combining the best of symbolic and statistical AI techniques, and going forward towards a more hybrid and distributed reasoning system.

II. Integration of standard conflict detection, conflict resolution and sensor data contextualization strategies to improve the performances of SHRSs.

In chapter 3, based on the first suggestion of the SLR, we proposed probabilistic multi-agent system architecture for reasoning under uncertainty in smart homes. The presented work was based on a probabilistic logic programming technique called ProbLog and multi-agent system technologies. ProbLog allows the design of a hybrid reasoning system, whereas MAS allows distributing the reasoning process over multiple nodes. In general, the chapter illustrated how the probabilistic reasoning technique enables intelligent agents to reason under uncertain situations using a vague inhabitant command as a scenario. Further, it discussed how the agent interaction protocols enhance the reasoning process by allowing the agents to exchange information about missing data or unobservable variables between the agents. Besides, it showed how an agent can take advantage of the interaction protocols and delegate its reasoning tasks for other agents in the system, when it lacks the necessary computational resources to accomplish it by itself. Therefore, this chapter indicated that the integration of MAS and probabilistic logic programming can help in building a reasoning system for AAL environments, which is capable of performing well under vague inhabitant commands, missing information, and in a partially observable environment. Moreover, our proof-of-concept implementation and experimental analysis suggested that this approach appears to be effective in counteracting uncertainties in the reasoning systems of these complex and dynamic environments. This work is the first study to our knowledge, which integrates the ProbLog reasoning engine into a multi-agent system framework and to utilize it for tackling uncertainty issues in smart home environments.

As per the second suggestion of the SLR, in chapter 4 of this thesis, we tackled inconsistency issues in SHRSs. Specifically, we defined, formalized and demonstrated how conflicting, duplicate, overlapping, self-looping and circular rules in SHRSs can be detected using satisfiability modulo theories. The proposed method is validated empirically using rules collected from a real-world SHRS as a model. The experimental results provide compelling evidence for the effectiveness of the proposed solution. This method can have multiple applications. First, it can be used to build a static rule-based reasoning system verification tool. Second, it can be integrated as a rule validation component of the reasoning system. This component allows verifying the consistency of the system after the homeowner inserted a new rule or updated an existing one. In addition, with some adaptation, the method can be directly used to verify the consistency properties of reasoning systems in other domains. In general, the chapter presented a reliable and effective solution to analyze and verify the consistency of a rule-based SHRS.

5.2 Future Research Directions

The structure and parameters of the probabilistic rules presented as part of the MAS based smart home architecture in chapter 3, are designed based on our subjective knowledge, yet can be learned from data. For this purpose, we plan to use a probabilistic rule learner tool such as ProbFOIL[87]. This activity can be facilitated by collecting a multimodal dataset, which comprises inhabitants ADL along with the state of each smart home devices in the environment. However, we noted the lack of a comprehensive dataset that can be used for this purpose. Nonetheless, as a continuation of the presented work, we are planning to try out the Sweet-Home [103] and MavHome [32] datasets. Further, we aim to test this architecture in real-world settings.

Likewise, the consistency verification method presented in chapter 4 of this thesis has some limitations. First, the proposed method can only be used to detect structural but not semantic inconsistencies. However, in smart homes, the execution of some non-structurally conflicting rules may still result in an unwanted effect in the environment. For instance, having noticed that the inhabitant is watching movie, to dim the brightness of the room, the reasoning system may trigger a rule which closes all the shutters, followed by another rule which perceives the darkness of the room and turns on the lights. These two rules may not be structurally conflicting, but have opposite effects in the environment. Thus, these kinds of semantically conflicting rules need to be detected and resolved. Second, as can be noted from the experimental results, the approach followed in this study is computationally expensive. Therefore, future work will address these limitations. Specifically, as part of our future work, we aim to extend the basic ECA based rule model by introducing new attributes. The new attributes should allow specifying properties, such as *when* (e.g. weekend, summer, night ...) and *where* the rule will be executed (e.g. bedroom, living room, garden ...), and *which* environmental parameter its execution affects (e.g. brightness, temperature ...). It is believed that this kind of rule structure enables analyzing and detecting semantically conflicting rules. Further, it will make the inconsistency verification more modular, hence it improves the analysis time. Precisely, the extended rule structure will allow grouping the rules based on their spatiotemporal properties, and to verify their consistency within these groups, unlike the presented study which evaluates a rule against all other rules in the system. In addition, using the formalization presented in the chapter, we are planning to develop a static consistency analysis tool for SHRSs.

