Analyzing the Interaction between Knowledge and Social Commitments in Multi-Agent Systems

Faisal Suleiman AL-Saqqar

A Thesis

in

The Department

of

Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University

Montréal, Québec, Canada

December, 2015

CONCORDIA UNIVERSITY

Division of Graduate Studies

This is to certify that the thesis prepared

By: **Faisal Suleiman AL-Saqqar**

Entitled: **Analyzing the Interaction between Knowledge and Social Commitments in Multi-Agent Systems**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

_____ Dr. Shahin Hashtrudi Zad (Chair)

_____ Dr. Chamseddine Talhi

_____ Dr. Ferhat Khendek

_____ Dr. Olga Ormandjieva

_____ Dr. Joey Paquet

_____ Dr. Jamal Bentahar

Approved by _____

Chair of the CSE Department

_____ 2015 _____

Dean of Engineering

ABSTRACT

Analyzing the Interaction between Knowledge and Social Commitments in Multi-Agent Systems

Faisal Suleiman AL-Saqqar, Ph.D.

Concordia University, 2015

Both knowledge and social commitments in Multi-Agent Systems (MASs) have long been under research independently, especially for agent communication. Plenty of work has been carried out to define their semantics. However, in concrete applications such as business settings and web-based applications, agents should reason about their knowledge and their social commitments at the same time, particularly when they are engaged in conversations. In fact, studying the interaction between knowledge and social commitments is still in its beginnings. Therefore, in this thesis, we aim to provide a practical and formal framework that analyzes the interaction between knowledge and communicative social commitments in MASs from the semantics, model checking, complexity, soundness and completeness perspectives.

To investigate such an interaction, we, first, combine CTLK (an extension of Computation Tree Logic (CTL) with modality for reasoning about knowledge) and CTLC (an extension of CTL with modalities for reasoning about commitments and their fulfillments) in one new logic named CTLKC. By so doing, we identify some paradoxes in the new logic showing that simply combining current versions of commitment and knowledge logics results in a language of logic that violates some fundamental intuitions. Consequently, we propose CTLKC^{+}, a new consistent logic of knowledge and commitments that fixes the identified paradoxes and allows us to reason about social commitments and knowledge

simultaneously in a consistent manner. Second, we use correspondence theory for modal logics to prove the soundness and completeness of CTLKC$^+$. To do so, we develop a set of reasoning postulates in CTLKC$^+$ and correspond them to certain classes of frames. The existence of such correspondence allows us to prove that the logic generated by any subset of these postulates is sound and complete, with respect to the models that are based on the corresponding frames. Third, we address the problem of model checking CTLKC$^+$ by transforming it to the problem of model checking GCTL* (a generalized version of Extended Computation Tree Logic (CTL*) with action formulas) and ARCTL (the combination of CTL with action formulas) in order to respectively use the CWB-NC automata-based model checker and the extended NuSMV symbolic model checker. Moreover, we prove that the transformation techniques are sound. Fourth, we analyze the complexity of the proposed model checking techniques. The results of this analysis reveal that the complexity of our transformation procedures is PSPACE-complete for local concurrent programs with respect to the size of these programs and the length of the formula being checked. From the time perspective, we prove that the complexity of the proposed approaches is P-complete with regard to the size of the model and length of the formula. Finally, we implement our model checking approaches and report some experimental results by verifying the well-known NetBell payment protocol against some desirable properties.

ACKNOWLEDGEMENTS

My deepest appreciation goes to my supervisor Dr. Jamal Bentahar. I really appreciate his trust on me, his encouragement, his valuable discussions and feedback, and his gentle guidance during the time of my study which helped me accomplish this thesis.

I would like to express my sincere appreciation to my examination committee members: Dr. Chamseddine Talhi, Dr. Ferhat Khendek, Dr. Olga Ormandjieva, and Dr. Joey Paquet for their valuable comments and suggestions.

I would also like to thank both AL al-Bayt University - Jordan and Concordia University - Canada for their financial support. Without their support, I will never had the chance to conduct this research.

From administrative staff, I would like to thank Halina Monkiewicz, the graduate advisor, for her valuable administrative guidance during the course of my study.

I would like to thank my colleagues at Concordia University for sharing nice times at our research lab.

Finally, immeasurable thanks go out to my family (Najah, Remah, Mohammad, Ali, Kefah, Huda, Ashraf, Muhannad and Asma), my wonderful wife (Lina) and my amazing children (Tasneem, Sedra, Yamen and Seham) whom surrounded me with their love and prayers. Without their help and encouragement, I would never finish this work.

To the memory of my parents and to my family,

with love and gratitude.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

ACL	Agent Communication Language
ARCTL	Action Restricted Computation Tree Logic
BDD	Binary Decision Diagram
BNF	Backus-Naur Form
CTL	Computation Tree Logic
CTL*	Extended Computation Tree Logic
CTLC	Computation Tree Logic of Commitment
CTLK	Computation Tree Logic of Knowledge
CTLKC	Computation Tree Logic of Knowledge and Commitment
CWB-NC	Concurrency WorkBench of New Century
GCTL*	Generalized version of Extended Computation Tree Logic
LTL	Linear Temporal Logic
MAS	Multi-Agent System
MCMAS	Model Checker for Multi-Agent Systems
NuSMV	New Symbolic Model Verifier
PRISM	PRobabilistIc Symbolic Model checker
PSPASE	Polynomial Space
SMV	Symbolic Model Verifier

Chapter 1

Introduction

This chapter provides a motivational introduction to the thesis. It starts by introducing the scope of the research. Thereafter, it presents the motivations, research problem, objectives and methodology. Finally, the chapter describes the research structure.

1.1 Scope of Research

Multi-Agent Systems (MASs) paradigm is a popular distributed approach to solve complex computational problems that are beyond the capabilities of individual software systems. Typically, a multi-agent system consists of a collection of autonomous and possibly heterogenous software agents that interact with each other through communication and coordinate their actions in order to reach their goals [100].

The increasing adoption of MASs in various real life applications, for instance, e-commerce, e-health and web services [56], highlighted the importance of a large dimension of agents' aspects that need to be properly addressed. Among the agents' units that proved to have a vital role in the development of effective MASs are knowledge, commitments, trust, reputation, uncertainty [8, 68]. Each of these significant aspects has been extensively

studied independently. In this thesis, we focus on the domain of reasoning about knowledge, communicative social commitments and their interactions in MASs.

1.2 Motivations

In the literature, plenty of research work has been carried out to define the semantic of knowledge [43, 50, 51, 62] and social commitments [104, 87, 7, 37, 20, 32] independently. However, in concrete applications such as business settings, agents should reason about their knowledge and their social commitments at the same time, especially when they are engaged in conversations [2]. For example, an agent should know what it is committing about, and should know what the other agent is committing about once this commitment is made public. In fact, knowledge and commitments are not independent, but influence each other. Certainly, they should co-exist and interact in any agent-based system. Therefore, their interaction need to be specified and verified in a systematic way.

In order to motivate our study of modeling, specifying and verifying knowledge and commitments in a single framework, we use the following situational examples that arise in practical setting of web-based applications.

Example 1.1. Consider the fish-market protocol [80] in which different agents (one seller and one or more buyers) are involved in interactions to reach agreement about the price of the offered fish. The protocol starts when the fisherman delivers the fish to the fish market. After that, the seller announces the prices of the available buckets of fish. Thereafter, the buyer(s) either accept the price by uttering *Yes* or reject the price by uttering *No*. No response from the buyer(s) is also considered as a rejection also. Consequently, if only one buyer accepts the price, then the seller will sell him, however, if no one accepts, then the seller lowers the price. On the other hand, if more than one buyer accepts the price, the seller will increase the price and so on. In such scenarios, when the buyer accepts the price

(i.e., commits to pay), the seller should know that. Otherwise, the seller will lower the price even though there is an acceptance. Moreover, the buyer should know that he accepted the price, which means he has the capability to pay in order to fulfill his commitment.

Example 1.2. Let us take the case of buying a book online from a certain publisher as a second example. Suppose that we asked a member from our team to buy a book for us last month. He made the online order and committed to pay. The credit card debit succeeded, meaning that the agent (our team member) knows that he fulfilled his commitment to pay. The publishing company committed to send the requested book to our address. Unfortunately, the book has never arrived. The publisher claimed she had sent it out, but the shipping company she dealt with could not find it in their records. As a result, we asked the publisher to send it again. However, knowing that the book is delivered (i.e., fulfilling the commitment of delivering the book) will help avoid such situations.

Thus, to address the above shortcomings and capture the interaction between knowledge and social commitments in MASs from the semantic and model checking perspectives, we need to define a consistent logic that combines these two concepts together and allows expressing both of them simultaneously.

1.3 Research Problem

In this research, we aim to explore the interaction between knowledge and communicative social commitments in MASs from the semantics, model checking, complexity, soundness and completeness perspectives.

In the domain of multi-agent communications, many frameworks have been introduced to specify social commitments [7, 37, 48, 32, 87] and knowledge [51, 62, 43, 95, 26] independently. However, there is no formal framework and semantics available that allows

us to investigate the interaction between these two modalities in MASs.

To address this challenge, we introduce some research questions and name them: Q1, Q2, ... etc. We start our research questions by asking the following question: **How can we develop a formal approach that can capture both knowledge and communicative social commitments simultaneously?** [Q1]. As we mentioned earlier, there is no research work that addresses the issue of reasoning about social commitments and knowledge in MASs simultaneously. Therefore, we start thinking about how to combine the current available approaches of knowledge and commitments into one new logic so that properties involving both of them simultaneously can be expressed. Therefore, we highlight the second research question: **Which logics of knowledge and commitments should we combine?** [Q2]. In this thesis, we combine the CTLK [76] (an extension of CTL [41] with modality for reasoning about knowledge) and CTLC [7] (an extension of CTL with modalities for reasoning about commitments and their fulfillments) logics into one new logic called CTLKC. There are many reasons that encouraged us to combine these two particular logics. First, the two logics are extensions of the CTL [41] logic, which has an efficient model checking procedure. Second, their models are defined over the interpreted system formalism [43] and its extended version [7], which are very suitable to model MASs and agent communication. Finally, both of them have grounded semantics [99] which means that they can be associated with computational models. Consequently, we will use the combined logic CTLKC as a language that expresses a set of postulates that are used to reason about the interaction between knowledge and social commitments. The third research question of the thesis is the following: **Is it enough to simply combine the current versions of knowledge and commitment logics in order to obtain a consistent logic?** [Q3]. To answer this question, we introduce a set of postulates combining both knowledge and commitments as they are currently defined in the literature. The idea is to show that the violation of those postulates

results in a number of paradoxes in the combined CTLKC logic which shows that simply combining the current versions of knowledge and commitment logics is not a satisfactory solution to this complicated problem. A new multi-modal logic is then to be defined. Thus, the forth research question is: **What are the characteristics of the new logic that will combine knowledge and commitments?** [Q4]. The new logic, CTLKC^{+}, should solve the identified paradoxes in the CTLKC logic and reason about knowledge and social commitment in a consistent manner. The idea is to evaluate it against our fundamental postulates and criteria of a consistent knowledge and commitment logic. Thereafter, the fifth research question is: **How can we prove the soundness and completeness of the proposed logic**? [Q5]. To address this issue, we use Benthem's correspondence theory for modal logics [92]. In particular, we develop a set of reasoning postulates using the CTLKC^{+} logic and correspond them to certain classes of frames. Using correspondence theory, we can prove that the logic generated by any subset of these postulates is sound and complete with respect to the models that are based on the corresponding frames. After proving the soundness and completeness, we need to verify CTLKC^{+}. Therefore, we have the following question: **Which verification approach should we adopt?** [Q6]. In this research, we adopt model checking approach to verify our systems. Model checking is an automated approach that can verify the whole system model against a given specifications and can establish whether such specifications are satisfied in this model or not, at design time. To do so, we propose two transformation (reduction)-based model checking techniques in which the problem of model checking CTLKC^{+} is reduced into the problem of model checking GCTL* [14] (a generalized version of CTL* [42] with action formulas) and ARCTL [75] (an extension of CTL [41] with action formulas). Therefore, we can benefit from the CWB-NC [1] and extended NuSMV [2] model checkers in implementing the proposed techniques. Finally, we

[1] http://sourceforge.net/projects/cwb-nc/

[2] http://lvl.info.ucl.ac.be/Tools/NuSMV-ARCTL-TLACE

need to answer the following research question: **What is the cost of the proposed model checking procedures?** [Q7]. To answer this question, we analyze the complexity of the proposed model checking procedures from the time and space perspectives.

1.4 Research Objectives

The ultimate goal of this research is to develop a formal and comprehensive framework that can investigate the interaction between knowledge and social commitment in MASs from the semantics, soundness and completeness, model checking and complexity perspectives. In particular, we aim to address the following issues:

1. Developing a new consistent logic that can reason about knowledge, communicative social commitments and their interactions in MASs simultaneously.
2. Proving the soundness and completeness of the proposed logic.
3. Developing an efficient model checking technique for verifying the proposed logic.
4. Computing the computational complexity of the proposed model checking algorithm.
5. Implementing the proposed model checking technique on top of a dedicated model checker and report some experimental results.

1.5 Methodology

At the beginning of this work, we analyzed the research work carried out in the domain of agent communication, formal methods using computational logics and reasoning about knowledge and social commitments in MASs. We have noticed that there are many studies that tackled the problem of defining the semantics of knowledge [43, 50, 51, 62, 76, 97] and

social commitments [7, 37, 12, 87, 33] in MASs independently, but none of them tried to capture the relationship between the two concepts. To capture such a relationship, we have selected two well-known logics CTLK [76] (an extension of CTL [41] with modality for reasoning about knowledge) and CTLC [7] (an extension of CTL with modalities for reasoning about commitments and their fulfillments) and combined them into one multi-modal logic, the CTLKC logic, using the *independent join* [45]. After that, we used the CTLKC logic to express some reasoning postulates that combine knowledge and commitments in MASs. After analyzing such postulates, we have identified some paradoxes that should be addressed in any consistent logic combining knowledge and commitments. Thus, we propose CTLKC$^+$, a new logic that fixes the identified paradoxes and allows us to reason about social commitments and knowledge simultaneously, in a consistent manner.

Thereafter, we prove the soundness and completeness of CTLKC$^+$ using Benthem's correspondence theory for modal logic [92]. In particular, we develop a set of reasoning postulates in CTLKC$^+$ and correspond them to certain classes of frames. We illustrate each reasoning postulate using a concrete application example (the NetBill protocol [90]). The existence of such correspondence allows us to prove the soundness and completeness of CTLKC$^+$.

After that, to verify the proposed logic, we first address the problem of model checking CTLKC$^+$ by transforming it into the problem of model checking GCTL* [14], a generalized version of CTL* [42] with action formulas. By doing so, we directly benefit from CWB-NC, the automata-based model checker of GCTL*. Furthermore, to enhance the scalability of the proposed model checking procedure, we develop another fully-automatic and transformation-based model checking procedure. Particularly, we transform the problem of model checking CTLKC$^+$ into the problem of model checking an existing logic of action called ARCTL [75], an extension of CTL [41] with action formulas. Thus, we get benefit

from the extended version of NuSMV symbolic model checker of ARCTL. Following that, we analyze the time and space complexity of the proposed model checking techniques. Finally, we implement our model checking approaches on top of the extended NuSMV and CWB-NC model checkers and report verification results for the verification of the NetBill protocol [90], against some desirable properties. Figure 1.1 depicts the proposed approach.

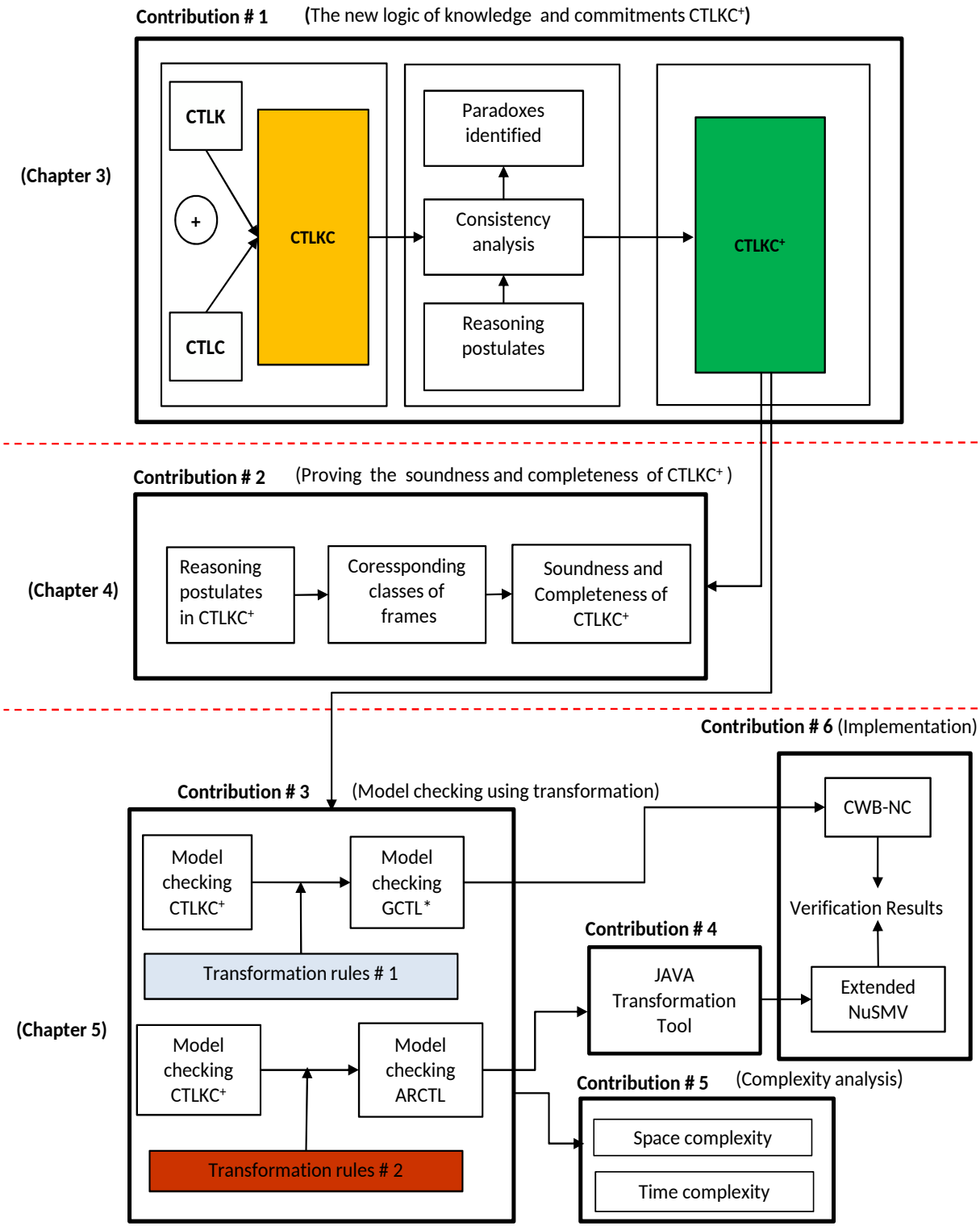

Figure 1.1: The proposed approach

1.6 Research Structure

The overall structure of this research is depicted in Figure 1.2 as follows. Chapter 1 gives a motivational introduction. Chapter 2 describes the background needed for our research. Chapter 3 presents a formal approach for capturing the interaction between knowledge and social commitments in MASs. Chapter 4 illustrates the usage of Benthem's correspondence theory for modal logics to prove the soundness and completeness of the proposed logic of knowledge and commitments (CTLKC$^+$). Chapter 5 presents two reduction-based procedures to verify CTLKC$^+$ with their implementations. Chapter 6 summarizes the obtained results in this thesis and highlights the possible extensions of this work.

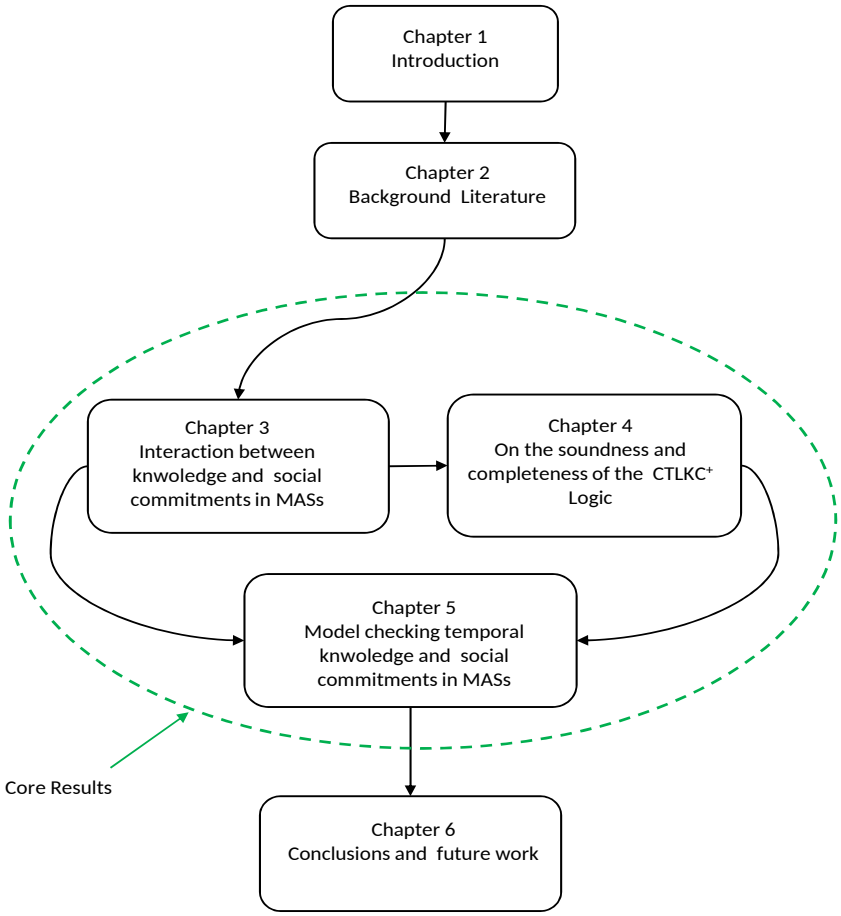

Figure 1.2: Research structure

Chapter 2

Background Literature

In this chapter, we present the necessary preliminaries for this research. In Section 2.1, we review reasoning about knowledge, knowledge axioms, and how to verify knowledge in MASs. In Section 2.2, reasoning about social commitments, commitment types, and commitment applications in MASs is discussed. In Section 2.3, the formalism of interpreted systems is presented as a tool to model MASs. Section 2.4 reviews the CTLK and CTLC logics. Thereafter, in Section 2.5, correspondence theory as a formal approach to prove the soundness and completeness of modal logics is reviewed.. In Section 2.6, model checking as a verification tool is presented. Section 2.7 describes the related work of this research. Finally, Section 2.8 summarizes the chapter.

2.1 Reasoning about Knowledge

Knowledge in MASs has long been modeled and reasoned about [43, 94, 73, 51, 62, 95]. Informally, it is captured through the interpretation that an agent knows something (say a fact) if it is true in all the worlds that the agent thinks possible [52]. Knowledge is formally

denoted by $K_i\varphi$ meaning that "agent i knows φ" where φ is the content of knowledge. Furthermore, the logic of knowledge (i.e., epistemic logic) has been successfully applied in many disciplines (e.g., philosophy and logic [62]). The first formal proposals toward defining epistemic framework were started by Hintikka [54] and later by Lenzen [60]. Their efforts were concentrated on defining some axioms of modal logic to capture some properties of knowledge in a reasonable way. The typical formalism is an $S5_n$ system built on top of the propositional calculus, where n is the number of agents [62]. The traditional axioms of knowledge are [43]:

- K: $(K_i p \wedge K_i(p \rightarrow q)) \rightarrow K_i q$,
- T: $K_i p \rightarrow p$,
- 4: $K_i p \rightarrow K_i K_i p$,
- 5: $\neg K_i p \rightarrow K_i \neg K_i p$,
- D: $\neg K_i(false)$

To capture the semantics of knowledge in epistemic systems, the formalism of Kripke models [57] was initially introduced. Kripke models are of the form $\mathscr{M} = (S, \{R_i\}_{i \in A}, \mathscr{V})$, where S is the set of "possible worlds", $R_i \subseteq S \times S$ is the epistemic accessibility relation between two possible worlds for each agent i in the set of all agents $\mathscr{A}$, and $\mathscr{V} : S \rightarrow 2^p$ is a function interpreting a set of propositions [62]. Later, to model the evolution of a system composed of autonomous agents and efficiently reason about temporal and epistemic properties, the "*Interpreted systems*" formalism was introduced [43]. Details about this formalism will be given later in Section 2.3.

2.2 Reasoning about Social Commitments

In an open MASs, to reach their goals, autonomous and heterogeneous agents interact with each other using Agent Communication Languages (ACLs). Therefore, we need to define an efficient and formal semantics for those ACLs [31]. The two main approaches for defining semantics for ACLs are the mental (cognitive) and social approaches. In fact, the first systematic proposal to define such a semantics was influenced by the Searle's *speech acts theory* [83]. Speech act theory treats communication as actions. This semantics is known as the *mental approach* which tries to capture core communication concepts such as: beliefs, desires, intentions, and goals. However, a major weakness in these approaches is being built based on the assumption that agents can access each other mind [85]. Therefore, mental approaches cannot verify whether an agent is acting according to a given semantics. This problem is commonly known as ACL semantics verification problem [100]. Another drawback associated with this kind of semantics is that "it does not allow ACLs to have enough interoperability among heterogenous agents" [85].

Therefore, MASs switch towards social approaches to overcome the shortcomings of ACLs semantics defined using mental approaches [87]. Social approaches are hired to define a formal semantics for ACLs in which no assumptions on the mental states of agents are made [44, 87, 104]. Social commitments are exploited in some of these social approaches that successfully provide a robust representation to model multi-agent interactions [87, 49]. In such approaches, commitments are considered as contracts between two or more autonomous agents to reach an agreement [10].

In the last decade, social commitments have been used successfully in a wide range of areas (e.g., Web-based applications, business contracts and commitment protocols [89, 30, 5]). Commitment-based social approaches for agent communication have the ability to treat commitments through a set of actions called "commitment actions", such as *creation,*

(discharge) fulfillment, violation, cancellation, release, delegation and assignment [86]. This ability to manipulate commitments is in fact an important characteristic that makes commitment-based approaches flexible and powerful. In this thesis, we use communicative social commitments, introduced in [7], as a means of communication using message passing. Those commitments are formally expressed as $C_{i \to j}\varphi$ which means that the debtor agent i commits toward the creditor agent j that the content of the commitment φ is satisfied [7].

Example 2.1. A customer (*Cus*) commits to send a payment (*Pay*) to a merchant (*Mer*). This commitment is formally denoted as $C_{Cus \to Mer} Pay$.

2.3 Interpreted Systems

The formalism of interpreted systems was introduced by Fagin et al. [43] to model the temporal evolution of a MAS in order to capture epistemic and temporal properties. Interpreted systems formalism has the capability to model different classes of MASs such as synchronous and asynchronous systems [36].

This formalism consists of a set of n agents $\mathscr{A} = \{1, \ldots, n\}$. Each agent $i \in \mathscr{A}$ is described by a set of local states L_i. Each local state of an agent represents a given moment of information for the system. The local state of agent i is denoted by $l_i \in L_i$. A global state $\mathrm{g} \in G$ is a state that consists of the local states of all agents in the system (i.e., $\mathrm{g} = (l_1, \ldots, l_n)$). The set of all global states $G = L_1 \times \cdots \times L_n$ is the Cartesian product of all local states of n agents. The local state of each agent i in the global state g is characterized by the notion $l_i(\mathrm{g})$. The set of initial global states of the system is represented by $I \subseteq G$. In this formalism, the set of local actions of each agent i, which is needed to model the temporal evolution of the system, is denoted by Act_i. $P_i : L \to 2^{Act_i}$ represents the local protocol of agent i which shows the set of available actions that could be carried out at a

given local state. The global transition function of the system can be defined as follows: $\tau : G \times ACT \rightarrow G$, where $ACT = Act_1 \times \cdots \times Act_n$ and each component $a \in ACT$ is a *joint action* (i.e., an action for each agent). τ_i is a local transition function for each agent i that points out the transitions between its local states and it is defined as follows: $\tau_i : L_i \times Act_i \rightarrow L_i$.

Bentahar et al. [7] and El-Menshawy et al. [36] extended the formalism of interpreted systems to capture communication between interacting agents during the execution of MASs. In their extension, they associated each agent with a set of variables (communication channels) such that for the two agents i and j, in order to communicate, they should share a variable (communication channel).

2.4 CTLK and CTLC logics

In this section, we present the models, syntax and semantics of CTLK [76] and CTLC [7] logics, which are the combination of branching time CTL [41] with modalities for reasoning about knowledge and social commitments.

The model ($\mathscr{M}_{\mathscr{K}}$) of CTLK is defined as follows [76]:

Definition 2.1 (Model of CTLK)**.** A model $\mathscr{M}_{\mathscr{K}} = (S, I, R_t, \{\approx_i | i \in \mathscr{A}\}, \mathscr{V})$ which is a member of the set of all models $\mathbb{M}$ is a tuple, where:

- $S \subseteq L_1 \times \cdots \times L_n$ is the set of reachable global states for the system.
- $I \subseteq S$ is a set of initial global states for the system.
- $R_t \subseteq S \times S$ is the transition relation between two global states s and s' in the system.
- $\approx_i \subseteq S \times S$ is the epistemic accessibility relation denoted by $s \approx_i s'$ iff $l_i(s) = l_i(s')$.

- $\mathscr{V} : S \to 2^{\Phi_p}$ is a valuation function, where $\Phi_p = \{p, q, ...\}$ and $p, q, ...$ are atomic propositions.

The epistemic accessibility relation $\approx_i$ is an equivalence relation, which means:

- $\approx_i$ is reflexive: for each $i \in \mathscr{A}$, we have $s \approx_i s$ for all $s \in S$.
- $\approx_i$ is symmetric: for each $i \in \mathscr{A}$, if $s \approx_i s'$ then $s' \approx_i s$ for all $s, s' \in S$.
- $\approx_i$ is transitive: for each $i \in \mathscr{A}$, if $s \approx_i s'$ and $s' \approx_i s''$ then $s \approx_i s''$ for all $s, s', s'' \in S$.

The syntax of CTLK, in BNF format, is defined as follows [76]:

Definition 2.2 (Syntax of CTLK)**.**

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid EX\varphi \mid E(\varphi U \varphi) \mid EG\varphi \mid K_i\varphi$$

Where:

- $p \in \Phi_p$;
- The boolean connectives $\neg$, and $\vee$ are defined in the usual way;
- E is the existential quantifier on paths;
- X, U, and G are CTL "path modal connectives" which stand for "next", "until", and "globally" respectively; and
- The modal connective K_i stands for "knowledge for agent i".

In this logic, $K_i\varphi$ is read as "agent i knows φ". For the other modalities, e.g., F (future), and A (universal path quantifier), they can be defined as usual (see for example [24]).

A path $\pi = (s_0, s_1, \dots)$ in a model $\mathscr{M}_{\mathscr{K}}$ is an infinite sequence of reachable global states in S such that $(s_i, s_{i+1}) \in R_t$ for all $i \geq 0$ [4]. $\Pi(s)$ represents the set of all paths that starts from a given global state s [36].

Definition 2.3 (Satisfaction of CTLK)**.**

The satisfaction of a CTLK formula φ in the model $\mathscr{M}_{\mathscr{K}}$ at the global state s, denoted by $(\mathscr{M}_{\mathscr{K}}, s) \models \varphi$, is defined recursively as follows [76]:

- $(\mathscr{M}_{\mathscr{K}}, s) \models p$ iff $p \in \mathscr{V}(s)$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models \neg\varphi$ iff $(\mathscr{M}_{\mathscr{K}}, s) \not\models \varphi$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models \varphi \vee \psi$ iff $(\mathscr{M}_{\mathscr{K}}, s) \models \varphi$ or $(\mathscr{M}_{\mathscr{K}}, s) \models \psi$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models EX\varphi$ iff $\exists\, \pi \in \Pi(s)$ such that $(\mathscr{M}_{\mathscr{K}}, \pi(1)) \models \varphi$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models E(\varphi U \psi)$ iff $\exists\, \pi \in \Pi(s)$ such that $\exists\, k \geq 0$, $(\mathscr{M}_{\mathscr{K}}, \pi(k)) \models \psi$ and $(\mathscr{M}_{\mathscr{K}}, \pi(j)) \models \varphi$ for all $0 \leq j < k$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models EG\varphi$ iff $\exists\, \pi \in \Pi(s)$ such that $\forall\, k \geq 0$ $(\mathscr{M}_{\mathscr{K}}, \pi(k)) \models \varphi$;
- $(\mathscr{M}_{\mathscr{K}}, s) \models K_i\varphi$ iff for $\forall\, s' \in S$ such that $s \approx_i s'$, $(\mathscr{M}_{\mathscr{K}}, s') \models \varphi$.

Since CTLK extends CTL [41] with modality for reasoning about knowledge, CTLK has the same semantics of CTL in addition to the modal operator of knowledge. In this logic, the formula $K_i\varphi$ holds in the model $\mathscr{M}_{\mathscr{K}}$ at state s iff φ is satisfied in all accessible state s' obtained by $\approx_i$ [76].

