University of Verona

DEPARTMENT OF COMPUTER SCIENCE
GRADUATE SCHOOL OF SCIENCE AND ENGINEERING
DOCTORAL PROGRAM IN COMPUTER SCIENCE
S.S.D. INF/01

DOCTORAL THESIS

Attack States Identification in a Logical Framework of Communicating Agents

Doctoral Student:
Katia Santacà

Tutor:
Prof. Matteo Cristani

Coordinator:
Prof. Massimo Merro

Cycle XXX, 2014

Series N°: **TD-09-18**

Università di Verona
Dipartimento di Informatica
Strada le Grazie 15, 37134 Verona
Italy

To my family

Abstract. A channel is a logical space where agents make announcements publicly. Examples of such objects are forums, wikis and social networks. Several questions arise about the nature of such a statement as well as about the attitude of the agent herself in doing these announcements. Does the agent know whether the statement is true? Is this agent announcing that statement or its opposite in any other channel?

Extensions to Dynamic Epistemic Logics have been proposed in the recent past that give account to public announcements. One major limit of these logics is that announcements are always considered truthful. It is however clear that, in real life, incompetent agents may announce false things, while deceitful agents may even announce things they do not believe in.

In this thesis, we provide a logical framework, called Multiple Channel Logic (MCL), able to relate true statements, agent beliefs, and announcements on communication channels. We discuss syntax and semantics of this logic and show the behavior of the proposed deduction system. We then propose a topological categorization of agents that makes use of the MCL framework. We introduce a complete formalization of prejudices on agents' attitudes and propose an extension of the rules of the MCL framework. We then use RCC5 (the Region Connection Calculus) to categorize different agents in Multi-Agent Systems (MAS) based on the collaboration, competence, and honesty of agents. We discuss the possibility of using RCC3 and RCC8 and generalize our results to define an upper bound on the number of different types of agents in MAS. Finally, we extend the categorization to systems of communicating agents and we provide a tool to support the reasoning on systems defined in MCL. We apply our topological categorization and tool to a specific MAS that describes a Cyber-Physical System, for which we define, categorize, and discuss the resulting attack states.

Acknowledgments. I would like to thank my supervisor, Matteo Cristani, for encouraging my research and for allowing me to grow as a research scientist.

I also really benefited from spending one year at SUTD, for which I thank Martín Ochoa, and all the “Information Systems Group”.

I am grateful for spending six months to at University of Luxembourg, for which I thank Leon van Der Torre and all the “Computer Science and Communications research unit”. A special thanks to Luca Viganò for the important suggestions and help.

A special thanks to my family and my friends for your constant support.

Contents

Part I: Thesis overview

1 Introduction . . . 3

2 About this thesis . . . 5
2.1 Collaborations and publications related to this thesis . . . 5
2.2 Synopsis . . . 5

Part II: Multiple Channel Logic

3 Introduction . . . 9

4 Multiple Channel Logic . . . 11
4.1 A deduction system for MCL . . . 12
4.2 The semantics of MCL . . . 16
4.3 Formal properties of MCL . . . 18

Part III: Assertion, Belief, and Fact - A Logical Framework for the Formal Definition of Communicating Agents

5 Introduction . . . 23

6 Agents and Systems of Agents in MAS . . . 25
6.1 Categorization of Agents . . . 27
6.1.1 Collaboration . . . 27
6.1.2 Competence . . . 29
6.1.3 Honesty . . . 30
6.2 On the Topology of MAS . . . 31
6.2.1 RCC3, RCC5, and RCC8 . . . 31
6.2.2 An Upper bound on the Number of Different Types of Agents 31
6.3 Systems of Communicating Agents . . . 33

6.3.1 Cyber-Physical Systems 33
6.3.2 Single-Channel Attack states 34
6.3.3 Multiple-Channel Attack States 35
6.3.4 Channels 35
6.3.5 Channel properties 37

7 ABF Tool 41
7.1 Implementation of the Region Connection Calculus 42
7.2 Experiment and Results 44
7.2.1 Two agents - unidirectional channel 45

Part IV: State of the art

Part V: Conclusion and Future Work

References i

Implementation of the ABF tool v

List of Figures

6.1 The rules for prejudice in MCL 26

6.2 Representation of the test case 33

6.3 The architecture of $\lambda_S \to \lambda_R$. 37

7.1 The ABF tool. 41

List of Tables

6.1 RCC3, RCC5, and RCC8 relations between spatial regions X, Y and Z 26

6.2 RCC3 composition table with respect to 3 sets. $T(X, Z) = \{DR(X, Z), EQ(X, Z), ONE(X, Z)\}$ 32

6.3 RCC5 composition table over 3 sets. The results show that there exist 54 possible relations. $T(X,Z) = \{DR(X,Z), PO(X,Z), EQ(X,Z), PP(X,Z), PPi(X,Z)\}$ 32

6.4 Number of agents with respect to different RCC 33

6.5 Example of attack states for the water level sensor 34

6.6 Channel Properties 38

6.7 RCC8 composition table over 3 sets. The results shows that there exist 193 possible relation. $T(X,Z)=\{DR(X, Z),PO(X,Z),EQ(X,Z),\mathrm{PP}(X,Z),PPi(X,Z)\}$ 39

7.1 ABF tool System Model configurations 45

7.2 ABF – Properties of agents 45

7.3 Selective forwards attack states 46

Part I

Thesis overview

1

Introduction

In several recent approaches to reasoning about web processes, two different matters tend to intersect: the attitudes of the agents using the web for communication activities, and the beliefs of those agents. Consider, in particular, a situation in which a set of agents use many different communication channels, such as blogs, social networks, forums. It may happen that these agents use those channels in an incoherent way: for instance one agent may announce one statement in a channel while omitting it in another one, or he can announce a statement in a channel and the opposite statement in another one.

In multiple agent settings, there are two different forms of incoherence. We can look at different agents announcing opposite statements, or to one agent announcing opposite things. Both types of incoherence lie on the same ground: contradictory statements are made. The former type involves different agents making such statements (possibly on the same channel), while the latter type sees one single agent making contradictory announcements (possibly on distinct channels, for certain cases).

In the first part of this thesis, we look at incoherences generated by announcements made by one single agent, within a multiple agent logical framework. We then investigate the set of possible communication attitudes of the agents. A communication attitude is the relation between reality facts and agent beliefs, or the relation between beliefs and announcements the agent makes. It is rather common that agents communicating on channels result not always competent on the matter they are talking about. Some agents can also be insincere, or they can hide some (possibly private) information. The attitudes agents assume when communicating, or the ability to know true facts about the reality are important aspects of the communication processes. When we observe agents communicating, we typically have *prejudices* about their attitudes, where prejudices value an agent behavior before observing what he announces.

From this investigation we define a framework to reason on communicating agents. We then apply the framework for the classification of communicating agents. In particular, we apply our framework for reasoning about security configurations of a MAS (Multi Agent System). In fact, reasoning about about properties of an agent (e.g., about his honesty) is particularly relevant for the security research community, where dishonest agents are used to formalize attacks to the systems under consideration. As a result, a number of research thesis have focused their attention to spotting unintended or even malicious behavior in MAS. We specifically focus on *Cyber-Physical Systems (CPS)* as examples of such problems as reported widely in [15, 24] and in [16], where an agent-based model of CPS is considered. In order to reason and classify CPS (as systems of communicating agents in MCL) we provide a tool called ABF Tool.

Distinguishing between the different types of agents in a MAS is a difficult task. This is witnessed by the fact that although a characterization of agents would obviously play a crucial role in the understanding of different aspects and facets in MAS, a proper definition is still missing.

2

About this thesis

The aim of this chapter is to list all the collaborations, projects and scientific articles related to this thesis and to give a roadmap for the thesis, highlighting the main contents of each chapter.

2.1 Collaborations and publications related to this thesis

My PhD has been supported by the University of Verona. In particular, during the PhD I spent twelve months in Singapore (from September 2015 to September 2016) at the Singapore University of Technology and Design (SUTD) and six months in Luxembourg (from the 28th of December 2017 to the 26th of May 2017) at the University of Luxembourg.

List of publications related to this thesis:

1. Matteo Cristani, Elisa Burato, Katia Santacà, Claudio Tomazzoli - "The Spiderman Behavior Protocol: Exploring Both Public and Dark Social Networks for Fake Identity Detection in Terrorism Informatics". In the proceedings of the International Workshop on Knowledge Discovery on the WEB (2015), pages 77–88. [8].
2. Matteo Cristani, Francesco Olivieri, Katia Santacà - "A logical model of communication channels". In the proceedings of In Intelligent and Evolutionary Systems (2016), pages 57-71. [9]
3. Katia Santacà, Matteo Cristani, Marco Rocchetto, Luca Viganò - "A Topological Categorization of Agents for the Definition of Attack States in Multi-Agent Systems". In the proceedings of Multi-Agent Systems and Agreement Technologies (2016), pages 261-276. [25]

2.2 Synopsis

Part II - Multiple Channel Logic.

- In Chapter 3, we look at incoherences generated by announcements made by one single agent, within a multiple agent logical framework.

- In Chapter 4, we present a logical formalism, the Multiple Channel Logic.

Part III - Assertion, Belief, and Fact - A Logical Framework for the Formal Definition of Communicating Agents.

- In Chapter 5, we introduce the MCL application for the categorization of agents.
- In Chapter 6, we classify the possible behaviors of an agent defined in MCL.
- In Chapter 7, we describe the tool that we have applied to the identification of all (or a selection of) the possible configurations of a system of agents in MCL.

Part IV - State of the Art

We give a general overview of state-of-the-art.

Part V - Conclusion and Future Work

We summarize the contents of the thesis and discuss some possible future directions.

Part II

Multiple Channel Logic

3

Introduction

In this thesis, we investigate how to combine public announcements with beliefs that are also not necessarily aligned with the reality. Somebody can have a false belief or he can believe something that is not known as true or false. Moreover agents can announce things they do not believe, or avoid to announce things they actually believe.

To clarify what we mean with this research boundaries, we provide a general example of announcements and their relations with truth and beliefs.

Example 3.1. Alice, Bob and Charlie travel quite often for work. They are also passionate about good food and love visiting nice restaurants. To choose the best hotels and restaurants in town, they use as channels the social networks C_1 and C_2, where they are also active users by posting reviews and feedbacks. During their last business trip, they all stayed at the hotel H and they ate at the restaurants R_1, R_2 and R_3. Once back home, Bob posts a review on channel C_1 *announcing* that hotel H was dirty (saying $dirty(H)$), whilst on channel C_2 he announces that H was clean (saying $\neg dirty(H)$). Alice agrees with his announcement on C_1 but does not post any comments on C_2, while Charlie announces $\neg dirty(H)$ both on C_1 and C_2.

We assume that hotel H being clean or dirty is not a matter of opinions but a *provable* fact. It follows that we may draw some conclusions on the statements announced by Alice, Bob and Charlie on C_1 and C_2. First, Bob is not truthful since he announces $dirty(H)$ on C_1 and $\neg dirty(H)$ on C_2. It may be the case that he believes in only one of the two announces (and possibly in none of them) and, consequently, he is lying in one of the two channels.