Finally, to realize a truly intelligent smart home system more research on the handling of multiple inhabitants ADLs need to be performed. Besides, relatively simple and cost-effective ways to test SHRSs in real-world settings need to be studied. Moreover, future work should explore decision explanation methods, privacy and security issues in smart home reasoning systems.

Bibliography

[1] D. Calvaresi, D. Cesarini, P. Sernani, M. Marinoni, A. F. Dragoni, and A. Sturm, "Exploring the ambient assisted living domain: a systematic review," *Journal of Ambient Intelligence and Humanized Computing*, vol. 8, no. 2, pp. 239–257, 2017.

[2] W. He, D. Goodkind, P. R. Kowal *et al.*, "An aging world: 2015," 2016.

[3] W. H. Organization, *World report on ageing and health.* World Health Organization, 2015.

[4] A. M. Omer, "Energy, environment and sustainable development," *Renewable and sustainable energy reviews*, vol. 12, no. 9, pp. 2265–2300, 2008.

[5] I. IEA, "International energy agency. key world energy statistics," 2015.

[6] G. Demiris and B. K. Hensel, "Technologies for an aging society: a systematic review of "smart home" applications," *Yearbook of medical informatics*, vol. 17, no. 01, pp. 33–40, 2008.

[7] C.-H. Lu, C.-L. Wu, M.-Y. Weng, W.-C. Chen, and L.-C. Fu, "Context-aware energy saving system with multiple comfort-constrained optimization in m2m-based home environment," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 3, pp. 1400–1414, 2017.

[8] N. Falcionelli, P. Sernani, A. Brugués, D. N. Mekuria, D. Calvaresi, M. Schumacher, A. F. Dragoni, and S. Bromuri, "Indexing the event calculus: Towards practical human-readable personal health systems," *Artificial intelligence in medicine*, 2018.

[9] H. Si, Y. Kawahara, H. Morikawa, and T. Aoyama, "A stochastic approach for creating context-aware services based on context histories in smart home," *COGNITIVE SCIENCE RESEARCH PAPER-UNIVERSITY OF SUSSEX CSRP*, vol. 577, p. 37, 2005.

[10] J. R. Velasco, I. Marsá-Maestre, A. Navarro, M. A. López-Carmona, A. J. de Vicente, E. De La Hoz, A. Paricio, and M. Machuca, "Location-aware services and interfaces in smart homes using multiagent systems." *PSC*, vol. 5, pp. 104–110, 2005.

[11] D. N. Mekuria, P. Sernani, N. Falcionelli, and A. F. Dragoni, "Reasoning in multi-agent based smart homes: A systematic literature review," in *Proceedings of the 9th Italian Ambient Assisted Forum.* Springer, 2018.

[12] N. F. A. F. D. Dagmawi Neway Mekuria, Paolo Sernani, "Smart home reasoning systems: a systematic literature review," *Journal of Ambient Intelligence and Humanized Computing*, 2019.

[13] D. N. Mekuria, P. Sernani, N. Falcionelli, and A. F. Dragoni, "Probabilistic logic reasoning in multi-agent based smart home environment," in *Italian Forum of Ambient Assisted Living.* Springer, 2019, pp. 161–179.