The model ($\mathscr{M}_{\mathscr{C}}$) of CTLC is defined as follows [7]:

Definition 2.4 (Model of CTLC)**.** A model $\mathscr{M}_{\mathscr{C}} = (S, I, R_t, \{\sim_{i\to j} | (i,j) \in \mathscr{A}^2\}, \mathscr{V})$ which is a member of the set of all models $\mathbb{M}$ is a tuple, where:

- S, I, R_t and $\mathscr{V}$ are the same as in Definition 2.1.
- $\sim_{i\to j} \subseteq S \times S$ is the social accessibility relation denoted by $s \sim_{i\to j} s'$ iff
  1) $l_i(s) = l_i(s')$,

2) $l_i^x(s) = l_j^x(s')$ if $Var_i \cap Var_j \neq \emptyset \; \forall x \in Var_i \cap Var_j$,

3)$l_j^y(s) = l_j^y(s') \; \forall y \in Var_j - Var_i$.

The intuition behind $\sim_{i\to j}$ from global state s to global state s' is that there is a shared variable x, which represents a communication channel, between agent i and agent j such that agent i sends the information (message) in s, and in s' agent j receives the information. After receiving the information, all the shared variables between i and j will have the same values [7, 36]. Figure 2.1 from [7] illustrates $\sim_{i\to j}$ using shared and unshared variables. In this figure, agent i has the set of variables $Var_i = \{x_1, x_2\}$ and agent j has the set $Var_j = \{x_1, x_3\}$ where x_1 is the shared variable between i and j and x_2 and x_3 are the unshared variables. After receiving the message, the value of the shared variable x_1 for agent j and agent i in state s' is the same. However, the values of the unshared variables x_2 and x_3 in states s and s' are not changed.

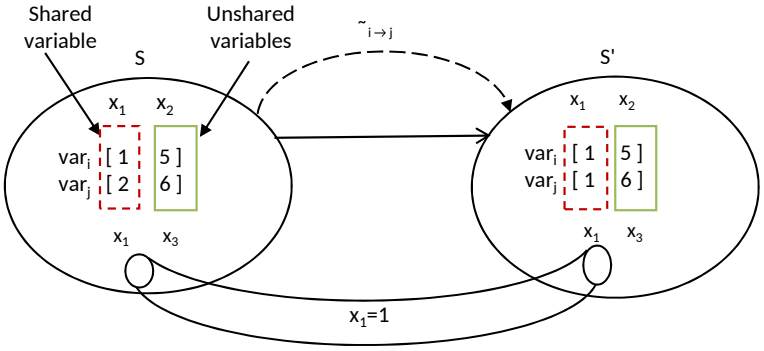

Figure 2.1: An example of social accessibility relation $\sim_{i\to j}$ [7].

The social accessibility relation $\sim_{i\to j}$ is serial, transitive and Euclidean [7, 36].

The syntax of CTLC, which extends CTL [41] with two modalities for commitment and fulfillment, is defined as follows [7]:

Definition 2.5 (Syntax of CTLC)**.**

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid EX\varphi \mid E(\varphi U \varphi) \mid EG\varphi \mid C_{i\to j}\varphi \mid Fu(C_{i\to j}\varphi).$$

Where:

- $p, E, X, G, U, \neg$ and $\vee$ are defined as in Definition 2.2;
- The modal connective $C_{i\to j}$ stands for "commitment from i to j"; and
- The modal connective $Fu(C_{i\to j})$ stands for "fulfillment of commitment".

In this logic, $C_{i\to j}\varphi$ is read as "agent i commits towards agent j to bring about φ". $Fu(C_{i\to j}\varphi)$ is read as "the commitment $C_{i\to j}\varphi$ is fulfilled".
As the main fragment of both CTLK and CTLC logics is the CTL logic, hereafter, we only recall the semantics of the commitment and fulfillment modalities [7].

- $(\mathscr{M}_{\mathscr{C}}, s) \models C_{i\to j}\varphi$ iff $\forall\, s' \in S$ such that $s \sim_{i\to j} s'$ $(\mathscr{M}_{\mathscr{C}}, s') \models \varphi$;
- $(\mathscr{M}_{\mathscr{C}}, s) \models Fu(C_{i\to j}\varphi)$ iff $\exists\, s' \in S$ such that $s' \sim_{i\to j} s$ and $(\mathscr{M}_{\mathscr{C}}, s') \models C_{i\to j}\varphi$.

In this semantics, the formula $C_{i\to j}\varphi$ is satisfied in the model $\mathscr{M}_{\mathscr{C}}$ at global state s iff the content of the commitment φ is satisfied in all global states s' accessible from s using $\sim_{i\to j}$. The formula $Fu(C_{i\to j}\varphi)$ is satisfied in the model $\mathscr{M}_{\mathscr{C}}$ at s if the commitment is satisfied at any state s' where s is accessible from s' using $\sim_{i\to j}$.

It is worth noticing that both logics CTLK and CTLC have grounded semantics, which means they can be associated to computational models [99].

2.5 Correspondence Theory

Correspondence theory for modal logic was introduced by van Benthem [92]. It exhibits a formal analysis of the relationship (correspondence) between classes of frames and modal languages [55]. One of the most key aspects of possible worlds (*Kripke*) semantics and modal logic is highlighted by the fact that modal axioms are kind of representations for the properties of the accessibility relations [15]. For example, a typical "modal completeness theorem" read as follows:

"*A formula is provable in S*4 *iff it is true in all models based on frames whose accessibility relation is transitive and reflexive*" [15]. This means that, the theorems of S4 (i.e., the axioms K, T and 4 are traditionally been called S4 [43]) must hold in all models with a transitive and reflexive relations [15]. To analyze the relationship between classes of frames and modal languages, we have to answer the following *semantics questions: "what can modal formulas say about the frames, and how do they say it ?"*[15].

To answer these questions, let us first define frame, model, frame validity, frame property, frame definability (correspondence), and then show how to use frame definability in capturing the correspondence between a given modal formula and a class of frames.

Definition 2.6 (Frame)**.** A tuple $(W, R_1, ..., R_n)$ with W is a nonempty set of states (worlds) and for each i $(1 <= i <= n)$, R_i is a binary (accessibility) relation on W is called a frame.

Definition 2.7 (Model)**.** Given a frame $\mathscr{F} = (W, R_1, ..., R_n)$, we say the model $\mathscr{M}$ is based on the frame $\mathscr{F} = (W, R_1, ..., R_n)$ if $\mathscr{M} = (W, R_1, ..., R_n, \mathscr{V})$ for some valuation function $\mathscr{V}$, where $\mathscr{V}$ is defined as follows: $\mathscr{V} : W \times \Phi_p \to \{T, F\}$ over the set of atomic propositions Φ_p.

Definition 2.8 (Frame Validity)**.** Given a frame $\mathscr{F} = (W, R_1, ..., R_n)$, we say that a modal formula φ is valid on $\mathscr{F}$, denoted by $\mathscr{F} \models \varphi$, if $\mathscr{M} \models \varphi$ for all models $\mathscr{M}$ based on $\mathscr{F}$. A modal formula φ is valid on a class of frames $\mathbb{F}$ if it is valid on each frame $\mathscr{F}$ in $\mathbb{F}$ [15].

Remark 2.1. "*Note that if* $\mathscr{F} \models \varphi$ *where* φ *is some modal formula, then* $\mathscr{F} \models \varphi^*$ *where* φ^* *is any substitution instance of* φ*. That is,* φ^* *is obtained by replacing sentence letters in* φ *with modal formulas. In particular, this means, for example, in order to show that* $\mathscr{F} \not\models \Box\varphi \to \varphi$*, it is enough to show that* $\mathscr{F} \not\models \Box p \to p$ *where p is a sentence letter*" [74].

Definition 2.9 (Frame Property)**.** Suppose that P_r is a property of an accessability relation (e.g., symmetry or seriality). We say a frame $\mathscr{F} = (W, R_1, ..., R_n)$ has property P_r w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i has property P_r.

Thus, we introduce the following frames:

- $\mathscr{F} = (W, R_1, ..., R_n)$ is called a serial frame w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i is serial, i.e., for all $w \in W$ there exists $v \in W$, such that wR_iv.

- $\mathscr{F} = (W, R_1, ..., R_n)$ is called a reflexive frame w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i is reflexive, i.e., for all $w \in W$, wR_iw.

- $\mathscr{F} = (W, R_1, ..., R_n)$ is called a transitive frame w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i is transitive, i.e., for all $w, x, v \in W$, if wR_ix and xR_iv then wR_iv.

- $\mathscr{F} = (W, R_1, ..., R_n)$ is called an Euclidean frame w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i is Euclidean, i.e., for all $w, x, v \in W$, if wR_iv and wR_ix then vR_ix.

- $\mathscr{F} = (W, R_1, ..., R_n)$ is called a symmetric frame w.r.t. a particular R_i $(1 <= i <= n)$ provided R_i is symmetric, i.e., for all $w, v \in W$, if wR_iv then vR_iw.

We also introduce a new particular frame

- $\mathscr{F} = (W, R_1, R_2)$ is called epistemic social frame (ES) provided R_1, R_2 are the epistemic and social accessibility relations such that for all $w, v, x \in W$ if wR_1v and vR_2x then wR_2x.

Definition 2.10 (Frame Correspondence)**.** "*A modal formula φ defines a class of frames $\mathbb{F}$ iff it is valid on precisely the frames in $\mathbb{F}$. That is, not only must φ be valid on every frame in $\mathbb{F}$, it must also be possible to falsify φ on any frame that is not in $\mathbb{F}$* "[15].

Hereafter are some examples, from [74], of what classes of frames can a modal language define.

EXAMPLE 1. $\Box\varphi \rightarrow \varphi$ corresponds to the class of reflexive frames.

Proof. ($\Leftarrow$) Assume that $\mathscr{F} = (W,R)$ is reflexive and let $\mathscr{M} = (W,R,\mathscr{V})$ be any model based on $\mathscr{F}$. Given $w \in W$, we need to prove $(\mathscr{M},w) \models \Box\varphi \rightarrow \varphi$. Suppose that $(\mathscr{M},w) \models \Box\varphi$. Then $\forall\ v \in W$, if wRv then $(\mathscr{M},v) \models \varphi$. Since R is reflexive, we have wRw. Thus, $(\mathscr{M},w) \models \varphi$. Consequently, $(\mathscr{M},w) \models \Box\varphi \rightarrow \varphi$.

($\Rightarrow$) We argue by contraposition. Assume that $\mathscr{F}$ is not reflexive. We need to prove that $\mathscr{F} \nvDash \Box\varphi \rightarrow \varphi$. Using Remark 2.1, it is enough to show $\mathscr{F} \nvDash \Box p \rightarrow p$ for any sentence letter p. As $\mathscr{F}$ is not reflexive, $\exists\ w \in W$ s.t. it is not the case that wRw. Consider the model $\mathscr{M} = (W,R,\mathscr{V})$ based on $\mathscr{F}$ with $\mathscr{V}(v,p) = T\ \forall\ v \in W$ s.t. $v \neq w$. Then $(\mathscr{M},w) \models \Box p$ since, by assumption, $\forall\ v \in W$ if wRv, then $v \neq w$ and so $\mathscr{V}(v,p) = T$. Further, notice that by the definition of $\mathscr{V}$, $(\mathscr{M},w) \nvDash p$. Consequently, $(\mathscr{M},w) \models \Box p \wedge \neg p$, and so, $\mathscr{F} \nvDash \Box p \rightarrow p$. $\Box$

EXAMPLE 2. $\Box\varphi \rightarrow \Box\Box\varphi$ corresponds to the class of transitive frames.

Proof. ($\Leftarrow$) Assume that $\mathscr{F} = (W,R)$ is transitive and let $\mathscr{M} = (W,R,\mathscr{V})$ be a model based on $\mathscr{F}$. Given $w \in W$, we need to prove that $(\mathscr{M},w) \models \Box\varphi \rightarrow \Box\Box\varphi$. Let us assume that $(\mathscr{M},w) \models \Box\varphi$. We have to show $(\mathscr{M},w) \models \Box\Box\varphi$. Suppose that $v \in W$ and wRv. We need to prove that $(\mathscr{M},v) \models \Box\varphi$. Let $x \in W$ be any state with vRx. Since R is transitive and wRv and vRx, then wRx. Since $(\mathscr{M},w) \models \Box\varphi$, we have $(\mathscr{M},x) \models \varphi$. Therefore, since x is an arbitrary state accessible from v, $(\mathscr{M},v) \models \Box\varphi$, Hence $(\mathscr{M},w) \models \Box\Box\varphi$. Consequently, $(\mathscr{M},w) \models \Box\varphi \rightarrow \Box\Box\varphi$.

($\Rightarrow$) We argue by contraposition. Assume that $\mathscr{F}$ is not transitive. We need to prove that $\mathscr{F} \nvDash \Box\varphi \rightarrow \Box\Box\varphi$. Based on Remark 2.1, it is enough to show $\mathscr{F} \nvDash \Box p \rightarrow \Box\Box p$ for any sentence letter p. Since $\mathscr{F}$ is not transitive, there are states $w,v,x \in W$ with wRv and vRx but it is not the case that wRx. Consider the model $\mathscr{M} = (W,R,\mathscr{V})$ based on $\mathscr{F}$ with $\mathscr{V}(y,p) = T\ \forall\ y \in W$ s.t. $y \neq x$. Since $(\mathscr{M},x) \nvDash p$ and wRv and vRx, we have $(\mathscr{M},w) \nvDash \Box\Box p$. Furthermore, $(\mathscr{M},w) \models \Box p$ since the only state where p is false is x and it is assumed that it is not the case that wRx. Thus, $(\mathscr{M},w) \models \Box p \wedge \neg\Box\Box p$, and so, $\mathscr{F} \nvDash \Box p \rightarrow \Box\Box p$. $\Box$

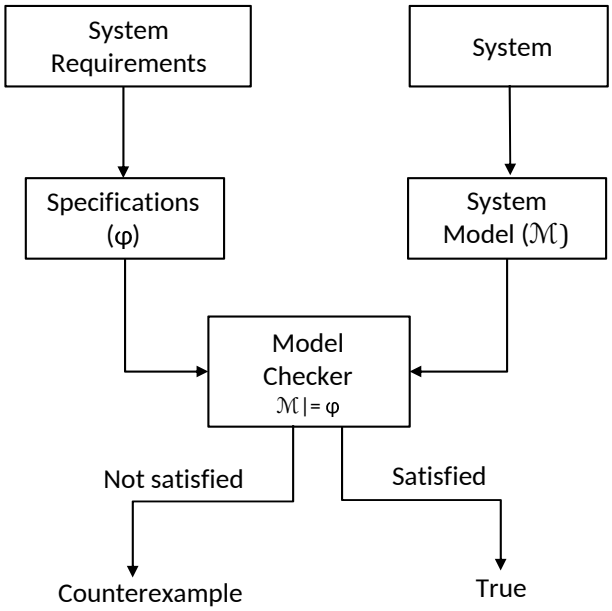

Figure 2.2: A typical model checker

2.6 Model Checking

Model checking is a formal verification approach that automatically verifies whether a given system model satisfies system specification or not [24]. Technically, the problem of model checking can be summarized as follows: given a model $\mathfrak{M}$ that represents a real system and a formula φ that represents a system specification (property) in certain temporal logic, does this model satisfy that specification or not? If the model does not satisfy the specification (i.e., $\mathfrak{M} \nvDash \varphi$), then the process returns a "counterexample" showing the steps where the error was encountered. Otherwise, it will return "true" which means the model satisfies the specification (i.e., $\mathfrak{M} \vDash \varphi$). Figure 2.2 depicts a typical model checking procedure.

Model checking algorithms can be implemented explicitly using Kripke structures. This type of algorithms is normally known as automata-based algorithms (e.g., [27, 47]). The other type of algorithms is developed implicitly using Boolean functions [17] and it is known as symbolic-based algorithms [17, 69, 33].

Generally speaking, model checking approaches suffer from the "state explosion problem" [25]. Symbolic approaches are considered efficient verification approaches since they "alleviate" the problem [63, 78]. In particular, their verification algorithms are implemented using Boolean functions (Bfs), which are represented by Ordered Binary Decision Diagrams OBDDs [16], rather than explicit Kripke structures. Therefore, symbolic-based algorithms consume less memory than automata-based algorithms [25]. In this research, to verify our proposed logic, we develop both algorithms and compare their results.

2.7 Related Work

In this thesis, we classify the work related to this research into three different categories. First, research work directed towards modeling and verifying knowledge only. Second, research work directed towards modeling and verifying commitments only. Third, research work directed towards both knowledge and commitments.

2.7.1 Knowledge in MASs

Penczek and Lomuscio [76] introduced Computation Tree Logic of Knowledge (CTLK) which extends CTL [41] with modalities to reason about knowledge and time. In this work, the authors adopt interpreted systems as modeling formalism. In particular, knowledge modalities are interpreted using epistemic accessibility relations. After that, Lomuscio et al. [61] addressed the problem of model checking CTLK by transforming it into the problem of model checking ARCTL. Concretely, they transformed the CTLK model into ARCTL model and CTLK formulas into ARCTL formulas. Later, they developed an OBDD-based symbolic model checker MCMAS [63] and used it to verify agents' epistemic properties in

MASs. Hereafter, they computed the complexity of model checking CTLK [64]. Nevertheless, the authors did not analyze the interaction between CTLK and social commitments. Furthermore, they did not address the soundness and completeness of CTLK.

Raimondi and Lomuscio [79] developed an OBDD-based procedure to verify deontic interpreted systems [65]. After that, Wozna et al. [102], introduced a bounded model checking algorithm for verifying deontic interpreted systems. In particular, the authors presented a new logical language, CTLKD, which extends CTL with modalities to reason about temporal evolution of epistemic state and their correct and incorrect behavior. In this work, local states are classified into correct and incorrect states. Furthermore, a modality $O_i\varphi$ for representing the correct states of every agent is also developed. Finally, Lomuscio et al. [66] extended this approach to include explicit knowledge. However, no interaction with commitments has been investigated. Moreover, the soundness and completeness of the proposed approaches are not considered.

Meyden et al. [93] put forward a language that captures knowledge and time in systems with perfect recall. Using the proposed language which is based on linear time logic, systems remember all their past states. In this work, interpreted systems were adopted as modeling tools. Moreover, the authors used the MCK model checker [46] to verify their proposed logic. Nevertheless, they did not address the interaction between their language and commitments.

Wan et al. [97] developed a new logic to reason about probabilistic knowledge called Probabilistic Computation Tree Logic of Knowledge (PCTLK). In this work, MASs are modeled using probabilistic interpreted systems. After that, the problem of model checking PCTLK is tackled by transforming it into the problem of model checking Probabilistic Computation Tree Logic (PCTL) [53]. Hereafter, the authors implemented their approach

on top of the PRISM model checker [59]. However, the interaction of PCTLK with commitments is not addressed. Furthermore, the soundness and completeness of PCTLK is not proven.

2.7.2 Social Commitments in MASs

Bentahar et al. [11] introduced a logic to reason about social commitments and argumentations. In particular, they developed a modal logic called DCTL$^*_{CAN}$ built over CTL* [24] and dynamic logic. The main objective of this work is to show how commitments are changed during conversations. After that, the authors developed several reasoning postulates to specify commitments and argumentations. However, they did not prove the soundness and completeness of DCTL$^*_{CAN}$. Moreover, they did not study the interaction between their proposed logic and knowledge. Later, in [9], Bentahar et al. proposed a model checking approach to verify communicative MASs. In this work, agents are linked with knowledge and beliefs and interact with each other according to a set of logical rules. The authors put forward a tableau-based algorithm for model checking their logic and proved the soundness and completeness of their verification procedure, but not of the logic itself. Moreover, they did not investigate the interaction between commitments and knowledge from the semantics perspective.

Singh [88] proposed a temporal logic to reason about practical and dialectical conditional commitments. In this proposal, the author developed a set of reasoning postulates and set of semantics constraints. Thereafter, Singh used Benthem's correspondence theory for modal logic to prove the soundness and completeness of the proposed logic. However, instead of corresponding the reasoning postulates to classes of frames, Singh corresponded the reasoning postulates to a set of semantics constraints. Moreover, no interaction between the proposed logic and knowledge is provided.

In [7], Bentahar et al. developed a new temporal logic called CTLC, an extension of CTL with modalities to reason about commitments and their fulfillment. In this work, the authors extended the formalism of interpreted systems by associating a set of variables for each agent. By so doing, the authors modeled the communication between the interacting agents. Following that, they developed a symbolic model checking algorithm to verify CTLC. Thereafter, they computed the complexity of CTLC model checking procedure. Later, El-Menshawy et al. [36] refined the semantics of CTLC and introduced CTLC^{+}, a temporal logic of commitments and their fulfillment. To verify this logic, the problem of model checking CTLC^{+} is transformed into the problem of model checking ARCTL and GCTL* in order to benefit from the extended NuSMV and CWB-NC model checkers respectively. However, they neither investigated the soundness and completeness of their logics, nor analyzed the interaction between their logics and knowledge.

Chesani et al. [20] introduced a commitment logic that allows the domain modeler to determine the roles of the debtor and creditor in handling social commitments. Furthermore, the authors proposed a set of axioms for commitment operations. However, they did not pursue the soundness and completeness of the proposed logic. Moreover, they did not address the interaction between the proposed logic and knowledge.

El Kholy et al. [32] put forward a new temporal logic, CTL^{cc}, which extends CTL [41] with modalities to specify conditional commitments and their fulfillments. In this work, the authors adopt the formalism of interpreted systems as a modeling tool. Furthermore, they developed a set of rules to reason about conditional commitments and their fulfillments. After that, they addressed the model checking problem of CTL^{cc} by developing a symbolic model checking algorithm and analyzed the complexity of the proposed algorithm. Hereafter, they implemented the developed algorithm on top of MCMAS. Nevertheless, they did not investigate the soundness and completeness of CTL^{cc}. Further, they

did not study the interaction between the proposed logic and knowledge.

2.7.3 Interaction between Knowledge and Commitments

Schmidt et al. [81] introduced a new temporal logic called Agent Dynamic Logic (ADL) to study the interaction between knowledge, actions and commitments. In this work, the authors integrated a different type of commitments called "internal commitments". An internal commitment is referred by Castelfranchi [18] as " a relation between an agent and action" not between two or more agents as in the case with social commitments. Moreover, the authors introduced some axioms that exhibit agents' knowledge about its commitments. However, they proved that their proposed logic is sound and complete with respect to a Kripke-style semantics rather than using correspondence theory. Moreover, internal commitment suffered from the semantics verification problem similar to the mental approaches.

Woźna [101] proposed CDCTL*K, a new temporal logic that extends CTL* [24] with modalities to reason about knowledge, correct functioning behaviour, and social commitments in MASs. In this work, Communication Deontic Interpreted Systems (CDIS) was adopted to define the semantics of the proposed logic. Furthermore, a Sat-based bounded model checking algorithm was developed to verify the CDCTL*K logic. However, neither the interaction between knowledge and commitment in MASs nor the soundness and completeness of the proposed logic are addressed. In addition to this, the author did not investigate the complexity analysis of the proposed logic.

2.8 Summary

In this chapter, we presented the preliminaries needed to go throughout the thesis. Moreover, we reviewed the research work related to our work highlighting the differences between them. In the next chapter, we will introduce a new formal and consistent approach to capture the interaction between knowledge and social commitments in MASs.

Chapter 3

Interaction between Knowledge and Social Commitments in MASs

In this chapter[1], we present a comprehensive and systematic technique to formally reason about knowledge, communicative social commitments and their interactions in MASs. In particular, we develop a new consistent temporal logic called the logic of knowledge and commitments (CTLKC$^+$). CTLKC$^+$ extends CTL [41] with modalities to reason about knowledge, commitments and their fulfillments. This logic captures the interaction between knowledge and communicative social commitments from the semantics perspective. To model MASs using CTLKC$^+$, we develop a new version of interpreted systems based on a new social accessibility relation [2].

3.1 Introduction

In this chapter, we aim to figure out the relationship between knowledge and social commitments from the semantics perspective. To do so, we combined two established logics,

[1]The results of this chapter are published in [2].

CTLK [76] (an extension of CTL logic [41] with a modality for reasoning about knowledge) and CTLC [7] (an extension of CTL logic with modalities for reasoning about commitments and their fulfillments) in one logic that we call CTLKC. This combination is designed so that it keeps the same semantics of the knowledge and commitment modalities as they are in the original logics. This combination allows us to express formulas merging the two modalities and check if some basic and principle intuitions are satisfied in the resulting logic. Such intuitions about the interaction between knowledge and commitments are represented as a set of reasoning postulates yielding some paradoxes. The purpose of identifying such paradoxes is to motivate the need for a new way of making commitments and knowledge interact in a consistent logic. Thus, we propose a new logic called CTLKC$^+$ that fixes the identified paradoxes and allows us to reason about knowledge and commitments in a consistent way.

The remainder of this chapter is organized as follows. In Section 3.2, we introduce the model, syntax and semantics of the CTLKC logic. In Section 3.3, we present a set of reasoning postulates written in CTLKC logic to capture the relationship between knowledge and social commitments. We use the NetBill payment protocol [90] as a running example to illustrate our reasoning postulates. In Section 3.4, we present some conclusive remarks about CTLKC. In Section 3.5, we introduce the new consistent logic CTLKC$^+$ that overcomes all the problems raised in CTLKC. Finally, we conclude the chapter in Section 3.6.

3.2 CTLKC Logic

In this section, we develop a unified model for both CTLK [76] and CTLC [7] defined over the formalism of the extended version of interpreted systems [7]. This model will be used later on to define the semantics of the CTLKC logic. Before that, we present how to construct a multi-modal logic using independent join [45].

3.2.1 Combining Logics

There are many ways to construct a multi-modal logic by combining logics (see for example [6, 45]). To construct our multi-modal logic, the CTLKC logic, we use the *independent join* "fusion" [45] for combining the two logics CTLK and CTLC. Typically, the independent join of two logics is denoted by $A_1 \otimes A_2$. Given two logics A_1 and A_2, their languages $\mathscr{L}_1$ and $\mathscr{L}_2$, and their axiomatic systems $\mathscr{X}_1$ and $\mathscr{X}_2$, the logic $A_1 \otimes A_2$ is the "smallest" logic with the following properties [78]:

- The language $\mathscr{L}_{A_1 \otimes A_2} = \mathscr{L}_{A_1} \cup \mathscr{L}_{A_2}$.
- The logic $A_1 \otimes A_2$ has the set of axioms in both logics.

If the logics A_1 and A_2 are interpreted using Kripke frames $F_{r_1} = (S, R_1^1, \ldots, R_n^1)$ and $F_{r_2} = (S, R_1^2, \ldots, R_m^2)$ where $R^1_{x/x\in[1,n]}$ and $R^2_{y/y\in[1,m]}$ are accessibility relations, then the semantics for the combined logic $A_1 \otimes A_2$ can be defined in the Kripke frame $F = (S, R_1^1, \ldots, R_n^1, R_1^2, \ldots, R_m^2)$ which results from applying the independent join on the frames F_{r_1} and F_{r_2}.

The reason behind selecting the independent join to construct our combined logic is the fact that it "keeps the same semantics and axioms" of the combined logics [45]. Consequently, we can analyze the relationship between knowledge and commitments exactly as they are advocated in the two independent logics CTLK and CTLC. Precisely, in this chapter, we introduce a set of reasoning postulates to figure out the relationship between knowledge and social commitment in MASs from the semantics perspective. The new logic can express such postulates since both modalities can be expressed. *By so doing, we answer the first and second research questions (i.e., [Q1] and [Q2]).*

3.2.2 CTLKC Model

Definition 3.1 (CTLKC Model)**.** A model $\mathscr{M} = (S, I, R_t, \{\approx_i | i \in \mathscr{A}\},$

$\{\sim_{i\to j} | (i,j) \in \mathscr{A}^2\}, \mathscr{V})$ which is a member of the set of all models $\mathbb{M}$ is a tuple, where:

- $S, I, R_t, \mathscr{V}, \approx_i$ and $\sim_{i\to j}$ are the same as in Definitions 2.1 and 2.4.

In this model, we assume that each accessible state is reachable. The syntax of CTLKC , in BNF format, is defined as follows:

Definition 3.2 (Syntax of CTLKC)**.**

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid EX\varphi \mid E(\varphi U \varphi) \mid EG\varphi \mid K_i\varphi \mid C_{i\to j}\varphi \mid Fu(C_{i\to j}\varphi).$$

CTLKC is at least as expressive as CTLK and CTLC. CTLK, CTLC and CTLKC can all express CTL properties since they are all built over the CTL logic. However, CTLK logic can express the CTL properties in addition to knowledge modality (i.e., $K_i p$). On the other hand, CTLC can express commitment modality and its fulfillment (i.e., $C_{i\to j}p$ and $Fu(C_{i\to j}p)$) which cannot be expressed in CTLK. By introducing the CTLKC combined logic, we can express properties on both knowledge and social commitment (e.g., $K_i(C_{i\to j}p)$).

3.3 Knowledge and Commitments Analysis

3.3.1 Running Example

To capture the relationship between knowledge and social commitments in MASs from the semantics perspective, we identify some reasoning postulates and study their validity in the combined CTLKC logic. Before that, we present the NetBill payment protocol [90], from the business domain, as a running example to illustrate the reasoning postulates.

The NetBill payment protocol (see Figure 3.1 from [38]) is developed for buying and selling encrypted software goods on the Internet [28, 90]. This protocol consists of eight steps [90]. First, the customer requests a quote from the merchant. The merchant sends the quote to the customer (i.e., Present Quote) . If the customer accepts the quote, then the merchant sends the software (deliver the goods) encrypted and keeps the key. The customer creates an electronic payment order (EPO) including a description for the received goods and send it to the merchant. The merchant verifies the EPO and sends it to the NetBill server. The NetBill server checks the customer bank account and deposits the payment on the merchant account. Then, a receipt including the key to decrypt the goods is sent to the merchant first and then to the customer. Finally, the customer decrypts the purchased software (goods).

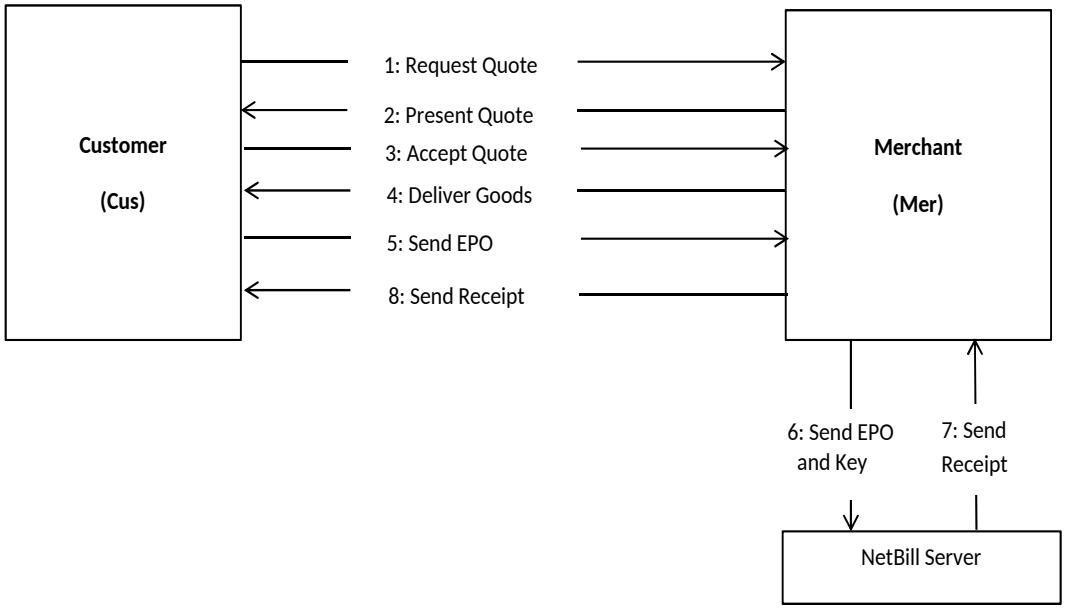

Figure 3.1: The NetBill protocol [38]

In the rest of this thesis, we consider the following commitments to represent the interactions between the customer (Cus) and the merchant (Mer) agents in the NetBill protocol:

1. $C_{Cus \rightarrow Mer}$ *sendPayment*, which means that the customer (Cus) commits toward the merchant (Mer) to send the agreed amount of payment $(sendPayment)$ (*sendPayment* is an atomic proposition that represents the content of the commitment).

2. $C_{Mer \rightarrow Cus}$ $deliverGoods$, which means that the merchant (Mer) commits toward the customer (Cus) to deliver the required goods $(deliverGoods)$ ($deliverGoods$ is an atomic proposition that represents the content of the commitment).

3. $C_{Mer \rightarrow Cus}$ $sendReciept$, which means that the merchant (Mer) commits toward the customer (Cus) to send the receipt $(sendReciept)$ ($sendReciept$ is an atomic proposition that represents the content of the commitment).

3.3.2 Selection Criteria

To identify reasoning postulates that capture the interaction between knowledge and commitments in MASs simultaneously, there should be clear criteria for selecting such postulates. Since this kind of reasoning has not been investigated yet (i.e., reasoning about knowledge and social commitments in MASs in the same logic), we propose the following criteria based on common intuitions:

- It should not be the case that an agent commits about everything it knows to others.
- It should not be the case that an agent should commit about everything known by others.
- An agent should know about its own commitment.
- An agent should know the content of its fulfilled commitment.
- An agent should know the content of any fulfilled commitment directed towards it.