In this thesis, we assume atemporal channels, namely announcements are made in a channel and hold forever and eversince, with respect to the moment in which the announcement is made. When an agent observes a particular channel and another agent contradicts herself in that channel the observer finds it out. Consequently we assume that agents make coherent announcements in every single channel, though it is possible that they make opposite announcements in distinct channels. In temporal channels, an agent might announce a statement, and, later, announce the opposite statement. Provided that he is not lying, this implies a belief revision process he passed through in order to change his point of view.

In general, we include agents that can do more real-life things than just truthful and sincere announcements. They can lie, but it is also possible that they just announce things they simply are not informed about. An agent can make an announcement that corresponds to his belief, or he can claim the opposite of his belief. Moreover he can behave in a combination of the three above basic attitudes on different channels. On the same way, an agent can believe things that are true, that are false, and that are neither true or false.

We assume *consistent* agents, that is agents that either believe in the truthfulness of a given statement, or believe in the truthfulness of the opposite statement; naturally, we allow that an agent may be not competent on a certain topic and, as such, believe in neither of them, but never to believe in both at the same.

Back to the example, while Bob and Charlie announce on every channel, Alice decides to express his opinions just on channel C_1. Finally, Alice and Charlie are consistent with themselves even if not with one another. Therefore, they can be both sincere but one of them is not competent. We shall assume that *competent* agents do *generally* know any topic discussed in every channel, while – admittedly a strong assumption – if the agent is ignorant in at least one topic, then he is considered incompetent.

The structure of this part is as follows. In Section 4 we define the logical language. In Chapter 4.2, we introduce the semantics of the logic and in Chapter 4.1 we provide a inference rules of the framework. Section 4.3 provides the prove of soundness for the introduced logic. We conclude in Chapter V with a summary of the results obtained in the investigation and a proposal of a few further investigations.

4

Multiple Channel Logic

In this section, we present our logical formalism, the *Multiple Channel Logic* (hereafter MCL); MCL is specified in terms of language, semantics and inference rules.

MCL is a three-layered labelled, modal logical framework. The first layer of MCL is a propositional calculus. The second layer is a multi-modal calculus, where we can use three distinct modalities: one modality of *belief*, permitting to assert that an agent believes in a proposition, one modality for stating that a given proposition is asserted by an agent in every channel, and one modality to state that an agent asserts a proposition in at least one channel. The last two are henceforth named *communication modalities*.

The second layer does not allow skolemization, permitting, in particular, only to assert that when an agent believes in a proposition, he cannot believe in the opposite, and that he cannot simultaneously believe in something and not believe in it. The same holds for the communication modalities, combined in a dual fashion. A proposition can not be asserted by an agent in every channel when its opposite is asserted in one channel, and, on the other hand, we cannot assert a proposition in every channel and not asserting it in one channel (and vice versa). Not asserting anywhere a proposition does not imply that the opposite of this proposition is asserted somewhere (and vice versa). When we deal with the deduction rules, in section 4.1, we shall mark those rules that guarantee this forms of duality.

Within the third layer, we habilitate *agent tagging* with the explicit purpose of allowing assertion of *prejudices* about agent communicative attitudes. In particular, we tag an agent as sincere, collaborative, and other positive or negative tags. The statement of an agent tag associated to a given agent is named a *prejudice*. Prejudices are employed as means to make certain deduction rules apply.

An MCL theory $\mathcal{T}$ is a triple $\mathcal{T} = \langle \mathcal{W}, \mathcal{A}, \mathcal{R} \rangle$, where $\mathcal{W}$ is the logical language, $\mathcal{A}$ is the set of axioms, and $\mathcal{R}$ is the set of inference rules, in our specific case, rewriting rules. When the set $\mathcal{A}$ is empty, then we call $\mathcal{T}$ a *calculus*. For a given MCL theory M we denote by $\mathcal{R}_M$ the set of axiom of M. To represent the set of rules $\mathcal{R}$, that is common to any MCL theory, we also use, for symmetry with $\mathcal{A}_M$ the notation $\mathcal{R}_M$.

We employ the alphabet $\sum = \mathcal{L} \cup \mathcal{C} \cup \mathcal{M} \cup \mathcal{S} \cup \Lambda \cup \mathcal{T}$ where:
$\mathcal{L}$ is a finite non-empty set of propositional letters $\mathcal{L} = \{A_1, A_2, \ldots, A_n\}$,
$\mathcal{C}$ is the set of connectives $\mathcal{C} = \{\neg, \wedge, \vee \sim, -, \}$,
$\mathcal{M}$ is the set of modalities $\mathcal{M} = \{\mathrm{B}, \mathrm{T}_\Box, \mathrm{T}_\Diamond\}$,

$\mathcal{S}$ is the set of logical signs $\mathcal{S} = \{(,),[,],\bot\}$,
Λ is a finite non-empty set of agents labels $\Lambda = \{\lambda_1,\ldots,\lambda_m\}$,
$\mathcal{T}$ is the set of agent tags $\mathcal{T} = \{Co, S, SCl, WCl, O\}$.

A propositional formula φ is defined by $\varphi := A \mid \neg\varphi \mid \varphi\wedge\varphi \mid \varphi\vee\varphi$ where A denotes a letter. The second layer of MCL is a modal logic where the modal operators are B for beliefs, while $\mathrm{T}_\Box$ and $\mathrm{T}_\Diamond$ are the operators for the communication channels. We use $\mathrm{T}_\Box$ ($\mathrm{T}_\Diamond$) to denote that an agent *tells* the embedded formula in every (at least one) communication channel. A modal formula is $\mu ::= \mathrm{B}[\lambda : \varphi] \mid \mathrm{T}_\Box[\lambda : \varphi] \mid \mathrm{T}_\Diamond[\lambda : \varphi] \mid {\sim}\mu$ where φ denotes a propositional formula. The modal formula for belief $\mathrm{B}[\lambda : \varphi]$ is intended to denote that the agent λ believes φ. For the purpose of this thesis, the intended notion of belief embeds the notion of knowledge as meant in Epistemic Logic.

The modal formula $\mathrm{T}_\Box[\lambda : \varphi]$ denotes that the agent λ announces φ everywhere, namely, when λ announces φ in a channel C, then he announces φ in any channel C' that is accessible from C. When we provide the semantics of MCL we shall relate the notion of accessibility to the notion of *observation*. A channel C' is accessible from a channel C when the observer of C, also observes C'.

On the third layer of the logical framework we make use of the agent tags. An agent tag formula has one of the two formats $\alpha ::= +(X)\lambda \mid -(X)\lambda$ where $X \in \{Co, S, SCl, WCl, O\}$ is an agent tag, and λ is the label representing the agent. The intended meaning of the tags is *competent* for Co, *sincere* for S, *strongly (weakly) collaborative* for SCl and WCl, respectively and *omniscient* for O.

4.1 A deduction system for MCL

In this Section we provide a set of inference rules for MCL. These rules are redundant. In the stream of rules below, we prove that some of the rules can be reduced.

The rules can be of three distinct types:

- **Introduction rules** use the elements appearing in the antecedent for building those elements that appear in the subsequent;
- **Elimination rules** de-construct elements appearing in the antecedent into elements in the subsequent;
- $\bot$ **rules** introduce a contradiction, deriving $\bot$ in the subsequent.

The first group of rules, defined below, manage inference on the propositional layer of MCL. We adapted the classical presentation of Prawitz [21] for propositional calculus to our needs. We shall then introduce a few other specific $\bot$ rules while providing the single contexts for the inference rules of the propositional layer, for the belief layer and finally for the announcement layer.

Without loss of generality we assume that axioms in a MCL theory M are all written in *Conjunctive Normal Form*, namely as conjunctions of disjunctions of positive and negative literals.

Below, Introduction is shortened to In., Elimination to El., D.N. for Double Negation, Left is shortened to L and Right to R, and finally Non Contradiction to N.C. The

"Ex falso sequitur quodlibet in beliefs" rule is shortened to EFSQ, and Modus Ponens is shortened to MP

$$\text{R.1}\ \frac{\varphi \quad \psi}{\varphi \wedge \psi}\ [\text{In.}\ \wedge] \qquad \text{R.2}\ \frac{\varphi \wedge \psi}{\varphi}\ [\text{R el.}\ \wedge]$$

$$\text{R.3}\ \frac{\varphi \wedge \psi}{\psi}\ [\text{L el.}\ \wedge] \qquad \text{R.4}\ \frac{\varphi}{\varphi \vee \psi}\ [\text{In.}\ \vee]$$

$$\text{R.5}\ \frac{\varphi \vee \psi}{\psi}\ [\text{R el. of}\ \vee] \qquad \text{R.6}\ \frac{\varphi \vee \psi}{\varphi}\ [\text{L el. of}\ \vee]$$

$$\text{R.7}\ \frac{\varphi}{\neg\neg\varphi}\ [\text{In. of D.N.}] \qquad \text{R.8}\ \frac{\neg\neg\varphi}{\varphi}\ [\text{El. of D.N.}]$$

$$\text{RC.1}\ \frac{\varphi \quad \neg\varphi}{\bot}\ [\text{N.C. principle}] \qquad \text{RC.2}\ \frac{\bot}{\varphi}\ [\text{EFSQ}]$$

The last operation we need in this group to provide for inferential mechanism is the *Modus Ponens* rule, that we introduce as follows:

$$\text{MP.1}\ \frac{\varphi \ \left(\neg\varphi \vee \psi\right)}{\psi}\ [\text{Propositional modus ponens}]$$

The second group of rules manage inference on the layer of beliefs. The last rule manages the behavior of combined negations ~ and ¬. This is a *quasi*-skolemization, in the sense that it introduces the concept that someone cannot believe that a fact is true and simultaneously not believe that the negation of that fact is false.

$$\text{R.9}\ \frac{\mathrm{B}[\lambda : \varphi] \quad \mathrm{B}[\lambda : \psi]}{\mathrm{B}[\lambda : \varphi \wedge \psi]}\ [\text{In.}\ \wedge\ \text{in B}\ ] \qquad \text{R.10}\ \frac{\mathrm{B}[\lambda : \varphi \wedge \psi]}{\mathrm{B}[\lambda : \varphi]}\ [\text{R el.}\ \wedge\ \text{in B}\ ]$$

$$\text{R.11}\ \frac{\mathrm{B}[\lambda : \varphi \wedge \psi]}{\mathrm{B}[\lambda : \psi]}\ [\text{L el.}\ \wedge\ \text{in B}\ ] \qquad \text{R.12}\ \frac{\mathrm{B}[\lambda : \varphi]}{\mathrm{B}[\lambda : \varphi \vee \psi]}\ [\text{In.}\ \vee\ \text{in B}\ ]$$

$$\text{R.13}\ \frac{\mathrm{B}[\lambda : \varphi]}{\mathrm{B}[\lambda : \neg\neg\varphi]}\ [\text{In. of D.N. in B}\ ] \qquad \text{R.14}\ \frac{\mathrm{B}[\lambda : \neg\neg\varphi]}{\mathrm{B}[\lambda : \varphi]}\ [\text{El. of D.N. in B}\ ]$$

$$\text{R.15}\ \frac{\mathrm{B}[\lambda : \varphi]}{\sim \mathrm{B}[\lambda : \neg\varphi]}\ [\text{Coherence of disbeliefs}]$$

We provide the ⊥ rules for the belief layer. The first rule is used for belief of contradiction, while the second rule is used for belief contradiction. A belief contradiction occurs when the claim of belief is contradicted by the claim of corresponding disbelief. We also have a belief contradiction rule corresponding to "Ex falso sequitur quodlibet in beliefs". We then introduce the Modus Ponens rule for beliefs.

$$\text{RC.3}\ \frac{\mathrm{B}[\lambda : \varphi] \quad \mathrm{B}[\lambda : (\neg\varphi)]}{\bot}\ [\text{B of contr.}] \qquad \text{RC.4}\ \frac{\mathrm{B}[\lambda : \bot]}{\mathrm{B}[\lambda : \varphi]}\ [\text{B EFSQ}]$$

$$\text{RC.5}\ \frac{\mathrm{B}[\lambda : \varphi] \quad \sim \mathrm{B}[\lambda : \varphi]}{\bot}\ [\text{B contr.}] \qquad \text{MP.2}\ \frac{\mathrm{B}[\lambda : \varphi] \quad \mathrm{B}[\lambda : \neg\varphi \vee \psi]}{\psi}\ [\text{B MP}]$$

We obtain the $\perp$ when we believe in φ and either we believe in the opposite, or we do not believe in it.