[14] D. N. Mekuria, P. Sernani, N. Falcionelli, and A. F. Drazoni, "A probabilistic multi-agent system architecture for reasoning in smart homes," in *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA).* IEEE, 2019, pp. 1–6.

[15] M. Chan, D. Estève, C. Escriba, and E. Campo, "A review of smart homes-present state and future challenges," *Computer methods and programs in biomedicine*, vol. 91, no. 1, pp. 55–81, 2008.

[16] L. Liu, E. Stroulia, I. Nikolaidis, A. Miguel-Cruz, and A. R. Rincon, "Smart homes and home health monitoring technologies for older adults: A systematic review," *International journal of medical informatics*, vol. 91, pp. 44–59, 2016.

[17] C. Wilson, T. Hargreaves, and R. Hauxwell-Baldwin, "Smart homes and their users: a systematic analysis and key challenges," *Personal and Ubiquitous Computing*, vol. 19, no. 2, pp. 463–476, 2015.

[18] T. K. Hui, R. S. Sherratt, and D. D. Sánchez, "Major requirements for building smart homes in smart cities based on internet of things technologies," *Future Generation Computer Systems*, vol. 76, pp. 358–369, 2017.

[19] M. Marschollek, "Decision support at home (ds@ home)–system architectures and requirements," *BMC medical informatics and decision making*, vol. 12, no. 1, p. 43, 2012.

[20] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, no. 2004, pp. 1–26, 2004.

[21] R. Van Solingen, V. Basili, G. Caldiera, and H. D. Rombach, "Goal question metric (gqm) approach," *Encyclopedia of software engineering*, 2002.

[22] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of systems and software*, vol. 80, no. 4, pp. 571–583, 2007.

[23] F. Corno and F. Razzak, "Intelligent energy optimization for user intelligible goals in smart home environments," *IEEE transactions on Smart Grid*, vol. 3, no. 4, pp. 2128–2135, 2012.

[24] P. Chahuara, F. Portet, and M. Vacher, "Context-aware decision making under uncertainty for voice-based control of smart home," *Expert Systems with Applications*, vol. 75, pp. 63–79, 2017.

[25] C. Li, L. Suna, and X. Hua, "A context-aware lighting control system for smart meeting rooms," *Systems Engineering Procedia*, vol. 4, pp. 314–323, 2012.

[26] S. Zhang, S. McClean, B. Scotney, and C. Nugent, "Learning under uncertainty in smart home environments," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE.* IEEE, 2008, pp. 2083–2086.

[27] J. C. Augusto, J. Liu, P. McCullagh, H. Wang, and J.-B. Yang, "Management of uncertainty and spatio-temporal aspects for monitoring and diagnosis in a smart home," *International Journal of Computational Intelligence Systems*, vol. 1, no. 4, pp. 361–378, 2008.

[28] T. Van Nguyen, Y. C. Woo, and D. Choi, "Ccbr: Chaining case based reasoning in context-aware smart home," in / *2009 First Asian Conference on Intelligent Information and Database Systems.* IEEE, 2009, pp. 453–458.

[29] R. Velik, G. Zucker, and D. Dietrich, "Towards automation 2.0: a neurocognitive model for environment recognition, decision-making, and action execution," *EURASIP Journal on Embedded Systems*, vol. 2011, p. 4, 2011.

[30] M. Alirezaie, J. Renoux, U. Köckemann, A. Kristoffersson, L. Karlsson, E. Blomqvist, N. Tsiftes, T. Voigt, and A. Loutfi, "An ontology-based context-aware system for smart homes: E-care@ home," *Sensors*, vol. 17, no. 7, p. 1586, 2017.

[31] P. Carner, "Beyond home automation: designing more effective smart home systems," in *9th. IT & T Conference*, 2009, p. 14.

[32] B. Das, C. Chen, N. Dasgupta, D. J. Cook, and A. M. Seelye, "Automated prompting in a smart home environment," in *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on.* IEEE, 2010, pp. 1045–1052.