- An agent should know about the fulfillment of its commitment.
- An agent should know about the fulfillment of any commitment directed towards it.
- A sincere agent should know (i.e., believe) the content of its own commitment.
- A sincere agent should fulfill its commitment.
- A liar agent could not know (i.e., believe) the content of its commitment.
- If an agent knows about its own commitment, then this commitment should exist, which means an agent should not know about a non-existing commitment.

3.3.3 Desiderata of Paradoxes

In this section, we identify some paradoxes that result from combining CTLK and CTLC into the new CTLKC logic. Those paradoxes are represented as postulates that are 1) not reasonable in open MAS, but valid in CTLKC; or 2) desirable for the interaction between knowledge and commitments, but can get violated in CTLKC because they are not valid. For each paradox, a validity proof or a counter-example is provided. In the rest of this chapter, all the postulates are $AG\varphi$ formulas, which means they are considered globally (i.e., in all states) in all paths. However, to simplify the notation, AG is omitted. For instance, instead of $AG(K_i\varphi \rightarrow C_{i\rightarrow j}\varphi)$, we simply write $K_i\varphi \rightarrow C_{i\rightarrow j}\varphi$.

P1. **[Committing everything known to others]**

$K_i\varphi \rightarrow C_{i\rightarrow j}\varphi$, for all $j \in A$ where $i \neq j$.

Meaning: An agent (the debtor) commits toward all other agents (the creditors) about what it knows.

Proof. Assume that $(\mathscr{M},s) \models K_i\varphi$. From the semantics of $K_i\varphi$, for all states $s' \in S$ such that $s \approx_i s'$, we have $(\mathscr{M},s') \models \varphi$. Now, suppose there is a state $s' \in S$ such that $s \sim_{i\to j} s'$ and $(\mathscr{M},s') \models \neg\varphi$. With respect to the semantics of a commitment modality, the state s will not be labeled with the commitment (i.e., $(\mathscr{M},s) \models \neg C_{i\to j}\varphi$). Furthermore, from condition 1 of the definition of the social accessibility relation $s \sim_{i\to j} s'$ (i.e., $l_i(s) = l_i(s')$), we conclude that the state s' is epistemically accessible from s (i.e., $s \approx_i s'$) and since the current state (i.e., s) holds the knowledge modality (i.e., $(\mathscr{M},s) \models K_i\varphi$), we have $(\mathscr{M},s') \models \varphi$, which is contradiction. Thus, $(\mathscr{M},s) \models C_{i\to j}\varphi$, so we are done. □

Discussion: The idea of this postulate is to study the effect of agent's knowledge on its commitments. It simply says whenever an agent knows something, it should commit its knowledge to all other agents in the system. In terms of agent communication, this would mean an agent tells other agents whatever it knows. The validity of this postulate is based on the fact that by establishing communication channels, the epistemic relation is also established. The postulate is not reasonable in open MASs where agents are selfish. The following is an example illustrating the paradox.

Example 3.1. From the NetBill payment protocol, assume that the merchant knows the goods that it will deliver. By applying the postulate, $K_{Mer}\ deliverGoods \to C_{Mer\to Cus}\ deliverGoods$, the merchant will commit towards all the customers to deliver them the goods, which they might not have requested, but it happens that they— for example— established communication channels with that merchant and rejected the price quote for some reasons.

P2. [**Committing everything known by others**]

$K_iK_j\varphi \to C_{i\to j}\varphi$ where $i \neq j$.

Meaning: An agent (the debtor) commits toward another agent (the creditor) to bring about what it (i.e., the debtor) knows about the creditor's knowledge.

Proof. Assume $(\mathscr{M},s) \models K_iK_j\varphi$. From the semantics of $K_iK_j\varphi$, for all states $s' \in S$ such that $s \approx_i s'$, we have $(\mathscr{M},s') \models K_j\varphi$. Suppose that there exists a global state $s' \in S$ such that $s \sim_{i\to j} s'$ and $(\mathscr{M},s') \models \neg\varphi$. According to the semantics of commitment modality, the current state (i.e., s) will not be labeled with the commitment (i.e., $(\mathscr{M},s) \models \neg C_{i\to j}\varphi$). Since $\approx_j$ is reflexive, we conclude that $(\mathscr{M},s') \models \neg K_j\varphi$. On the other hand, since an accessible state via $\sim_{i\to j}$ is also accessible via $\approx_i$, it follows that $(\mathscr{M},s') \models K_j\varphi$, so the contradiction. Consequently, $(\mathscr{M},s) \models C_{i\to j}\varphi$. □

Discussion: In this postulate, the purpose is to study the effect of an agent's knowledge about another agent's knowledge on the commitment of the first agent towards the second. The formula says that an agent should commit towards another agent about what the first agent knows about the knowledge of the second. In other words, whenever an agent knows that the other agent knows something, it will commit to bring about this information towards that agent. In terms of agent communication, this formula would mean that an agent should tell the other agent what it knows the other knows. The following example shows that this postulate is not reasonable in MASs.

Example 3.2. Let us assume that the customer knows that the merchant will deliver the goods. Applying this postulate, $K_{Cus}K_{Mer}\,deliverGoods \to C_{Cus\to Mer}\,deliverGoods$, the customer will commit towards the merchant to deliver the goods which contradicts what actually has to be done (i.e., the merchant is responsible of delivering the goods to the customer).

P3. [**Committing everything known from others**]

$K_iK_j\varphi \rightarrow C_{j\rightarrow i}\varphi$ where $i \neq j$.

Meaning: An agent (the debtor) commits toward another agent (the creditor) to bring about what the creditor knows about the debtor's knowledge.

Proof. Assume $(\mathscr{M},s) \models K_iK_j\varphi$. From the semantics of $K_i\varphi$, for all states $s' \in S$ such that $s \approx_i s'$, we have $(\mathscr{M},s') \models K_j\varphi$. Because $\approx_i$ is reflexive, then $(\mathscr{M},s) \models K_j\varphi$. Suppose there exists a global state $s' \in S$ such that $s \sim_{j\rightarrow i} s'$ and $(\mathscr{M},s') \models \neg\varphi$. According to the semantics of commitment modality, the current state (i.e., s) will not be labeled with the commitment (i.e., $(\mathscr{M},s) \models \neg C_{i\rightarrow j}\varphi$). On the other hand, since an accessible state via $\sim_{j\rightarrow i}$ is also accessible via $\approx_j$, it follows that $(\mathscr{M},s') \models \varphi$, so the contradiction. Consequently, $(\mathscr{M},s) \models C_{j\rightarrow i}\varphi$. □

Example 3.3. To clarify the paradox, let us consider the following example. Assume that the customer knows that the merchant will deliver the goods. By applying this postulate, $K_{Cus}K_{Mer}\ deliverGoods \rightarrow C_{Mer\rightarrow Cus}\ deliverGoods$, the merchant will commit towards the customer to deliver the goods. Thus, only a customer's knowledge influences and even obliges the merchant's commitment, which is counter-intuitive. In fact, the customer could be uncertain about its knowledge, even though the commitment should be established. This postulate can result in serious circumstances if the customer agent is malicious, so it can express wrong knowledge about the merchant, obliging that merchant to establish unwanted commitment, such as knowing that the delivery is free of charge.

P4. [**Knowing about its own commitment**]

$C_{i\rightarrow j}\varphi \rightarrow K_i(C_{i\rightarrow j}\varphi)$ where $i \neq j$.

Meaning: An agent knows about its commitment.

This postulate is a reasonable one as agents should be aware of their own and intentional commitments, but it results in a paradox as the formula is not valid. The following model depicted in Figure 3.2 shows a counterexample.

Model 1

In this model, assume that the global state s_0 is labeled by the commitment (i.e., $(\mathscr{M}, s_0) \models (C_{i\to j}\varphi)$) and there exists a global state s_1 accessible from s_0 using social accessibility relation (i.e., $s_0 \sim_{i\to j} s_1$). From the semantics of social commitments, s_1 is labeled by φ (i.e., $(\mathscr{M}, s_1) \models \varphi$) and since the commitments modality is shift reflexive, then s_1 will be labeled by the commitment (i.e., $(\mathscr{M}, s_1) \models (C_{i\to j}\varphi)$). Moreover, suppose there exists a global state s_2 accessible from s_0 using epistemic accessibility relation (i.e., $s_0 \approx_i s_2$). Let us assume that there exists two global states s_3 and s_4 accessible from s_2 using social accessibility relation (i.e., $s_2 \sim_{i\to j} s_3$ and $s_2 \sim_{i\to j} s_4$). Suppose that s_3 is labeled by φ (i.e., $(\mathscr{M}, s_3) \models \varphi$) and s_4 is labeled by $\neg\varphi$ (i.e., $(\mathscr{M}, s_4) \models \neg\varphi$). From the semantics of social commitments, s_2 will not be labeled by the commitments (i.e.,$(\mathscr{M}, s_2) \models \neg(C_{i\to j}\varphi)$). Thus, from the semantics of knowledge, s_0 will not be labeled by the knowledge of commitment (i.e., $(\mathscr{M}, s_0) \models \neg K_i(C_{i\to j}\varphi)$).

Discussion: The reason why this postulate is not valid is due to the fact that a state accessible through the epistemic accessibility relation $\approx_i$ is not necessarily accessible though the social accessibility $\sim_{i\to j}$. Consequently, we can find a state accessible through $\approx_i$ that does not satisfy the commitment, although this case cannot happen in a state accessible via $\sim_{i\to j}$. Let us consider the following example.

Example 3.4. Assuming the customer commits toward the merchant to send the agreed amount of payment. Applying this postulate, $C_{Cus\to Mer}\ sendPayment \to K_{Cus}$

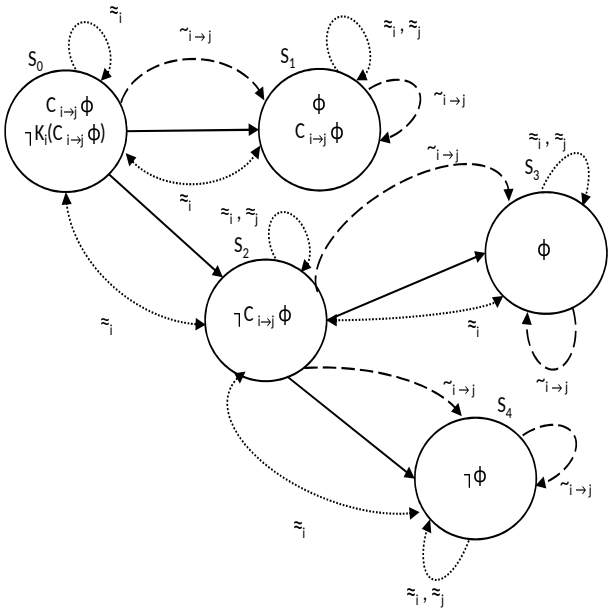

Figure 3.2: Model 1

($C_{Cus \to Mer}$ *sendPayment*), it is obvious that the customer should know about this commitment and about the amount to be sent. Otherwise, this would mean that the commitment is made accidently, without the agent being conscious.

P5. [**Knowing the content of its own fulfilled commitment**]

$Fu(C_{i \to j}\varphi) \to K_i\varphi$ where $i \neq j$.

Meaning: An agent knows the content of its fulfilled commitment.

This postulate is a reasonable one, but as for the previous postulate, it results in another paradox as it is not valid. The following model depicted in Figure 3.3 shows a counterexample.

Model 2

In this model, assume that $(\mathscr{M}, s) \models Fu(C_{i \to j}\varphi)$. From the semantics of fulfillment, there exists a global state $s' \in S$ such that $s' \sim_{i \to j} s$ and $(\mathscr{M}, s') \models C_{i \to j}\varphi$. Suppose

$(\mathscr{M}, s') \models \neg\varphi$. From the first condition of the definition of the social accessibility relation $s' \sim_{i\to j} s$ (i.e., $l_i(s') = l_i(s)$), we conclude that the state s' is epistemically accessible from s (i.e, $s \approx_i s'$). Since the global state s' is not labeled by φ (i.e., $(\mathscr{M}, s') \models \neg\varphi$), then, $(\mathscr{M}, s) \models \neg K_i\varphi$. So, the result.

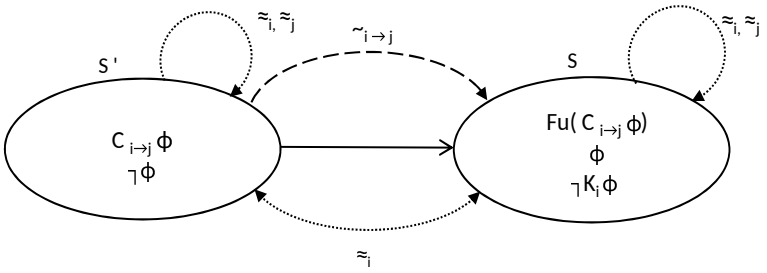

Figure 3.3: Model 2

Discussion: This postulate intuitively reflects the fact that an agent knows the fulfillment of its commitment. The main reason behind its non-validity is the non-reflexiveness of the social accessibility relation. This allows the commitment state not to satisfy the commitment content. Thus, the result follows from the fact that this commitment state is epistemically accessible from the fulfillment state. It is extremely important for this postulate to be valid. The following example clarifies the case.

Example 3.5. Assume that the customer sends the agreed amount of payment to the merchant (i.e., fulfills its commitment of sending the amount). Applying the postulate, $Fu(C_{Cus\to Mer} sendPayment) \to K_{Cus}\ sendPayment$, and because the formula is not valid, it could happen that the agent does not know (or may be forgot) about the sending action. Hence, the agent can send the amount again and again.

P6. [**Knowing the content of the other's fulfilled commitment**]

$Fu(C_{i\to j}\varphi) \to K_j\varphi$ where $i \neq j$.

Meaning: The creditor knows the content of the debtor's fulfilled commitment.

This postulate should be satisfied in all the models (i.e., valid), as the creditor should be aware of the satisfaction of the commitment directed to it once this fulfillment happens. Otherwise, the creditor can ask the debtor to satisfy the commitment again. The following model depicted in Figure 3.4 shows a counterexample.

Model 3

In this model, assume that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi)$. From the semantics of fulfillment, there exists a global state $s_0 \in S$ such that $s_0 \sim_{i\to j} s_1$ and $(\mathscr{M}, s_0) \models C_{i\to j}\varphi$. Assume the existence of another global state s_2 accessible from s_1 via $\approx_j$ and holding $\neg\varphi$ (i.e., $(\mathscr{M}, s_2) \models \neg\varphi$). From the semantics of the knowledge operator, it follows that $\neg K_j\varphi$ holds in S_1, so we are done.

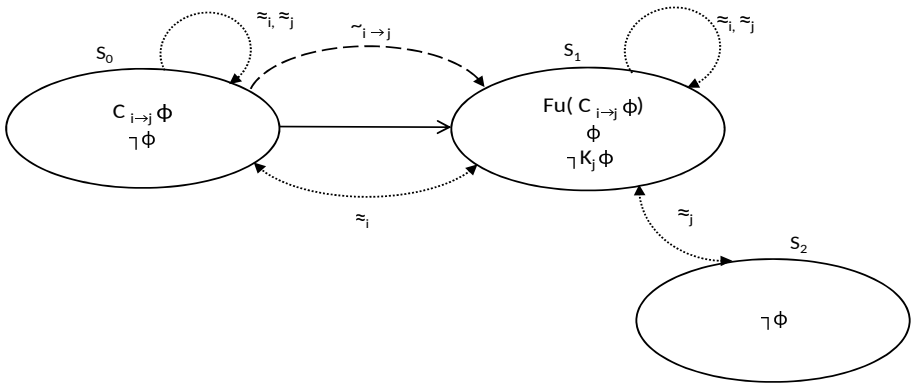

Figure 3.4: Model 3

Discussion: The reason why this postulate is not valid, so it can be violated is because the social accessibility relation $\sim_{i\to j}$ is independent from the epistemic relation for the creditor $\approx_j$. Thus, it is always possible to connect an undistinguishable state for the creditor that does not satisfy the commitment content to the fulfillment state. The

following example shows the consequences of the underlying paradox.

Example 3.6. Assume that the customer sends the agreed amount of payment to the merchant (i.e., fulfills its commitment). If the postulate is valid, we should have: $Fu(C_{cust \to Mer}\ sendPayment) \to K_{Mer} sendPayment$. Thus, the merchant should know that the payment has been sent, because fulfilling the commitment means a proof of payment has been issued. Since the formula is not valid, we could simply imagine a case where the merchant claims that no payment has been sent, so it will require that a new payment should be issued, and the situation can be repeated infinitely often.

P7. [**Knowing the fulfillment of its own commitment**]

$Fu(C_{i \to j}\varphi) \to K_i Fu(C_{i \to j}\varphi)$ where $i \neq j$.

Meaning: The agent knows that it fulfills its commitment.

This postulate is similar to P4, except that P4 is about knowing the commitment not its fulfillment. However, both of them should be valid as the agent should be aware of both its commitment and fulfillment actions.

Model 4

It is easy to find a model violating this formula. We only need to link a state s_2 to a state s_1 using $\approx_i$ (i.e., $s_1 \approx_i s_2$) such that $s_1 \models Fu(C_{i \to j}\varphi)$ and $s_2 \models \neg Fu(C_{i \to j}\varphi)$. Consequently, $s_2 \models \neg K_i Fu(C_{i \to j}\varphi)$. State s_2 should not be accessible from s_1 through the social accessibility relation $\sim_{i \to j}$. This allows s_2 not to be labeled by $Fu(C_{i \to j}\varphi)$. The reason is because if s_2 is also accessible from s_1 using $\sim_{i \to j}$, and knowing that s_1 is socially accessible from the commitment state s_0, then s_2 is also socially accessible from s_0 since $\sim_{i \to j}$ is transitive. In that case, we will have $s_2 \models Fu(C_{i \to j}\varphi)$.

Discussion: As shown in Model 4, the satisfiability of the negation of P7 is guaranteed since a state accessible through the epistemic accessibility relation $\approx_i$ is not

necessarily accessible through the social accessibility relation from i to j (i.e., $\sim_{i\to j}$). As for P4, this postulate is very significant in open MASs and should be satisfied in all possible models. This would reflect the agent's intentionality of its action to fulfill its own commitment. Having a model violating this postulate (such as Model 4) would mean that the agent is not aware about its own action of fulfilling its own commitment. Thus, the agent could fulfill the commitment many times, maybe infinitely often. The following example illustrates the paradox.

Example 3.7. Let us assume that the customer fulfills its commitment of sending the payment to the merchant. Being aware of the fulfillment action performed by the customer is formally expressed as follows:
$Fu(C_{Cus\to Mer}sendPayment) \to K_{Cus}Fu(C_{Cus\to Mer}sendPayment)$. Since this formula is not valid, we could have a model in which the customer fulfills its commitment but ignores its own fulfillment action. This inconsistency could have devastating consequences if the agent is honest with itself, so it could send the payment many times.

P8. [**Knowing the fulfillment of the debtor's commitment**]

$Fu(C_{i\to j}\varphi) \to K_jFu(C_{i\to j}\varphi)$ where $i \neq j$.

Meaning: The creditor knows that the debtor fulfills its commitment.

This postulate is about knowing the fulfillment of the commitment by the creditor.

Model 5

A similar model to Model 4 can be used with a small modification. We simply need to link a state s_2 to a state s_1 using $\approx_j$ (i.e., $s_1 \approx_j s_2$) such that $s_1 \models Fu(C_{i\to j}\varphi)$ and $s_2 \models \neg Fu(C_{i\to j}\varphi)$. Thus, $s_2 \models \neg K_jFu(C_{i\to j}\varphi)$. Like in Model 4, state s_2 should not

be accessible from s_1 through the social accessibility relation $\sim_{i\to j}$ in order to have $s_2 \models \neg Fu(C_{i\to j})\varphi$.

Discussion: As illustrated in Model 5, the violation of the postulate is made possible because from a given state s, the set of states that are accessible via $\approx_j$ and the set of states that are accessible via $\sim_{i\to j}$ could be disjoint. The validity of the postulate is of a great importance in open MASs. This would reflect that, fulfilling the commitment is public. The following example illustrates the underlying paradox.

Example 3.8. Let us assume that the customer fulfills its commitment of sending the payment to the merchant. This fulfillment should be public so that the merchant can observe it, which means:
$Fu(C_{Cus\to Mer}sendPayment) \to K_{Mer}Fu(C_{Cus\to Mer}sendPayment)$. Violating this postulate would be source of problematic scenarios as the merchant can require the payment to be resent, probably many times until it achieves a state in which the merchant can get the information that the payment has been sent.

P9. [**Commitment-knowledge modus ponens**]

$(C_{i\to j}\varphi \wedge K_i(\varphi \to \psi)) \to C_{i\to j}\psi$ where $i \neq j$.

Meaning: The debtor commits to the conclusion of an implication it knows if it commits to the premise.

Proof. Let s be a state such that $s \models (C_{i\to j}\varphi \wedge K_i(\varphi \to \psi))$. Since any state socially accessible from s is also epistemically accessible from the same state, all the states socially accessible from s will satisfy φ (using the semantics of $C_{i\to j}\varphi$) and will also satisfy $\varphi \to \psi$ (using the semantics of $K_i(\varphi \to \psi)$). Using modus ponens, any accessible state from s via $\sim_{i\to j}$ will satisfy ψ. Thus, the result follows from the semantics of $C_{i\to j}\psi$. □

Discussion: We consider this postulate as a paradox as it over-commits the debtor to the consequences of all the implications it might know. The over-commitment becomes more apparent if the creditor ignores the implications. Instead of having the postulate as a valid formula in all the possible models, it is more desirable to be a satisfiable formula that could be forced in some specific applications where agents should be sincere.

3.4 Conclusive Remarks about CTLKC

In the previous section, we identified nine paradoxes related to the interaction between knowledge and social commitments. Four of which (i.e., P1, P2, P3, and P9) are valid postulates in CTLKC but represent undesirable and unreasonable properties. The remaining five paradoxes (i.e., P4, P5, P6, P7, and P8) are non-valid formulas in CTLKC that should be valid in real settings. The former group are paired with their validity proofs, while the latter ones are followed by counterexamples showing the satisfiability of their negations.

The main reason of having the first fifth paradoxes (P1 to P5) and the paradoxes P7 and P9 in CTLKC is that the set of states that are accessible from a given state s using the social accessibility relation $\sim_{i\to j}$ is, in general, included in or equal to the set of states that are accessible from the same state s using the epistemic accessibility relation $\approx_i$. Thus, each state accessible from s using $\sim_{i\to j}$ is also accessible from the same state using $\approx_i$. However, it is possible to find a state accessible from s through $\approx_i$ without being among the states accessible from the same state using $\sim_{i\to j}$. This means that in this case, the set of states that are accessible from a given state s using the social accessibility relation $\sim_{i\to j}$ is strictly included in the set of states that are accessible from the same state s using the epistemic accessibility relation $\approx_i$. Four direct consequences of that are: 1) whenever we have knowledge (Paradox P1) or knowledge of knowledge (Paradoxes P2 and P3), we should

have commitment about that knowledge; 2) it is possible to have commitment, without having knowledge of it (Paradox P4); 3) it is possible to have fulfillment of a commitment without having knowledge of its content (Paradox P5) and of the fulfillment of the commitment (Paradox P7); and 4) whenever we have commitment and knowledge about an implication and its antecedent, we should have commitment about the consequence (Paradox P9). In the next section (Section 3.5), we will propose a solution to those paradoxes by removing this inclusion constraint.

For the remaining paradoxes (P6 and P8), the main reason behind them is that the set of states accessible from a given state s using the social accessibility relation $\sim_{i\to j}$ from the debtor i to the creditor j is independent from the epistemic relation $\approx_j$ of the creditor j. Thus, it is possible to have an accessible state from the fulfillment state using $\approx_j$ that does not satisfy the commitment content without being accessible from the commitment state using $\sim_{i\to j}$. Consequently, we will have fulfillment of a commitment without knowing its content by the creditor (Paradox P6). Finally, it is possible to have an accessible state from the fulfillment state using $\approx_j$ without being accessible from the commitment state using $\sim_{i\to j}$. Consequently, we could have fulfillment without the knowledge of the creditor (Paradox P8). In the next section, we will solve those two paradoxes by linking the semantics of commitment and its fulfillment to the creditor.

What is particularly interesting to observe is that the two logics, namely CTLK for knowledge and CTLC for commitments, are consistent and working properly if taken individually. This has been proven through many applications and case studies that have been successfully modeled using these two logics (see for instance [7, 37, 43, 50, 51, 62]). The paradoxes only arise when the two logics are merged together. *Having such paradoxes in the combined logic (CTLKC) answers the third research question [Q3].*

3.5 CTLKC$^+$ Logic

To fix the identified paradoxes, we introduce CTLKC$^+$, the new logic of knowledge and commitments. In this section, we first present the model of CTLKC$^+$ and introduce the syntax and semantics of this logic. Then, we trace each paradox and show its solution.

3.5.1 Model of CTLKC$^+$

Definition 3.3 (Model of CTLKC$^+$)**.** A model $\mathfrak{M} = (S, I, R_t, \{\approx_i | i \in \mathscr{A}\}, \{\approx_{i\to j} | (i,j) \in \mathscr{A}^2\}, \mathscr{V})$ which is a member of the set of all models $\mathbb{M}$ is a tuple, where:

- $S, I, R_t, \approx_i$ and $\mathscr{V}$ are the same as in Definition 3.1.
- For each pair $(i,j) \in \mathscr{A}^2$, $\approx_{i\to j} \subseteq S \times S$ is the social accessibility relation defined by $s \approx_{i\to j} s'$ iff $Var_i \cap Var_j \neq \emptyset$ such that $\forall x \in Var_i \cap Var_j$, we have $l_i^x(s) = l_i^x(s') = l_j^x(s')$.

The intuition behind the new social accessibility relation $\approx_{i\to j}$ from one global state s to another global state s' ($s \approx_{i\to j} s'$) is that there are some shared variables (i.e., communication channels) between the interacting agents i and j such that agent i sends the message (information) through the channel in s, and agent j receives the message in s'. After receiving the message, all the shared variables between i and j will have the same values (i.e., $l_i^x(s) = l_i^x(s') = l_j^x(s')\ \forall x \in Var_i \cap Var_j$). Unlike the previous social accessibility relation $\sim_{i\to j}$, there is no restriction on the content of the unshared variables for both agents, as they can both receive, at the same time, information from other agents through other channels involving other variables for each one of them. Those variables can be thus different from s to s'. This idea is illustrated in Figure 3.5. where two agents i and j have some shared variables to communicate as follows: Agent i has the set of variables $Var_i = \{x_1, x_2\}$ and Agent j has the set of variables $Var_j = \{x_1, x_2'\}$. The variable x_1 is the shared variable where the

variables x_2 and x_2' are the unshared variables between the two agents. When the message is sent, the value of x_1 for agent j in s is changed to be equal the value of variable x_1 for agent i in s'.

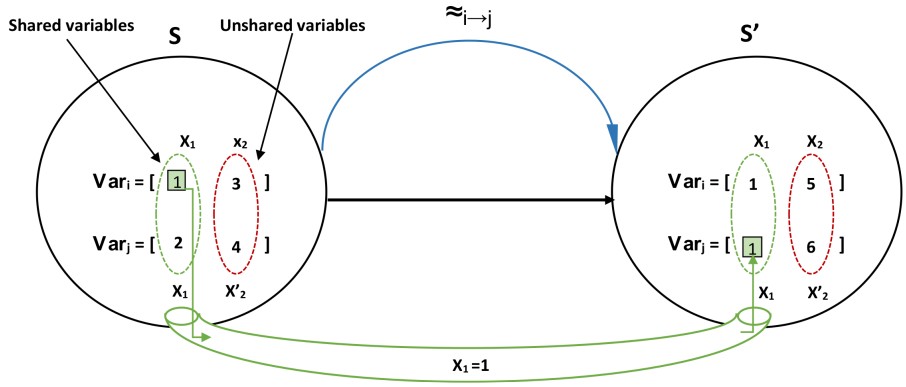

Figure 3.5: An example of the new social accessibility relation $\approx_{i\to j}$

Thus, the main motivation behind this new definition of $\approx_{i\to j}$ is that the previous definition (i.e., $\sim_{i\to j}$) over specifies and over constrains the concept of illocutionary communication in the sense that the accessible state for the debtor i should be the same. We argue that only the shared variables should be the same as not only the creditor j gains more information when moving from one state to an accessible one, but the debtor as well could change its local state by probably gaining new information through other communications. Furthermore, by relaxing this condition of having the same values for unshared variables, we allow the agent to establish and participate in multiple communications while being in the same state. The social accessibility relation $\approx_{i\to j}$ has the following properties:

Proposition 3.1. *$\approx_{i\to j}$ is transitive and Euclidean.*

- *$\approx_{i\to j}$ is transitive: for any pair $i, j \in \mathscr{A}$, if $s \approx_{i\to j} s'$ and $s' \approx_{i\to j} s''$ then $s \approx_{i\to j} s''$ for all $s, s', s'' \in S$.*

  Proof. Assume $s \approx_{i\to j} s'$ and $s' \approx_{i\to j} s''$, for any pair $i, j \in \mathscr{A}$. According to the definition of $\approx_{i\to j}$, it is the case that $s \approx_{i\to j} s''$ as $\forall x \in Var_i \cap Var_j$, we have $l_i^x(s) =$

$l_i^x(s') = l_i^x(s'') = l_j^x(s'')$. □

- $\approx_{i\to j}$ *is Euclidean: for any pair* $i, j \in \mathscr{A}$*, if* $s \approx_{i\to j} s'$ *and* $s \approx_{i\to j} s''$ *then* $s' \approx_{i\to j} s''$ *for all* $s, s', s'' \in S$.

  Proof. Assume $s \approx_{i\to j} s'$ and $s \approx_{i\to j} s''$, for any pair $i, j \in \mathscr{A}$. According to the definition of $\approx_{i\to j}$, we have $s' \approx_{i\to j} s''$ as $l_i^x(s') = l_i^x(s) = l_i^x(s'') = l_j^x(s'')\ \forall x \in Var_i \cap Var_j$. □

From the properties of the social and epistemic accessibility relations the following lemmas hold.

Lemma 3.1. *If* $s_1 \approx_i s_2$ *and* $s_2 \approx_{i\to j} s_3$ *then* $s_1 \approx_{i\to j} s_3$.

Proof. Assume $s_1 \approx_i s_2$ and $s_2 \approx_{i\to j} s_3$ for any pair $i, j \in \mathscr{A}$. According to the definition of $\approx_i$, it is the case that $l_i(s_1) = l_i(s_2)$. Therefore, $l_i^x(s_1) = l_i^x(s_2)\ \forall x \in Var_i \cap Var_j$. From the definition of $\approx_{i\to j}$, it is the case that $l_i^x(s_2) = l_i^x(s_3) = l_j^x(s_3)\ \forall x \in Var_i \cap Var_j$. Consequently, $s_1 \approx_{i\to j} s_3$. □

Lemma 3.2. *If* $s_1 \approx_i s_2$ *and* $s_1 \approx_{i\to j} s_3$ *then* $s_2 \approx_{i\to j} s_3$.

Proof. From $s_1 \approx_i s_2$ and $s_1 \approx_{i\to j} s_3$ for any pair $i, j \in A$, we get $l_i^x(s_1) = l_i^x(s_2)$ and $l_i^x(s_1) = l_i^x(s_3) = l_j^x(s_3)\ \forall x \in Var_i \cap Var_j$. Consequently, $s_2 \approx_{i\to j} s_3$. □

The following theorem is direct from Lemmas 3.1 and 3.2:

Theorem 3.1. *If* $s_1 \approx_i s_2$ *then* $s_1 \approx_{i\to j} s_3$ *iff* $s_2 \approx_{i\to j} s_3$.

This result is reasonable as for agent i, s_1 and s_2 are indistinguishable, so that socially accessing to s_3 from s_1 or s_2 should be the same.

3.5.2 Semantics of CTLKC$^+$

CTLKC$^+$ has the same syntax as CTLKC. Its semantics is as follows.

Definition 3.4 (Satisfaction of CTLKC$^+$)**.**
Given the model $\mathfrak{M}$, the satisfaction of a CTLKC$^+$ formula φ in a global state s, denoted by $(\mathfrak{M},s) \models \varphi$, is recursively defined as follows:

- $(\mathfrak{M},s) \models p$ iff $p \in \mathscr{V}(s)$;
- $(\mathfrak{M},s) \models \neg\varphi$ iff $(\mathfrak{M},s) \not\models \varphi$;
- $(\mathfrak{M},s) \models \varphi \vee \psi$ iff $(\mathfrak{M},s) \models \varphi$ or $(\mathfrak{M},s) \models \psi$;
- $(\mathfrak{M},s) \models EX\varphi$ iff there exists a path π starting at s such that $(\mathfrak{M},\pi(1)) \models \varphi$;
- $(\mathfrak{M},s) \models E(\varphi U \psi)$ iff there exists a path π starting at s such that for some $k \geq 0$, $(\mathfrak{M},\pi(k)) \models \psi$ and $(\mathfrak{M},\pi(j)) \models \varphi$ for all $0 \leq j < k$;
- $(\mathfrak{M},s) \models EG\varphi$ iff there exists a path π starting at s such that $(\mathfrak{M},\pi(k)) \models \varphi$ for all $k \geq 0$;
- $(\mathfrak{M},s) \models K_i\varphi$ iff for all global states $s' \in S$ such that $s \approx_i s'$, we have $(\mathfrak{M},s') \models \varphi$;
- $(\mathfrak{M},s) \models C_{i\to j}\varphi$ iff for all global states $s' \in S$ such that $s \approx_{i\to j} s'$, we have $(\mathfrak{M},s') \models K_i\varphi$ and $(\mathfrak{M},s') \models K_j\varphi$;
- $(\mathfrak{M},s) \models Fu(C_{i\to j}\varphi)$ iff there exists $s' \in S$ such that $s' \approx_{i\to j} s$ and $(\mathfrak{M},s') \models C_{i\to j}\varphi$ or there exists $s'' \in S$ and $s'' \approx_i s$ such that $(\mathfrak{M},s'') \models Fu(C_{i\to j}\varphi)$ or there exists $s'' \in S$ and $s'' \approx_j s$ such that $(\mathfrak{M},s'') \models Fu(C_{i\to j}\varphi)$.