The third group of rules is introduced to manage inference on $T_{\Box}$ and $T_{\Diamond}$ modalities. The first subgroup, formed by the rules R.12–R.16, supplies the five classical rules for introduction, left and right elimination for $\wedge$, introduction and elimination for double negation for $T_{\Box}$. The rules R.28 and R.28 represent, respectively, Coherence of missing announcements (COM) and Coherence of provided announcements (COP), and correspond to the quasi-skolemization of the modalities $T_{\Box}$ and $T_{\Diamond}$. Seriality is shortened below to Ser.

$$\text{R.16}\ \frac{T_{\Box}[\lambda:\varphi]\quad T_{\Box}[\lambda:\psi]}{T_{\Box}[\lambda:\varphi\wedge\psi]}\ [\text{In. } \wedge \text{ on } T_{\Box}] \qquad \text{R.17}\ \frac{T_{\Box}[\lambda:\varphi\wedge\psi]}{T_{\Box}[\lambda:\varphi]}\ [\text{R el. } \wedge \text{ on } T_{\Box}]$$

$$\text{R.18}\ \frac{T_{\Box}[\lambda:\varphi\wedge\psi]}{T_{\Box}[\lambda:\psi]}\ [\text{L el. } \wedge \text{ on } T_{\Box}] \qquad \text{R.19}\ \frac{T_{\Box}[\lambda:\varphi]}{T_{\Box}[\lambda:\varphi\vee\psi]}\ [\text{In. } \vee \text{ in } T_{\Box}]$$

$$\text{R.20}\ \frac{T_{\Box}\varphi}{T_{\Box}[\lambda:\neg\neg\varphi]}\ [\text{In. of D.N. on } T_{\Box}] \qquad \text{R.21}\ \frac{T_{\Box}[\lambda:\neg\neg\varphi]}{T_{\Box}[\lambda:\varphi]}\ [\text{El. of D.N. on } T_{\Box}]$$

$$\text{R.22}\ \frac{T_{\Diamond}[\lambda:\varphi\wedge\psi]}{T_{\Diamond}[\lambda:\varphi]}\ [\text{R el. } \wedge \text{ on } T_{\Diamond}] \qquad \text{R.23}\ \frac{T_{\Diamond}[\lambda:\varphi\wedge\psi]}{T_{\Diamond}[\lambda:\psi]}\ [\text{L el. } \wedge \text{ on } T_{\Diamond}]$$

$$\text{R.24}\ \frac{T_{\Diamond}[\lambda:\varphi]}{T_{\Diamond}[\lambda:\varphi\vee\psi]}\ [\text{In. } \vee \text{ in } T_{\Diamond}] \qquad \text{R.25}\ \frac{T_{\Diamond}\varphi}{T_{\Diamond}[\lambda:\neg\neg\varphi]}\ [\text{In. of D.N. on } T_{\Diamond}]$$

$$\text{R.26}\ \frac{T_{\Diamond}[\lambda:\neg\neg\varphi]}{T_{\Diamond}[\lambda:\varphi]}\ [\text{El. of D.N. on } T_{\Diamond}] \qquad \text{R.27}\ \frac{T_{\Box}[\lambda:\varphi]}{T_{\Diamond}[\lambda:\varphi]}\ [\text{Ser. } T_{\Box} \text{ on } T_{\Diamond}]$$

$$\text{R.28}\ \frac{T_{\Box}[\lambda:\varphi]}{\sim T_{\Diamond}[\lambda:\neg\varphi]}\ [\text{COM}] \qquad \text{R.29}\ \frac{T_{\Diamond}[\lambda:\varphi]}{\sim T_{\Box}[\lambda:\neg\varphi]}\ [\text{COP}]$$

If agent λ announces φ on every channel, it is straightforward that he announces it on at least one channel (R.27). If λ announces φ on every channel, then in no channel he may announces the opposite (R.28). Lastly, if λ announces φ on at least one channel, then he cannot announce the opposite on every channel (R.29).

We now present the $\perp$ rules for announcements. Announcement contradictions are shortened below to AC.

$$\text{RC.6}\ \frac{T_{\Diamond}[\lambda:(\varphi\wedge\neg\varphi)]}{\perp}\ [\text{AC on } T_{\Box}] \qquad \text{RC.7}\ \frac{T_{\Box}[\lambda:(\varphi\wedge\neg\varphi)]}{\perp}\ [\text{AC}]$$

$$\text{RC.8}\ \frac{T_{\Box}[\lambda:\varphi]\quad T_{\Diamond}[\lambda:\neg\varphi]}{\perp}\ [\text{AC on } T_{\Diamond}]$$

RC.8 is derived from R.28–R.29: announcing φ on every channel contradicts with announcing $\neg\varphi$ somewhere. Again we provide a modus ponens rule for $T_{\Box}$ and $T_{\Diamond}$. The first is Channel existential Modus Ponens (CEMP), the second is Channel Universal Modus Ponens (CUMP).

$$\text{MP.3}\ \frac{T_{\Box}[\lambda:\varphi]\quad T_{\Diamond}[\neg\varphi\vee\psi]}{T_{\Diamond}:[\lambda:\psi]}\ [\text{CEMP}] \qquad \text{MP.4}\ \frac{T_{\Diamond}[\lambda:\varphi]\quad T_{\Box}[\neg\varphi\vee\psi]}{T_{\Diamond}:[\lambda:\psi]}\ [\text{CUMP}]$$

We need to manage introduction and elimination of missing beliefs and missing announces. We summarize the above by using the expression μ to denote a modal formula of MCL.

$$\text{R.30}\ \frac{\mu[\lambda:\varphi]}{\sim\sim\mu[\lambda:\varphi]}\ \text{[In. of double} \sim\text{]} \qquad \text{R.31}\ \frac{\sim\sim\mu[\lambda:\varphi]}{\mu[\lambda:\varphi]}\ \text{[El. of double} \sim\text{]}$$

We can introduce the rules for the agent tags. We have two rules that relate reality and beliefs, based on the expressed prejudice of omniscience and competence. Moreover, we have two pairs of rules for weak and strong collaboration tags, and one rule for sincereness which relates beliefs and communication channels.

All the above mentioned rules are employed for positive tags. For every positive tag, we have the dual rule for the negative counterpart.

$$\text{R.32}\ \frac{\varphi\ \ +(O)\lambda}{\mathrm{B}[\lambda:\varphi]}\ \text{[Ext. +}(O)\text{]} \qquad \text{R.33}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ +(Co)\lambda}{\varphi}\ \text{[Ext. +}(Co)\text{]}$$

$$\text{R.34}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ +(Wcl)\lambda}{\mathrm{T}_{\Diamond}[\lambda:\varphi]}\ \text{[Ext. +}(WCl)\text{]} \qquad \text{R.35}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ +(SCl)\lambda}{\mathrm{T}_{\Box}[\lambda:\varphi]}\ \text{[Ext. +}(SCl)\text{]}$$

$$\text{R.36}\ \frac{\mathrm{T}_{\Diamond}[\lambda:\varphi]\ \ +(S)\lambda}{\mathrm{B}[\lambda:\varphi]}\ \text{[Ext. +}(S)\text{]} \qquad \text{R.37}\ \frac{\sim\mathrm{B}[\lambda:\varphi]\ \ \varphi}{\text{-}(O)\lambda}\ \text{[In. -}(O)\text{]}$$

$$\text{R.38}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ \neg\varphi}{\text{-}(Co)\lambda}\ \text{[In. -}(Co)\text{]} \qquad \text{R.39}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ \sim\mathrm{T}_{\Diamond}[\lambda:\varphi]}{\text{-}(Wcl)\lambda}\ \text{[In. -}(WCl)\text{]}$$

$$\text{R.40}\ \frac{\mathrm{B}[\lambda:\varphi]\ \ \sim\mathrm{T}_{\Box}[\lambda:\varphi]}{\text{-}(Scl)\lambda}\ \text{[In. -}(SCl)\text{]} \qquad \text{R.41}\ \frac{\mathrm{T}_{\Diamond}[\lambda:\varphi]\ \ \sim\mathrm{B}[\lambda:\varphi]}{\text{-}(S)\lambda}\ \text{[In. -}(S)\text{]}$$

An agent is omniscient if he knows every true formula. An agent is competent is every formula he knows is true. Notice that in the case of omniscience, the set of formulae believed true by the agent is a superset of the (actually) true formulae: the agent knows all what is true but he may also believe in some formulae proven neither true, nor false. On the contrary, in case of competence, the agent's beliefs are a subset of the true formulae; as such, all the agent's beliefs are proven to be true but there may be true formulae "out" of his knowledge base.

If an agent believes that φ is true and he is weakly collaborative, then he will announces φ in at least one channel. Sincerity relates the communication of an agent with his beliefs. As such, a sincere agent that tells φ on a channel, then he believes φ to be true. We might be tempted to formulate sincerity from "the other agents' perspective" and state that occurs whenever an agent announces φ somewhere while $\neg\varphi$ anywhere. The proposed formulation of R.36 has the advantage that a sincere agent cannot contradict herself. This would lead to a contradiction due to λ being sincere and the application of R.36 to both $\mathrm{T}_{\Diamond}[\lambda:\varphi]$ and $\mathrm{T}_{\Diamond}[\lambda:\overline{\varphi}]$, and the subsequent application of RC.3.

Sincerity and collaboration do not derive one another. Assume three channels and that agent λ believes both formulae φ and ψ to be true. Suppose that λ announces φ on the first channel, $\neg\varphi$ on the second one, and ψ on the third one. In this setting, λ is collaborative but not sincere. Suppose now that λ solely announces ψ on the first channel. In this case, λ is sincere but not collaborative. It is straightforward to notice that R.35 subsumes R.34 through seriality. In addition, it implies a subtle form of sincerity. A strongly collaborative agent cannot announce anything he does believe to be true, even if he might announce something he believes to be neither true nor false. (Indeed, whenever the strongly collaborative λ has $\sim$Bφ as well as $\sim$B$\neg\varphi$, nothing prevent his to announce either φ or $\neg\varphi$ somewhere.)