[33] T. Tiberghien, M. Mokhtari, H. Aloulou, and J. Biswas, "Semantic reasoning in context-aware assistive environments to support ageing with dementia," in *International Semantic Web Conference.* Springer, 2012, pp. 212–227.

[34] A. Sanchez, R. Tercero, D. Saldaña, and L. Burnell-Ball, "Sdbi: An ontology based smart home lab environment," *Proceedings WILE-8th MICAI*, pp. 217–226, 2009.

[35] V. Degeler and A. Lazovik, "Dynamic constraint reasoning in smart environments," in *Tools with Artificial Intelligence (ICTAI), 2013 IEEE 25th International Conference on.* IEEE, 2013, pp. 167–174.

[36] A.-M. Vainio, M. Valtonen, and J. Vanhala, "Learning and adaptive fuzzy control system for smart home," in *Developing Ambient Intelligence.* Springer, 2006, pp. 28–47.

[37] B. Alhafidh and W. Allen, "Design and simulation of a smart home managed by an intelligent self-adaptive system," *International Journal of Engineering Research and Applications*, vol. 6, no. 8, pp. 64–90, 2016.

[38] B. Xiao, L. Chen, M. Liu, Y. Cao, and Y. Yang, "Design and implementation of rule-based uncertainty reasoning in smart house," in *Communication Technology (ICCT), 2015 IEEE 16th International Conference on.* IEEE, 2015, pp. 441–448.

[39] G. Acampora, V. Loia, and A. Vitiello, "Distributing emotional services in ambient intelligence through cognitive agents," *Service Oriented Computing and Applications*, vol. 5, no. 1, pp. 17–35, 2011.

[40] H. Lee, J. S. Choi, and R. Elmasri, "A dynamic evidential network for multisensor context reasoning in home-based care," in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on.* IEEE, 2009, pp. 4994–4999.

[41] C. Vivekananthan, Y. Mishra, and F. Li, "Real-time price based home energy management scheduler," *IEEE Transactions on Power Systems*, vol. 30, no. 4, pp. 2149–2159, 2015.

[42] L. Chen, C. D. Nugent, M. Mulvenna, D. Finlay, X. Hong, and M. Poland, "A logical framework for behaviour reasoning and assistance in a smart

home," *International Journal of Assistive Robotics and Mechatronics*, vol. 9, no. 4, pp. 20–34, 2008.

[43] S. Zhang, P. McCullagh, C. Nugent, H. Zheng, and N. Black, "An ontological framework for activity monitoring and reminder reasoning in an assisted environment," *Journal of Ambient Intelligence and Humanized Computing*, vol. 4, no. 2, pp. 157–168, 2013.

[44] B. Xu, Y. Ge, I. Chen, Z. Chen, and Y. Ling, "Elderly personal safety monitoring in smart home based on host space and travelling pattern identification," *Information Technology Journal*, vol. 11, no. 8, 2012.

[45] C. H. Lim, P. Anthony, and L. C. Fan, "Applying multi-agent system in a context aware smart home," *Learning*, vol. 24, pp. 53–64, 2009.

[46] V. Degeler and E. Curry, "Human-assisted rule satisfaction in partially observable environments," in *Ubiquitous Intelligence and Computing, 2014 IEEE 11th Intl Conf on and IEEE 11th Intl Conf on and Autonomic and Trusted Computing, and IEEE 14th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UTC-ATC-ScalCom).* IEEE, 2014, pp. 171–178.

[47] C. Doebel and C. Ament, "Enkos–a smart home control system basing on learning classifier systems," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 1465–1470, 2014.

[48] V. R. Jakkula and D. J. Cook, "Detecting anomalous sensor events in smart home data for enhancing the living experience." *Artificial intelligence and smarter living*, vol. 11, no. 201, p. 1, 2011.

[49] N. Alshammari, T. Alshammari, M. Sedky, J. Champion, and C. Bauer, "Openshs: Open smart home simulator," *Sensors*, vol. 17, no. 5, p. 1003, 2017.