The semantics of CTLKC$^+$ state formulas is defined in the model $\mathfrak{M}$ as usual (semantics of CTL, see for example [41]) with modalities for reasoning about knowledge and social commitments and their fulfillments respectively. The state formula $K_i\varphi$ is satisfied in the model $\mathfrak{M}$ iff the content φ holds in every accessible state s' obtained by the epistemic accessibility relation $\approx_i$. The state formula $C_{i\to j}\varphi$ is satisfied in the model $\mathfrak{M}$ iff the modalities $K_i\varphi$ and $K_j\varphi$ hold in every accessible state s' obtained by the social accessibility relation $\approx_{i\to j}$. The intuition of this semantics is as follows: when i commits to j using an illocution, the two agents become aware of the content in the accessible states. The state formula $Fu(C_{i\to j}\varphi)$ is satisfied in the model $\mathfrak{M}$ iff, first there exists a state s' satisfying the commitment from which s can be seen using the social accessibility relation $\approx_{i\to j}$, or there exists a state s'' indistinguishable from s either for i or for j that satisfies the fulfillment of the commitment. The intuition behind this semantics is as follows. First, a state s is a fulfillment state if it is socially accessible from the commitment state. Once fulfillment states are determined using this first option, fulfillment will be propagated to all the states that are equivalent for each agent. This is because when a fulfillment is achieved, the situation should be reflected in all equivalent states for the two interacting agents (i.e., debtor and creditor). This idea of fulfillment propagation is shown in the following theorem and corollary.

Theorem 3.2. *Let* $\rho \in \{i, j\}$. *If* $(\mathfrak{M}, s) \models Fu(C_{i\to j}\varphi)$ *then* $\forall s'$ *s.t.* $s' \approx_\rho s$, *we have* $(\mathfrak{M}, s') \models Fu(C_{i\to j}\varphi)$

Proof. We prove the theorem for $\approx_i$, the proof for $\approx_j$ is similar. Assume that $(\mathfrak{M}, s) \models Fu(C_{i\to j}\varphi)$ and $\exists s'$ s.t. $s' \approx_i s$ and $(\mathfrak{M}, s') \models \neg Fu(C_{i\to j}\varphi)$. From the semantics of $Fu(C_{i\to j}\varphi)$, the three or-conditions are not satisfied. Consequently, $\forall s''$ s.t. $s'' \approx_i s'$, we have $(\mathfrak{M}, s'') \models \neg Fu(C_{i\to j}\varphi)$. Since $s \approx_i s'$ as $\approx_i$ is symmetric, we have contradiction with the assumption when $s'' = s$, so we are done. □

Corollary 3.1. *Let* $\rho \in \{i, j\}$. *If* $(\mathfrak{M}, s) \models Fu(C_{i\to j}\varphi)$ *then* $\forall s'$ *s.t.* $s \approx_\rho s'$, *we have* $(\mathfrak{M}, s') \models Fu(C_{i\to j}\varphi)$

Proof. The result is direct from Theorem 3.2 since the accessibility relation $\approx_\rho$ is symmetric. □

3.5.3 Fixing Paradoxes

As discussed in Section 3.4, the first problem in CTLKC is that each state accessible from s using $\sim_{i\to j}$ is also accessible from the same state using $\approx_i$. This problem causes paradoxes P1 to P5, P7 and P9. Using the new model of CTLKC$^+$, each state accessible from s by $\approx_{i\to j}$ is not necessarily accessible from that state using $\approx_i$. The second problem that causes P6 and P8 in CTLKC is that the set of states accessible from a given state s by the social accessibility relation $\sim_{i\to j}$ is independent from the epistemic accessibility relation $\approx_j$ of the creditor j. In CTLKC$^+$, this problem is fixed using the new semantics of commitments and fulfillment. In the rest of this section, each agent is associated with two variables, where the first one is shared.

1. $K_i\varphi \to C_{i\to j}\varphi$, for all $j \in A$ where $i \neq j$ is not valid.

   Model 6

   Figure 3.6 depicts a counterexample for paradox P1. In this model, the global state s_0 is labeled by the knowledge (i.e., $(\mathfrak{M}, s_0) \models K_i\varphi$) because all the accessible states using the epistemic accessibility relation $\approx_i$ satisfy φ. However, s_0 is not labeled by the commitment (i.e., $(\mathfrak{M}, s_0) \models \neg C_{i\to j}\varphi$) since the socially accessible state s_2 does not satisfy $K_j\varphi$ because of the existence of s_4, which is epistemically accessible from s_2 for j (i.e., $s_2 \approx_j s_4$).

2. $K_iK_j\varphi \to C_{i\to j}\varphi$ such that $i \neq j$ is not valid.

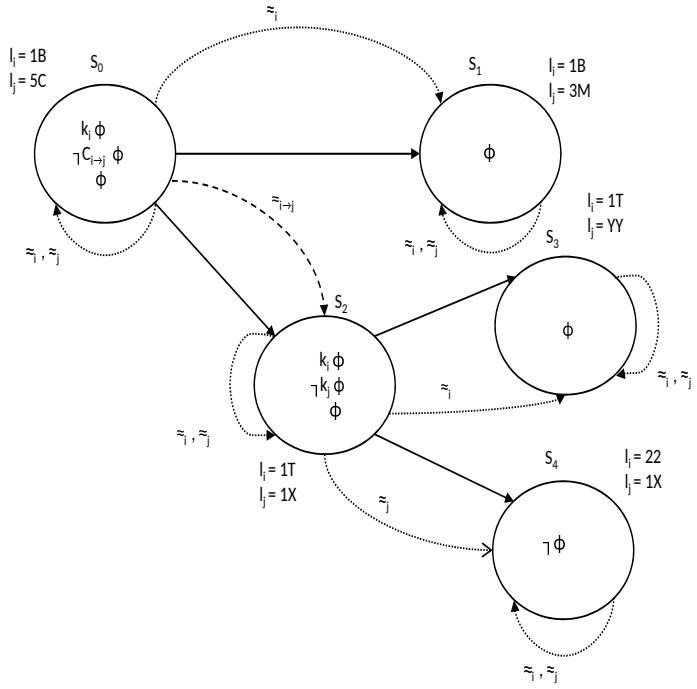

Figure 3.6: Model 6

Model 7

Figure 3.7 depicts a counterexample for paradox P2. In this model, $(\mathfrak{M}, s_0) \models K_i K_j \varphi$ since all the accessible states using $\approx_i$ satisfy $K_j\varphi$. However, $(\mathfrak{M}, s_0) \models \neg C_{i\to j}\varphi$ because of the existence of the socially accessible state s_3 that does not satisfy, for instance, $K_i\varphi$.

3. $K_i K_j \varphi \to C_{j\to i}\varphi$ such that $i \neq j$ is not valid.

Model 8

Figure 3.8 depicts a counterexample for paradox P3. In this model, $(\mathfrak{M}, s_0) \models K_i K_j \varphi$, but $(\mathfrak{M}, s_0) \models \neg C_{i\to j}\varphi$ because it is not the case that $(\mathfrak{M}, s_3) \models K_i\varphi$.

4. $C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$ where $i \neq j$.

Proof. Assume $(\mathfrak{M}, s) \models C_{i\to j}\varphi \wedge \neg K_i(C_{i\to j}\varphi)$. Consequently, from the semantics

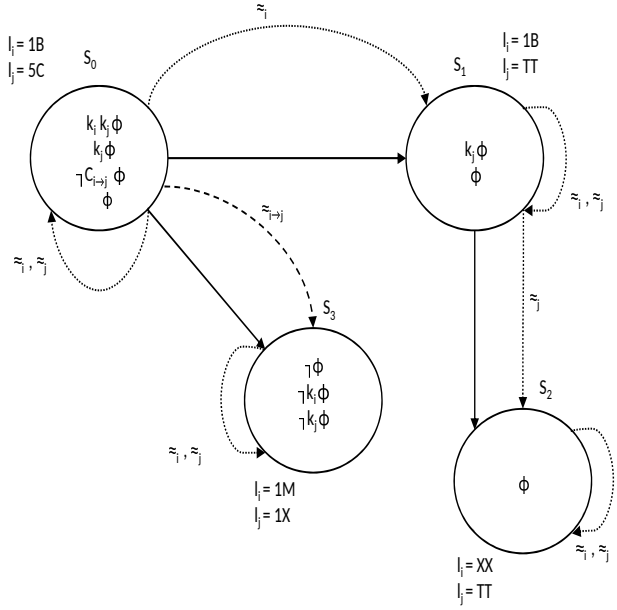

Figure 3.7: Model 7

of knowledge, there exists a state s' accessible from s using $\approx_i$ such that $(\mathfrak{M}, s') \models \neg C_{i\to j}\varphi$. Thus, from the semantics of commitments, there exists a state s'' accessible from s' using $\approx_{i\to j}$ such that $(\mathfrak{M}, s') \models \neg K_i\varphi \vee \neg K_j\varphi$. Using Theorem 3.1, s'' is also accessible from s through $\approx_{i\to j}$, thus the contradiction as $(\mathfrak{M}, s) \models C_{i\to j}\varphi$. □

5. $Fu(C_{i\to j}\varphi) \to K_i\varphi$ where $i \neq j$.

Proof. Assume that $(\mathfrak{M}, s) \models Fu(C_{i\to j}\varphi)$. From the semantics of $Fu(C_{i\to j}\varphi)$, three options are to be considered. Let us consider the first option, the second and third options will follow from Theorem 3.2 and Corollary 3.1. According to the first option, there exists a global state $s' \in S$ such that $s' \approx_{i\to j} s$ and $(\mathfrak{M}, s') \models C_{i\to j}\varphi$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathfrak{M}, s) \models K_i\varphi$, so we are done with the first option.

According to the propagation property established from Theorem 3.2 and Corollary 3.1, $(\mathfrak{M}, s) \models Fu(C_{i\to j}\varphi)$ iff $\forall s' \in S$ such that $s \approx_i s'$, $(\mathfrak{M}, s') \models Fu(C_{i\to j}\varphi)$. Assume

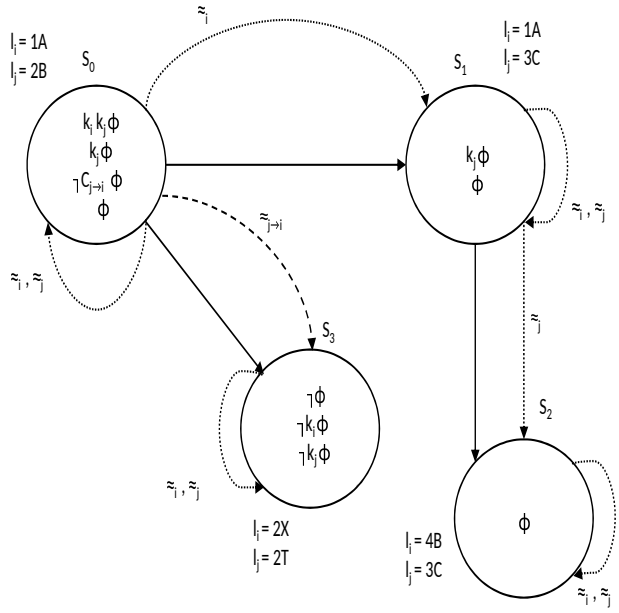

Figure 3.8: Model 8

that for one of those s' states $(\mathfrak{M}, s') \models \neg K_i \varphi$. Since $\approx_i$ is an equivalence relation, it follows that $\forall s' \in S$ such that $s \approx_i s'$, $(\mathfrak{M}, s') \models \neg K_i \varphi$, which is contradictory with the result of the first option; thus the postulate. □

6. $Fu(C_{i \to j} \varphi) \to K_j \varphi$ where $i \neq j$.

   Proof. The proof is similar to the previous one, but with respect to j. □

7. $Fu(C_{i \to j} \varphi) \to K_i Fu(C_{i \to j} \varphi)$ where $i \neq j$.

   Proof. The postulate is direct from Corollary 3.1 and the semantics of $K_i \varphi$. □

8. $Fu(C_{i \to j} \varphi) \to K_j Fu(C_{i \to j} \varphi)$ where $i \neq j$.

   Proof. The postulate is direct from Corollary 3.1 and the semantics of $K_j \varphi$. □

9. $(C_{i \to j} \varphi \wedge K_i(\varphi \to \psi)) \to C_{i \to j} \psi$ is not valid.

Proof. The non-validity follows from the fact that from a given state, states accessible via $\approx_{i\to j}$ are distinct from those accessible via $\approx_i$. Thus, an accessible state via $\approx_{i\to j}$ from a state s satisfying $C_{i\to j}\varphi \wedge K_i(\varphi \to \psi)$ could satisfy $(\varphi \wedge \neg\psi)$. Thus, using the semantics of commitments it follows that $s \models \neg C_{i\to j}\psi$. □

Moreover, in addition to the properties of knowledge (in CTLK) and commitments (in CTLC) such as $K_i\varphi \to \varphi$ and $Fu(C_{i\to j}\varphi) \to \varphi$ that hold in CTLKC$^+$ (the proofs are straightforward), the following properties are also satisfied in this logic:

1. Property 1: $C_{i\to j}\varphi \to \varphi$ is satisfiable but not valid.

   Proof. The non-validity follows from the fact that $\approx_{i\to j}$ is not reflexive. □

   This property says that when an agent commits to bring about a proposition, the proposition does not have to be true. This reflects the limited capability of the agent as it could not be aware about what is true in the current state.

2. Property 2: $C_{i\to j}\varphi \to K_i\varphi$ is satisfiable but not valid.

   Proof. The non-validity follows from the fact that a state can be accessible via $\approx_i$ without being accessible via $\approx_{i\to j}$, such that this state satisfies $\neg\varphi$. □

   This property conveys the fact that committing to bring about a proposition φ should not, in all the cases, imply the knowledge of φ. This means, the agent's sincerity should not be taken as granted. In the models where the formula is satisfiable (globally in all paths), agents will be sincere; otherwise, they are not all the time sincere, which is the general case in open MASs.

3. Property 3: $(C_{i\to j}\varphi \wedge K_j(\varphi \to \psi)) \to C_{i\to j}\psi$ is satisfiable but not valid.

Proof. The non-validity follows from the fact that from a given state, states accessible via $\approx_{i\to j}$ are distinct from those accessible via $\approx_j$. Thus, an accessible state via $\approx_{i\to j}$ from a state s satisfying $C_{i\to j}\varphi \wedge K_j(\varphi \to \psi)$ could satisfy $(\varphi \wedge \neg\psi)$. Therefore, using the semantics of commitments it follows that $s \models \neg C_{i\to j}\psi$. □

4. Property 4: $(C_{i\to j}(\varphi \to \psi) \wedge K_j\varphi) \to C_{i\to j}\psi$ is satisfiable but not valid.

This property is very similar to Property 3 and can be proved in the same way. *By defining the consistent logic (CTLKC$^+$) that can reason about knowledge and commitments simultaneously and fixes the identified paradoxes, we answer the fourth research question [Q4].*

3.6 Summary

To capture the interaction between knowledge and social commitments from the semantics perspective, a new combined temporal logic, called CTLKC, is first introduced in this chapter. CTLKC logic simply combines the logic of knowledge CTLK and the logic of communicative commitments CTLC as presented in the literature. This logic served as a language to express a set of postulates that are used to reason about both knowledge and social commitments. By analyzing such postulates, we identified some paradoxes that should be addressed in any consistent logic combining these two modalities. To overcome and solve the paradoxes identified in CTLKC, we introduced CTLKC$^+$, a new consistent logic for knowledge, communicative commitments and their interactions. We presented a new semantics of commitments and their fulfillment based on a new social accessibility relation. In the next chapter, we will investigate the soundness and completeness of the logic of knowledge and commitments (CTLKC$^+$) using correspondence theory for modal logic [92].

Chapter 4

On the Soundness and Completeness of the CTLKC$^+$ Logic

In this chapter[1], we develop a formal and systematic approach to prove the soundness and completeness of the CTLKC$^+$ logic using correspondence theory for modal logics [92]. To do so, we introduce a set of reasoning postulates in CTLKC$^+$ and correspond them to certain classes of frames providing the required proofs. We illustrate each reasoning postulate using a concrete application example. We adopted the interpreted systems as an underlying formalism over which our developed postulates are interpreted. The existence of such correspondence allows us to prove that the logic generated by any subset of these postulates is sound and complete with respect to the models that are based on the corresponding frames.

4.1 Introduction

Soundness and completeness of a given modal logic can be summarized as follows: Given a set of axioms constructed from a modal logic, is there any relationship between those

[1]The results of this chapter are published in [1].

axioms and certain frames?

To answer this question, in this chapter, we use *Correspondence theory* for modal logic, which was introduced by van Benthem [92], to prove the soundness and completeness of the CTLKC$^+$ logic. Correspondence theory, a subfield of the model theory, reflects a formal analysis of the relationship between classes of frames and modal logics [55].

In [84] Segerberg introduced the early completeness theorems in modal logic as follows: "*modal logic* **L** *is determined by a class* $\mathfrak{R}$ *of Kripke frames*". Those theorems were first applied to the minimum modal logic **K** [92]. In this context, two perspectives emerge here. First, given a class $\mathfrak{R}$ of Kripke frames, we have to find an axiomatisation in a given modal logic **L**. Second, given a modal logic **L**, we have to find a certain class $\mathfrak{R}$ of Kripke frames in which the given logic is complete. Note that, the latter perspective represents the current direction in modal logic.

In this chapter, we use Benthem's correspondence theory for modal logic to prove the soundness and completeness of the logic of knowledge and commitments (CTLKC$^+$). This process of proving the soundness and completeness of CTLKC$^+$ can be seen as a step towards demonstrating and evaluating the efficiency and consistency of CTLKC$^+$ from a new perspective. In particular, as depicted in Figure 4.1, we develop a set of reasoning postulates in CTLKC$^+$ and correspond them to certain classes of frames providing the required proofs. Consequently, we prove that the logic generated by any subset of those postulates is sound and complete with respect to the models that are based on the corresponding frames.

The main strength of the proposed approach lies in the fact that all the results are obtained through the use of solid formal theories. Moreover, the developed axioms are associated with their corresponding proofs based on models defined over the frames. This shows that the proposed logical model is correct and any application built on it would not suffer from any inconsistency. However, the fact that the model is being built using a

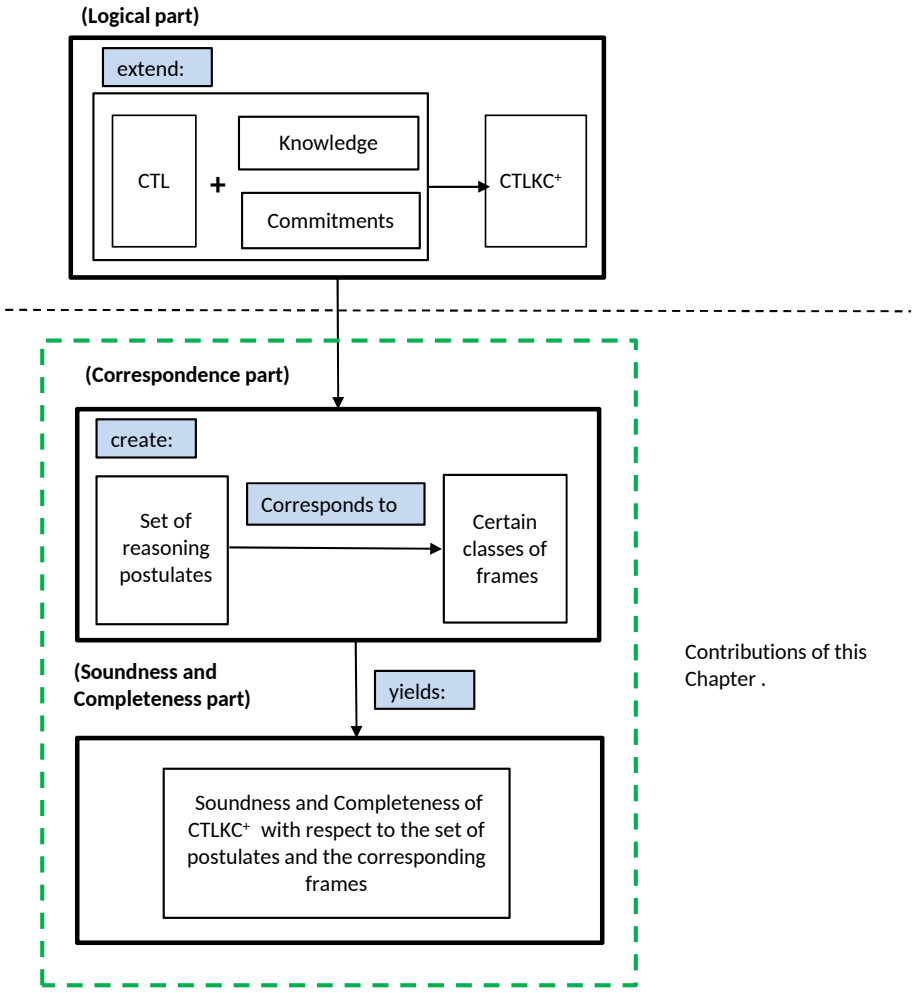

Figure 4.1: A schematic view of our approach

purely theoretical foundation could be seen as a potential weakness as the model could be considered abstract and far from any real and concrete application. To handle this issue and facilitate the reader's understanding of this chapter, we provide a concrete application , the NetBill protocol [90], as a running example throughout the chapter over which the axioms are illustrated.

The rest of this chapter is organized as follows. In Section 4.2, we address the problem of corresponding the reasoning postulates to certain classes of frames and then we prove the soundness and completeness of the CTLKC$^+$ logic. Finally, we conclude the chapter in Section 4.3.

4.2 Corresponding Reasoning Postulates

In this section, we use the NetBill payment protocol [90] which was presented in Section 3.3.1 as an application example, taken from the e-commerce domain, to illustrate the proposed reasoning postulates. After that, we consider the reasoning postulates along with their corresponding classes of frames. Finally, we explain how to use the correspondence results in proving the soundness and completeness of CTLKC$^+$. It is worth noticing that, to simplify the proofs of the correspondence between the proposed reasoning postulates and their related frames, the semantics of the $Fu(C_{i\rightarrow j}\varphi)$ in the CTLKC$^+$ logic is redefined in such a way to avoid the recursion that appeared in the second and third options of the semantics of the fulfillment in Chapter 3 (Section 3.5) as follows: [2]

- $(\mathscr{M},s) \models Fu(C_{i\rightarrow j}\varphi)$ iff there exists $s' \in S$ such that $s' \approx_{i\rightarrow j} s$ and $(\mathscr{M},s') \models C_{i\rightarrow j}\varphi$ or there exists $s' \in S$ such that $s \approx_i s'$ and there exists $s'' \in S$ such that $s'' \approx_{i\rightarrow j} s'$ and $(\mathscr{M},s'') \models C_{i\rightarrow j}\varphi$ or there exists $s' \in S$ such that $s \approx_j s'$ and there exists $s'' \in S$ such that $s'' \approx_{i\rightarrow j} s'$ and $(\mathscr{M},s'') \models C_{i\rightarrow j}\varphi$.

  However, the two semantics are equivalent.

4.2.1 Reasoning Postulates and Corresponding Frames

In this section, we follow Benthem's [92] notion of correspondence theory for modal logic in proving the correspondence between our proposed reasoning postulates and their related classes of frames. In our approach, we first give a name, formalization and meaning for each postulate. After that, we correspond the postulates to certain classes of frames and

[2]The semantics of $Fu(C_{i\rightarrow j}\varphi)$ in Chapter 3 (Section 3.5) was defined as follows:
$(\mathscr{M},s) \models Fu(C_{i\rightarrow j}\varphi)$ iff there exists $s' \in S$ such that $s' \approx_{i\rightarrow j} s$ and $(\mathscr{M},s') \models C_{i\rightarrow j}\varphi$ or
there exists $s'' \in S$ and $s'' \approx_i s$ such that $(\mathscr{M},s'') \models Fu(C_{i\rightarrow j}\varphi)$ or
there exists $s'' \in S$ and $s'' \approx_j s$ such that $(\mathscr{M},s'') \models Fu(C_{i\rightarrow j}\varphi)$

provide the required proofs. Thereafter, we present a discussion that illustrates the importance of each postulate in MASs and how they were addressed in the literature. It is worth mentioning that for valid formulas in any frame, we will not discuss the correspondence since they correspond to all possible frames.

P1. **[Fulfillment]**

Formalization: $Fu(C_{i\to j}\varphi) \to \varphi$.

Meaning: When a commitment is fulfilled, its content holds.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models Fu(C_{i\to j}\varphi) \to \varphi$ iff $\mathscr{F}$ is reflexive and symmetric with respect to $\approx_i$ or $\approx_j$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is reflexive and symmetric with respect to $\approx_i$ or $\approx_j$ and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi) \to \varphi$. Suppose that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi)$. We must show $(\mathscr{M}, s_1) \models \varphi$. From the semantics of $Fu(C_{i\to j}\varphi)$, three options are to be considered. According to the first option, there exists $s_2 \in W$ such that $s_2 \approx_{i\to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i\to j}\varphi$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_1) \models K_i\varphi \wedge K_j\varphi$. From the semantics of $K_i\varphi$, for all $s_3 \in W$ such that $s_1 \approx_i s_3$ we have $(\mathscr{M}, s_3) \models \varphi$. Since $\mathscr{F}$ is reflexive, we have $s_1 \approx_i s_1$. Thus, $(\mathscr{M}, s_1) \models \varphi$. So, we are done for the first option.

For the second and third options, there exists two states s_2 and $s_3 \in W$ such that $s_1 \approx_i s_2$, $s_3 \approx_{i\to j} s_2$, and $(\mathscr{M}, s_3) \models (C_{i\to j}\varphi)$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_2) \models K_i\varphi \wedge K_j\varphi$. Since $\mathscr{F}$ is symmetric, then $s_2 \approx_i s_1$. Therefore, from the semantics of $K_i\varphi$, $(\mathscr{M}, s_1) \models \varphi$ as desired.

($\Rightarrow$) We argue by contraposition. Suppose that $\mathscr{F}$ is not reflexive and not symmetric. We must show $\mathscr{F} \not\models Fu(C_{i\to j}\varphi) \to \varphi$. Using Remark 2.1, it is enough to show $\mathscr{F} \not\models$

$Fu(C_{i\to j}p) \to p$ for some sentence letter p. Consider the model $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ based on $\mathscr{F}$. Assume $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p)$. From the semantics of $Fu(C_{i\to j}p)$, there exists $s_2 \in W$ such that $s_2 \approx_{i\to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i\to j}p$. From the semantics of $C_{i\to j}p$, we have $(\mathscr{M}, s_1) \models K_i p \wedge K_j p$. From the semantics of $K_i p$, for all $s_3 \in W$ such that $s_1 \approx_i s_3$ we have $(\mathscr{M}, s_3) \models p$. Assume $(\mathscr{M}, s_1) \models \neg p$. Since $\mathscr{F}$ is not reflexive, then it might not be the case that $s_1 \approx_i s_1$ and so $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p) \wedge \neg p$. Thus, $\mathscr{F} \not\models Fu(C_{i\to j}p) \to p$.

□

Discussion: It is obvious that this postulate is reasonable and realistic because in general when an agent fulfills its commitment, the content of this commitment holds at the same state. For example, with respect to the NetBill protocol, once the customer pays the agreed amount of money (i.e., fulfills its commitment), then the payment (i.e., the content of the commitment) holds. Formally,
$Fu(C_{Cus\to Mer}\ sendPayment) \to sendPayment$. This postulate is incorporated in the axioms of fulfillment introduced in [7]. Furthermore, a similar postulate is also incorporated in [104, 20, 32, 21].

P2. [**Knowing the content of its own fulfilled commitment**]

Formalization: $Fu(C_{i\to j}\varphi) \to K_i\varphi$.

Meaning: An agent knows the content of its fulfilled commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models Fu(C_{i\to j}\varphi) \to K_i\varphi$ iff $\mathscr{F}$ is symmetric and transitive with respect to $\approx_i$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is symmetric and transitive with respect to $\approx_i$ and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given

$s_1 \in W$, we must show that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi) \to K_i\varphi$. Suppose that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi)$. We must show $(\mathscr{M}, s_1) \models K_i\varphi$. From the semantics of $Fu(C_{i\to j}\varphi)$, three options are to be considered. According to the first option, there exists $s_2 \in W$ such that $s_2 \approx_{i\to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i\to j}\varphi$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_1) \models K_i\varphi \wedge K_j\varphi$. So, we are done for the first option.

For the second and third options, there exists two states s_2 and $s_3 \in W$ such that $s_1 \approx_i s_2$, $s_3 \approx_{i\to j} s_2$, and $(\mathscr{M}, s_3) \models (C_{i\to j}\varphi)$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_2) \models K_i\varphi \wedge K_j\varphi$. Assume, $(\mathscr{M}, s_1) \models \neg K_i\varphi$. From the semantics of $K_i\varphi$, there exists $s_4 \in W$ such that $s_1 \approx_i s_4$ and $(\mathscr{M}, s_4) \models \neg\varphi$. Since $\mathscr{F}$ is symmetric, then $s_2 \approx_i s_1$. Further, since $\mathscr{F}$ is transitive, then $s_2 \approx_i s_4$. Therefore, from the semantics of $K_i\varphi$, we have $(\mathscr{M}, s_4) \models \varphi$. Thus, we have contradiction and so $(\mathscr{M}, s_1) \models K_i\varphi$ as desired.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not symmetric and not transitive. We must show $\mathscr{F} \not\models Fu(C_{i\to j}\varphi) \to K_i\varphi$. We argue by contraposition. Consider the model $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ based on $\mathscr{F}$. Suppose that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p)$. From the semantics of fulfillment, there exists two states s_2 and $s_3 \in W$ such that $s_1 \approx_i s_2$, $s_3 \approx_{i\to j} s_2$, and $(\mathscr{M}, s_3) \models (C_{i\to j}p)$. From the semantics of $C_{i\to j}p$, we have $(\mathscr{M}, s_2) \models K_ip \wedge K_jp$. Assume $(\mathscr{M}, s_1) \models \neg K_ip$. From the semantics of K_ip, there exists $s_4 \in W$ such that $s_1 \approx_i s_4$ and $(\mathscr{M}, s_4) \models \neg p$. Since $\mathscr{F}$ is not symmetric, then it might not be the case that $s_2 \approx_i s_1$. Further, since $\mathscr{F}$ not transitive, then it might not be the case that $s_2 \approx_i s_4$. Therefore, $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p) \wedge \neg K_ip$. Consequently, $\mathscr{F} \not\models Fu(C_{i\to j}p) \to K_ip$.

□

Discussion: This postulate highlights the fact that an agent knows the content of its fulfilled commitment. Otherwise, it could happen that the agent does not know about

its action. Hence, the agent might repeat it again and re-fulfill its commitment. For instance, from the NetBill protocol, assume that the merchant delivers the required goods to the customer (i.e., fulfills its commitment of delivering the goods). Applying the postulate, $Fu(C_{Mer \to Cus} deliverGoods) \to K_{Mer} deliverGoods$, reveals that the merchant should be aware about the content of its commitment after having fulfilled it. This postulate is incorporated in our reasoning postulates in Chapter 3.

P3. [**Knowing the content of the fulfilled commitment**]

Formalization: $Fu(C_{i \to j}\varphi) \to K_j\varphi$.

Meaning: An agent knows the content of the fulfilled commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i \to j})$, $\mathscr{F} \models Fu(C_{i \to j}\varphi) \to K_j\varphi$ iff $\mathscr{F}$ is symmetric and transitive with respect to $\approx_j$.

Proof. The proof is similar to the proof of P2, but with respect to j. □

Discussion: This postulate conveys the fact that the creditor knows the content of the fulfilled commitment. The postulate is reasonable as the creditor should be aware of the satisfaction of the commitment directed to it once this fulfillment occurs. Otherwise, the creditor might require the debtor to re-discharge the commitment. In the previous example, assume that the merchant delivers the required goods to the customer (i.e., fulfills its commitment). Thus, the customer should know that the goods has been delivered. Formally, $Fu(C_{Mer \to Cus}\ deliverGoods) \to K_{Cus}\ deliverGoods$. Otherwise, the customer could argue that no goods has been delivered, so it will require that a new delivery be performed, and the situation can be repeated. This postulate is incorporated in our reasoning postulates in Chapter 3.

P4. [**Knowing the fulfillment of its own commitment**]

Formalization: $Fu(C_{i\to j}\varphi) \to K_i(Fu(C_{i\to j}\varphi))$.