Trivially, an agent: (R.37) is non-omniscient whenever he does not believe true an actually true formula, (R.38) is incompetent whenever he believe true a false formula. An agent is not weakly (strongly) collaborative if he knows something which he does not announce in at least one channel. Finally, an insincere agent announces, in at least one channel, a formula he does not believe to be true. In our formulation, it is not necessary that the agent believes in the truthfulness of a formula φ while announcing $\neg\varphi$ to be considered insincere.

R.38 depends on R.37. In fact, given the premises of R.37, by applying R.15 to B[$\lambda : \varphi$] we obtain $\sim$B[$\lambda : \neg\varphi$].

Finally we introduce the rules that provide contradictions between prejudices (positive and negative). The set of rules can be summarized by the expressions $+(P)\lambda$ as the assertion of the prejudice P on λ, and $-(P)\lambda$ as the negation of the prejudice P on λ.

$$\text{RC.9}\ \frac{+(P)\lambda \quad -(P)\lambda}{\perp}\ \text{[Prejudice contradiction]}$$

4.2 The semantics of MCL

In MCL, the announcement of a formula by one agent cannot be bound to appear on a single, specific channel. We can only bind an announcement to appear in either all channels, or at least one. We therefore employ a semantics for the modalities that follows Kripke's modeling guidelines.

In order to build the semantics of a MCL theory we provide the interpretation of the signature, of first-layer formulae, of second-layer formulae and of agent tags. Accordingly, the semantics of a MCL theory is a tuple $\mathcal{M}$=$\langle\mathcal{W}_C,\mathcal{W}_\mathcal{A},\mathcal{W}_\mathcal{R},\mathcal{W}_\mathcal{L},\mathcal{R}_C,\mathcal{I}\rangle$. where $\mathcal{W}_C$ is the domain of *Channels*, $\mathcal{W}_\mathcal{A}$ is the domain of *Agents*, $\mathcal{W}_\mathcal{R}$ is the Boolean twofold interpretation domain $\{true, false\}$, $\mathcal{W}_\mathcal{L}$ is the the domain of the *Letters*, a finite non-empty set, as numerous as the letters in the signature of MCL, $\mathcal{R}_C$ is the accessibility relation between elements of $\mathcal{W}_C$, $\mathcal{I}$ is the *Interpretation* function.

Given a MCL theory L, the interpretation function maps every propositional letter of the signature of L in one element of $\mathcal{W}_\mathcal{R}$, every agent letter in one element of $\mathcal{W}_\mathcal{A}$, every channel letter in one element of $\mathcal{W}_C$, and finally every agent tag in a subset of $\mathcal{W}_\mathcal{A}$. The accessibility relation is defined in $\mathcal{W}_C$. The accessibility relation captures the idea that a channel C_1 is related to a channel C_2 *iff* the external observer representing the theory can observe C_2 whenever he observes C_1. The accessibility relation is assumed therefore to be reflexive and transitive, while we do not make any assumption about symmetry.

The interpretation of $\perp$ is $\mathcal{I}(\perp) = false$. The truth of a propositional formula follows the classic interpretation of $\wedge$, $\vee$ and $\neg$ operators.

Interpretations are models when they provide consistent evaluations for the propositional layer, the beliefs, the announcements and the relationships between the above determined by the expression of prejudices.

In particular, we assume that for a model the following hold:

1. The interpretation of the set of propositional axioms is consistent with the interpretation of letters;
2. The interpretation of each agent belief set is a consistent propositional theory with respect to the interpretation of pairs formed by agent and propositional letters;
3. The set of announcements for each agent in every single channel is a consistent propositional theory with respect to the interpretation of triples formed by agent, channel and propositional letters;
4. For every modal formula $\mathrm{T}_{\Box}[\lambda : \varphi]$ asserted axiomatically in L and for every channel C in which φ is announced by λ, φ is announced by λ in every channel related from C by the accessibility relation;
5. For every modal formula $\mathrm{T}_{\Diamond}[\lambda : \varphi]$ asserted axiomatically in L and for every channel C in which φ is announced by λ, φ is announced by λ in at least one channel related from C by the accessibility relation;
6. For every *omniscient* agent λ and every formula φ that is interpreted true, then φ is also a belief of λ;
7. For every *competent* agent λ and every formula φ believed by λ, then φ is interpreted true;
8. For every *sincere* agent λ, if a formula φ is announced in one channel by λ, then φ is a belief of λ;
9. For every *strongly collaborative* agent λ, and every belief φ of λ, λ announces φ in every channel;
10. For every *weakly collaborative* agent λ, and every belief φ of λ, λ announces φ in at least one channel.

As usual, when a MCL theory L has a model, then L is called *satisfiable*. Conversely, when it has no model is called *unsatisfiable*. A set of axioms containing the symbol $\perp$ is unsatisfiable.

When we value semantics as defined above, we name such a model a MCL-model, and we say that this holds for a MCL-semantics.

To explain how the interpretation of a theory works we introduce here an example.

Example 4.1. Consider three agents. The construction of the model is performed as follows: starting from a set of axioms $A_1, A_2, \ldots A_n$, corresponding to the interpretation of the letters $A_1^{\mathcal{I}} \ldots A_n^{\mathcal{I}}$.

An interpretation is, in fact, a set of literals assumed true: $A_1, A_3, \overline{A_5}, A_6$.

An assignments for the beliefs of the agents is something like:

$$\begin{array}{ll} \lambda_1 & A_1,\overline{A_3},A_6 \\ \\ \lambda_2 & \overline{A_1},A_6 \\ \\ \lambda_3 & A_2 \end{array}$$

and we can define for each C, where C is a channel, an assignment as in the scheme below.

$$\begin{array}{lllllll} \lambda_{1,C1} \; A_1,A_2 & \lambda_{2,C1} & \ldots & \lambda_{3,C1} & \ldots & \lambda_{N,C1} & \ldots \\ \lambda_{1,C2} \; A_1,A_2 & \lambda_{2,C2} & \ldots & \lambda_{3,C2} & \ldots & \lambda_{N,C1} & \ldots \\ \vdots \; \vdots & \vdots \; \vdots & & \vdots \; \vdots & & \vdots \; \vdots & \\ \\ \lambda_{1,Cm} \; A_1,A_2 & \lambda_{2,Cm} & \ldots & \lambda_{3,Cm} & \ldots & \lambda_{N,CM} & \vdots \end{array}$$

On the other hand, a channel is a space for interpreting announcements of the agents. The underlying idea, again, is that a single agent in a single channel announces things in a coherent way, being irrational that an agent contradicts herself in a completely observable channel. Thus, the set of axioms corresponding to agent assertion quantifications are mapped to channels by choosing a set of literals for each agent label that, assumed true, makes coherent the statements of the agent mapped in that particular channel.

4.3 Formal properties of MCL

In this section we prove that MCL is sound and complete with respect to the introduced semantics. For the sake of space, the proofs of simplest results are omitted, and we only deal with the main ones.

The task we look at is *consistency checking*. Given a MCL theory L, we aim at establishing whether the set of axioms in L are consistent with each other, or, in other terms, whether they can or cannot derive a contradiction.

The first property we prove is that the set of deduction rules introduced for MCL theories preserve satisfiability. When employing the deduction rules of MCL we transform a theory L into other theories, called *derived from L*. We specifically say that a theory L is deductively closed when applying the deduction rules to L we obtain L. Conversely, when applying rules to a theory L leads to a theory L', and further on, to a theory L^*, that is deductively closed, we name L^* the *deductive closure* of L.

Lemma 4.2. *Given a satisfiable MCL theory L, if L' is derived from L, then L' is satisfiable.*

Proof. The proof is a direct consequence of the rules R.1-R.40 and of the rules MP.1-MP.4, along with the definition of MCL-models.

Lemma 4.3. *Given an unsatisfiable MCL theory L, if L' is derived from L, then L' is unsatisfiable.*

Proof. The proof is a direct consequence of Lemma 4.2 and of rules RC.1-RC.9, along with the definition of MCL-models..

An immediate, straightforward consequence of the above mentioned lemmas is claimed in Corollary 4.4, whose proof is omitted.

Corollary 4.4. *The deductive closure L^* of a MCL theory L is consistent* iff *L is consistent.*

Based on Lemma 4.2, 4.3 and Corollary 4.4, we can conclude Theorem 4.5.

Theorem 4.5. *The rules of MCL preserve satisfiability.*

Once we have proved that the rules preserve satisfiability, we are now able to claim a soundness result. The soundness result is obtained by means of the rules, the standard interpretation of $\perp$ and the notion of MCL-model.

First of all, we claim that when a theory is contradictory, its deductive closure contains $\perp$. Secondly, we prove the inverse: if a MCL theory L' contains $\perp$, when another theory L can be transformed by the rules onto L', then L contains either $\perp$ or a contradiction. When, given a theory L', there is not a theory L such that L' is derived from L, we say that L' is *primary*.

Lemma 4.6. *The deductive closure of a contradictory theory contains $\perp$.*

Lemma 4.7. *Every theory L' containing $\perp$ that is not primary, can be derived from a different theory L that either contains $\perp$ or is contradictory.*

As a consequence of the above lemmas we conclude the following theorem.

Theorem 4.8. *The deduction system of MCL is sound.*

In order to prove the completeness of the MCL deduction rules with respect to the semantics introduced in Section 4.2 we firstly show the following lemma.

Lemma 4.9. *If a MCL theory L is satisfiable, then it is consistent.*

Proof. Lemmas 4.6 and 4.7 prove that deductively closed satisfiable theories do not contain $\perp$. As a consequence, if a theory had a model, then it would be consistent.

Based on Lemma 4.9 we can finally prove the following theorem.

Theorem 4.10. *The deduction system of MCL is complete.*

Part III

Assertion, Belief, and Fact - A Logical Framework for the Formal Definition of Communicating Agents

5

Introduction

Much effort has been devoted to the characterization of different agents in *Multi-Agent Systems (MAS)*, ranging from works that employ Dynamic Epistemic Logic and Public Announcement Logics (PAL) [30] to more recent approaches such as [1]. These works have studied an agent's beliefs and announcements, typically under the assumption that agents are always truthful and sincere. However, as discussed in, e.g., [9], this assumption is an oversimplification since most MAS contain a number of agents that are clearly neither sincere nor truthful.

General problems with agency and norms, namely social regulations, are presented in [12], where many open problems are discussed. Further investigations, including those in *Public Announcement Logic* [1], have devised a pathway to follow, with many problems in the definitions still open. A step in this direction has been carried out in [9], which introduced a general logical framework called *Multiple Channel Logic Framework (MCL)*. However, the focus of [9] is on the definition of the framework and little attention is payed to the definition of a general categorization of all the possible agents that could be defined using MCL.

The overall goal of this part of the thesis is the definition of a general categorization of agents, based on MCL. We focus, in particular, on the application of MAS for reasoning about security systems, such as CPS. More specifically, our contributions are:

1. We define a topological categorization of agents in MAS, obtaining 50 new rules in the MCL framework.
2. We identify a theoretical limit to the maximum number of different types of agents in a MAS (defined using MCL).
3. We extend our results to systems of communicating agents defined in MCL by defining and characterize channels.
4. We implement a tool to reason on systems of agents defined in MCL.
5. As an example of a concrete application, we apply our topological categorization to define attack states for a MAS that describes a general CPS. Our case study ultimately allows us to show that our categorization of agents can be used to reason about the security of CPS and, more generally, MAS.