[50] T. Van Nguyen, J. G. Kim, and D. Choi, "Iss: the interactive smart home simulator," in *2009 11th International Conference on Advanced Communication Technology*, vol. 3. IEEE, 2009, pp. 1828–1833.

[51] P. Sernani, A. Claudi, P. Calvaresi, D. Accattoli, R. Tofani, and A. F. Dragoni, "Using 3d simulators for the ambient assisted living," in *Proceedings of the 3rd International Conference on Artificial Intelligence and Assistive Medicine-Volume 1213.* Citeseer, 2014, pp. 16–20.

[52] M. H. Kabir, M. R. Hoque, H. Seo, and S.-H. Yang, "Machine learning based adaptive context-aware system for smart home environment," *International Journal of Smart Home*, vol. 9, no. 11, pp. 55–62, 2015.

[53] B. Qela and H. T. Mouftah, "Observe, learn, and adapt (ola)-an algorithm for energy management in smart homes using wireless sensors and artificial intelligence," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 2262–2272, 2012.

[54] M. Antunes, D. Gomes, and R. Aguiar, "Towards behaviour inference in smart environments," in *Future Internet Communications (CFIC), 2013 Conference on.* IEEE, 2013, pp. 1–8.

[55] L. Chen, C. Nugent, and A. Al-Bashrawi, "Semantic data management for situation-aware assistance in ambient assisted living," in *Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services.* ACM, 2009, pp. 298–305.

[56] J. Al-Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas, "Cerberus: a context-aware security scheme for smart spaces," in *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on.* IEEE, 2003, pp. 489–496.

[57] J. Dahmen, B. L. Thomas, D. J. Cook, and X. Wang, "Activity learning as a foundation for security monitoring in smart homes," *Sensors*, vol. 17, no. 4, p. 737, 2017.

[58] K. Gayathri and K. Easwarakumar, "Intelligent decision support system for dementia care through smart home," *Procedia Computer Science*, vol. 93, pp. 947–955, 2016.

[59] C. Chen, B. Das, and D. J. Cook, "A data mining framework for activity recognition in smart environments," in *2010 Sixth International Conference on Intelligent Environments.* IEEE, 2010, pp. 80–83.

[60] O. A. Sianaki and M. A. Masoum, "A fuzzy topsis approach for home energy management in smart grid with considering householders' preferences," in *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT).* IEEE, 2013, pp. 1–6.

[61] G. M. Youngblood, D. J. Cook, and L. B. Holder, "Managing adaptive versatile environments," *Pervasive and Mobile Computing*, vol. 1, no. 4, pp. 373–403, 2005.

[62] V. Gokul, P. Kannan, S. Kumar, and S. G. Jacob, "Deep q-learning for home automation," *International Journal of Computer Applications*, vol. 152, no. 6, pp. 1–5, 2016.

[63] Y. Chen, I. Chen, and W. Chang, "Context-aware services based on osgi for smart homes. ubi-media computing (u-media)," in *2010 3rd IEEE International Conference on*, vol. 11, 2010, p. 392.

[64] A. K. Ray, G. Leng, T. M. McGinnity, S. A. Coleman, and L. P. Maguire, "Dynamically reconfigurable online self-organising fuzzy neural network with variable number of inputs for smart home application." in *IJCCI*, 2013, pp. 507–514.

[65] C. Panagiotou, T. Panagiotakopoulos, and A. Kameas, "A multi: modal decision making system for an ambient assisted living environment," in *Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments.* ACM, 2015, p. 44.

[66] Z. Lu, J. Augusto, J. Liu, and H. Wang, "A linguistic truth-value temporal reasoning (ltr) system and its application to the design of an intelligent environment," *International Journal of Computational Intelligence Systems*, vol. 5, no. 1, pp. 173–196, 2012.