Meaning: An agent knows that it fulfills its own commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$,

$\mathscr{F} \models Fu(C_{i\to j}\varphi) \to K_i(Fu(C_{i\to j}\varphi))$ iff $\mathscr{F}$ is symmetric with respect to $\approx_i$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is symmetric with respect to $\approx_i$ and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi) \to K_i(Fu(C_{i\to j}\varphi))$. Suppose that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\varphi)$. We must show $(\mathscr{M}, s_1) \models K_i(Fu(C_{i\to j}\varphi))$. From the semantics of $Fu(C_{i\to j}\varphi)$, three options are to be considered. According to the first option, there exists $s_2 \in W$ such that $s_2 \approx_{i\to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i\to j}\varphi$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_1) \models K_i\varphi \wedge K_j\varphi$. Assume $(\mathscr{M}, s_1) \models \neg K_i(Fu(C_{i\to j}\varphi))$. From the semantics of knowledge, there exists $s_3 \in W$ such that $s_1 \approx_i s_3$ and $(\mathscr{M}, s_3) \models \neg Fu(C_{i\to j}\varphi)$. Using the semantics of $Fu(C_{i\to j}\varphi)$, the three or-conditions are not satisfied. Therefore, using Corollary 3.1, $\forall s_4 \in W$ such that $s_3 \approx_i s_4$, we have $(\mathscr{M}, s_4) \models \neg Fu(C_{i\to j}\varphi)$. Since $\mathscr{F}$ is symmetric, then $s_3 \approx_i s_1$. Thus, we have contradiction with the assumption when $s_1 = s_4$, and so $(\mathscr{M}, s_3) \models Fu(C_{i\to j}\varphi)$. Consequently, $(\mathscr{M}, s_1) \models K_i(Fu(C_{i\to j}\varphi))$.

For the second and third options, there exists two states s_2 and $s_3 \in W$ such that $s_1 \approx_i s_2$, $s_3 \approx_{i\to j} s_2$, and $(\mathscr{M}, s_3) \models (C_{i\to j}\varphi)$. From the semantics of $C_{i\to j}\varphi$, we have $(\mathscr{M}, s_2) \models K_i\varphi \wedge K_j\varphi$. Assume, $(\mathscr{M}, s_1) \models \neg K_i(Fu(C_{i\to j}\varphi))$. According to the semantics of knowledge, there exists $s_4 \in W$ such that $s_1 \approx_i s_4$ and $(\mathscr{M}, s_4) \models \neg Fu(C_{i\to j}\varphi)$. From the semantics of $Fu(C_{i\to j}\varphi)$, the three or-conditions are not satisfied. Therefore, using Corollary 3.1, $\forall s_5 \in W$ such that $s_4 \approx_i s_5$, we have $(\mathscr{M}, s_5) \models \neg Fu(C_{i\to j}\varphi)$. Since $\mathscr{F}$ is symmetric, then $s_4 \approx_i s_1$. Thus, we have contradiction with the assumption when $s_1 = s_5$, and so $(\mathscr{M}, s_4) \models Fu(C_{i\to j}\varphi)$. Consequently,

$(\mathscr{M}, s_1) \models K_i(Fu(C_{i\to j}\varphi))$ as desired.

($\Rightarrow$) We argue by contraposition. Suppose that $\mathscr{F}$ is not symmetric. We must show $\mathscr{F} \not\models Fu(C_{i\to j}\varphi) \to K_i(Fu(C_{i\to j}\varphi))$. Consider the model $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ based on $\mathscr{F}$. Assume $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p)$. From the semantics of $Fu(C_{i\to j}p)$, there exists $s_2 \in W$ such that $s_2 \approx_{i\to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i\to j}p$. Using the semantics of $C_{i\to j}p$, we have $(\mathscr{M}, s_1) \models K_i p \wedge K_j p$. To this end, assume $(\mathscr{M}, s_1) \models \neg K_i(Fu(C_{i\to j}p))$. According to the semantics of knowledge, there exists $s_3 \in W$ such that $s_1 \approx_i s_3$ and $(\mathscr{M}, s_3) \models \neg Fu(C_{i\to j}p)$. From the semantics of $Fu(C_{i\to j}\varphi)$, the three or-conditions are not satisfied. Therefore, using Corollary 3.1, for all s_4 such that $s_3 \approx_i s_4$, we have $(\mathscr{M}, s_4) \models \neg Fu(C_{i\to j}\varphi)$. Since $\mathscr{F}$ is not symmetric, then it might not be the case that $s_3 \approx_i s_1$. Thus, $(\mathscr{M}, s_1) \models Fu(C_{i\to j}p) \wedge \neg K_i(Fu(C_{i\to j}p))$. So, we are done.

□

Discussion: It is clear that this postulate has certain importance in open MASs. Actually, it reflects the agent's intentionality of its action to fulfill its own commitment. The validity of this postulate is significant as violating this postulates would mean that the agent is not aware about its action of discharging (fulfilling) its own commitment. Consequently, the agent could discharge the commitment many times. For example, assume that the customer fulfills its commitment of paying the required amount of money to the merchant, then the customer should know that. This is formally denoted as: $Fu(C_{Cus\to Mer}\ sendPayment) \to K_{Cus}Fu(C_{Cus\to Mer}\ sendPayment)$. This postulate is incorporated in our reasoning postulates in Chapter 3.

P5. [**Knowing the fulfillment of the commitment**]

Formalization: $Fu(C_{i\to j}\varphi) \to K_j Fu(C_{i\to j}\varphi)$.

Meaning: An agent knows the fulfillment of the commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$,

$\mathscr{F} \models Fu(C_{i\to j}\varphi) \to K_j(Fu(C_{i\to j}\varphi))$ iff $\mathscr{F}$ is symmetric with respect with $\approx_j$.

Proof. The proof is similar to the previous one, but with respect to j. □

Discussion: This postulate emphasizes the awareness of the creditor about the fulfillment of the debtor's commitment. In particular, it captures the intuition that, in open MASs, fulfilling the commitment is public. For example, assume that the merchant fulfills its commitment of delivering the required goods to the customer. This fulfillment should be public so that the customer can recognize it. It can be formally expressed as: $Fu(C_{Mer\to Cus}deliverGoods) \to K_{Cus}Fu(C_{Mer\to Cus}\ deliverGoods)$. If this postulate is violated, then serious problems could happen. Indeed, being aware about the fulfillment of the commitment, prevents the creditor from asking the debtor to fulfill it again. This postulate is incorporated in our reasoning postulates in Chapter 3.

P6. [**Knowing its own commitment**]

Formalization: $C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$.

Meaning: An agent knows about its own commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$ iff $\mathscr{F}$ is ES.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is ES and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$. Suppose that $(\mathscr{M}, s_1) \models C_{i\to j}\varphi$. We must show $(\mathscr{M}, s_1) \models K_i(C_{i\to j}\varphi)$.

Further, we must show $(\mathscr{M}, s_2) \models C_{i\to j}\varphi$ for any $s_2 \in W$ and $s_1 \approx_i s_2$. Assume $(\mathscr{M}, s_2) \models \neg C_{i\to j}\varphi$. From the semantic of $C_{i\to j}\varphi$, there exists $s_3 \in W$ such that $s_2 \approx_{i\to j} s_3$ and $(\mathscr{M}, s_3) \models \neg K_i\varphi \vee \neg K_j\varphi$. Based on the definition of ES, we have $s_1 \approx_{i\to j} s_3$ and so $(\mathscr{M}, s_3) \models K_i\varphi \wedge K_j\varphi$ which is a contradiction. Thus, $(\mathscr{M}, s_2) \models C_{i\to j}\varphi$ and therefore, $(\mathscr{M}, s_1) \models K_i(C_{i\to j}\varphi)$. Consequently, $(\mathscr{M}, s_1) \models C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$ as desired.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not ES. We must show $\mathscr{F} \nvdash C_{i\to j}\varphi \to K_i(C_{i\to j}\varphi)$. Consider the model $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ based on $\mathscr{F}$. Using contraposition, suppose that $(\mathscr{M}, s_1) \models C_{i\to j}p$. Furthermore, assume there exists $s_2 \in W$ such that $s_1 \approx_i s_2$ and $(\mathscr{M}, s_2) \models \neg C_{i\to j}p$. From the semantics of commitment, there exists $s_3 \in W$ such that $s_2 \approx_{i\to j} s_3$ and $(\mathscr{M}, s_3) \models \neg K_i p \vee \neg K_j p$. Since $\mathscr{F}$ is not ES, then it might not the case that $s_1 \approx_{i\to j} s_3$. Therefore, $(\mathscr{M}, s_1) \nvdash K_i(C_{i\to j}p)$. Consequently, $(\mathscr{M}, s_1) \models C_{i\to j}p \wedge \neg K_i(C_{i\to j}p)$ and so, $\mathscr{F} \nvdash C_{i\to j}p \to K_i(C_{i\to j}p)$, as desired. □

Discussion: Being committed to do something, the agent (i.e., the debtor) has to be aware about this action. This postulate is reasonable to be applied in MASs as agents should realize their own placed commitments. To illustrate the importance of this postulate, let us assume that the merchant commits toward the customer to deliver the required goods. Applying this postulate,

$C_{Mer\to Cus}\ deliverGoods \to K_{Mer}\ (C_{Mer\to Cus}\ deliverGoods)$, it is obvious that the committing agent (i.e., the merchant) should know about this particular commitment as it does not make sense for an agent to create a commitment and in the same time it is not aware about the consequences of this action. Otherwise, this would mean that the commitment is made accidently. This postulate is incorporated in our reasoning postulates in Chapter 3. Furthermore, a similar postulate is also incorporated in [81].

P7. [**R-Conjoin**]

Formalization: $(C_{i\to j}\varphi_1)\wedge(C_{i\to j}\varphi_2)\to C_{i\to j}(\varphi_1\wedge\varphi_2)$.

Meaning: Agent i would become committed towards agent j that both φ_1 and φ_2 are held if i individually commits toward j that φ_1 holds and i commits toward j that φ_2 holds.

Proof. Given $s_1\in W$, we must show $(\mathscr{M},s_1)\models(C_{i\to j}\varphi_1)\wedge(C_{i\to j}\varphi_2)\to C_{i\to j}(\varphi_1\wedge\varphi_2)$. Assume that $(\mathscr{M},s_1)\models((C_{i\to j}\varphi_1)\wedge(C_{i\to j}\varphi_2))\wedge\neg(C_{i\to j}(\varphi_1\wedge\varphi_2)$. From the semantics of $C_{i\to j}\varphi$, for all global states $s_2\in S$ such that $s_1\approx_{i\to j}s_2$, we have $(\mathscr{M},s_2)\models(K_i\varphi_1\wedge K_j\varphi_1)\wedge(K_i\varphi_2\wedge K_j\varphi_2)\wedge(\neg K_i(\varphi_1\wedge\varphi_2)\vee\neg K_j(\varphi_1\wedge\varphi_2))$. Further, from the semantics of knowledge, there exists $s_3\in S$ such that $s_2\approx_i s_3$ and $(\mathscr{M},s_3)\models(\varphi_1\wedge\neg\varphi_1)\vee(\varphi_2\wedge\neg\varphi_2)$. Thus, we have contradiction. Consequently, $(C_{i\to j}\varphi_1)\wedge(C_{i\to j}\varphi_2)\to C_{i\to j}(\varphi_1\wedge\varphi_2)$. So, the postulates.

□

Discussion:

The validity of this postulate is captured from the fact that agent i has the ability to have more than one commitment towards the same agent j at the same time. Suppose, for instance, that the merchant commits toward the customer to deliver the goods and it also commits toward the same customer to send a receipt, then the merchant would be committed towards the customer to deliver the goods and send the receipt. Formally, $(C_{Mer\to Cus}\ deliverGoods)\wedge(C_{Mer\to Cus}\ sendReciept)\to C_{Mer\to Cus}\ (deliverGoods\wedge sendReciept)$. This postulate is incorporated in [88, 32, 21].

P8. [**Knowing its R-conjoin commitment**]

Formalization: $(C_{i\to j}\varphi_1)\wedge(C_{i\to j}\varphi_2)\to K_i(C_{i\to j}(\varphi_1\wedge\varphi_2))$.

Meaning: An agent knows about its conjoin commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$,

$\mathscr{F} \models (C_{i\to j}\varphi_1) \wedge (C_{i\to j}\varphi_2) \to K_i(C_{i\to j}(\varphi_1 \wedge \varphi_2))$ iff $\mathscr{F}$ is ES.

Proof. Based on Postulate P7, $(\mathscr{M}, s_1) \models (C_{i\to j}\varphi_1) \wedge (C_{i\to j}\varphi_2) \to C_{i\to j}(\varphi_1 \wedge \varphi_2)$. Thus, it is enough to prove $(\mathscr{M}, s_1) \models C_{i\to j}(\varphi_1 \wedge \varphi_2) \to K_i(C_{i\to j}(\varphi_1 \wedge \varphi_2))$ which can be proved in a way similar to P6.

□

Discussion: The validity of this postulate is captured from the fact that an agent knows about its own commitment [2]. Consequently, in the previous example, the merchant should know that it would be committed towards the same customer to deliver the goods and send the receipt. Formally, $(C_{Mer\to Cus}\, deliverGoods) \wedge (C_{Mer\to Cus}\, sendReciept) \to K_{Mer}(C_{Mer\to Cus}\, (deliverGoods \wedge\, sendReceipt))$.

P9. [**Commitment's chain**]

Formalization: $(C_{i\to j}\varphi) \wedge (C_{i\to j}(\varphi \to \psi)) \to C_{i\to j}\psi$.

Meaning: Agent i can commit to a chain (implication).

Proof. Assume that $(\mathscr{M}, s_1) \models (C_{i\to j}\varphi) \wedge (C_{i\to j}(\varphi \to \psi)) \wedge \neg(C_{i\to j}\psi)$. From the semantics of commitment, for all $s_2 \in S$ such that $s_1 \approx_{i\to j} s_2$, we have $(\mathscr{M}, s_2) \models (K_i\varphi \wedge K_j\varphi) \wedge (K_i(\varphi \to \psi) \wedge K_j(\varphi \to \psi))$. Using the K axiom of knowledge, $(\mathscr{M}, s_2) \models K_i\psi \wedge K_j\psi$. Since $s_1 \approx_{i\to j} s_2$, then $(\mathscr{M}, s_1) \models C_{i\to j}\psi$ which contradicts our assumption.

□

Discussion: This postulate shows that CTLKC$^+$ is closed under strict implication. This postulate is integrated in [88, 32].

P10. [**Knowing its commitment's chain**]

Formalization: $(C_{i\to j}\varphi) \wedge (C_{i\to j}(\varphi \to \psi)) \to K_i(C_{i\to j}\psi)$.

Meaning: Knowledge of the commitments is closed under strict implication.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$,

$\mathscr{F} \models (C_{i\to j}\varphi) \wedge (C_{i\to j}(\varphi \to \psi)) \to K_i(C_{i\to j}\psi)$ iff $\mathscr{F}$ is ES.

Proof. Based on Postulate P9, $(\mathscr{M}, s_1) \models (C_{i\to j}\varphi) \wedge (C_{i\to j}(\varphi \to \psi)) \to C_{i\to j}\psi$. Thus, it is enough to prove $(\mathscr{M}, s_1) \models C_{i\to j}\psi \to K_i(C_{i\to j}\psi)$ which was proved in P6. □

Discussion: This postulate conveys the fact that an agent knows about its implied commitment.

P11. [**Weaken commitment**]

Formalization: $C_{i\to j}(\varphi_1 \wedge \varphi_2) \to C_{i\to j}\varphi_1$.

Meaning: Committing to a conjunction implies committing to each part of the conjunction.

Proof. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models C_{i\to j}(\varphi_1 \wedge \varphi_2) \to C_{i\to j}\varphi_1$. Assume that $(\mathscr{M}, s_1) \models C_{i\to j}(\varphi_1 \wedge \varphi_2) \wedge \neg(C_{i\to j}\varphi_1)$. From the semantics of $C_{i\to j}\varphi$, for all global states $s_2 \in S$ such that $s_1 \approx_{i\to j} s_2$, we have $(\mathscr{M}, s_2) \models K_i(\varphi_1 \wedge \varphi_2) \wedge K_j(\varphi_1 \wedge \varphi_2)$. Thus, $(\mathscr{M}, s_2) \models K_i\varphi_1 \wedge K_j\varphi_1$. Therefore, $(\mathscr{M}, s_1) \models C_{i\to j}\varphi_1$ which contradicts our assumption. So, the postulate.

□

Discussion: To clarify this postulate, consider the following example. In the NetBill protocol, if the merchant commits to send both the required goods and the receipt, then the merchant commits to send the goods. Formally, $C_{Mer \to Cus}\ (deliverGoods \wedge sendReceipt) \rightarrow C_{Mer \to Cus}\ deliverGoods$. This postulate is incorporated in [88, 32, 21].

P12. [**Knowing its weakened commitment**]

Formalization: $C_{i \to j}(\varphi_1 \wedge \varphi_2) \rightarrow K_i(C_{i \to j}\varphi_1)$.

Meaning: An agent knows about each part of its conjuncted commitment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i \to j})$, $\mathscr{F} \models C_{i \to j}(\varphi_1 \wedge \varphi_2) \rightarrow K_i(C_{i \to j}\varphi_1)$ iff $\mathscr{F}$ is ES.

Proof. Based on Postulate P11, $(\mathscr{M}, s_1) \models C_{i \to j}(\varphi_1 \wedge \varphi_2) \rightarrow C_{i \to j}\varphi_1$. So, it is enough to prove $(\mathscr{M}, s_1) \models C_{i \to j}\varphi_1 \rightarrow K_i(C_{i \to j}\varphi_1)$ which was proved in P6. □

Discussion: This postulate shows that if an agent commits to a conjunction, then it will be aware of its own commitment to each part of the conjunction.

P13. [**Weaken fulfillment**]

Formalization: $Fu(C_{i \to j}(\varphi_1 \wedge \varphi_2)) \rightarrow Fu(C_{i \to j}\varphi_1)$.

Meaning: If i fulfills a conjunction, i is also fulfills each part of the conjunction.

Proof. Assume that $(\mathscr{M}, s_1) \models Fu(C_{i \to j}(\varphi_1 \wedge \varphi_2)) \wedge \neg Fu(C_{i \to j}\varphi_1)$. From the semantics of $Fu(C_{i \to j}\varphi)$, the three fulfillment options are to be considered. According to the first option, there exists $s_2 \in W$ such that $s_2 \approx_{i \to j} s_1$ and $(\mathscr{M}, s_2) \models C_{i \to j}(\varphi_1 \wedge \varphi_2)$

and for all $s \in W$ such that $s \approx_{i\to j} s_1$ $(\mathscr{M},s) \models \neg C_{i\to j}\varphi_1$. Using P11, we have contradiction when $s = s_2$. Thus, $(\mathscr{M},s_1) \models Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \wedge Fu(C_{i\to j}\varphi_1)$. So, we are done for the first option.

For the second and third options, there exists two states s_2 and $s_3 \in W$ such that $s_1 \approx_i s_2$, $s_3 \approx_{i\to j} s_2$, and $(\mathscr{M},s_3) \models C_{i\to j}(\varphi_1 \wedge \varphi_2)$ and for all $s \in W$ such that $s \approx_{i\to j}$ s_2 $(\mathscr{M},s) \models \neg C_{i\to j}\varphi_1$. Using P11, we have contradiction when $s = s_3$. Therefore, $(\mathscr{M},s_3) \models C_{i\to j}(\varphi_1 \wedge \varphi_2) \wedge C_{i\to j}\varphi_1$ and so $(\mathscr{M},s_2) \models Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \wedge Fu((C_{i\to j}\varphi_1))$. Using Corollary 3.1, $(\mathscr{M},s_1) \models Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \wedge Fu((C_{i\to j}\varphi_1))$ as desired.

□

Discussion: To clarify the postulate, consider the following example if the merchant fulfills its commitment of sending both the required goods and the receipt, then the merchant fulfills sending the goods. Formally, $Fu(C_{Mer\to Cus}\ (deliverGoods \wedge sendReceipt)) \to Fu(C_{Mer\to Cus}\ deliverGoods)$.

P14. [**Knowing its weakened fulfillment**]

Formalization: $Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \to K_i(Fu(C_{i\to j}\varphi_1))$.

Meaning: An agent knows about each part of its conjuncted fulfillment.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \to K_i(Fu(C_{i\to j}\varphi_1))$ iff $\mathscr{F}$ is ES.

Proof. Based on Postulate P13, $Fu(C_{i\to j}(\varphi_1 \wedge \varphi_2)) \to Fu(C_{i\to j}\varphi_1)$. So, it is enough to prove $(\mathscr{M},s_1) \models Fu(C_{i\to j}\varphi_1) \to K_i(Fu(C_{i\to j}\varphi_1))$ which was proved in P4. □

Discussion: This postulate reflects the fact that if an agent fulfilled a conjunction, then it would be aware of its own fulfillment of each part of the conjunction.

P15. [**Strong consistency**]

Formalization: $C_{i\to j}\varphi \to \neg C_{i\to j}\neg\varphi$.

Meaning: If commitment is satisfied, then committing to the negation of its content never satisfied.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models C_{i\to j}\varphi \to \neg C_{i\to j}\neg\varphi$ iff $\mathscr{F}$ is serial with respect to $\approx_{i\to j}$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is serial with respect to $\approx_{i\to j}$, and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models C_{i\to j}\varphi \to \neg C_{i\to j}\neg\varphi$. Assume that $(\mathscr{M}, s_1) \models (C_{i\to j}\varphi) \wedge (C_{i\to j}\neg\varphi)$. From the semantics of $C_{i\to j}\varphi$, for all global states $s_2 \in S$ such that $s_1 \approx_{i\to j} s_2$, we have $(\mathscr{M}, s_2) \models K_i\varphi \wedge K_i\neg\varphi$. Thus, from the semantics of $K_i\varphi$, for all $s_3 \in S$ such that $s_2 \approx_i s_3$, we have $(\mathscr{M}, s_3) \models \varphi \wedge \neg\varphi$. Therefore, the contradiction.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not serial with respect to $\approx_{i\to j}$. We must show $\mathscr{F} \not\models C_{i\to j}\varphi \to \neg C_{i\to j}\neg\varphi$. Since $\mathscr{F}$ is not serial, using an argument by contraposition, then it might be the case that $(\mathscr{M}, s_1) \models C_{i\to j}p \wedge C_{i\to j}\neg p$. Therefore, $\mathscr{F} \not\models C_{i\to j}p \to \neg C_{i\to j}\neg p$, as desired.

□

Discussion: The validity of this postulate is captured from the fact that an agent cannot commit to bring about φ and $\neg\varphi$ at the same time. This postulate is incorporated in [88, 32].

P16. [**Knowing strong consistency**]

Formalization: $C_{i\to j}\varphi \to K_i(\neg C_{i\to j}\neg\varphi)$.

Meaning: When a commitment holds, then the debtor knows that there is no possibility to commit about the the negation of the commitment content.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$,

$\mathscr{F} \models C_{i\to j}\varphi \to K_i(\neg C_{i\to j}\neg\varphi)$ iff $\mathscr{F}$ is ES.

Proof. Based on Postulate P15, $(\mathscr{M}, s_1) \models C_{i\to j}\varphi \to \neg C_{i\to j}\neg\varphi$. Consequently, it is enough to prove that $(\mathscr{M}, s_1) \models \neg C_{i\to j}\neg\varphi \to K_i(\neg C_{i\to j}\neg\varphi)$ which is similar to the proof of P6. □

Discussion: This postulate is reasonable to be applied in open MASs since it does not make sense for an agent to commit and reason about φ and $\neg\varphi$ at the same time.

P17. [**Nonexistence**]

Formalization: $AG\ \neg\varphi \to \neg C_{i\to j}\varphi$.

Meaning: If the content of the commitment does not hold globally, then the commitment itself does not hold too.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models AG\ \neg\varphi \to \neg C_{i\to j}\varphi$ iff $\mathscr{F}$ is serial with respect to $\approx_{i\to j}$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is serial with respect to $\approx_{i\to j}$ and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models AG\ \neg\varphi \to \neg C_{i\to j}\varphi$. Assume that $(\mathscr{M}, s_1) \models AG\ \neg\varphi \wedge C_{i\to j}\varphi$. From the semantics of $C_{i\to j}\varphi$, for all global states $s_2 \in S$ such that $s_1 \approx_{i\to j} s_2$, we have $(\mathscr{M}, s_2) \models K_i\varphi \wedge K_j\varphi$. Further, from the semantics of $K_i\varphi$ and $AG\varphi$, for all $s_3 \in S$ such that $s_2 \approx_i s_3$, we have $(\mathscr{M}, s_3) \models \varphi \wedge \neg\varphi$. So, the contradiction.

($\Rightarrow$) We argue by contraposition. Suppose that $\mathscr{F}$ is not serial with respect to $\approx_{i\to j}$. We must show $\mathscr{F} \not\models AG\ \neg\varphi \to \neg C_{i\to j}\varphi$. Since $\mathscr{F}$ is not serial, then it might be the case that $(\mathscr{M}, s_1) \models AG\ \neg p \wedge C_{i\to j}p$. Therefore, $\mathscr{F} \not\models AG\ \neg p \to \neg C_{i\to j}p$, as desired. □

Discussion: This postulate illustrates the fact that, if the content of the commitment (i.e., φ) does not hold in all global states, then the commitment itself never holds. This seems reasonable since, from the semantics of the commitment, both agents (i.e., debtor and creditor) become aware of the content in the accessible states which are also reachable. This postulates is incorporated in [32].

P18. [**Knowing the nonexistence**]

Formalization: $AG\ \neg\varphi \to K_i\neg(C_{i\to j}\varphi)$.

Meaning: An agent knows that it is not the case that it commits to a content which never holds.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models AG\ \neg\varphi \to K_i\neg(C_{i\to j}\varphi)$ iff $\mathscr{F}$ is ES.

Proof. From Postulate P17, $(\mathscr{M}, s_1) \models AG\neg\varphi \to \neg C_{i\to j}\varphi$. So, it is enough to prove $(\mathscr{M}, s_1) \models \neg C_{i\to j}\varphi \to K_i\neg(C_{i\to j}\varphi)$ which has similar proof as P6.

□

Discussion: This postulate is reasonable to be applied in open MASs since agents should know that they cannot commit about an impossible content.

P19. [**Commitment consistency**]

Formalization: $\neg C_{i\to j}\bot$.

Meaning: Commuting to false never holds.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models \neg C_{i\to j}\bot$ iff $\mathscr{F}$ is serial with respect to $\approx_{i\to j}$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is serial with respect to $\approx_{i\to j}$ and let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models \neg C_{i\to j}\bot$. Assume that $(\mathscr{M}, s_1) \models C_{i\to j}\bot$. From the semantics of commitment, for all global states $s_2 \in S$ such that $s_1 \approx_{i\to j} s_2$, we have $(\mathscr{M}, s_2) \models K_i\bot \wedge K_j\bot$ which contradicts the consistency axiom of knowledge (i.e., Axiom D). Therefore, $(\mathscr{M}, s_1) \models \neg C_{i\to j}\bot$.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not serial with respect to $\approx_{i\to j}$. We must show $\mathscr{F} \not\models \neg C_{i\to j}\bot$. Since $\mathscr{F}$ is not serial, then it might be the case that $(\mathscr{M}, s_1) \models C_{i\to j}\bot$. Therefore, $\mathscr{F} \not\models \neg C_{i\to j}\bot$, as desired.

□

Discussion: The validity of this postulate is captured from the fact that an agent cannot know *false* [43]. Consequently, similar to knowledge, an agent cannot commit to false. This postulate is integrated in [88, 32, 21].

P20. [**Knowing committing to false**]

Formalization: $\neg K_i(C_{i\to j}\bot)$.

Meaning: Committing to false cannot be known.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models \neg K_i(C_{i\to j}\bot)$ iff $\mathscr{F}$ is serial with respect to $\approx_i$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is serial with respect to $\approx_i$. Let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show

$(\mathscr{M}, s_1) \models \neg K_i(C_{i\to j}\bot)$. Assume that $(\mathscr{M}, s_1) \models K_i(C_{i\to j}\bot)$. From the semantics of knowledge, for all global states $s_2 \in S$ such that $s_1 \approx_i s_2$, we have $(\mathscr{M}, s_2) \models C_{i\to j}\bot$ which is a contradiction with Postulate P19. Therefore, $(\mathscr{M}, s_1) \models \neg K_i(C_{i\to j}\bot)$.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not serial with respect to $\approx_i$. We must show $\mathscr{F} \not\models \neg K_i(C_{i\to j}\bot)$. Since $\mathscr{F}$ is not serial, then it might be the case that $(\mathscr{M}, s_1) \models K_i(C_{i\to j}\bot)$. Therefore, $\mathscr{F} \not\models \neg K_i(C_{i\to j}\bot)$, as desired.

□

Discussion: This postulate is reasonable to be applied in open MASs since it reflects the fact that an agent cannot know an impossible commitment. Again, the validity of this postulate is captured from the fact that an agent cannot know false (i.e., Axiom *D* of knowledge).

P21. [**Fulfillment consistency**]

Formalization: $\neg Fu(C_{i\to j}\bot)$.

Meaning: Agent *i* cannot fulfill a non existing commitment.

Proof. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models \neg Fu(C_{i\to j}\bot)$. Assume that $(\mathscr{M}, s_1) \models Fu(C_{i\to j}\bot)$. According to Postulate P2, $(\mathscr{M}, s_1) \models K_i\bot$ which contradicts the *D* axiom of knowledge. Thus, $(\mathscr{M}, s_1) \models \neg Fu(C_{i\to j}\bot)$.

□

Discussion: This postulate says that a non existing commitment cannot be fulfilled. It is reasonable to be applied in open MASs as agents cannot fulfill an impossible commitment. This postulate is incorporated in [32].

P22. [**Debtors' knowledge of fulfilling an impossible commitment**]

Formalization: $\neg K_i(Fu(C_{i\to j}\bot))$.

Meaning: Fulfillment for committing to false cannot be known by the debtor.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models \neg K_i(Fu(C_{i\to j}\bot))$ iff $\mathscr{F}$ is serial with respect to $\approx_i$.

Proof. ($\Leftarrow$) Suppose that $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$ is serial with respect to $\approx_i$. Let $\mathscr{M} = (W, \approx_i, \approx_j, \approx_{i\to j}, \mathscr{V})$ be any model based on $\mathscr{F}$. Given $s_1 \in W$, we must show $(\mathscr{M}, s_1) \models \neg K_i(Fu(C_{i\to j}\bot))$. Assume $(\mathscr{M}, s_1) \models K_i(Fu(C_{i\to j}\bot))$. From the semantics of knowledge, for all global states $s_2 \in S$ such that $s_1 \approx_i s_2$ we have $(\mathscr{M}, s_2) \models Fu(C_{i\to j}\bot)$ which is a contradiction with P21. Thus, $(\mathscr{M}, s_1) \models \neg K_i(Fu(C_{i\to j}\bot))$.

($\Rightarrow$) Suppose that $\mathscr{F}$ is not serial with respect to $\approx_i$. We must show that $\mathscr{F} \not\models \neg K_i(Fu(C_{i\to j}\bot))$. We argue by contraposition. Since $\mathscr{F}$ is not serial, then it might be the case that $(\mathscr{M}, s_1) \models K_i(Fu(C_{i\to j}\bot))$. Therefore, $\mathscr{F} \not\models \neg K_i(Fu(C_{i\to j}\bot))$, as desired.

□

Discussion: This postulate is reasonable to be applied in MASs as debtor cannot know a fulfillment for a non existing commitment.

P23. [**Creditors' knowledge of fulfilling an impossible commitment**]

Formalization: $\neg K_j(Fu(C_{i\to j}\bot))$.

Meaning: Fulfillment for committing to false cannot be known by the creditor.

Correspondence: For any frame $\mathscr{F} = (W, \approx_i, \approx_j, \approx_{i\to j})$, $\mathscr{F} \models \neg K_j(Fu(C_{i\to j}\bot))$ iff $\mathscr{F}$ is serial with respect to $\approx_j$.

Proof. The proof is similar to the previous one, but with respect j. □

Discussion: Similar to Postulate P22, the creditor cannot know a fulfillment to a non existing commitment.

4.2.2 Soundness and Completeness

Soundness and completeness of a "deduction system" illustrate its appropriateness for handling logic [96]. A logic is called sound, "*whenever its decision problem is solved for a given set of formulae, then this formula set has a special semantic property*" [96]. On the other hand, completeness of a deductive logic means that "*if a set of formulae has the semantic property given by soundness, then the calculus works successfully over that set of formulas*" [96].

As mentioned by [88], the existence of the correspondence between a given set of postulates and their related classes of frames provides the soundness and completeness for the logic under consideration. Thus, the Theorem below results directly from the proofs given in the aforementioned postulates.

Theorem 4.1. *The logic consists of any subset of the postulates* $\{P1 - P23\}$ *is sound and complete with regard to the models that are based on the corresponding classes of frames.*

By proving the soundness and completeness of CTLKC$^+$*, we answer the fifth research question [Q5].*

4.3 Summary

In this chapter, we proved the soundness and completeness of CTLKC$^+$ using Benthem's correspondence theory for modal logic. The main insightful and practical implication of this

chapter is that the combined CTLKC$^+$ logic is proven as a consistent, robust and powerful logical tool to model complex but realistic MASs where agents have knowledge and able to manipulate and reason about commitments. This will leverage the use of this logic in practice particularly because it is more expressive than the logics of knowledge and the logics of commitments taken separately.

In the next chapter, we will address the problem of model checking CTLKC$^+$. Then, we will compute the time and space complexity of the proposed model checking algorithms.