6

Agents and Systems of Agents in MAS

In MCL, agents are defined by using the three main components of the framework: the sets $\mathbb{A}_\lambda$, $\mathbb{B}_\lambda$ and $\mathbb{F}$ of announcements, beliefs and facts. A natural step is to define how many different types of agents can be defined out of these three sets. To do that, we first extend the results of [9] by considering the relations between these three sets and then use these relations to define agents in MCL.

Intuitively, we can define the following three relations:

- *Collaboration* $(\mathbb{A}_\lambda, \mathbb{B}_\lambda)$ is the relation between beliefs and announcements of an agent λ. This relation defines the level of collaboration of λ as the *quantity* of data an agent announces with respect to the data he believes. For example, if an agent asserts everything he believes, he is collaborative (recall that belief can be false, in which case the agent might not be competent).
- *Competence* $(\mathbb{B}_\lambda, \mathbb{F})$ is the relation between beliefs of an agent λ and true facts. This relation defines the level of competence of λ and is related to the *quality* of data an agent produces. For example, if everything an agent believes is also true, he is competent (note that this is not the definition of knowledge since an agent could believe in false formulae).
- *Honesty* $(\mathbb{A}_\lambda, \mathbb{F})$ is the relation between announcements made by an agent λ and true facts. This relation defines the level of honesty of λ. For example, if everything an agent shares on a channel is also true, then he is honest.[1]

Given that these three relations are over sets, they express *mereological* relations. We use the *Region Connection Calculus* (RCC) to reason on the different "levels" of collaboration/competence/honesty and to identify which are the different possible relations between the three sets $\mathbb{A}_\lambda$, $\mathbb{B}_\lambda$ and $\mathbb{F}$. This ultimately defines how many different types of agents we can theoretically consider.

RCC, as defined in [13, 17], is an axiomatization of certain spatial concepts and relations in first-order logic. In its broader definition, the RCC theory is composed by eight axioms, and is known as RCC8, but here we restrict to RCC5 by not considering tangential connections between spatial regions. We discuss the choice of RCC5 in more detail in Section 6.2.

[1] Note that honesty is not necessary related to correctness. In fact, we define an agent as honest if he asserts the truth even if he does not believe in what he asserts.

$$\text{R.32}\ \frac{\varphi \quad +(O)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\text{E. }+(O)] \qquad \text{R.33}\ \frac{\mathrm{B}[\lambda:\varphi] \quad +(Co)\lambda}{\varphi}\ [\text{E. }+(Co)]$$

$$\text{R.34}\ \frac{\mathrm{B}[\lambda:\varphi] \quad +(W_{Cl})\lambda}{\mathrm{T}_{\Diamond}[\lambda:\varphi]}\ [\text{E.}+(W_{Cl})] \qquad \text{R.35}\ \frac{\mathrm{B}[\lambda:\varphi] \quad +(S_{Cl})\lambda}{\mathrm{T}_{\Box}[\lambda:\varphi]}\ [\text{E. }+(S_{Cl})]$$

$$\text{R.36}\ \frac{\mathrm{T}_{\Diamond}[\lambda:\varphi] \quad +(S)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\text{E. }+(S)] \qquad \text{R.37}\ \frac{\sim\mathrm{B}[\lambda:\varphi] \quad \varphi}{-(O)\lambda}\ [\text{I. }-(O)]$$

$$\text{R.38}\ \frac{\mathrm{B}[\lambda:\varphi] \quad \neg\varphi}{-(Co)\lambda}\ [\text{I. }-(Co)] \qquad \text{R.39}\ \frac{\mathrm{B}[\lambda:\varphi] \quad \sim\mathrm{T}_{\Diamond}[\lambda:\varphi]}{-(W_{Cl})\lambda}\ [\text{I. }-(W_{Cl})]$$

$$\text{R.40}\ \frac{\mathrm{B}[\lambda:\varphi] \quad \sim\mathrm{T}_{\Box}[\lambda:\varphi]}{-(S_{Cl})\lambda}\ [\text{I.}-(S_{Cl})] \qquad \text{R.41}\ \frac{\mathrm{T}_{\Diamond}[\lambda:\varphi] \quad \sim\mathrm{B}[\lambda:\varphi]}{-(S)\lambda}\ [\text{I.}-(S)]$$

Fig. 6.1. The rules for prejudice in MCL

Table 6.1. RCC3, RCC5, and RCC8 relations between spatial regions X, Y and Z

RCC3	RCC5	RCC8	Name	Notation	Definition
			Connects with	$C(X,Y)$	$X \subseteq Y$
			Disconnected from	$\neg C(X,Y)$	$X \nsubseteq Y$
			Part of	$P(X,Y)$	$\forall Z\ C(Z,X) \rightarrow C(Z,Y)$
			Overlaps	$O(X,Y)$	$\exists Z\ P(Z,X) \wedge P(Z,Y)$
●			**Overlaps Not Equal**	$ONE(X,Y)$	$O(X,Y) \wedge \neg EQ(X,Y)$
●	●	●	**Equal to**	$EQ(X,Y)$	$P(X,Y) \wedge P(Y,X)$
●	●	●	**DiscRete from**	$DR(X,Y)$	$\neg O(X,Y)$
	●	●	**Partial-Overlap**	$PO(X,Y)$	$O(X,Y) \wedge \neg P(X,Y) \wedge \neg P(Y,X)$
	●		**Proper-part-of**	$PP(X,Y)$	$P(X,Y) \wedge \neg P(Y,X)$
	●		**Proper-part-of-inverse**	$PPi(X,Y)$	$P(Y,X) \wedge \neg P(X,Y)$
		●	**Externally Connected**	$EC(X,Y)$	$C(X,Y) \wedge \neg O(X,Y)$
		●	**Tangential PP**	$TPP(X,Y)$	$PP(X,Y) \wedge \exists Z\ [EC(Z,X), EC(Z,Y)]$
		●	**Tangential PPi**	$TPPi(X,Y)$	$TPP(Y,X)$
		●	**Non-tangential PP**	$NTPP(X,Y)$	$PP(X,Y) \wedge \neg\exists Z\ [EC(Z,X), EC(Z,Y)]$
		●	**Non-tangential PPi**	$NTPPi(X,Y)$	$NTPP(Y,X)$

We define *parthood* as the primitive binary inclusion relation $\subseteq$, which is reflexive, antisymmetric and transitive. In Table 6.1, we define the relations of RCC3, RCC5 and RCC8, where X, Y and Z are sets (spatial regions) of formulae and *Connects with* expresses the parthood relation. By applying these relations to the pairs $(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$, $(\mathbb{B}_\lambda,\mathbb{F})$ and $(\mathbb{A}_\lambda,\mathbb{F})$, we can distinguish between different levels of collaboration, competence and honesty. Every tuple representing the combination of the three relations defines a different type of agent.

$$Agent = \langle RCC5_1(\mathbb{A}_\lambda,\mathbb{B}_\lambda), RCC5_2(\mathbb{B}_\lambda,\mathbb{F}), RCC5_3(\mathbb{A}_\lambda,\mathbb{F})\rangle,$$

where $RCC5_1$, $RCC5_2$ and $RCC5_3$ are relations in RCC-5. As we discuss in Section 6.2, some combinations of $RCC5_1$, $RCC5_2$ and $RCC5_3$ are topologically incorrect.

6.1 Categorization of Agents

We now consider the details of every RCC5 relation between each pair of $\mathbb{A}_\lambda$, $\mathbb{B}_\lambda$ and $\mathbb{F}$ and we define 15 different prejudices. Our list is complete with respect to RCC5, i.e., no other relations can be considered. We will use overline numbers to identify the new rules we introduce, whereas the decimals for the rules were already defined in [9].

6.1.1 Collaboration

Sincere $PP(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$**.** A sincere agent λ is defined by the proper part of his announcements with respect to his beliefs. More formally, for any propositional formula φ,

$$\text{if } \mathrm{T}_*[\lambda:\varphi] \text{ then } \mathrm{B}[\lambda:\varphi],$$

where $*$ identifies one of the two modalities in MCL, i.e., $* \in \{\Box, \Diamond\}$.

This type of agent announces *only* what he believes ($\Rightarrow$) but does not announce everything he believes ($\not\Leftarrow$). As already defined in [9], we can negate the formula of a sincere agent and provide deduction rules to define when an agent is *not* sincere as follows. For a non-sincere agent λ[2], there exists a propositional formula φ such that

$$\mathrm{T}_*[\lambda:\varphi] \text{ and } \sim \mathrm{B}[\lambda:\varphi].$$

We can then define rules that formalize that if an agent asserts, even only once, something that he does not believe in, then he is non-sincere:

$$\text{R.41}\ \frac{\mathrm{T}_\Diamond[\lambda:\varphi] \quad \sim \mathrm{B}[\lambda:\varphi]}{-(W_S)\lambda}\ [\text{I.}-(W_S)] \qquad\qquad \text{R.}\overline{1}\ \frac{\mathrm{T}_\Box[\lambda:\varphi] \quad \sim \mathrm{B}[\lambda:\varphi]}{-(S_S)\lambda}\ [\text{I.}-(S_S)]$$

As we discussed in Section 6, the notion of weak and strong is only applied to the notion of collaborative agent in MCL. We avoid this asymmetry and we introduce the notion of weak and strong for all the prejudices involving a relation with announcements. This explains why we have now used W_S in R.41 instead of S of MCL (as in Fig. 6.1). We extend the elimination rules accordingly:

$$\text{R.}\overline{2}\ \frac{\sim \mathrm{B}[\lambda:\varphi] \quad +(W_S)\lambda}{\sim \mathrm{T}_\Diamond[\lambda:\varphi]}\ [\text{E.}+(W_S)] \qquad\qquad \text{R.}\overline{3}\ \frac{\sim \mathrm{B}[\lambda:\varphi] \quad +(S_S)\lambda}{\sim \mathrm{T}_\Box[\lambda:\varphi]}\ [\text{E.}+(S_S)]$$

$$\text{R.36}\ \frac{\mathrm{T}_\Diamond[\lambda:\varphi] \quad +(W_S)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\text{E.}+(W_S)] \qquad\qquad \text{R.}\overline{4}\ \frac{\mathrm{T}_\Box[\lambda:\varphi] \quad +(S_S)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\text{E.}+(S_S)]$$

Collaborative $PPi(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$**.** Symmetrically to a sincere agent, a collaborative agent λ is defined by the proper part of his beliefs with respect to his announcements: for any propositional formula φ,

$$\text{if } \mathrm{B}[\lambda:\varphi] \text{ then } \mathrm{T}_*[\lambda:\varphi].$$

This type of agent announces everything he believes ($\Rightarrow$) but what he says is *not only* what he believes ($\not\Leftarrow$). Hence, some of the announcements are intentionally

[2] Slightly abusing notation, we are using λ for both a sincere and non-sincere agent.