[67] Q. Ni, A. B. García Hernando, and I. Pau de la Cruz, "A context-aware system infrastructure for monitoring activities of daily living in smart home," *Journal of Sensors*, vol. 2016, 2016.

[68] P. Sukanya and K. Gayathri, "An unsupervised pattern clustering approach for identifying abnormal user behaviors in smart homes 1," *International Journal of Computer Science and Network*, 2013.

[69] T. Van Nguyen, W. Lim, H. Nguyen, D. Choi, and C. Lee, "Context ontology implementation for smart home," *arXiv preprint arXiv:1007.1273*, 2010.

[70] B. Ahmed, G. Abdelouahed, and O. Kazar, "Semantic-based approach to context management in ubiquitous environment," *Procedia Computer Science*, vol. 109, pp. 592–599, 2017.

[71] A. M. Elmahalawy, "Intelligent agent and multi agent systems," *Journal of Engineering and Technology*, vol. 2, no. 1, 2009.

[72] D. N. Mekuria, P. Sernani, N. Falcionelli, and A. F. Dragoni, "Reasoning in multi-agent based smart homes: A systematic literature review," in *Italian Forum of Ambient Assisted Living.* Springer, 2018, pp. 161–179.

[73] M. Wooldridge and N. R. Jennings, "Intelligent agents: Theory and practice," *The knowledge engineering review*, vol. 10, no. 2, pp. 115–152, 1995.

[74] N. J. Nilsson, "Probabilistic logic," *Artificial intelligence*, vol. 28, no. 1, pp. 71–87, 1986.

[75] P. Sernani, A. Claudi, L. Palazzo, G. Dolcini, and A. F. Dragoni, "Home care expert systems for ambient assisted living: A multi-agent approach," in *Proceedings of the Workshop on The Challenge of Ageing Society: Technological Roles and Opportunities for Artificial Intelligence, Turin, Italy*, vol. 6, 2013.

[76] C. Reinisch, M. J. Kofler, F. Iglesias, and W. Kastner, "Thinkhome energy efficiency in future smart homes," *EURASIP Journal on Embedded Systems*, vol. 2011, p. 1, 2011.

[77] S. Abras, S. Ploix, S. Pesty, and M. Jacomino, "A multi-agent home automation system for power management," in *Informatics in Control Automation and Robotics.* Springer, 2008, pp. 59–68.

[78] D. J. Cook, M. Youngblood, and S. K. Das, "A multi-agent approach to controlling a smart environment," in *Designing smart homes.* Springer, 2006, pp. 165–182.

[79] A. Machado, V. Maran, I. Augustin, J. C. Lima, L. K. Wives, and J. P. M. de Oliveira, "Reasoning on uncertainty in smart environments," in *Proceedings of the 18th International Conference on Enterprise Information Systems.* SCITEPRESS-Science and Technology Publications, Lda, 2016, pp. 240–250.

[80] M. Bruynooghe, T. Mantadelis, A. Kimmig, B. Gutmann, J. Vennekens, G. Janssens, and L. De Raedt, "Problog technology for inference in a probabilistic first order logic." in *ECAI*, vol. 215, 2010, pp. 719–724.

[81] T. Sztyler, G. Civitarese, and H. Stuckenschmidt, "Modeling and reasoning with problog: an application in recognizing complex activities," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops).* IEEE, 2018, pp. 259–264.

[82] L. De Raedt, A. Kimmig, and H. Toivonen, "Problog: A probabilistic prolog and its application in link discovery." in *IJCAI*, vol. 7. Hyderabad, 2007, pp. 2462–2467.

[83] (2002) Fipa request interaction protocol specification. [Online]. Available: http://fipa.org/specs/fipa00026/SC00026H.html

[84] R. G. Smith, "The contract net protocol: High-level communication and control in a distributed problem solver," *IEEE Transactions on computers*, no. 12, pp. 1104–1113, 1980.