Chapter 5

Model Checking Temporal Knowledge and Social Commitments in MASs

In this chapter[1], we address the problem of model checking CTLKC$^+$ by transforming it to the problem of model checking GCTL* [14] and ARCTL [75] in order to respectively use the CWB-NC automata-based model checker and the extended NuSMV symbolic model checker. We also prove that the transformation (reduction) techniques are sound. After that, we analyze the space and time complexity of the proposed model checking techniques. The results of this analysis reveal that the space complexity of our procedures is PSPACE-complete for local concurrent programs with respect to the size of these programs and the length of the formula being checked. Finally, we implement our model checking procedures on top of the extended NuSMV and CWB-NC model checkers and report some verification results.

[1]The results of this chapter are published in [3] and [2].

5.1 Introduction

Multi-Agent Systems (MASs) have noticed an increase in their use in numerous real world applications since their emergence. They have been extensively and successfully used in a variety of industrial, commercial, governmental, military, and entertainment applications [70, 98, 56]. Such systems have long been under focus by researchers to develop systematic techniques to model them and ensure their compliance against their specifications. In fact, various approaches have been carried out to model and represent MASs. Kripke structures [57] and interpreted systems [43] are the most prominent frameworks for this purpose. These underlying models are used to traditionally interpret some logics that are used to specify and reason about desirable properties of MASs.

In this chapter, we exploit model checking paradigm to formally model and automatically verify MASs with respect to certain properties related to agents knowledge and their commitments in the system. In Chapter 3, we studied the interactions between knowledge and social commitments in MASs from formal semantics perspective. Concretely, we introduced the CTLKC$^+$ logic, an extension of CTL [41] with modalities for knowledge and commitments. This logic has the ability to express and reason not only about knowledge and social commitments independently, but also about formulas combining the two modalities. Moreover, we develop, a new version of interpreted systems, originally introduced in [43], as the formal model of CTLKC$^+$ over which formulas can be interpreted. This extension allows us to model agents as well as their interactions. The developed approach proposes a new definition of the social accessibility relation needed for commitments, which was introduced in [7, 36] in such a way that it does not include the epistemic accessibility for knowledge in any way, yet keeping the intuition of having communication channels between interacting components. This new definition makes the logic consistent when it comes to express relationships between knowledge and commitments. After that,

in Chapter 4, we proved the soundness and completeness of the CTLKC$^+$ logic using correspondence theory for modal logic [92].

In this chapter, we aim to investigate the relationship between knowledge and social commitments from model checking and complexity perspectives. To do so, we first use a direct and intuitive reduction technique of the model checking problem of CTLKC$^+$ logic into the problem of model checking an action logic called GCTL* [14] that extends branching temporal logic. The technique is direct in the sense that GCTL* models include general action transitions that are directly mapped to the accessibility relations. Consequently, we are able to use the automata-based model checker CWB-NC as verification tool. After that, we develop a symbolic model checking approach by transforming the problem of model checking CTLKC$^+$ into the problem of model checking ARCTL [75], that extends the branching time logic CTL with actions. Thus, we are able to benefit from the extended NuSMV symbolic model checker as verification tool.

Figure 5.1 illustrates the overall approach, which consists of four phases. In the first phase, we recall the CTLKC$^+$ logic which was introduced in Chapter 3. In the second phase, we introduce our formal verification techniques based on transforming the problem of model checking CTLKC$^+$ into the problem of model checking GCTL* [14] and ARCTL [75]. In the third phase, we analyze the complexity of our transformation-based procedures of model checking CTLKC$^+$. To check the effectiveness of the proposed approach, in the fourth phase we implement our reduction techniques on top of the CWB-NC and extended NuSMV model checkers and report the verification results of verifying the NetBill protocol [90] against some desirable properties expressed in our logic.

The rest of this chapter is organized as follows. In Section 5.2, we address the

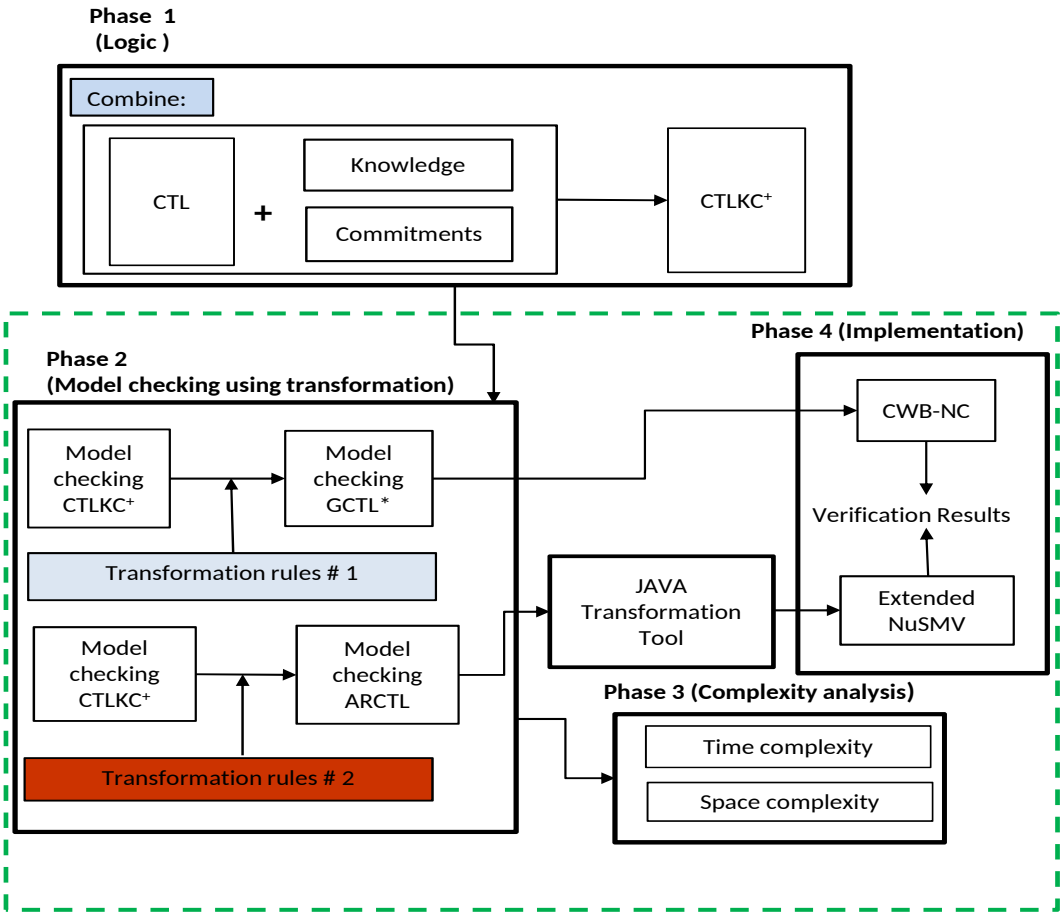

Figure 5.1: A schematic view of the proposed approach

automata-based model checking of CTLKC$^+$. In Section 5.3, we introduce the symbolic-based model checking of CTLKC$^+$. In Section 5.4, we compute the computational complexity of our transformation-based model checking techniques. We discuss the implementation of the proposed techniques and report the verification results in Section 5.5. Finally, we conclude the chapter in Section 5.6.

5.2 Model Checking CTLKC$^+$ using Transformation to GCTL*

In this section, we introduce a new technique to model check CTLKC$^+$ (Phase 2 of Figure 5.1). In a nutshell, the technical formulation of the CTLKC$^+$ model checking problem is as follows: Given a MAS represented as an interpreted systems model $\mathfrak{M}$ and a formula φ in CTLKC$^+$ describing a property, the problem can be defined as verifying whether or not $\mathfrak{M} \models \varphi$.

Model checking the CTLKC$^+$ logic can be carried out in two ways:

- A direct method by either developing a proper model checker from scratch or extending an existing model checker with new algorithms for the needed modalities as in [7, 39, 62, 40].

- By a formal reduction (transformation) method into an existing model checker as in [13, 61, 36, 97]. In this chapter, we follow this approach by transforming the problem of model checking CTLKC$^+$ into the problem of model checking GCTL* (a generalized version of CTL* with action formulas) [14] and ARCTL (an extension of CTL with action formulas) [75].

5.2.1 Transformation Procedure

In this section, we will show how our CTLKC$^+$ logic can be model checked by formally transforming the problem of model checking CTLKC$^+$ into the problem of model checking an existing logic called Generalized CTL* (or simply GCTL*) [14]. We first review GCTL* that extends CTL* by allowing formulas to constrain not only states, but also actions. We then present the transformation (also called reduction) procedure of the model checking

problem. The following BNF grammar defines the syntax of GCTL* as proposed in [14]:

$$\mathrm{S} ::= p \mid \neg \mathrm{S} \mid \mathrm{S} \vee \mathrm{S} \mid E\ \mathrm{P}$$

$$\mathrm{P} ::= \theta \mid \neg \mathrm{P} \mid \mathrm{S} \mid \mathrm{P} \vee \mathrm{P} \mid X\ \mathrm{P} \mid \mathrm{P}\ U\ \mathrm{P}$$

where p is an atomic proposition from the set Φ_p and θ is an atomic action proposition from the set Φ_a. Two types of formulas are distinguished: (1) state formulas S that hold on a given state; and (2) path formulas P that express temporal properties of paths. State formulas are the legal GCTL* formulas. The model of GCTL* is defined as follows.

Definition 5.1 (Model of GCTL*)**.** A model $\mathrm{M}_{GC} = (S_{GC}, Acc, l_{SC}, l_{Acc}, T_r, I_{GC})$ is a tuple where S_{GC} is a set of global states; Acc is a set of actions; $l_{SC} : S_{GC} \to 2^{\Phi_p}$ is a state labeling function; $l_{Acc} : Acc \to 2^{\Phi_a}$ is an action labeling function; $T_r \subseteq S_{GC} \times Acc \times S_{GC}$ is a labeled transition relation; and $I_{GC} \subseteq S_{GC}$ is a set of initial states.

Intuitively, S_{GC} contains the reachable states of the system, and Acc are the possible actions for the system. The label function l_{SC} indicates which atomic proposition is satisfied on a given state where the function l_{Acc} indicates which atomic action is satisfied on a given state.

The GCTL* semantics as a temporal logic is given as follows [14]. A state satisfies $A\varphi$ "if every path starting from the state satisfies φ". On the other hand, a state satisfies $E\varphi$ "if some paths starting from the state satisfies φ". A path satisfies a state formula if its initial state does, and a path satisfies θ if the label of the first transition on this path satisfies θ. The time operators X and U are as usual.

Our transformation procedure from the problem of model checking CTLKC$^+$ to the problem of model checking GCTL* is defined as follows: given a CTLKC$^+$ model $\mathfrak{M} = (S, I, R_t, \{\approx_i \mid i \in \mathscr{A}\}, \{\approx_{i\to j} \mid (i,j) \in \mathscr{A}^2\}, \mathscr{V})$, we need to define a GCTL* model $\mathrm{M}_{GC} =$

$\mathscr{F}(\mathfrak{M})$. Further, given a CTLKC$^+$ formula φ, we need to define a GCTL* formula $\mathscr{F}(\varphi)$ using a transformation function $\mathscr{F}$ such that $\mathfrak{M} \models \varphi$ iff $\mathscr{F}(\mathfrak{M}) \models \mathscr{F}(\varphi)$. The model $\mathscr{F}(\mathfrak{M})$ is defined as a GCTL* model $\mathrm{M}_{GC}(S_{GC}, Acc, l_{SC}, l_{Acc}, T_r, I_{GC})$ as follows:

- $S_{GC} = S$;
- $I_{GC} = I$;
- $l_{SC} = \mathscr{V}$;
- To define the set of actions (Acc), let us first define the set Φ_a of atomic action propositions from three types of actions: one for the social accessibility relation $\approx_{i\to j}$ to capture the semantics of commitment; one for the epistemic accessibility relation $\approx_i$ to capture the semantics of knowledge, and one from the symmetric closure of the social accessibility relation $\approx_{i\to j}$ to capture the semantics of fulfillment. $\Phi_a = \{\varepsilon, \alpha_{1\to 1}, \alpha_{1\to 2}, \ldots, \alpha_{n\to n}\} \cup \{\beta_1, \beta_2, \ldots, \beta_n\} \cup \{\gamma_{1\to 1}, \gamma_{1\to 2}, \ldots, \gamma_{n\to n}\}$, then $Acc = \{\alpha^o, \alpha^{11}, \alpha^{12}, \ldots, \alpha^{nn}\} \cup \{\beta^1, \beta^2, \ldots, \beta^n\} \cup \{\gamma^{11}, \gamma^{12}, \ldots, \gamma^{nn}\}$ where α^o is the action labeling transition defined from the transition relation (T_r) and α^{ij} is the action labeling transition defined from the social accessibility relation $\approx_{i\to j}$, β^i is the action labeling the transition obtained from the epistemic accessibility relation $\approx_i$, and γ^{ij} is the action labeling the symmetric transition added when there exists a transition labeled with α^{ij} and needed to define transformation of the formula $Fu(C_{i\to j}\varphi)$;
- The function l_{Acc} is defined as follows:
  1. If $\alpha^o \in Acc$, then $l_{Acc}(\alpha^o) = \{\varepsilon\}$,
  2. $l_{Acc}(\alpha^{ij}) = \{\alpha_{i\to j}\}$ for $1 \leq i \leq n$ and $1 \leq j \leq n$,
  3. $l_{Acc}(\beta^i) = \{\beta_i\}$ for $1 \leq i \leq n$,

4. $l_{Acc}(\gamma^{ij}) = \{\gamma_{i\to j}\}$ for $1 \le i \le n$ and $1 \le j \le n$; and

- The labeled transition relation T_r combines the temporal labeled transition R_t, the accessibility relations $\approx_{i\to j}$ and $\approx_i$, and the symmetric closure of the social accessibility relation $\approx_{i\to j}$ as follows:

  1. $(s, \alpha^o, s') \in T_r$ if $(s, s') \in R_t$,
  2. $(s, \alpha^{ij}, s') \in T_r$ if $s \approx_{i\to j} s'$,
  3. $(s, \beta^i, s') \in T_r$ if $s \approx_i s'$,
  4. $(s, \gamma^{ij}, s') \in T_r$ if $(s', \alpha^{ij}, s) \in T_r$.

Let us now define $\mathscr{F}(\varphi)$ as a GCTL* formula by induction on the form of the CTLKC$^+$ formula φ.

- $\mathscr{F}(p)$=p, if $p \in \Phi_p$;
- $\mathscr{F}(\neg\varphi) = \neg\mathscr{F}(\varphi)$;
- $\mathscr{F}(\varphi \vee \psi) = \mathscr{F}(\varphi) \vee \mathscr{F}(\psi)$;
- $\mathscr{F}(EX\varphi) = EX\mathscr{F}(\varphi)$;
- $\mathscr{F}(E(\varphi\ U\ \psi)) = E(\mathscr{F}(\varphi)\ U\ \mathscr{F}(\psi))$;
- $\mathscr{F}(EG\varphi) = EG\mathscr{F}(\varphi)$;
- $\mathscr{F}(K_i\varphi) = A(\beta_i \wedge X\mathscr{F}(\varphi))$;
- $\mathscr{F}(C_{i\to j}\varphi) = A(\alpha_{i\to j} \wedge X\mathscr{F}(K_i\varphi \wedge K_j\varphi))$;
- $\mathscr{F}(Fu(C_{i\to j}\varphi)) = E(\gamma_{i\to j} \wedge X\mathscr{F}(C_{i\to j}\varphi)) \vee$
  $E(\beta_i \wedge X\mathscr{F}(Fu(C_{i\to j}\varphi))) \vee$
  $E(\beta_j \wedge X\mathscr{F}(Fu(C_{i\to j}\varphi)))$.

Theorem 5.1 (Soundness of $\mathscr{F}$)**.** *Let $\mathfrak{M}$ and φ be respectively a CTLKC$^+$ model and formula and let $\mathscr{F}(\mathfrak{M})$ and $\mathscr{F}(\varphi)$ be the corresponding model and formula in GCTL*. We have $\mathfrak{M} \models \varphi$ iff $\mathscr{F}(\mathfrak{M}) \models \mathscr{F}(\varphi)$.*

Proof. We prove this theorem by induction on the structure of the formula φ. All the cases are straightforward, except the following three cases:

- $\varphi = K_i\psi$. We have $(\mathfrak{M},s) \models K_i\psi$ iff $(\mathfrak{M},s') \models \psi$ for every $s' \in S$ such that $s \approx_i s'$. Consequently, $(\mathfrak{M},s) \models K_i\psi$ iff $(\mathscr{F}(\mathfrak{M}),s') \models \mathscr{F}(\psi)$ for every $s' \in S_{GC}$ such that $(s,\beta^i,s') \in T_r$. By semantics of A and X, we obtain $(\mathscr{F}(\mathfrak{M}),s) \models A(\beta_i \wedge X\mathscr{F}(\psi)$.

- $\varphi = C_{i\to j}\psi$. We have $(\mathfrak{M},s) \models C_{i\to j}\psi$ iff $(\mathfrak{M},s') \models K_i\psi \wedge K_j\psi$ for every $s' \in S$ such that $s \approx_{i\to j} s'$. Consequently, $(\mathfrak{M},s) \models C_{i\to j}\psi$ iff $(\mathscr{F}(\mathfrak{M}),s') \models \mathscr{F}(K_i\psi \wedge K_j\psi)$ for every $s' \in S_{GC}$ such that $(s,\alpha^{ij},s') \in T_r$. By the semantics of A and X, we obtain $(\mathscr{F}(\mathfrak{M}),s) \models A(\alpha_{i\to j} \wedge X\mathscr{F}(K_i\psi \wedge K_j\psi))$.

- $\varphi = Fu(C_{i\to j}\psi)$. We have $(\mathfrak{M},s) \models Fu(C_{i\to j}\psi)$ iff (1) $(\mathfrak{M},s_1) \models C_{i\to j}\psi$ for a state $s_1 \in S$ such that $s_1 \approx_{i\to j} s$; or (2) $(\mathfrak{M},s_2) \models Fu(C_{i\to j}\psi)$ for a state $s_2 \in S$ such that $s_2 \approx_i s$; or (3) $(\mathfrak{M},s_3) \models Fu(C_{i\to j}\psi)$ for a state $s_3 \in S$ such that $s_3 \approx_j s$.

  Consequently, $(\mathfrak{M},s) \models Fu(C_{i\to j}\psi)$ iff (1) $(\mathscr{F}(\mathfrak{M}),s_1) \models \mathscr{F}(C_{i\to j}\psi)$ for $s_1 \in S_{GC}$ such that $(s,\gamma^{ij},s_1) \in T_r$; or (2) $(\mathscr{F}(\mathfrak{M}),s_2) \models \mathscr{F}(Fu(C_{i\to j}\psi))$ for $s_2 \in S_{GC}$ such that $(s_2,\beta^i,s) \in T_r$; or (3) $(\mathscr{F}(\mathfrak{M}),s_3) \models \mathscr{F}(Fu(C_{i\to j}\psi))$ for $s_3 \in S_{GC}$ such that (s_3,β^j,s) $\in T_r$. Using the semantics of E and X, we obtain: $(\mathscr{F}(\mathfrak{M}),s) \models E(\gamma_{i\to j} \wedge X\mathscr{F}(C_{i\to j}\varphi))$ $\vee\, E(\beta_i \wedge X\mathscr{F}(Fu(C_{i\to j}\varphi))) \vee E(\beta_j \wedge X\mathscr{F}(Fu(C_{i\to j}\varphi)))$.

Therefore, the theorem. □

5.3 Model Checking CTLKC$^+$ using Transformation to ARCTL

5.3.1 Transformation Procedure

In this section, we briefly review ARCTL logic [75]. Then, we show how the problem of model checking CTLKC$^+$ can be reduced to the problem of model checking ARCTL. The main advantage of using this reduction is to benefit from the efficient model checking procedure already integrated in the extended NuSMV model checker [22]. Figure 5.2 depicts the workflow of such a reduction technique which consists of the following processes. First, we transform our model $\mathfrak{M}$ into an ARCTL model, which is automatically translated into an extended NuSMV model. Second, we transform CTLKC$^+$ formulas into ARCTL formulas. Finally, both ARCTL formulas and the obtained extended NuSMV model will be the input of the extended NuSMV model checker to obtain our verification results. This approach is carried out automatically using a JAVA transformation tool. More details about this tool will be given in Section 5.5.2.

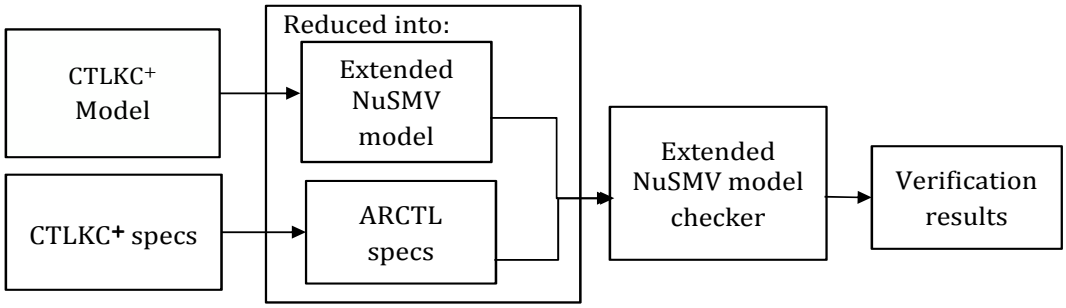

Figure 5.2: Reduction technique workflow

The syntax of ARCTL, as proposed in [75], is defined by the following BNF grammar:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \varphi \mid E_\alpha X\varphi \mid E_\alpha(\varphi\ U\ \varphi) \mid E_\alpha G\varphi$$
$$\alpha ::= b \mid \neg\alpha \mid \alpha\ \vee\ \alpha$$

where $p \in \Phi_p$, φ is a state formula, α is an action formula, and $b \in \Phi_a$ is an atomic action proposition.

Definition 5.2 (Model of ARCTL)**.** A model $\mathrm{M}_{AR} = (S_{AR}, I_{AR}, AC_{AR}, T_{AR}, VS_{AR}, VA_{AR})$ is a tuple where S_{AR} is a set of global states; $I_{AR} \subseteq S_{AR}$ is a set of initial states; AC_{AR} is a set of actions; $T_{AR} \subseteq S_{AR} \times AC_{AR} \times S_{AR}$ is a labeled transition relation; $VS_{AR} : S_{AR} \to 2^{\Phi_p}$ is an interpretation for atomic propositions; $VA_{AR} : AC_{AR} \to 2^{\Phi_a}$ is an interpretation for atomic action propositions.

To define the semantics of ARCTL [75], we need to define the α-restriction of M_{AR} as follows $\mathrm{M}^{\alpha}_{AR} = (S_{AR}, I_{AR}, AC_{AR}, T^{\alpha}_{AR}, VS_{AR}, VA_{AR})$ where T^{α}_{AR} is a transition relation such that $(s,a,s') \in T^{\alpha}_{AR}$ iff $(s,a,s') \in T_{AR}$ and $a \models \alpha$ wherein $\models$ is defined as follows:

- $a \models b$ iff $b \in VA_{AR}(a)$;
- $a \models \neg\alpha$ iff not $(a \models \alpha)$ and;
- $a \models \alpha \vee \alpha'$ iff $a \models \alpha$ or $a \models \alpha'$.

The α-restriction motivates us to concentrate each time on specific transitions whose labels hold on a given action formula. Consequently, when we want to verify a formula, we consider the relevant transitions only. In this perspective, $\Pi^{\alpha}(s)$ defines the set of paths whose actions satisfy an action formula α which starts at state s.

By omitting the semantics of Boolean connectives and propositional atoms, the satisfaction relation $(\mathrm{M}^{\alpha}_{AR}, s) \models \varphi$ is given as follows [75]:

- $(\mathrm{M}^{\alpha}_{AR}, s) \models E_{\alpha} X \varphi$ iff $\exists\ \pi \in \Pi^{\alpha}(s)$ and $\pi(1) \models \varphi$,

- $(\mathrm{M}^{\alpha}_{AR}, s) \models E_{\alpha}(\varphi\ U\ \psi)$ iff $\exists\ \pi \in \Pi^{\alpha}(s)$ s.t. for some $k \geq 0, \pi(k) \models \psi$ and $\pi(j) \models \varphi$ for all $0 \leq j < k$,

- $(\mathrm{M}^{\alpha}_{AR}, s) \models\ E_{\alpha} G \varphi$ iff $\exists\ \pi \in \Pi^{\alpha}(s)$ such that $\pi(k) \models \varphi$ for all $k \geq 0$.

We define the reduction process as follows: given a CTLKC^{+} model $\mathfrak{M} = (S, I, R_t, \{\approx_i | i \in \mathrm{A}\}, \{\approx_{i \to j} | (i, j) \in \mathscr{A}^2\}, \mathscr{V})$ and a CTLKC^{+} formula φ, we need to define an α-restricted ARCTL model $\mathrm{M}^{\alpha}_{AR} = \mathscr{H}(\mathfrak{M})$ and an ARCTL formula $\varphi = \mathscr{H}(\varphi)$ using a transformation function $\mathscr{H}$ such that $\mathfrak{M} \models \varphi$ iff $\mathscr{H}(\mathfrak{M}) \models \mathscr{H}(\varphi)$.

The model $\mathscr{H}(\mathfrak{M})$ is defined as an ARCTL model $\mathrm{M}^{\alpha}_{AR} = (S_{AR}, I_{AR}, AC_{AR}, T^{\alpha}_{AR}, VS_{AR}, VA_{AR})$ as follows:

- $S_{AR} = S$;

- $I_{AR} = I$;

- $VS_{AR} = \mathscr{V}$;

- We define the set of atomic action propositions Φ_a as follows. Each relation in $\mathfrak{M}$ is translated into a labeled transition in the transformed model $\mathscr{H}(\mathfrak{M})$ (i.e., M^{α}_{AR}). So doing provides us with four types of actions: one for the transition relation which already exists in the model; one for the social accessibility relation $\approx_{i \to j}$ to capture the semantics of commitment; one for the epistemic accessibility relation $\approx_i$

to capture the semantics of knowledge; the last one from the symmetric closure of the social accessibility relation $\approx_{i\to j}$ to capture the semantics of fulfillment. Thus, $\Phi_a = \{\varepsilon, \alpha_{1\to 1}, \alpha_{1\to 2}, \ldots, \alpha_{n\to n}\} \cup \{\beta_1, \beta_2, \ldots, \beta_n\} \cup \{\gamma_{1\to 1}, \gamma_{1\to 2}, \ldots, \gamma_{n\to n}\}$. Consequently, the set AC_{AR} of actions is defined as follows: $AC_{AR} = \{\alpha^o, \alpha^{11}, \alpha^{12}, \ldots, \alpha^{nn}\} \cup \{\beta^1, \beta^2, \ldots, \beta^n\} \cup \{\gamma^{11}, \gamma^{12}, \ldots, \gamma^{nn}\}$ where α^o is the action labeling transition defined from the transition relation R_t. α^{ij} is the action labeling transition defined the social accessibility relation $\approx_{i\to j}$, β^i is the action labeling the transition obtained from the epistemic accessibility relation $\approx_i$, and γ^{ij} is the action labeling the symmetric transition added when there exists a transition labeled with α^{ij} and needed to define transformation of the formula $Fu(C_{i\to j}\varphi)$;

- The function VA_{AR} is defined as follows:

  1. If $\alpha^o \in AC_{AR}$, then $VA_{AR}(\alpha^o) = \{\varepsilon\}$,
  2. $VA_{AR}(\alpha^{ij}) = \{\alpha_{i\to j}\}$ for $1 \leq i \leq n$ and $1 \leq j \leq n$,
  3. $VA_{AR}(\beta^i) = \{\beta_i\}$ for $1 \leq i \leq n$,
  4. $VA_{AR}(\gamma^{ij}) = \{\gamma_{i\to j}\}$ for $1 \leq i \leq n$ and $1 \leq j \leq n$.

- The labeled transition relation T^{α}_{AR} merges the temporal labeled transition R_t, the accessibility relations $\approx_{i\to j}$ and $\approx_i$, and the symmetric closure of the social accessibility relation $\approx_{i\to j}$ as follows,

  1. $(s, \alpha^o, s') \in T^{\varepsilon}_{AR}$ if $(s, s') \in R_t$;
  2. $(s, \alpha^{ij}, s') \in T^{\alpha_{i\to j}}_{AR}$ if $s \approx_{i\to j} s'$;
  3. $(s, \beta^i, s') \in T^{\beta_i}_{AR}$ if $s \approx_i s'$; and
  4. $(s, \gamma^{ij}, s') \in T^{\gamma_{i\to j}}_{AR}$ if $(s', \alpha^{ij}, s) \in T^{\alpha_{i\to j}}_{AR}$.

  Hereafter, let us define $\mathscr{H}(\varphi)$ by induction on the form of the CTLKC$^+$ formula φ.

1. $\mathscr{H}(p) = p$, if $p \in \Phi_p$;

2. $\mathscr{H}(\neg\varphi) = \neg\mathscr{H}(\varphi)$;

3. $\mathscr{H}(\varphi \vee \psi) = \mathscr{H}(\varphi) \vee \mathscr{H}(\psi)$;

4. $\mathscr{H}(EX\varphi) = E_{\varepsilon}(X\mathscr{H}(\varphi))$;

5. $\mathscr{H}(E(\varphi\ U\ \psi)) = E_{\varepsilon}(\mathscr{H}(\varphi)\ U\ \mathscr{H}(\psi))$;

6. $\mathscr{H}(EG\varphi) = E_{\varepsilon}(G\mathscr{H}(\varphi))$;

7. $\mathscr{H}(K_i\varphi) = A_{\beta_i}(X\mathscr{H}(\varphi))$;

8. $\mathscr{H}(C_{i\to j}\varphi) = A_{\alpha_{i\to j}}(X\mathscr{H}(K_i\varphi \wedge K_j\varphi))$;

9. $\mathscr{H}(Fu(C_{i\to j}\varphi)) = E_{\gamma_{i\to j}}(X\mathscr{H}(C_{i\to j}\varphi)) \vee$
$E_{\beta_i}(X\mathscr{H}(E_{\gamma_{i\to j}}(X\mathscr{H}(C_{i\to j}\varphi)))) \vee$
$E_{\beta_j}(X\mathscr{H}(E_{\gamma_{i\to j}}(X\mathscr{H}(C_{i\to j}\varphi)))).$

Therefore, we can verify CTLKC$^+$ formulas by verifying their transformed form in ARCTL using the extended NuSMV tool [61]. Figure 5.3 depicts an example illustrating the transformation function $\mathscr{H}$.

In this figure, the CTLKC$^+$ model $\mathfrak{M}$, on the left side of the figure, will be transformed into the ARCTL model M^{α}_{AR}, on the right side of the figure. In fact, the model $\mathfrak{M}$ consists of two global states s_0 and s_1. The state s_1 is socially accessible from the state s_0 (i.e., $s_0 \approx_{i\to j} s_1$). Furthermore, the state formulas $K_i\varphi$ and $K_j\varphi$ hold on s_1 (i.e., $(\mathfrak{M}, s_1) \models K_i\varphi$, $(\mathfrak{M}, s_1) \models K_j\varphi$). Thus, according to the semantics, $(\mathfrak{M}, s_0) \models C_{i\to j}\varphi$. Since the commitment modality holds on s_0 (i.e., $(\mathfrak{M}, s_0) \models C_{i\to j}\varphi$) and s_1 is socially accessible from s_0 (i.e., $s_0 \approx_{i\to j} s_1$) then, according to the semantics, the fulfillment modality will hold on s_1 (i.e., $(\mathfrak{M}, s_1) \models Fu(C_{i\to j}\varphi)$). Moreover, the state s_1 is epistemically accessible from

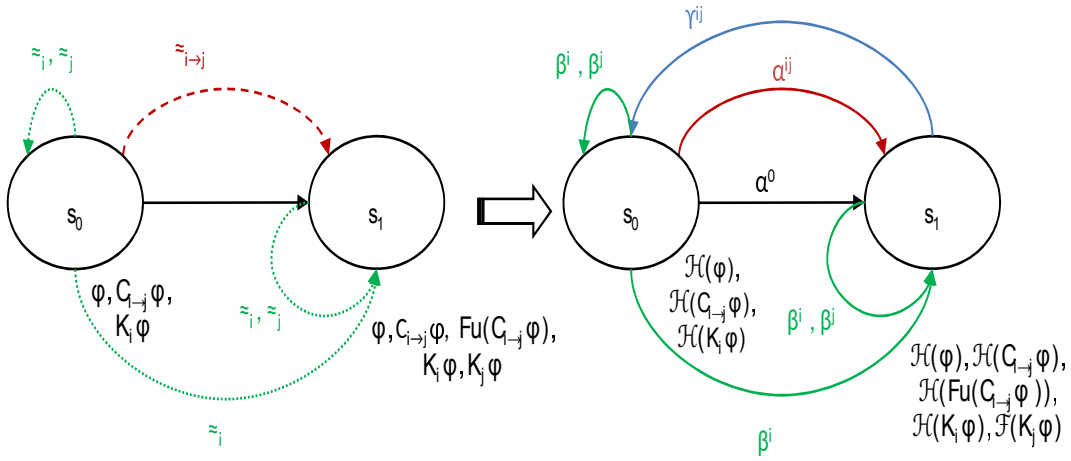

Figure 5.3: Example of the transformation function $\mathscr{H}$

the state s_0 (i.e., $s_0 \approx_i s_1$) and φ holds on both s_0 and s_1. Consequently, $(\mathfrak{M}, s_0) \models K_i\varphi$. Using the transformation technique described above, the model $\mathfrak{M}$ will be transformed into the ARCTL model M^{α}_{AR} as follows. The transition relation (i.e., $(s_0, s_1) \in R_t$) is transformed into a labeled transition (s_0, α^o, s_1). Furthermore, the social accessibility relation ($\approx_{i\to j}$) between the states s_0 and s_1 is transformed into a labeled transition (s_0, α^{ij}, s_1). Moreover, the epistemic accessibility relation ($\approx_i$) between the states s_0 and s_1 is transformed into a labeled transition (s_0, β^i, s_1). The symmetric closure of the social accessibility relation ($\approx_{i\to j}$) between the states s_0 and s_1 is transformed into a labeled transition (s_1, γ^{ij}, s_0). Finally, each state formula in CTLKC$^+$ will be transformed into ARCTL formula using the transformation function $\mathscr{H}$. In fact, the formulas φ, $C_{i\to j}\varphi$, $Fu(C_{i\to j}\varphi)$, $K_i\varphi$ and $K_j\varphi$ will be transformed into $\mathscr{H}(\varphi)$, $\mathscr{H}(C_{i\to j}\varphi)$, $\mathscr{H}(Fu(C_{i\to j}\varphi))$, $\mathscr{H}(K_i\varphi)$ and $\mathscr{H}(K_j\varphi)$ respectively.