against his beliefs (these announcements might be accidentally true facts but we will discuss this case later in this section). If we negate the definition of collaborative, we obtain that if an λ's belief has not been announced (i.e., there exists φ such that $\mathrm{B}[\lambda : \varphi]$ and $\sim \mathrm{T}_*[\varphi : \lambda]$), then λ is *not* collaborative. As for the sincere agent, we define strong and weak prejudice with $\Box$ and $\Diamond$, respectively:

$$\mathrm{R}.39\ \frac{\mathrm{B}[\lambda:\varphi]\quad \sim \mathrm{T}_{\Diamond}[\lambda:\varphi]}{-(W_{CI})\lambda}\ [\mathrm{I}.-(W_{CI})] \qquad \mathrm{R}.40\ \frac{\mathrm{B}[\lambda:\varphi]\quad \sim \mathrm{T}_{\Box}[\lambda:\varphi]}{-(S_{CI})\lambda}\ [\mathrm{I}.-(S_{CI})]$$

$$\mathrm{R}.34\ \frac{\mathrm{B}[\lambda:\varphi]\quad +(W_{CI})\lambda}{\mathrm{T}_{\Diamond}[\lambda:\varphi]}\ [\mathrm{E}.+(W_{CI})] \qquad \mathrm{R}.35\ \frac{\mathrm{B}[\lambda:\varphi]\quad +(S_{CI})\lambda}{\mathrm{T}_{\Box}[\lambda:\varphi]}\ [\mathrm{E}.+(S_{CI})]$$

$$\mathrm{R}.\overline{5}\ \frac{\sim \mathrm{T}_{\Diamond}[\lambda:\varphi]\quad +(W_{CI})\lambda}{\sim \mathrm{B}[\lambda:\varphi]}\ [\mathrm{E}.+(W_{CI})] \qquad \mathrm{R}.\overline{6}\ \frac{\sim \mathrm{T}_{\Box}[\lambda:\varphi]\quad +(S_{CI})\lambda}{\sim \mathrm{B}[\lambda:\varphi]}\ [\mathrm{E}.+(S_{CI})]$$

Fair $EQ(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$**.** A fair agent λ is defined by the equality between the sets of his announcements and beliefs: for any propositional formula φ,

$$\mathrm{T}_*[\lambda : \varphi] \text{ if and only if } \mathrm{B}[\lambda : \varphi]\,.$$

Hence, a fair agent is an agent who believes in *everything* he announces ($\Rightarrow$) and who announces *only* what he believes ($\Leftarrow$). As before, in order to give the rules for MCL, we first negate the definition of the fair agent. For a non-fair agent λ, there exists a propositional formula φ such that

$$(\sim \mathrm{T}_*[\lambda : \varphi] \text{ and } \mathrm{B}[\lambda : \varphi]) \text{ or } (\sim \mathrm{B}[\lambda : \varphi] \text{ and } \mathrm{T}_*[\lambda : \varphi])\,.$$

The left and right disjuncts are exactly the definitions of *PPi* and *PP*, respectively. Hence, the introduction and elimination rules have been already considered in the previous two cases.

Saboteur $PO(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$**.** A saboteur agent λ is defined by the partial overlap of his announcements with respect to his beliefs: for any propositional formula φ,

$$\mathrm{B}[\lambda : \varphi] \text{ or } \mathrm{T}_*[\lambda : \varphi]\,.$$

This type of agent may announce something that he believes but also that he does not believe, or does not announce something he believes.

$$\mathrm{R}.\overline{7}\ \frac{\sim \mathrm{B}[\lambda:\varphi]\quad \sim \mathrm{T}_{\Diamond}[\lambda:\varphi]}{-(W_I)\lambda}\ [\mathrm{I}.-(W_I)] \qquad \mathrm{R}.\overline{8}\ \frac{\sim \mathrm{B}[\lambda:\varphi]\quad \sim \mathrm{T}_{\Box}[\lambda:\varphi]}{-(S_I)\lambda}\ [\mathrm{I}.-(S_I)]$$

$$\mathrm{R}.\overline{9}\ \frac{\sim \mathrm{T}_{\Diamond}[\lambda:\varphi]\quad +(S_I)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\mathrm{E}.+(S_I)] \qquad \mathrm{R}.\overline{10}\ \frac{\sim \mathrm{T}_{\Box}[\lambda:\varphi]\quad +(S_I)\lambda}{\mathrm{B}[\lambda:\varphi]}\ [\mathrm{E}.+(S_I)]$$

$$\mathrm{R}.\overline{11}\ \frac{\sim \mathrm{B}[\lambda:\varphi]\quad +(W_I)\lambda}{\mathrm{T}_{\Diamond}[\lambda:\varphi]}\ [\mathrm{E}.+(W_I)] \qquad \mathrm{R}.\overline{12}\ \frac{\sim \mathrm{B}[\lambda:\varphi]\quad +(W_I)\lambda}{\mathrm{T}_{\Box}[\lambda:\varphi]}\ [\mathrm{E}.+(W_I)]$$

Braggart $DR(\mathbb{A}_\lambda,\mathbb{B}_\lambda)$**.** A braggart agent λ is defined by the discrete-from relation between his announcements and beliefs: for any propositional formula φ,

$$\sim \mathrm{T}_*[\lambda : \varphi] \text{ or } \sim \mathrm{B}[\lambda : \varphi]\,.$$

This agent *only* announces what he does not believe and he does not announce what he believes. Reasoning on the negated definition (i.e., on a non-braggart agent λ for which there exists a propositional formula φ such that $T_*[\lambda : \varphi]$ and $B[\lambda : \varphi]$), we can define that if (at least once) the agent states something he believes in, then he is non-braggart.

$$R.\overline{13}\ \frac{T_\Diamond[\lambda:\varphi]\quad B[\lambda:\varphi]}{-(W_B)\lambda}\ [I.-(W_B)] \qquad R.\overline{14}\ \frac{T_\Box[\lambda:\varphi]\quad B[\lambda:\varphi]}{-(S_B)\lambda}\ [I.-(S_B)]$$

$$R.\overline{15}\ \frac{T_\Diamond[\lambda:\varphi]\quad +(S_B)\lambda}{\sim B[\lambda:\varphi]}\ [E.+(S_B)] \qquad R.\overline{16}\ \frac{T_\Box[\lambda:\varphi]\quad +(S_B)\lambda}{\sim B[\lambda:\varphi]}\ [E.+(S_B)]$$

$$R.\overline{17}\ \frac{B[\lambda:\varphi]\quad +(W_B)\lambda}{\sim T_\Diamond[\lambda:\varphi]}\ [E.+(W_B)] \qquad R.\overline{18}\ \frac{B[\lambda:\varphi]\quad +(W_B)\lambda}{\sim T_\Box[\lambda:\varphi]}\ [E.+(W_B)]$$

6.1.2 Competence

Competent ***PP*****($\mathbb{B}_\lambda$,$\mathbb{F}$).** An agent's beliefs are a subset of the true formulae. Hence, all the agent's beliefs are facts but there may be true formulae "out" of his beliefs. An agent λ is competent if, for every propositional formula φ, if $B[\lambda : \varphi]$ then $\varphi \in \mathbb{F}$.

$$R.38\ \frac{B[\lambda:\varphi]\quad \neg\varphi}{-(Co)\lambda}\ [I.\ -(Co)]$$

$$R.33\ \frac{B[\lambda:\varphi]\quad +(Co)\lambda}{\varphi}\ [E.\ +(Co)] \qquad R.\overline{19}\ \frac{\sim B[\lambda:\varphi]\quad +(Co)\lambda}{\neg\varphi}\ [E.\ +(Co)]$$

Omniscient ***PPi*****($\mathbb{B}_\lambda$,$\mathbb{F}$).** An agent λ is omniscient if the set of formulae he believes is a superset of the actually true formulae: for any propositional formula φ, if $\varphi \in \mathbb{F}$ then $B[\lambda : \varphi]$.

$$R.37\ \frac{\sim B[\lambda:\varphi]\quad \varphi}{-(O)\lambda}\ [I.\ -(O)]$$

$$R.32\ \frac{\varphi\quad +(O)\lambda}{B[\lambda:\varphi]}\ [E.\ +(O)] \qquad R.\overline{20}\ \frac{\sim B[\lambda:\varphi]\quad +(O)\lambda}{\neg\varphi}\ [E.\ +(O)]$$

Wise ***EQ*****($\mathbb{B}_\lambda$,$\mathbb{F}$).** A wise agent λ is defined by the equality between the sets of his beliefs and facts, i.e., he *only* believes in true formulae and knows *all* the true facts: for any propositional formula φ, $\varphi \in \mathbb{F}$ if and only if $B[\lambda : \varphi]$. The rules generated are exactly the rules of PP_i and PP.

Incompetent ***PO*****($\mathbb{B}_\lambda$,$\mathbb{F}$).** An incompetent agent λ is defined by the partial overlap of his beliefs with the true facts, therefore part of his belief are not facts, and this makes the agent incompetent: for any propositional formula φ, $\varphi \in \mathbb{F}$ or $B[\lambda : \varphi]$. This type of agent believes in true and false formulae, and there exist facts that he does not believe in, but he won't believe a false formula φ.

$$R.\overline{21}\ \frac{\neg\varphi\quad \sim B[\lambda:\varphi]}{-(In)\lambda}\ [I.\ -(In)]$$

$$R.\overline{22}\ \frac{\neg\varphi\quad +(In)\lambda}{B[\lambda:\varphi]}\ [E.+(In)] \qquad R.\overline{23}\ \frac{\sim B[\lambda:\varphi]\quad +(In)\lambda}{\varphi}\ [E.+(In)]$$

Ignorant *DR*($\mathbb{B}_\lambda$,$\mathbb{F}$)**.** An ignorant agent λ is defined by the discrete-from relation between true formulae and beliefs: for any propositional formula φ, $\neg\varphi \in \mathbb{F}$ or $\sim$ B$[\lambda : \varphi]$. Therefore, this agent *only* believes in false formulae.

$$\mathrm{R}.\overline{24}\ \frac{\varphi \quad \mathrm{B}[\lambda:\varphi]}{-(Ig)\lambda}\ [\mathrm{I.}\ -(Ig)]$$

$$\mathrm{R}.\overline{25}\ \frac{\varphi \quad +(Ig)\lambda}{\sim \mathrm{B}[\lambda:\varphi]}\ [\mathrm{E.}+(Ig)]$$

$$\mathrm{R}.\overline{26}\ \frac{\mathrm{B}[\lambda:\varphi] \quad +(Ig)\lambda}{\neg\varphi}\ [\mathrm{E.}+(Ig)]$$

6.1.3 Honesty

Honest *PP*($\mathbb{A}_\lambda$,$\mathbb{F}$)**.** An agent is honest if every formula he asserts is a fact, and the agent's assertion are a subset of the true formulae: for any propositional formula φ, if φ then $\mathrm{T}_*[\lambda : \varphi]$.

$$\mathrm{R}.\overline{27}\ \frac{\varphi \quad \sim \mathrm{T}_\diamond[\lambda:\varphi]}{\text{-}(W_H)\lambda}\ [\mathrm{I.}\ \text{-}(W_H)]$$

$$\mathrm{R}.\overline{28}\ \frac{\varphi \quad \sim \mathrm{T}_\square[\lambda:\varphi]}{\text{-}(S_H)\lambda}\ [\mathrm{I.}\text{-}(S_H)\ ]$$

$$\mathrm{R}.\overline{29}\ \frac{\varphi \quad +(W_H)\lambda}{\mathrm{T}_\diamond[\lambda:\varphi]}\ [\mathrm{E.}+(W_H)]$$

$$\mathrm{R}.\overline{30}\ \frac{\varphi \quad +(S_H)\lambda}{\mathrm{T}_\square[\lambda:\varphi]}\ [\mathrm{E.}+(S_H)]$$

$$\mathrm{R}.\overline{31}\ \frac{\sim \mathrm{T}_\diamond[\lambda:\varphi] \quad +(W_H)\lambda}{\neg\varphi}\ [\mathrm{E.}+(W_H)]$$

$$\mathrm{R}.\overline{32}\ \frac{\sim \mathrm{T}_\square[\lambda:\varphi] \quad +(S_H)\lambda}{\neg\varphi}\ [\mathrm{E.}+(S_H)]$$