[85] F. Bellifemine, F. Bergenti, G. Caire, and A. Poggi, “Jade- a java agent development framework,” in *Multi-Agent Programming.* Springer, 2005, pp. 125–147.

[86] G. Shafer, *A mathematical theory of evidence.* Princeton university press, 1976, vol. 42.

[87] L. De Raedt and I. Thon, “Probabilistic rule learning,” in *International Conference on Inductive Logic Programming.* Springer, 2010, pp. 47–58.

[88] P. Sernani, M. Biagiola, N. Falcionelli, D. N. Mekuria, S. Cremonini, and A. F. Dragoni, “Time aware task delegation in agent interactions for video-surveillance.” in *RTcMAS@ IJCAI*, 2018, pp. 16–30.

[89] A. Dries, A. Kimmig, W. Meert, J. Renkens, G. Van den Broeck, J. Vlasselaer, and L. De Raedt, “Problog2: Probabilistic logic programming,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases.* Springer, 2015, pp. 312–315.

[90] M. Suwa, A. C. Scott, and E. H. Shortliffe, “An approach to verifying completeness and consistency in a rule-based expert system,” *Ai Magazine*, vol. 3, no. 4, pp. 16–16, 1982.

[91] Q. Zhang, G. Zeng, C. Xiao, and Y. Yue, “A rule conflict resolution method based on vague set,” *Soft Computing*, vol. 18, no. 3, pp. 549–555, 2014.

[92] C. Chang, J. Combs, and R. Stachowitz, “A report on the expert systems validation associate (eva),” *Expert Systems with Applications*, vol. 1, no. 3, pp. 217–230, 1990.

[93] D. Zhang and D. Nguyen, “A tool for knowledge base verification,” in *Knowledge Engineering Shells: Systems and Techniques.* World Scientific, 1993, pp. 455–486.

[94] D. Xu, K. Xia, D. Zhang, and H. Zhang, “Model checking the inconsistency and circularity in rule-based expert systems.” *Computer and Information Science*, vol. 2, no. 1, pp. 12–17, 2009.

[95] X. Jin, Y. Lembachar, and G. Ciardo, “Symbolic verification of eca rules.” in *PNSE+ ModPE*, 2013, pp. 41–59.

[96] J. Cano, G. Delaval, and E. Rutten, “Coordination of eca rules by verification and control,” in *International Conference on Coordination Languages and Models.* Springer, 2014, pp. 33–48.

[97] Y. Sun, X. Wang, H. Luo, and X. Li, "Conflict detection scheme based on formal rule model for smart building systems," *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 2, pp. 215–227, 2014.

[98] C. Vannucchi, M. Diamanti, G. Mazzante, D. Cacciagrano, R. Culmone, N. Gorogiannis, L. Mostarda, and F. Raimondi, "Symbolic verification of event–condition–action rules in intelligent environments," *Journal of Reliable Intelligent Environments*, vol. 3, no. 2, pp. 117–130, 2017.

[99] L. De Moura and N. Bjørner, "Satisfiability modulo theories: introduction and applications," *Communications of the ACM*, vol. 54, no. 9, pp. 69–77, 2011.

[100] R. Sebastiani, "Lazy satisfiability modulo theories," *Journal on Satisfiability, Boolean Modeling and Computation*, vol. 3, pp. 141–224, 2007.

[101] C. M. Nguyen, R. Sebastiani, P. Giorgini, and J. Mylopoulos, "Multi-objective reasoning with constrained goal models," *Requirements Engineering*, vol. 23, no. 2, pp. 189–225, 2018.

[102] L. De Moura and N. Bjørner, "Z3: An efficient smt solver," in *International conference on Tools and Algorithms for the Construction and Analysis of Systems.* Springer, 2008, pp. 337–340.

[103] M. Vacher, B. Lecouteux, P. Chahuara, F. Portet, B. Meillon, and N. Bonnefond, "The sweet-home speech and multimodal corpus for home automation interaction," 2014.