The following theorem proves the soundness of our reduction from CTLKC$^+$ to ARCTL.

Theorem 5.2 (Soundness of $\mathscr{H}$)**.** *Let $\mathfrak{M}$ and φ be respectively a CTLKC$^+$ model and formula and let $\mathscr{H}(\mathfrak{M})$ and $\mathscr{H}(\varphi)$ be the corresponding model and formula in ARCTL.*

We have $\mathfrak{M} \models \varphi$ *iff* $\mathscr{H}(\mathfrak{M}) \models \mathscr{H}(\varphi)$.

Proof. We prove this theorem by induction on the structure of the formula φ. All the cases are straightforward, except the following three cases:

- $\varphi = K_i\psi$. We have $(\mathfrak{M}, s) \models K_i\psi$ iff $(\mathfrak{M}, s') \models \psi$ for every $s' \in S$ such that $s \approx_i s'$. Consequently, $(\mathfrak{M}, s) \models K_i\psi$ iff $(\mathscr{H}(\mathfrak{M}), s') \models \mathscr{H}(\psi)$ for every $s' \in S_{AR}$ such that $(s, \beta^i, s') \in T_{AR}^{\beta^i}$. By semantics of A and X, we obtain $(\mathscr{H}(\mathfrak{M}), s) \models A_{\beta_i}(X\mathscr{H}(\psi))$.

- $\varphi = C_{i\to j}\psi$. We have $(\mathfrak{M}, s) \models C_{i\to j}\psi$ iff $(\mathfrak{M}, s') \models K_i\psi \wedge K_j\psi$ for every $s' \in S$ such that $s \approx_{i\to j} s'$. Consequently, $(\mathfrak{M}, s) \models C_{i\to j}\psi$ iff $(\mathscr{H}(\mathfrak{M}), s') \models \mathscr{H}(K_i\psi \wedge K_j\psi)$ for every $s' \in S_{AR}$ such that $(s, \alpha^{ij}, s') \in T_{AR}^{\alpha_{i\to j}}$. By semantics of A and X, we obtain $(\mathscr{H}(\mathfrak{M}), s) \models A_{\alpha_{i\to j}}(X\mathscr{H}(K_i\psi \wedge K_j\psi)$.

- $\varphi = Fu(C_{i\to j}\psi)$. We have $(\mathfrak{M}, s) \models Fu(C_{i\to j}\psi)$ iff (1) $(\mathfrak{M}, s_1) \models C_{i\to j}\psi$ for a state $s_1 \in S$ such that $s_1 \approx_{i\to j} s$; or (2) $(\mathfrak{M}, s_2) \models Fu(C_{i\to j}\psi)$ for a state $s_2 \in S$ such that $s_2 \approx_i s$; or (3) $(\mathfrak{M}, s_3) \models Fu(C_{i\to j}\psi)$ for a state $s_3 \in S$ such that $s_3 \approx_j s$.

  Consequently, $(\mathfrak{M}, s) \models Fu(C_{i\to j}\psi)$ iff (1) $(\mathscr{H}(\mathfrak{M}), s_1) \models \mathscr{H}(C_{i\to j}\psi)$ for $s_1 \in S_{AR}$ such that $(s, \gamma^{ij}, s_1) \in T_{AR}^{\alpha_{i\to j}}$; or (2) $(\mathscr{H}(\mathfrak{M}), s_2) \models \mathscr{H}(Fu(C_{i\to j}\psi))$ for $s_2 \in S_{AR}$ such that $(s_2, \beta^i, s) \in T_{AR}^{\beta_i}$; or (3) $(\mathscr{H}(\mathfrak{M}), s_3) \models \mathscr{H}(Fu(C_{i\to j}\psi))$ for $s_3 \in S_{AR}$ such that $(s_3, \beta^j, s) \in T_{AR}^{\beta_j}$. Using the semantics of E and X, we obtain: $(\mathscr{H}(\mathfrak{M}), s) \models$

  $E_{\gamma_{i\to j}}(X\mathscr{H}(C_{i\to j}\psi)) \vee E_{\beta_i}(X\mathscr{H}(Fu(C_{i\to j}\psi))) \vee E_{\beta_j}(X\mathscr{H}(Fu(C_{i\to j}\psi)))$.

Therefore, the theorem. □

By developing the reduction-approaches in Sections 5.2 and 5.3, we are answering the sixth research question [Q6].

5.4 Complexity Analysis

In this section, we will first analyze the time complexity of model checking CTLKC$^+$ with regard to the size of the explicit model $\mathfrak{M}$ and length of the formula to be checked. Thereafter, we will analyze the space complexity of model checking CTLKC$^+$ for concurrent programs with respect to the size of the components of these programs and length of the formula (Phase 3 of Figure 5.1).

5.4.1 Time Complexity

In this subsection, we will prove that model checking CTLKC$^+$ is P-complete, so it can be done in polynomial running time in the size of the model and length of the formula.

Theorem 5.3. *The time complexity for model checking CTLKC$^+$ is $O(|\mathfrak{M}| \times |\psi|)$ where $|\mathfrak{M}|$ is the size of the model and $|\psi|$ is the length of the formula.*

Proof. CTLKC$^+$ extends CTL with modalities for knowledge, commitments and fulfillment. Further, in [23] Clarke proved that the problem of model checking CTL is linear in the size of the model and the length of the formula, which gives us the lower bound. We just need to analyze the time complexity of the reduction procedure presented in Section 5.3. In this procedure, it is easy to show that reducing CTLKC$^+$ model $\mathfrak{M}$ into ARCTL model $\mathrm{M}^{\alpha}_{\mathrm{AR}}$ can be done in linear running time in the size of the model, as procedure steps are simply performing transformation operations on states and transitions. For transforming CTLKC$^+$ formula into ARCTL formula, this can be shown as follows. First, it is known form [91] that the reachability test between two global states s and s', s.t. $(s,s') \in R_t$, can be done in linear time. Moreover, Step 7 in defining $\mathscr{H}(\varphi)$ calls the procedure recursively on the subformula φ of the formula $\psi = K_i\varphi$. Further, step 8 calls the procedure recursively on the subformulas $K_i\varphi$ and $K_j\varphi$ of the formula $\psi = C_{i\to j}\varphi$. Finally, step 9 calls

the procedure recursively on the subformula $C_{i\to j}\varphi$ of the formula $\psi = Fu(C_{i\to j}\varphi)$. The procedure is recursively called until a CTL subformula is reached. Consequently, the depth of the recursion is limited to the length of the formulas ψ which is linear. Therefore, we conclude that the complexity of the proposed transformation procedure is linear in both the size of the model $|\mathfrak{M}|$ and the length of the formula $|\psi|$.

□

Theorem 5.4. *The model checking problem for CTLKC$^+$ is P-complete.*

Proof. `Membership` in P (i.e., upper bound) follows from Theorem 5.3.

`Hardness` in P (i.e., lower bound) follows by a reduction from model checking CTL proved to be P-complete in [82]. □

5.4.2 Space Complexity

The methodology of analyzing the space complexity of our reduction technique is as follows. We first analyze the computational complexity of model checking ARCTL in concurrent programs and show that our reduction procedure is polynomial with respect to the size of those programs. Consequently, the proposed verification technique has the same complexity as the one of model checking the original logic, namely ARCTL in concurrent programs. In the rest of the chapter, $\preceq_{\log}$ denotes log-space reduction and $\preceq_{\mathrm{p}}$ denotes polynomial-space reduction. Moreover, let $Mod(L)$ be the model of a logical language L and $Con(L)$ the concurrent program model of the language L.

Lemma 5.1. *Model checking ARCTL for concurrent programs is PSPACE-hard with respect to the size of the local processes of these programs and the length of the formula being checked.*

Proof. ARCTL is an extension of CTL. In fact, any model of CTL can be translated to a model of ARCTL by simply labeling the transitions by ε, the empty action symbol and

specify the interpretation of atomic actions by associating to ε an empty atomic action formula, which always holds for any action a. Hence, we can imagine a deterministic Turing machine that looks at the input CTL model and writes in its output tape, one by one, the same states and transitions, where each transition will be labeled by ε. Thus, the Turing machine can compute this reduction in space $O(\log n)$ where n is the size of the input CTL model as there is no need to store the whole model beforehand. So, we obtain $Mod(\text{CTL}) \preceq_{\log} Mod(\text{ARCTL})$. Since the explicit model is obtained as the product of the components of a concurrent program and this product is at most exponentially larger than the program, the state space is exponential in the length of the program. Consequently, $Con(\text{CTL}) \preceq_{\text{p}} Con(\text{ARCTL})$. Since any formula of CTL is also a formula of ARCTL (existential CTL formulas $E\varphi$ are $E_\varepsilon\varphi$ ARCTL formulas[2]) and Model checking CTL for concurrent programs is proven to be PSPACE-hard in [58], the hardness of model checking ARCTL in PSPACE follows. □

To prove the upper bound complexity of ARCTL model checking in concurrent programs, we use reduction to GCTL* a generalized branching temporal logic with actions [14].

Lemma 5.2. *Model checking ARCTL for concurrent programs with respect to the size of the local processes of these programs and the length of the formula being checked is in PSPACE.*

Proof. From Definitions 5.2 and 5.1, the models of ARCTL and GCTL* are similar, so we can easily observe that a deterministic Turing machine can reduce the model of ARCTL to the model of GCTL* by simply looking at the input and writing in the output tape the same states, transitions, set of actions, and state and action valuation functions one by one. Using a similar argument as in the proof of Lemma 5.1, we obtain $Mod(\text{ARCTL}) \preceq_{\log}$

[2]The soundness of this transformation is trivial, so the proof is omitted.

$Mod(\text{GCTL}^*)$ and thus $Con(\text{ARCTL}) \preceq_p Con(\text{GCTL}^*)$. Let $\mathscr{H}$ be the reduction function from ARCTL to GCTL* such that $M^{\alpha}_{AR} \models \varphi$ iff $\mathscr{H}(M^{\alpha}_{AR}) \models \mathscr{H}(\varphi)$. $\mathscr{H}(M^{\alpha}_{AR})$ is a GCTL* model where $S_{GC} = S_{AR}$, $I_{GC} = I_{AR}$, $AC_{GC} = AC_{AR}$, $T_{GC} = T_{AR}$, $VS_{GC} = VS_{AR}$, and $VA_{GC} = VA_{AR}$. For the common formulas in ARCTL and GCTL*, $\mathscr{H}$ is defined in the same way as the function $\mathscr{F}$. The reductions of the non-common formulas are as follows:

- $\mathscr{H}(E_{\alpha}X\varphi) = E(G\alpha \wedge X\varphi)$
- $\mathscr{H}(E_{\alpha}(\varphi\ U\ \psi)) = E(G\alpha \wedge \varphi\ U\ \psi)$
- $\mathscr{H}(E_{\alpha}G\varphi) = E(G\alpha \wedge G\varphi)$

Soundness of this reduction follows from the semantics of the two languages. In ARCTL, for $E_{\alpha}X\varphi$ to be satisfied, we need to find a path whose transition actions should satisfy α and the next state of the path satisfies φ. In GCTL*, this means the action formula α should be global, so that it is satisfied by all the transition actions and $X\varphi$ is satisfied in the next state. The two other formulas follow the same reasoning. GCTL* turns out to be then more expressive than ARCTL. It is obvious that a deterministic Turing machine can compute these reductions polynomially in the size of the formulas. Consequently, ARCTL $\preceq_p$ GCTL*. Thus, the membership in PSPACE follows from the facts that model checking GCTL* is PSPACE-complete from [36], so in PSPACE, and complexity hierarchical classes are inclusive.

□

Theorem 5.5. *Model checking ARCTL is PSPACE-complete for concurrent programs with respect to the size of the local processes of these programs and the length of the formula being checked.*

Proof. The proof is direct from Lemmas 5.1 and 5.2. □

Lemma 5.3. *Model checking CTLKC$^+$ for concurrent programs with respect to the size of the local processes of these programs and the length of the formula being checked is PSPACE-hard.*

Proof. CTL is a subset of CTLKC$^+$ as CTLKC$^+$ extends CTL with additional operators of knowledge, commitment, and fulfilment. Thus, any model of CTL is a model of CTLKC$^+$ and all CTL formulas are also CTLKC$^+$ formulas. Consequently, the result follows from CTL $\preceq_p$ CTLKC$^+$ and model checking CTL for concurrent programs is PSPACE-hard with respect to the size of the local processes of these programs and the length of the formula being checked [58]. □

Lemma 5.4. *Model checking CTLKC$^+$ for concurrent programs with respect to the size of the local processes of these programs and the length of the formula being checked is in PSPACE.*

Proof. In Section 5.3, a reduction from CTLKC$^+$ to ARCTL has been described. We can imagine a deterministic Turing machine that looks in the input tape at the CTLKC$^+$ model and writes in the output tape, one by one, all the states and for each transition in the input tape, a transition labeled by α^o is written in the output tape. Moreover, for each $\approx_i$ accessibility relation, a transition labeled by β^i is written in the output tape. Finally, for each $\approx_{i\to j}$ accessibility relation in the input tape, a transition labeled by α^{ij} and a reverse transition labeled by γ^{ij} are written in the output tape. Since there is no need to store the whole model, this reduction can be computed in space $\log(n)$ where n is the size of the input model. Consequently, $Mod(\text{CTLKC}^+) \preceq_{\log} Mod(\text{ARCTL})$ and thus $Con(\text{CTLKC}^+) \preceq_p Con(\text{ARCTL})$ (using the same argument as in the proof of Lemma 5.1). With regard to the formulas, the transformation described in Section 5.3 is clearly polynomial in the size of the formula. As a result, CTLKC$^+$ $\preceq_p$ ARCTL. Therefore, the result flows from Theorem 5.1. □

From Lemmas 5.3 and 5.4, the following theorem follows.

Theorem 5.6. *Model checking CTLKC$^+$ is PSPACE-complete for concurrent programs with respect to the size of the local processes of these programs and the length of the formula being checked.*

By computing the time and space complexity of the CTLKC$^+$ logic, we answer the seventh research question [Q7].

5.5 Case Study

In this section, we implement the reduction techniques presented in Sections 5.2 and 5.3 on top of the CWB-NC and extended NuSMV model checkers to verify the interaction between knowledge and commitment in MASs (Phase 4 of Figure 5.1). The case study for which we have been able to carry out this motivation is the NetBill protocol [90]. This protocol has been applied in many research work to show how to specify commitment protocols in MASs (see for example [35, 36, 104]).

5.5.1 Modeling the NetBill Protocol

The NetBill protocol is developed for buying and selling encrypted software goods on the Internet [28, 90]. This protocol encompasses two interacting agents: the customer (Cus) and the merchant (Mer) [67, 103]. To model the NetBill protocol using our transformation tool, the designer should insert the local states of each agent, their actions, which agent performs which action, what commitments, if any, hold in each state and where each of those commitments is fulfilled. Based on these specifications, the global system is thereafter generated automatically with the relevant epistemic and social accessibility relations. In this protocol, the actions performed by agents are labeling the edges and modeled using the notation *Action* and *'Action* to express, respectively, the sending and receiving of messages.

The use of *Action* and *'Action* is to only distinguish who is performing the action. Thus, in Figure 5.4 (the Customer model), when the customer agent performs an action, we model it by the name of the action directly (e.g., *Request*, *Accept*, *Reject*, etc.). On the other hand, when the merchant agent performs an action, we model it by *'Action* (e.g., *'Quote*, *'Refund*, *'Receipt*, etc.). In a similar way, in Figure 5.5, the notation *Action* is used for merchant's actions and the notation *'Action* is reserved for customer's actions.

Figure 5.4 depicts the customer model. In this model, the customer (*Cus*) requests a quote (i.e., action *Request*) from the merchant (*Mer*) for a certain goods at state c_0. The merchant replies to this request by presenting a quote for the requested goods at state c_1 (i.e., action *'Quote*). The customer at state c_2 can reject the offer (i.e., action *Reject*) and the protocol moves to the initial state c_0 after passing the failure state c_4, or accept the offer (i.e., action *Accept*), which means the customer commits to send the payment to the merchant at state c_3, formally, $C_{Cus \to Mer}\ Pay$ where *Pay* is the content of the commitment and the customer should be aware of its commitment (i.e., $K_{Cus}(C_{Cus \to Mer} Pay)$). After that, if the customer accepts the offer, then it has two choices: (1) send the payment to the merchant at state c_5, which means that the customer fulfills its commitment (i.e., action *Payment*), formally, $Fu(C_{Cus \to Mer}\ Pay)$; or (2) violates its commitment (action *notPayment*) and the protocol moves to state c_0 after passing state c_4. When the merchant receives the payment, it has two choices: (1) not delivering the goods (i.e., action *'notDelivery*) and the protocol moves to the initial state c_0 after refunding the customer (i.e., action *'Refund*) at state c_8; or (2) delivering the requested goods (i.e., action *'Delivery*) to the customer at state c_7, and then the protocol moves to the acceptance state c_9 after sending the receipt (i.e., action *'Receipt*) to the customer, and finally the protocol moves to c_0. Moreover, the atomic proposition p in states c_1, c_5, c_6 and c_7 is indicating that the customer is waiting for an action from the merchant. For instance, in state c_1, the customer is waiting to receive

the quote and in state c_7, it is waiting the receipt. The atomic proposition *Init* in state c_0 is simply indicating that this state is the initial state. These two propositions will be used latter to formulate the deadlock property (see Section 5.5.3).

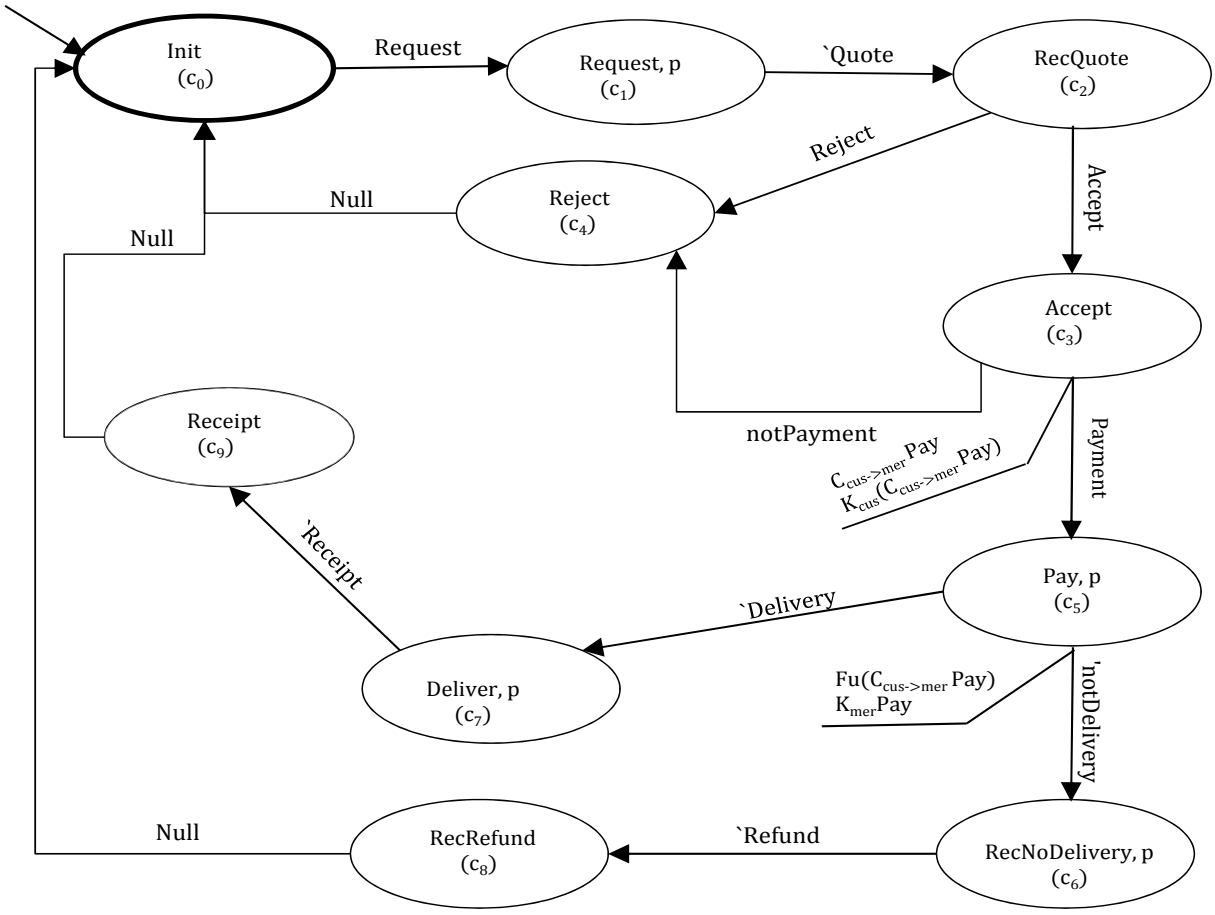

Figure 5.4: Customer model

Figure 5.5 depicts the Merchant model. In this model, if the merchant (*Mer*) receives the payment from the customer (*Cus*) (i.e., action *'Payment*), after presenting the quote (i.e., action *Quote*), then it commits towards the customer to deliver the required goods at state m_5 (i.e., $C_{Mer \rightarrow Cus}Deliver$) where *Deliver* is the content of the commitment. Furthermore, the merchant should be aware of this commitment at the same state (m_5) (i.e.,

$K_{Mer}(C_{Mer \to Cus} Deliver)$). Thereafter, if the merchant delivers the goods (i.e., action *Delivery*), then it fulfills its commitment at state m_7 (i.e., $Fu(C_{Mer \to Cus} Deliver)$) and it should be aware of the content of the commitment (i.e., $K_{Mer} Deliver$). Then the protocol moves to the acceptance state m_9 after sending the receipt to the customer (i.e., action *Receipt*), and finally the protocol moves to state m_0, or it has to refund the customer (i.e., action *Refund*) in case the delivery fails (i.e., action *notDelivery*). Similar to the customer model, the atomic proposition q in states m_0, m_2 and m_3 is indicating that the merchant is waiting for a customer's action. The atomic proposition *Init* in state m_0 has the same meaning as in the customer model.

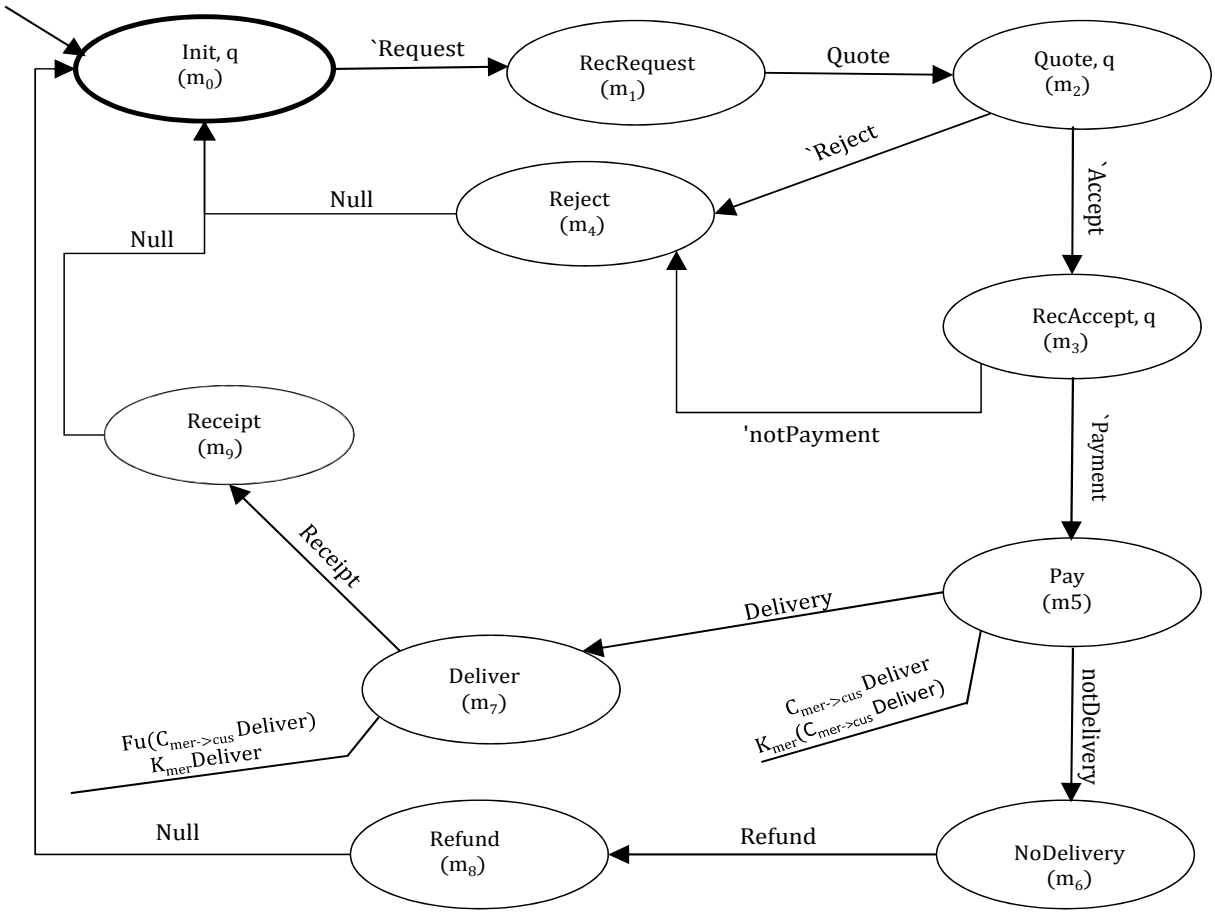

Figure 5.5: Merchant model

5.5.2 Implementation

We have implemented the reduction technique (that transforms the problem of model checking CTLKC$^+$ into the problem of model checking GCTL*) presented in Section 5.2 on top of the automata-based model checker CWB-NC. Furthermore, we have implemented the reduction technique (that transforms the problem of model checking CTLKC$^+$ into the problem of model checking ARCTL) presented in Section 5.3 on top of the extended NuSMV model checker.

CWB-NC (Concurrency WorkBench of the New Century) is an automata-based model checker. It has been used to model check different types of temporal logics (e.g., CTL, CTL*, and GCTL*). This tool was developed at Stony Brook in the middle of 1990s. The interesting feature of this tool is that it adopts an on-the-fly technique. In this technique, the algorithm only searches the part of the state space relevant to a certain formula (i.e., the state space does not need to be constructed before). Recently, CWB-NC has been used to verify commitment protocols [10, 34]. To use CWB-NC, we exploited the language CCS [71] to encode the specification of the NetBill protocol and formalized by the model $\mathfrak{M} = (S, I, R_t, \{\approx_i | i \in \mathscr{A}\}, \{\approx_{i \to j} | (i, j) \in \mathscr{A}^2\}, \mathscr{V})$.

The two agents customer (Cus) and merchant (Mer) of the NetBill protocol are encoded in the language CCS by defining each agent as a set of processes. Each process contains the local states of an agent. Technically, the commitment, fulfillment, and knowledge states of each agent are defined as variables in the `proc` statement. Details about encoding the NetBill protocol using CWB-NC can be found on the open source projects web site SourceForge[3].

On the other hand, NuSMV [22] is a symbolic-based model checker used to verify both Linear Temporal Logic (LTL) [77] and Computation Tree Logic (CTL) [41]. It has

[3]https://sourceforge.net/projects/knowledgencommitment/files/NuSMV/?.

been applied to model check various applications of MASs (e.g., Web-based services and commitment protocols [36]). However, the original version of NuSMV does not support verifying actions and knowledge properties in MASs. To overcome this limitation, the extended version of NuSMV is developed [75]. Extended version of NuSMV can verify ARCTL formulas and epistemic properties. In NuSMV, models are encoded using a language called (extended SMV) [75]. To model ARCTL actions, Pecheur et al. [75] and Lomuscio et al. [61] proposed to use existing NuSMV input variables. This model checker can be downloaded online [4].

To implement our model $\mathfrak{M}$ and CTLKC$^+$ specifications using extended NuSMV, we developed a JAVA Transformation Tool[5] (JTT) as shown in Figure 5.6. This tool accepts, as input, a CTKLC$^+$ model $\mathfrak{M}$ and its specifications, then automatically generates the equivalent extended NuSMV model and ARCTL specification using the proposed reduction technique presented in Section 5.3. After that, the extended NuSMV model checker is used to verify the extended NuSMV model and ARCTL specifications. In this approach, we only need to provide the tool with the model and specifications. After that, everything will be done automatically.

Figure 5.7 illustrates the main components of coding the NetBill protocol in case of one merchant and one customer as depicted in Figures 5.4 and 5.5. We have uploaded the entire verification code on the open source projects web site SourceForge[6].

[4] http://www.kenmcmil.com/

[5] https://github.com/Marooned202/jtl

[6] http://sourceforge.net/projects/knowledgencommitment/files/NuSMV/?.

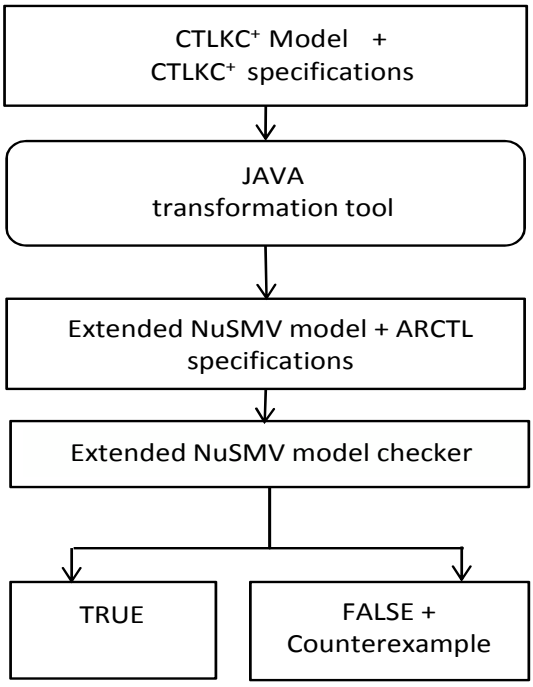

Figure 5.6: Verification workflow

5.5.3 Verification Results

Our verification experiments were performed using a Toshiba Protégé computer with 2.00 GHz Intel Core Duo T6400 processor and 3GB memory under 64-bit Windows Vista Operating System. We used extended NuSMV (NuSVM-ARCTL-TLACE[7]), which is based on a modified version of NuSMV 2.2.2 that is able to model check ARCTL formulas. We have reported the results of 8 experiments in Table 5.1. In this perspective, we started our experiments with only two agents: the merchant (*Mer*) and customer (*Cus*) that interact in a MAS to reason about their knowledge and commitments. In the second experiment, we have added one more customer. Moreover, in the rest of our experiments, we added a new agent each time up to 9 agents. In Table 5.1, number of reachable states (States), execution time (Time(sec)), and memory usage (Memory (MB)) are listed for various number of

[7]http://lvl.info.ucl.ac.be/Tools/NuSMV-ARCTL-TLACE

agents. We have noticed that the state space increases exponentially as the number of agents increases. However, memory usage increases only polynomially with the same number of agents, which shows the efficiency of our model checking approach when the system scales up (about $1.30413E+07$ states).

Table 5.1: Verification results of the NetBill protocol using extended NuSMV

Agents	States	Time (sec)	Memory (MB)
2	25	< 0.001	4.398
3	226	< 0.01	4.492
4	1,454	0.1	4.698
5	9,070	1.3	4.691
6	55,862	12.1	4.750
7	341,806	168.6	4.880
8	2,082,850	1753.5	4.969
9	$1.30413E+07$	27557.4	5.151

On the other hand, we also implemented the reduction technique in Section 5.2 and applied it on the NetBill protocol using the CWB-NC automata-based model checker. The verification results using this approach are shown in Table 5.2. From the scalability point of view, we notice that the automata-based approach reaches the state explosion problem relatively faster than the symbolic-based approach. As shown in Table 5.1, in the symbolic-based approach, with $1.30413E+7$ states and 9 agents, only 5.151 MB of memory is being used. However, in the automata-based approach in Table 5.2, with only $54,439$ states and 6 agents, a larger amount of memory (41.888 MB) is needed. It is worth noticing that both the number of states and memory usage in the automata-based technique (Table 5.2) increase exponentially as we add more agents to the system up to 6 agents, after that the model goes dramatically to the state explosion.