Oracle *PPi*($\mathbb{A}_\lambda$,$\mathbb{F}$)**.** An agent λ is an oracle if, for any propositional formula φ, if $\mathrm{T}_*[\lambda : \varphi]$ then $\varphi \in \mathbb{F}$.

$$\mathrm{R}.\overline{33}\ \frac{\mathrm{T}_\diamond[\lambda:\varphi] \quad \neg\varphi}{X}\ [\mathrm{I.}\ \text{-}(W_{Or})\ ]$$

$$\mathrm{R}.\overline{34}\ \frac{\mathrm{T}_\square[\lambda:\varphi] \quad \neg\varphi}{X}\ [\mathrm{I.}\text{-}(S_{Or})\ ]$$

$$\mathrm{R}.\overline{35}\ \frac{\mathrm{T}_\diamond[\lambda:\varphi] \quad +(W_{Or})\lambda}{\varphi}\ [\mathrm{E.}+(W_{Or})]$$

$$\mathrm{R}.\overline{36}\ \frac{\mathrm{T}_\square[\lambda:\varphi] \quad +(S_{Or})\lambda}{\varphi}\ [\mathrm{E.}+(S_{Or})]$$

$$\mathrm{R}.\overline{37}\ \frac{\neg\varphi \quad +(W_{Or})\lambda}{\sim \mathrm{T}_\diamond[\lambda:\varphi]}\ [\mathrm{E.}+(W_{Or})]$$

$$\mathrm{R}.\overline{38}\ \frac{\neg\varphi \quad +(S_{Or})\lambda}{\sim \mathrm{T}_\square[\lambda:\varphi]}\ [\mathrm{E.}+(S_{Or})]$$

Right *EQ*($\mathbb{A}_\lambda$,$\mathbb{F}$)**.** An agent λ is right if, for any propositional formula φ, $\varphi \in \mathbb{F}$ if and only if $\mathrm{T}_*[\lambda : \varphi]$. We omit the rules since they are the same as for PP and PP_i.

Incorrect *PO*($\mathbb{A}_\lambda$,$\mathbb{F}$)**.** An agent λ is incorrect if, for any propositional formula φ, $\varphi \in \mathbb{F}$ or $\mathrm{T}_*[\lambda : \varphi]$. The announcements of this type of agent might be true or false, and he only announces part of the facts (i.e., a subset of the facts will never be announced by him).

$$\mathrm{R}.\overline{39}\ \frac{\neg\varphi \quad \sim \mathrm{T}_\diamond[\lambda:\varphi]}{-(W_{Ir})\lambda}\ [\mathrm{I.}\ \text{-}(\mathrm{W}_{Ir})\ ]$$

$$\mathrm{R}.\overline{40}\ \frac{\neg\varphi \quad \sim \mathrm{T}_\square[\lambda:\varphi]}{-(S_{Ir})\lambda}\ [\mathrm{I.}\text{-}(\mathrm{S}_{Ir})\ ]$$

$$\mathrm{R}.\overline{41}\ \frac{\neg\varphi \quad +(W_{Ir})\lambda}{\mathrm{T}_\diamond[\lambda:\varphi]}\ [\mathrm{E.}+(W_{Ir})]$$

$$\mathrm{R}.\overline{42}\ \frac{\neg\varphi \quad +(S_{Ir})\lambda}{\mathrm{T}_\square[\lambda:\varphi]}\ [\mathrm{E.}+(S_{Ir})]$$

$$\mathrm{R}.\overline{43}\ \frac{\sim \mathrm{T}_\diamond[\lambda:\varphi] \quad +(W_{Ir})\lambda}{\varphi}\ [\mathrm{E.}+(W_{Ir})]$$

$$\mathrm{R}.\overline{44}\ \frac{\sim \mathrm{T}_\square[\lambda:\varphi] \quad +(S_{Ir})\lambda}{\varphi}[\mathrm{E.}+(S_{Ir})]$$

False *DR*$(\mathbb{A}_\lambda,\mathbb{F})$**.** A false agent λ is defined by the discrete-form relation between true formulae and his assertions, i.e., for any propositional formula φ, $\neg\varphi \in \mathbb{F}$ or $\sim \mathrm{T}_*[\lambda : \varphi]$. In other words, everything he announces is false.

$$\mathrm{R}.\overline{45}\ \frac{\varphi \quad \mathrm{T}_\Diamond[\lambda:\varphi]}{-(W_F)\lambda}\ [\mathrm{I}.-(W_F)\ ] \qquad \mathrm{R}.\overline{46}\ \frac{\varphi \quad \mathrm{T}_\Box[\lambda:\varphi]}{-(S_F)\lambda}\ [\mathrm{I}.-(S_F)]$$

$$\mathrm{R}.\overline{47}\ \frac{\varphi \quad +(W_F)\lambda}{\sim \mathrm{T}_\Diamond[\lambda:\varphi]}\ [\mathrm{E}.+(W_F)] \qquad \mathrm{R}.\overline{48}\ \frac{\varphi \quad +(S_F)\lambda}{\sim \mathrm{T}_\Box[\lambda:\varphi]}\ [\mathrm{E}.+(S_F)]$$

$$\mathrm{R}.\overline{49}\ \frac{\mathrm{T}_\Diamond[\lambda:\varphi] \quad +(W_F)\lambda}{\neg\varphi}\ [\mathrm{E}.+(W_F)] \qquad \mathrm{R}.\overline{50}\ \frac{\mathrm{T}_\Box[\lambda:\varphi] \quad +(S_F)\lambda}{\neg\varphi}\ [\mathrm{E}.+(S_F)]$$

6.2 On the Topology of MAS

In this section, we justify the use of RCC5 instead of RCC3 or RCC8, and discuss the relation between the topology we consider and the agent types.

6.2.1 RCC3, RCC5, and RCC8

There exist three different types of RCC, based on the number of topological relations considered: RCC3, RCC5, and RCC8. RCC3 considers the three different topological relations listed in Table 6.1: *ONE*, *EQ*, and *DR*. The topological relations *EQ* and *DR* are the same as in RCC5 (see Table 6.1), whereas *ONE* defines the overlap relation between two regions with the additional constraint that the regions cannot be fully overlapping (i.e., they cannot be two exact copies of the same region).

The relation *ONE* in RCC3 is detailed in RCC5 with the relations *PP*, *PPi*, and *PO*. Hence, considering RCC5 instead of RCC3 results in a more accurate and expressive categorization of agents. However, the same reasoning cannot be applied to RCC8. In fact, even if RCC8 is more detailed than RCC5 as it considers more topological relations, the additional topological relations considered by RCC8 cannot be applied for the categorization of agents in MCL. As showed in Table 6.1, RCC8 considers tangential connections, where, informally, two tangential regions are near enough so that no other region can fit between the two (without overlapping them), but are not overlapping at any point. This is formalized by the *EC* relation. In addition, in RCC8, each of the two relations *PP* and *PPi* is detailed into tangential and non-tangential.

In our work, the elements of the three sets $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{F}$ are not ordered. In other words, we are not considering the distance between those elements (or between regions containing those elements). Hence, given any pair of (sub-)sets between $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{F}$, regardless of the sets being near or far apart between each other, we consider them as disjoint (i.e., *DR*).

6.2.2 An Upper bound on the Number of Different Types of Agents

Applying RCC over a finite number of sets, we obtain a definite number of resulting combinations. Hence, applying RCC over $\mathbb{A}_\lambda,\mathbb{B}_\lambda,\mathbb{F}$, we obtain a definite number of

Table 6.2. RCC3 composition table with respect to 3 sets. $T(X, Z) = \{DR(X, Z), EQ(X, Z), ONE(X, Z)\}$

	$DR(X, Y)$	$EQ(X, Y)$	$ONE(X, Z)$
$DR(Y, Z)$	$T(X, Z)$	$DR(X, Z)$	$DR(X, Z)$ $ONE(X,Z)$
$EQ(Y, Z)$	$DR(X, Z)$	$EQ(Y, Z)$	$ONE(Y, Z)$
$ONE(Y, Z)$	$DR(X, Z)$ $ONE(X,Z)$	$ONE(X, Z)$	$T(X, Z)$

Table 6.3. RCC5 composition table over 3 sets. The results show that there exist 54 possible relations. $T(X,Z) = \{DR(X,Z), PO(X,Z), EQ(X,Z), PP(X,Z), PPi(X,Z)\}$

	$DR(X, Y)$	$PO(X, Y)$	$EQ(X, Y)$	$PPi(X, Y)$	PP(X, Y)
$DR(Y, Z)$	$T(X, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PP(X, Z)$	$DR(X, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PP(X, Z)$	$DR(X, Z)$
$PO(Y, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PP(X, Z)$	$T(X, Z)$	$PO(X, Z)$	$PO(X, Z)$ $PPi(X, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PP(X, Z)$
$EQ(X, Z)$	$DR(X, Z)$	$PO(X, Z)$	$EQ(X, Z)$	$PPi(X, Z)$	$PP(X, Z)$
$PP(Y, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PP(X, Z)$	$PO(X, Z)$ $PP(X, Z)$	$PP(X, Z)$	$PO(X, Z)$ $EQ(X, Z)$ $PP(X, Z)$ $PPi(X, Z)$	$PP(X, Z)$
$PPi(Y, Z)$	$DR(X, Z)$	$DR(X, Z)$ $PO(X, Z)$ $PPi(X, Z)$	$PPi(X, Z)$	$PPi(X, Z)$	$T(X, Z)$

different types of agents. In this section, we show the general upper bound on the number of different agents with respect to the type of RCC (RCC5, RCC3 or RCC8) considered.

The general formula to calculate the number of different types of agents is $r^{\binom{n}{k}}$, where r is the number of relations with arity k, between n different sets, where r^e is the number of permutation of r relations over e elements with repetitions, with e being the number of k-ary combinations of n sets, $\binom{n}{k}$. In our case, $\binom{n}{k} = 3$ since we consider 3 sets ($\mathbb{A},\mathbb{B},\mathbb{F}$), and all the relations considered in the RCC are binary. Hence, using RCC5 (with five different spatial relations) over three sets, we can theoretically define up to 125 different type of agents. However, only 54 of the 125 (as showed in [13] and derived by the composition table of RCC5 in Table 6.3) combinations are topologically correct with respect to the definition of the relations of RCC5. Generalizing to all the RCCs, in Table 6.4 we calculate the number of different agents with respect to all the variations of RCC (i.e., with 3, 5 or 8 spatial relations).