Another important motivation of this chapter is to use CTLKC$^+$ to verify properties of protocols that have interaction between knowledge and commitments. These properties usually express some protocol requirements (specifications) that should be met. In fact,

Table 5.2: Verification results of the NetBill protocol using CWB-NC

Agents	States	Transitions	Memory (MB)
2	37	98	5.112
3	223	1023	5.212
4	1369	10,352	6.708
5	9,079	77,377	13.652
6	54,439	570,500	41.888

there have been large number of properties introduced in the literature [9, 10, 19, 29, 39, 72, 97]. In this section, we check *Safety*, *Liveness*, *Reachability* and *Deadlock* properties in the NetBill protocol.

- `Safety property`.

  Safety property insists on preventing the occurrence of "bad" behaviors (situations) for a given modeled system (i.e., "Something bad will never occur"). For example, the case that the customer (*Cus*) sends the payment to the merchant (*Mer*) (i.e., fulfills its commitment), but the merchant does not know that. In this case, the merchant could ask the customer to send the payment again until the merchant becomes aware of that. This bad situation can be avoided using CTLKC$^+$ as follows:

  $\varphi_1 = AG\,\neg(Fu(C_{Cus\rightarrow Mer} sendPayment) \wedge AG(\neg K_{Mer} sendPayment))$.

  A similar formula is when the customer fulfills its commitments, but it turns out that it is not aware of:

  $\varphi_2 = AG\,\neg(Fu(C_{Cus\rightarrow Mer} sendPayment) \wedge AG(\neg K_{Cus} sendPayment))$.

  Another example of safety property is the case that, if a customer commits toward a merchant to send its payment, then the customer should be aware of that. This can be expressed as follows:

  $\varphi_3 = AG\,\neg((C_{Cus\rightarrow Mer} sendPayment) \wedge AG(\neg K_{Cus}(C_{Cus\rightarrow Mer} sendPayment)))$.

- `Liveness property`.

This property insists on achieving the system "good" situation in future (i.e., "Something good will eventually happen"). From the NetBill protocol, we can consider that case that: if the merchant commits to deliver the goods to the customer, it will eventually deliver them, as a good situation that should be obtained. This property is expressed as follows:

$\varphi_4 = AG(C_{Mer \to Cus} deliverGoods \to \text{EF}\ Fu(C_{Mer \to Cus} deliverGoods))$.

- `Reachability property`.

  This property highlights the fact that the system could reach a particular state (situation). For example, the merchant will eventually commit towards the customer to deliver the required goods, which should be reached from the initial state. This property can be expressed as follows:

  $\varphi_5 = \text{EF}\ C_{Mer \to Cus} deliverGoods$

- `Deadlock property`.

  As defined in [4], "*a deadlock occurs if the complete system is in a terminal state, although at least one component is in a (local) nonterminal state. The entire system has thus come to a halt, whereas at least one component has the possibility to continue to operate. A typical deadlock scenario occurs when components mutually wait for each other to progress*". That is, the system is in a deadlock state if it reaches a state in which no further progress could happen (halt state). To verify this property, we defined two atomic propositions in Figure 5.7: `DEF_cus.wait` and `DEF_mer.wait` to capture the cases where the customer is waiting for the merchant response and the merchant is waiting for the customer response (see Section 5.5.2). For example, one possible scenario of a deadlock that should be avoided in our system is the case that the customer is waiting for a quote from the merchant, while the merchant is waiting for a request from the customer at the same time, thus halting the entire system. This

property can be expressed in CTLKC$^+$ as follows: $\varphi_6 = \text{AG}\ \neg(p \wedge q)$

Moreover, since the final states in our models depicted in Figures 5.4 and 5.5 are also the initial states, we verify another instance of the deadlock property by checking that all computations are visiting the initial states (i.e., c_0 and m_0) in the future. To do so, we defined, in Figure 5.7, an atomic proposition `DEF_Init` which holds in the initial sates. This atomic proposition is represented by *Init* in Figures 5.4 and 5.5. Consequently, if this property is satisfied, then the system will not come to a halt, which entails that there is always a progress. This property can be expressed in CTLKC$^+$ as follows: $\varphi_7 = AG\ (EF\ (Init))$.

We notice that all the formulas hold in the model as shown in Figure 5.8, meaning that our approach is successful in expressing the protocol properties using CTLKC$^+$ and our reduction-based model checking techniques are working effectively.

5.6 Summary

In this chapter, we introduced two model checking techniques for verifying the logic of knowledge and communicative commitments, CTLKC$^+$. In those techniques, we developed a set of reduction rules to formally reduce the problem of model checking CTLKC$^+$ to the problem of model checking GCTL* and ARCTL respectively. Furthermore, we proved the soundness of the proposed reduction techniques. The complexity of model checking CTLKC$^+$ using our reduction techniques is addressed as well. After that, the effectiveness of the proposed techniques were evaluated through applying them on a real case study from the e-business domain, namely the NetBill protocol, and then implementing the reduction tools on top of the CWB-NC and extended NuSMV model checkers. We were successfully able to check some desirable protocol properties expressed in CTLKC$^+$.

In the next chapter, we will summarize the main contributions and present the future directions of this work.

```
MODULE main
VAR
cus  : process Customer(mer);
mer  : process Merchant(cus);
------------------------------
-- Atomic Propositions
------------------------------
DEFINE DEF_Pay := cus.state = c5 ;
DEFINE DEF_Deliver := cus.state = c7;
DEFINE DEF_cus.wait := (cus.state = c5 | cus.state = c6 | cus.state = c1 | cus.state = c7);
DEFINE DEF_mer.wait := (mer.state = m0 | mer.state =  m2 | mer.state = m3);
DEFINE DEF_Init := (cus.state = c0 | mer.state = m0);
-------------------------
-- Formulae
-----------------------
SPEC AG !(EAX(cus.action=Gamma_cus)(AAX(cus.action=Alpha_cus)(AAX (cus.action=Beta_cus)(DEF_Pay) &
AAX(mer.action=Beta_mer)(DEF_Pay))|EAX(cus.action=Beta_cus)(EAX(cus.action=Gamma_cus)(AAX(cus.action=Al
pha_cus)(AAX (cus.action=Beta_cus)(DEF_Pay) & AAX(mer.action=Beta_mer)(DEF_Pay))))
|EAX(mer.action=Beta_mer)(EAX(mer.action=Gamma_mer)(AAX(cus.action=Alpha_cus)(AAX
(cus.action=Beta_cus)(DEF_Pay) & AAX(mer.action=Beta_mer)(DEF_Pay)))))
&AG !AAX(mer.action=Beta_mer)(DEF_Pay));
….
----------------------------------------
-- The definition of Customer Agent (cus, mer)
----------------------------------------
MODULE Customer (arg1,arg2)
VAR state   : {c0,c1,c2,c3,c4,c5,c6,c7,c8,c9};
IVAR action : {Request, Accept, Reject, Null, notPayment, Payment, Alpha_cus, Beta_cus, Gamma_cus};
INIT (state = c0)
TRANS(next(state)= case
                  (state = c0 & arg1.action = Request) : c1;
                  (state = c1 & arg2.action = Quote) : c2;
                  (state = c2 & arg1.action = Accept) : c3;
                  (state = c2 & arg1.action = Reject) : c4;
                  (state = c3 & arg1.action = notPayment) : c4;
                  (state = c3 & arg1.action = Payment) : c5;
                  (state = c3 & arg1.action = Alpha_cus) : c5;
                  (state = c3 & arg1.action = Beta_cus) : c3;
                  (state = c4 & arg1.action = Null) : c0;
                  (state = c5 & arg2.action = Delivery) : c7;
                  (state = c5 & arg2.action = notDelivery) : c6;
                  (state = c5 & arg1.action = Beta_cus) : c5;
                  (state = c5 & arg1.action = Gamma_cus) : c3;
                  (state = c6 & arg2.action = Refund) : c8;
                  (state = c7 & arg2.action = Receipt) : c9;
                  (state = c8 & arg1.action = Null) : c0;
                  (state = c9 & arg1.action = Null) : c0;
esac)
----------------------------------------
-- The definition of Merchant Agent (mer,cus)
----------------------------------------
MODULE Merchant (arg1,arg2)
VAR state    : {m0,m1,m2,m3,m4,m5,m6,m7,m8,m9};
IVAR action  : {Quote, Delivery, notDelivery, Receipt, Beta_mer, Alpha_mer, Gamma_mer, Null, Refund};
INIT (state = m0)
TRANS(next(state)= case
                  (state = m0 & arg2.action = Request) : m1;
                  (state = m1 & arg1.action = Quote) : m2;
                  (state = m2 & arg2.action = Accept) : m3;
                  (state = m2 & arg2.action = Reject) : m4;
                  (state = m3 & arg2.action = notPayment) : m4;
                  (state = m3 & arg2.action = Payment) : m5;
                  (state = m4 & arg1.action = Null) : m0;
                  (state = m5 & arg1.action = Alpha_mer) : m7;
                  (state = m5 & arg1.action = Delivery) : m7;
                  (state = m5 & arg1.action = notDelivery) : m6;
                  (state = m5 & arg1.action = Beta_mer) : m5;
                  (state = m6 & arg1.action = Refund) : m8;
                  (state = m7 & arg1.action = Beta_mer) : m7;
                  (state = m7 & arg1.action = Receipt) : m9;
                  (state = m7 & arg1.action = Gamma_mer) : m5;
                  (state = m8 & arg1.action = Null) : m0;
                  (state = m9 & arg1.action = Null) : m0;
       esac)
```

Figure 5.7: Coding the NetBill protocol

Figure 5.8: Screenshot of verification results for the NetBill protocol

Chapter 6

Conclusions and Future Work

In this chapter, we conclude the thesis by highlighting the main contributions and listing the possible directions of future work.

6.1 Conclusions

In this thesis, we have developed a practical and formal approach that analyzes the interaction between knowledge and communicative social commitments in MASs from the semantics, soundness and completeness, model checking and complexity perspectives. In particular, the main contributions of this thesis are:

1. Developing a new consistent logic called the logic of knowledge and commitments (CTLKC^{+}) that captures the interaction between knowledge and social commitments in MASs. To do so, a new combined temporal logic, called CTLKC, is first introduced. The purpose of such a combination is to express and figure out some reasoning postulates merging both knowledge and commitments as they are currently defined in the literature. By analyzing the postulates, we identified some paradoxes that should

be addressed in any consistent logic combining these two modalities. Thus, to overcome and solve the paradoxes identified in CTLKC, we introduced CTLKC$^+$ which can be used to reason about knowledge, communicative social commitment and their interactions in a consistent manner [2].

2. Proving the soundness and completeness of CTLKC$^+$ using correspondence theory for modal logic. To do so, we developed a set of reasoning postulates that captures the interactions between knowledge and social commitments in MASs and corresponded them to certain classes of frames providing the required proofs. Consequently, we proved that the logic generated by any subset of these postulates is sound and complete with respect to the models that are based on the corresponding frames [1].

3. Introducing two transformation-based model checking techniques for verifying CTLKC$^+$ [2, 3]. In particular, we transformed the problem of model checking CTLKC$^+$ into the problem of model checking GCTL* (a generalized version of CTL* with action formulas) and ARCTL (the combination of CTL with action formulas). Concretely, we constructed a set of transformation rules to formally reduce the CTLKC$^+$ model into GCTL* and ARCTL models and the CTLKC$^+$ formulas into GCTL* and ARCTL formulas. Furthermore, we proved the soundness of the transformation techniques.

4. Developing a Java Transformation Tool (JTT)[1] that automatically performs the transformation process from the CTLKC$^+$ logic into the extended NuSMV logic. JTT accepts, as input, a CTLKC$^+$ model $\mathfrak{M}$ and its specifications, then generates the equivalent extended NuSMV model and ARCTL specification.

5. Computing the time and space complexity of the proposed model checking procedures [3]. This analysis confirms that the complexity of CTLKC$^+$ model checking

[1] https://github.com/Marooned202/jtl

for concurrent programs with respect to the size of the components of those programs and the length of the formula being checked is PSPACE-complete. From the time perspective, we proved that the complexity of the proposed approaches is P-complete with regard to the size of the model and length of the formula.

6. Implementing the proposed model checking techniques on top of the CWB-NC and extended NuSMV model checkers [3]. By so doing, we were successfully able to check some desirable properties for the NetBill protocol, a real case study from the e-business domain, expressed in CTLKC$^+$ and report verification results. The obtained results show the effectiveness of our model checking approaches when the system scales up.

6.2 Future Work

As future work, we intend to extend the proposed approach by addressing the following issues:

- Developing dedicated model checking algorithms for CTLKC$^+$ logic and implementing them on top of the MCMAS$^+$ symbolic model checker [7]. By so doing, we will be able to compare the verification results of both techniques (i.e., the reduction and model checking algorithms).

- Extending our java transformation tool to be a Graphical User Interface (GUI) tool and be able to perform transformation for more model checking problems.

- Integrating conditional social commitments in our approach. Thus, we will be able to study the interaction between knowledge and conditional social commitments from different perspectives.

- Introducing a new consistent logic for group social commitments. Therefore, we will be able to reason about many-to-one, one-to-many and many-to-many commitments.
- Investigating the interaction between group knowledge (i.e., every body in group knows, distributed knowledge and common knowledge) and group social commitments in MASs. The ultimate objective is to cover all possible interactions between knowledge and communicative social commitment in MASs.

Bibliography

[1] Faisal Al-Saqqar, Jamal Bentahar, and Khalid Sultan. On the soundness, completeness and applicability of knowledge and communicative commitments in multi-agent systems. *Expert Systems with applications*, 43:223 – 236, 2016.

[2] Faisal Al-Saqqar, Jamal Bentahar, Khalid Sultan, and Mohamed El-Menshawy. On the interaction between knowledge and social commitments in multi-agent systems. *Applied Intelligence*, 41(1):235–259, 2014.

[3] Faisal Al-Saqqar, Jamal Bentahar, Khalid Sultan, Wei Wan, and Ehsan Khosrowshahi Asl. Model checking temporal knowledge and commitments in multi-agent systems using reduction. *Simulation Modelling Practice and Theory*, 51:45 – 68, 2015.

[4] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. The MIT Press, 2008.

[5] Matteo Baldoni, Cristina Baroglio, and Elisa Marengo. Behavior-oriented commitment-based protocols. In *European Conference on Artificial Intelligence (ECAI)*, pages 137–142, 2010.

[6] Brandon Bennett, Anthony Cohn, Frank Wolter, and Michael Zakharyaschev. Multi-dimensional modal logic as a framework for spatio-temporal reasoning. *Applied Intelligence*, 17(3):239–251, 2002.

[7] Jamal Bentahar, Mohamed El-Menshawy, Hongyang Qu, and Rachida Dssouli. Communicative commitments: Model checking and complexity analysis. *Knowledge-Based Systems*, 35:21–34, 2012.

[8] Jamal Bentahar, Babak Khosravifar, Mohamed Adel Serhani, and Mahsa Alishahi. On the analysis of reputation for agent-based web services. *Expert Systems with Applications*, 39(16):12438–12450, 2012.

[9] Jamal Bentahar, John-Jules Meyer, and Wei Wan. Model checking communicative agent-based systems. *Knowledge-Based Systems*, 22(3):142–159, April 2009.

[10] Jamal Bentahar, John-Jules Meyer, and Wie Wan. Model checking agent communication. In *Specification and Verification of Multi-agent Systems*, pages 67–102. Springer, 2010.

[11] Jamal Bentahar, Bernard Moulin, John-Jules Meyer, and Brahim Chaib-draa. A modal semantics for an argumentation-based pragmatics for agent communication. In *Argumentation in Multi-Agent Systems (ArgMAS)*, pages 44–63, 2004.

[12] Jamal Bentahar, Bernard Moulin, John-Jules Meyer, and Yves Lespérance. A new logical semantics for agent communication. In *Proceedings of the 7th international conference on Computational logic in multi-agent systems*, pages 151–170, Berlin, Heidelberg, 2007. Springer-Verlag.

[13] Jamal Bentahar, Hamdi Yahyaoui, Melissa Kova, and Zakaria Maamar. Symbolic model checking composite web services using operational and control behaviors. *Expert Systems with Applications*, 40(2):508–522, 2013.

[14] Girish Bhat, Rance Cleaveland, and Alex Groce. Efficient model checking via Büchi tableau automata. In G. Berry, H. Comon, and A. Finkel, editors, *CAV*, Lecture Notes in Computer Science, pages 38–52. Springer, 2001.

[15] Patrick Blackburn, Johan van Benthem, and Frank Wolter. *Handbook of Modal Logic (Part 1), Volume 3 (Studies in Logic and Practical Reasoning)*. Elsevier Science Inc., New York, NY, USA, 2006.

[16] Randal Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, 35(8):677–691, August 1986.

[17] Jerry Burch, Edmund Clarke, Kenneth McMillan, David Dill, and L. Hwang. Symbolic model checking: 1020 states and beyond. *Information and Computation*, 98(2):142 – 170, 1992.

[18] Cristiano Castelfranchi. Commitments: From individual intentions to groups and organizations. In Victor R. Lesser and Les Gasser, editors, *International Conference on Multiagent Systems (ICMAS)*, pages 41–48. The MIT Press, 1995.

[19] Zhengang Cheng. *Verifying commitment-based business protocols and their compositions: model checking using promela and spin*. North Carolina State University, 2006. Ph.D. thesis.

[20] Federico Chesani, Paola Mello, Marco Montali, and Paolo Torroni. Representing and monitoring social commitments using the event calculus. *Autonomous Agents and Multi-Agent Systems(AAMAS)*, 27(1):85–130, July 2013.

[21] Amit Chopra and Munindar Singh. Generalized commitment alignment. In *Proceedings of the 14th Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 1–9, 2015.

[22] Alessandro Cimatti, Edmund Clarke, Enrico Giunchiglia, Fausto Giunchiglia, Marco Pistore, Marco Roveri, Roberto Sebastiani, and Armando Tacchella. Nusmv 2: An open source tool for symbolic model checking. In Ed Brinksma and Kim Guldstrand Larsen, editors, *CAV*, volume 2404 of *Lecture Notes in Computer Science*, pages 359–364. Springer, 2002.

[23] Edmund Clarke, Ernest Emerson, and Prasad Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Transactions on Programming Languages and Systems*, 8:244–263, 1986.

[24] Edmund Clarke, Orna Grumberg, and Doron Peled. *Model checking*. The MIT Press, Cambridge, 1999.

[25] Edmund Clarke, William Klieber, Milos Novacek, and Paolo Zuliani. Model checking and the state explosion problem. In Bertrand Meyer and Martin Nordio, editors, *Tools for Practical Software Verification*, volume 7682 of *Lecture Notes in Computer Science*, pages 1–30. Springer Berlin Heidelberg, 2012.

[26] Mika Cohen, Mads Dam, Alessio Lomuscio, and Hongyang Qu. A symmetry reduction technique for model checking temporal-epistemic logic. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 721–726, 2009.

[27] Costas Courcoubetis, Moshe Vardi, Pierre Wolper, and Mihalis Yannakakis. Memory-efficient algorithms for the verification of temporal properties. *Formal Methods in System Design*, 1(2/3):275–288, 1992.

[28] Benjamin Cox. Netbill security and transaction protocol. In *In first USENIX workshop on Electronic Commerce*, pages 77–88, 1995.

[29] Nirmit Desai, Zhengang Cheng, Amit Chopra, and Munindar Singh. Toward verification of commitment protocols and their compositions. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 144–146, 2007.

[30] Nirmit Desai, Amit Chopra, and Munindar Singh. Amoeba: A methodology for modeling and evolving cross-organizational business processes. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 19(2), 2009.

[31] Frank Dignum and Mark Greaves, editors. *Issues in Agent Communication*, volume 1916 of *Lecture Notes in Computer Science*. Springer, 2000.

[32] Warda EL Kholy, Jamal Bentahar, Mohamed El-Menshawy, Hongyang Qu, and Rachida Dssouli. Conditional commitments: Reasoning and model checking. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 24(2):9:1–9:49, December 2014.

[33] Warda EL Kholy, Jamal Bentahar, Mohamed El-Menshawy, Hongyang Qu, and Rachida Dssouli. Modeling and verifying choreographed multi-agent-based web service compositions regulated by commitment protocols. *Expert Systems with Applications*, 41(16):7478 – 7494, 2014.

[34] Mohamed El-Menshawy. *Model Checking Logics of Social Commitments for Agent Communication*. Concordia University, Montreal, Canada, 2012. PhD thesis.

[35] Mohamed El-Menshawy, Jamal Bentahar, and Rachida Dssouli. Symbolic model checking commitment protocols using reduction. In *Declarative Agent Languages and Technologies VIII - 8th International Workshop, DALT*, pages 185–203, 2010.

[36] Mohamed El-Menshawy, Jamal Bentahar, Warda El Kholy, and Rachida Dssouli. Reducing model checking commitments for agent communication to model checking ARCTL and GCTL*. *Autonomous Agents and Multi-Agent Systems*, 27(3):375–418, 2013.

[37] Mohamed El-Menshawy, Jamal Bentahar, Warda El Kholy, and Rachida Dssouli. Verifying conformance of multi-agent commitment-based protocols. *Expert Systems with Applications*, 40(1):122–138, 2013.

[38] Mohamed El-Menshawy, Jamal Bentahar, Hongyang Qu, and Rachida Dssouli. On the verification of social commitments and time. In *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3*, pages 483–490, 2011.

[39] Mohamed El-Menshawy, Jamal Bentahar, Hongyang Qu, and Rachida Dssouli. On the verification of social commitments and time. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 483–490, 2011.

[40] Mohamed El-Menshawy, Wei Wan, Jamal Bentahar, and Rachida Dssouli. Symbolic model checking for agent interactions. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 1555–1556, 2010.

[41] Ernest Emerson. Temporal and modal logic. In *Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B)*, pages 995–1072. 1990.

[42] Ernest Emerson and Joseph Halpern. "Sometimes" and "Not Never" revisited: On branching versus linear time temporal logic. *Journal of the ACM (JACM)*, 33(1):151–178, 1986.

[43] Ronald Fagin, Joseph Halpern, Yoram Moses, and Moshe Vardi. *Reasoning about Knowledge*. The MIT Press, Cambridge, 1995.

[44] Nicoletta Fornara and Marco Colombetti. A commitment-based approach to agent communication. *Applied Artificial Intelligence*, 18(9-10):853–866, 2004.

[45] Dov Gabbay. *Many-Dimensional Modal Logics: Theory and Applications*. Studies in Logic and the Foundations of Mathematics Series. Elsevier North Holland, 2003.

[46] Peter Gammie and Ron van der Meyden. Mck: Model checking the logic of knowledge. In Rajeev Alur and DoronA. Peled, editors, *Computer Aided Verification*, volume 3114 of *Lecture Notes in Computer Science*, pages 479–483. Springer Berlin Heidelberg, 2004.

[47] Rob Gerth, Doron Peled, Moshe Vardi, and Pierre Wolper. Simple on-the-fly automatic verification of linear temporal logic. In *Proceedings of the Fifteenth IFIP WG6.1 International Symposium on Protocol Specification, Testing and Verification XV*, pages 3–18, London, UK, UK, 1996. Chapman & Hall, Ltd.

[48] Laura Giordano, Alberto Martelli, and Camilla Schwind. Specifying and verifying interaction protocols in a temporal action logic. *Journal of Applied Logic*, 5(2):214 – 234, 2007. Logic-Based Agent Verification.

[49] Akın Günay and Pinar Yolum. Constraint satisfaction as a tool for modeling and checking feasibility of multiagent commitments. *Applied Intelligence*, 39(3):489–509, 2013.

[50] Joseph Halpern. Reasoning about knowledge: a survey. Technical Report 1995, IBM Almaden Research Center, 1995.

[51] Joseph Halpern and Leandro Chaves Rêgo. Reasoning about knowledge of unawareness revisited. *Mathematical Social Sciences*, 65(2):73–84, 2013.

[52] Joseph Halpern and Richard Shore. Reasoning about common knowledge with infinitely many agents. *Information and Computation*, 191(1):1–40, 2004.

[53] Hans Hansson and Bengt Jonsson. A logic for reasoning about time and reliability. *Formal Aspects of Computing*, 6(5):512–535, 1994.

[54] Jaakko Hintikka. *Knowledge and Belief: An Introduction to the Logic of the Two Notions*. Cornell University Press, 1962.

[55] Andrzej Indrzejczak. Correspondence theory in proof theory. *Bulletin of the Section of Logic*, 37:171–183, 2008.

[56] Nick Jennings and Michael Wooldridge. Applications of intelligent agents. In Nick Jennings and Michael Wooldridge, editors, *Agent Technology: Foundations, Applications, and Markets*, pages 3–28. Springer-Verlag, Heidelberg, 1998.

[57] Saul Kripke. Semantical Considerations on Modal Logic. *Acta Philosophica Fennica*, 16:83–94, 1963.

[58] Orna Kupferman, Moshe Vardi, and Pierre Wolper. An automata-theoretic approach to branching-time model checking. *Journal of the ACM*, 47(2):312–360, 2000.

[59] Marta Kwiatkowska, Gethin Norman, and David Parker. PRISM: Probabilistic symbolic model checker. In *Computer Performance Evaluation / TOOLS*, pages 200–204, 2002.

[60] Wolfgang Lenzen. *Recent work in epistemic logic*. Volume 30 of Acta Philosophica Fennica, North-Holland, Amsterdam, 1978.

[61] Alessio Lomuscio, Charles Pecheur, and Franco Raimondi. Automatic verification of knowledge and time with nusmv. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 1384–1389, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.

[62] Alessio Lomuscio and Wojciech Penczek. Symbolic model checking for temporal-epistemic logic. In *Logic Programs, Norms and Action*, pages 172–195, 2012.

[63] Alessio Lomuscio, Hongyang Qu, and Franco Raimondi. Mcmas: A model checker for the verification of multi-agent systems. In *Proceedings of the 21st International Conference on Computer Aided Verification*, CAV '09, pages 682–688, Berlin, Heidelberg, 2009. Springer-Verlag.

[64] Alessio Lomuscio and Franco Raimondi. The complexity of model checking concurrent programs against CTLK specifications. In *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, pages 548–550, 2006.

[65] Alessio Lomuscio and Marek J. Sergot. Deontic interpreted systems. *Studia Logica*, 75(1):63–92, 2003.

[66] Alessio Lomuscio and Bozena Wozna. A complete and decidable security-specialised logic and its application to the TESLA protocol. In *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, pages 145–152, 2006.

[67] Ashok Mallya and Munindar Singh. An algebra for commitment protocols. *Autonomous Agents and Multi-Agent Systems (AAMAS)*, 14(2):143–163, 2007.

[68] Omar Marey, Jamal Bentahar, Rachida Dssouli, and Mohamed Mbarki. Measuring and analyzing agents' uncertainty in argumentation-based negotiation dialogue games. *Expert Systems with Applications*, 41(2):306–320, 2014.

[69] Kenneth McMillan. *Symbolic Model Checking: An Approach to the State Explosion Problem*. PhD thesis, 1992. Carnegie Mellon University.

[70] David Meignan, Olivier Simonin, and Abderrafiaa Koukam. Simulation and evaluation of urban bus-networks using a multiagent approach. *Simulation Modelling Practice and Theory*, 15(6):659 – 671, 2007.

[71] Robin Milner. *A Calculus of Communicating Systems*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1982.

[72] Rabeb Mizouni and Aziz Salah. Towards a framework for estimating system NFRs on behavioral models. *Knowlegde-Based Systems*, 23(7):721–731, 2010.

[73] Nadim Obeid. A formalism for representing and reasoning with temporal information, event and change. *Applied Intelligence*, 23(2):109–119, 2005.

[74] Eric Pacuit. Notes for philosophy 151. In *Notes on Modal Logic*. Unpublished notes, found at http://web.pacuit.org/classes/logicai-cmu/ml-notes.pdf, 2009.

[75] Charles Pecheur and Franco Raimondi. Symbolic model checking of logics with actions. In Stefan Edelkamp and Alessio Lomuscio, editors, *Workshop on Model Checking and Artificial Intelligence (MOCHART)*, volume 4428, pages 113–128. Springer, 2006.

[76] Wojciech Penczek and Alessio Lomuscio. Verifying epistemic properties of multi-agent systems via bounded model checking. In *Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 209–216, 2003.

[77] Amir Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science*, pages 46–57, Washington, DC, USA, 1977. IEEE Computer Society.

[78] Franco Raimondi. *Model Checking Multi-Agent Systems*. University College London, London, 2006. Ph.D. thesis.

[79] Franco Raimondi and Alessio Lomuscio. Automatic verification of deontic properties of multi-agent systems. In *Deontic Logic in Computer Science, 7th International Workshop on Deontic Logic in Computer Science, DEON*, pages 228–242, 2004.

[80] Juan Rodriguez-Aguilar, Francisco Martin, Pablo Noriega, Pere Garcia, and Carles Sierra. Towards a test-bed for trading agents in electronic auction markets. *AI Communications*, 11(1):5–19, 1998.

[81] Renate Schmidt, Dmitry Tishkovsky, and Ullrich Hustadt. Interactions between knowledge, action, and commitment within agent dynamic logic. *Studia Logica*, 78(3):381–415, December 2004.

[82] Philippe Schnoebelen. The complexity of temporal logic model checking. In Philippe Balbiani, Nobu-Yuki Suzuki, Frank Wolter, and Michael Zakharyaschev, editors, *Proceedings of the 4th Workshop on Advances in Modal Logic (AIML'02)*, pages 481–517. King's College Publications, 2003.

[83] John Searle. *Speech acts: An essay in the philosophy of language*. Cambridge, Cambridge University Press, 1969.

[84] Krister Segerberg. *An Essay in Classic Modal Logic*. Filosofiska studier. Uppsala Universitet, 1971.

[85] Munindar Singh. Agent communication languages: Rethinking the principles. *Computer*, 31(12):40–47, 1998.

[86] Munindar Singh. An ontology for commitments in multiagent systems. *Artificial Intelligence and Law*, 7(1):97–113, 1999.

[87] Munindar Singh. A social semantics for agent communication languages. In *Issues in Agent Communication*, pages 31–45, 2000.

[88] Munindar Singh. Semantical considerations on dialectical and practical commitments. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 1*, AAAI'08, pages 176–181. AAAI Press, 2008.

[89] Munindar Singh and Michael Huhns. *Service-oriented computing - semantics, processes, agents*. Wiley, 2005.

[90] Marvin Sirbu. Credits and debits on the Internet. *IEEE Spectrum*, 34(2):23–29, 1997.

[91] Robert Endre Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[92] Johan van Benthem. Correspondence theory. In Dov Gabbay and Franz Guenthner, editors, *Handbook of Philosophical Logic: Volume II: Extensions of Classical Logic*, pages 167–247. Reidel, Dordrecht, 1984.

[93] Ron van der Meyden and Nikolay Shilov. Model checking knowledge and time in systems with perfect recall. In Pandu Rangan, V. Raman, and R. Ramanujam, editors, *Foundations of Software Technology and Theoretical Computer Science*, volume 1738 of *Lecture Notes in Computer Science*, pages 432–445. Springer Berlin Heidelberg, 1999.

[94] Ron van der Meyden and Ka shu Wong. Complete axiomatizations for reasoning about knowledge and branching time. *Studia Logica*, 75(1):93–123, 2003.

[95] Hans van Ditmarsch, Joseph Halpern, Wiebe van der Hoek, and Barteld Kooi. An introduction to logics of knowledge and belief. *Computing Research Repository (CoRR)*, abs/1503.00806, 2015.

[96] Katalin Pasztor Varga and Magda Varteresz. Languages of logic and their applications. *Computers & Mathematics with Applications*, 55(8):1660 – 1669, 2008.

[97] Wei Wan, Jamal Bentahar, and Abdessamad Ben Hamza. Model checking epistemic-probabilistic logic using probabilistic interpreted systems. *Knowledge-Based Systems*, 50:279–295, 2013.

[98] Fenghui Wang, Ming Yang, and Ruqing Yang. Simulation of multi-agent based cybernetic transportation system. *Simulation Modelling Practice and Theory*, 16(10):1606 – 1614, 2008.

[99] Michael Wooldridge. Computationally grounded theories of agency. In *International Conference on Multiagent Systems (ICMAS)*, pages 13–22, 2000.

[100] Michael Wooldridge. *Introduction to multiagent systems*. Wiley, 2002.

[101] Bozena Wozna. On the sat-based verification of communicative commitments. In *Proceedings of the 6th Podlasie Conference on Mathematics*, pages 175–186, 2014.

[102] Bozena Wozna, Alessio Lomuscio, and Wojciech Penczek. Bounded model checking for deontic interpreted systems. *Electronic Notes in Theoretical Computer Science*, 126:93–114, 2005.

[103] Pinar Yolum and Munindar Singh. Commitment machines. In *In Proceedings of the 8th International Workshop on Agent Theories, Architectures, and Languages (ATAL-01)*, pages 235–247. Springer, 2000.

[104] Pinar Yolum and Munindar Singh. Reasoning about commitments in the event calculus: An approach for specifying and executing protocols. *Annals of Mathematics and Artificial Intelligence*, 42(1-3):227–253, 2004.