- *RCC3* — theoretical: $3^3 = 27$, correct: 15. (see Table 6.2): 15
- *RCC5* — theoretical: $5^3 = 125$, correct: 54. (see Table 6.3): 54

	Theoretical	**Correct**
RCC3	$3^3 = 27$	15
RCC5	$5^3 = 125$	54
RCC8	$8^3 = 512$	193

Table 6.4. Number of agents with respect to different RCC

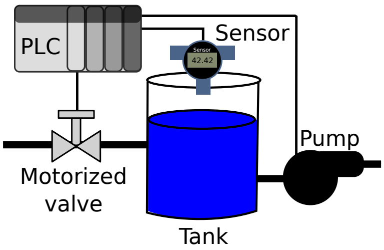


Fig. 6.2. Representation of the test case

- *RCC8* — theoretical: $8^3 = 512$, correct: 193. (see Table 6.7): 193

Hence, even if considering a different number of sets than the three $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{F}$ exponentially affects the number of theoretical agents, the application of RCC downscales that number of a factor that ranges from 1.8 to 2.5. In addition, using RCC5 we consider 3.6 times more (different) types of agents than RCC3, but using RCC8 would allow us to consider 3.5 times more different agents.

6.3 Systems of Communicating Agents

In this section, we show that both the framework and the categorization of agents that we have given can be applied to reason about the security of CPS.

6.3.1 Cyber-Physical Systems

We use the term CPS to refer to systems that consist of networked embedded systems, which are used to sense, actuate, and control physical processes. Examples of CPS include industrial water treatment facilities and power plants. CPS have seen a rapid increase in automation and connectivity, which threatens to increase their vulnerability to malicious attacks. Let us now use our approach to address the problem of defining security-related attack states for CPS.

Description of the Case Study. Similarly to [14, 22], we consider a CPS (depicted in Fig. 6.2) to be composed by five agents:

- A *tank* containing water.
- A *controller* (e.g., a PLC) that controls the water level so that the tank does not (underflow or) overflow.
- A *water level indicator* (e.g., a Sensor) that communicates the readings of the level of the water inside the tank to the PLC.
- A *motorized valve* and a *pump* that (controlled by the PLC) regulate the inflow and outflow of water respectively.

Mapping $\mathbb{A}$, $\mathbb{B}$, and $\mathbb{F}$ to CPS. It is possible that the three sets $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{F}$ contain at the same time different formulae that contain each element of the topological space φ. Hence, every assertion and belief must be objective (since it can be part of $\mathbb{F}$). This implies that formulae like $\varphi := highLevel(tank, water)$ cannot be considered in

Table 6.5. Example of attack states for the water level sensor

State of the sensor	$(\mathbb{A},\mathbb{B})$	$(\mathbb{B},\mathbb{F})$	$(\mathbb{A},\mathbb{F})$
optimal	*EQ*	*EQ*	*EQ*
sensor compromised	*EQ*	*DR*	*DR*
communication compromised	*DR*	*EQ*	*DR*
fully compromised	*DR*	*DR*	*DR*

our reasoning since “high” is considered to be subjective. In contrast, we can use objective formulae such as $\varphi := \mathit{level}(\mathit{tank}, \mathit{20L})$.

When considering a CPS (and security systems in general, e.g., security protocols) as a MAS, the message exchange between different agents can be formalized by means of *assertions*. In addition, redundant channels are often employed to reduce security treats (or assertions are required over multiple channels as, e.g., in two-factor authentication) and then it is fair to assume that assertions can be done over single or multiple channels. Finally, the inspection of the memory of any software/hardware of the CPS (supposing a white-box analysis) reveals the actual *beliefs*, while the *facts* in a CPS are defined by the physical laws of the physics. We can summarize our mapping as follows:

- $\mathbb{A}_\lambda$ defines the values communicated by the agent λ.
- $\mathbb{B}_\lambda$ defines the computational results of the agent λ.
- $\mathbb{F}$ defines the environmental values, i.e., the real values of the system.

6.3.2 Single-Channel Attack states

We are now in a position to show that we can directly apply our topological categorization to any agent in our CPS. For simplicity, we first use only the RCC5 relations *EQ* and *DR*, and then extend our results to all RCC5 relations.

Optimal System Status. Suppose that the tank contains 20L of water, e.g., $\mathit{level}(\mathit{tank}, \mathit{20L}) \in \mathbb{F}$, where *level* is a predicate, and *tank* and *20L* are propositional constants. For the sake of simplicity, we also suppose that the system is in idle (both the motorized valve and the pump are off). When the system is *not* compromised, the sensor correctly computes the level of the water in the tank (e.g., $\mathit{level}(\mathit{tank}, \mathit{20L}) \in \mathbb{B}_{\mathit{sensor}}$) and correctly communicates to the PLC the computed value of water in the tank (e.g., $\mathit{level}(\mathit{tank}, \mathit{20L}) \in \mathbb{A}_{\mathit{sensor}}$). We can then define the optimal status of the sensor as the triple $\langle EQ(\mathbb{A}_{\mathit{sensor}}, \mathbb{B}_{\mathit{sensor}}), EQ(\mathbb{B}_{\mathit{sensor}}, \mathbb{F}), EQ(\mathbb{A}_{\mathit{sensor}}, \mathbb{F})\rangle$.

System Under Attack. Suppose that the sensor is communicating wrong values to the PLC (i.e, $DR(\mathbb{A}_{\mathit{sensor}}, \mathbb{F})$). As showed in Table 6.5, we have three mutually exclusive cases:

1. The sensor is working properly $EQ(\mathbb{B}_{\mathit{sensor}}, \mathbb{F})$, therefore (topologically) the communication between the sensor and the PLC has been compromised, i.e., $DR(\mathbb{A}_{\mathit{sensor}}, \mathbb{B}_{\mathit{sensor}})$.
2. The communication between the sensor and the PLC has not been compromised $EQ(\mathbb{A}_{\mathit{sensor}}, \mathbb{B}_{\mathit{sensor}})$, therefore the sensor is *not* sending what it computes $DR(\mathbb{B}_{\mathit{sensor}}, \mathbb{F})$.

3. Both the communication and the sensor have been compromised.

As a consequence of the discussion in Section 6.2.2, between the optimal and the fully compromised status of the sensor there must be 52 other different statuses. We can generalize the attack states into three main categories, as follows:

- RCC5($\mathbb{A}$,$\mathbb{B}$) expresses the relation between the values communicated and the ones computed by an agent.
- RCC5($\mathbb{B}$,$\mathbb{F}$) expresses the relation between the values computed and the true environmental values.
- RCC5($\mathbb{A}$,$\mathbb{F}$) expresses the relation between the values communicated and the true environmental values.

Defense mechanisms that check sudden changes in physical readings (see [29] for an example of how this is defined in MAS with logical systems) are often adopted in CPS. To bypass the security mechanisms, during an attack, the optimal status will likely pass through most of the 52 intermediate statuses.

6.3.3 Multiple-Channel Attack States

A countermeasure often applied in CPS (but not limited to CPS) is the implementation of redundant channels. As proposed in [23], in our case study one could implement a dedicated system that interprets the readings of the sensor and directly closes the motorized valve if an upper threshold is reached. We can leverage the modal operators to define such communications and to define even more sophisticated attack states. For example, given a state $\mathbb{A}_{sensor}, \mathbb{B}_{sensor}, \mathbb{F}$ in MCL, we can check if one or all the channels that the sensor uses to communicate with the PLC have been compromised, as defined in (1) and (2) respectively:

$$\{\mathbb{A}_{sensor}, \mathbb{B}_{sensor}, \mathbb{F}\} \vdash -(S_{Fair})_{sensor} \tag{6.1}$$

$$\{\mathbb{A}_{sensor}, \mathbb{B}_{sensor}, \mathbb{F}\} \vdash -(W_{Fair})_{sensor} \tag{6.2}$$

Based on the approach we have proposed in this thesis, we can formalize the optimal/attack states of a CPS, reason on the properties of the CPS by means of prejudices in MCL, and obtain therefore a control upon the concept of redundancy as expressed above. Our approach is not specific to CPS but can potentially be applied to any MAS (as long as the elements of the topological space are objective).

6.3.4 Channels

In order to extend our work to consider systems of communicating agents, in this section, we introduce the notion of channel. Informally, a channel identifies the place where the communication takes place. In the following of this section we define “one-directional” and “bi-directional” channels.

Mono-directional channels are channels where the communications (i.e., the exchange of information) between two agents flows only from one agent to the other but not vice versa. We denote mono-directional channels with →. For example, $\lambda_S \rightarrow \lambda_R$ defines a mono-directional channel between the two agents λ_S, λ_R where only λ_S

communicates information to λ_R and λ_R only receives messages but does not communicate to λ_S. More formally, in MCL, we first categorize the two agents λ_S and λ_R with respect to the sets:$\mathbb{A}_S, \mathbb{B}_S, \mathbb{A}_R, \mathbb{B}_R, \mathbb{F}$; as follows.

$$\lambda_S = \{RCC5_1(\mathbb{A}_{\lambda_S}, \mathbb{B}_{\lambda_S}), RCC5_2(\mathbb{B}_{\lambda_S}, \mathbb{F}), RCC5_3(\mathbb{A}_{\lambda_S}, \mathbb{F})\}.$$
$$\lambda_R = \{RCC5_4(\mathbb{A}_{\lambda_R}, \mathbb{B}_{\lambda_R}), RCC5_5(\mathbb{B}_{\lambda_R}, \mathbb{F}), RCC5_6(\mathbb{A}_{\lambda_R}, \mathbb{F})\}.$$

However, the relations between $\mathbb{A}, \mathbb{B}, \mathbb{F}$ does not allow us to define the channel between the two agents. Therefore, in order to model the channel, we introduce the following notation (defined in MCL):

Definition 1 Mono-directional channel. *A mono-directional channel is defined by the message sent and received over the channel. $\mathbb{A}_{\lambda_S,\lambda_R}$, defines the sending of information over a mono-directional channel between λ_S and λ_R, written as $\lambda_S \rightarrow \lambda_R$. $\mathbb{A}_{\lambda_S,\lambda_R}$ is defined as a (improper) subset of the assertions of λ_S, i.e. $PP(\mathbb{A}_{\lambda_S,\lambda_R}, \mathbb{A}_{\lambda_S}) \vee EQ(\mathbb{A}_{\lambda_S,\lambda_R}, \mathbb{A}_{\lambda_S})$ (since λ_S may communicate with other agents). The receipt of messages over a mono-directional channel is defined as the topological boundary $\mathbb{B}_{S,R}$ (improper) subset of the beliefs of the receiver (i.e., the beliefs generated by the message exchange and by computations over the set of beliefs): $PP(\mathbb{B}_{S,R}, \mathbb{B}_R) \vee EQ(\mathbb{B}_{S,R}, \mathbb{B}_R)$.*

As an example, in an ideal system where there exist only two agents λ_S and λ_R that communicates over a mono-directional channel from $\lambda_S \rightarrow \lambda_R$, and where the agent R extends his beliefs only with the message it receives, the system of communicating agent is defined by the tuple $\langle \mathbb{A}_{\lambda_s,\lambda_R}, \mathbb{B}_{\lambda_S}, \mathbb{B}_{\lambda_S,\lambda_R}, \mathbb{F} \rangle$. We can exclude $\mathbb{A}_{\lambda_R}$ since λ_R does not communicate with any other agent.

Definition 2 System of communicating agents $\mathcal{S}_{abf}$.

$$\mathcal{S}_{abf} \triangleq \langle \mathbb{A}_{\lambda_0\lambda_j}, \ldots, \mathbb{A}_{\lambda_n\lambda_m}, \mathbb{B}\lambda_0, \ldots, \mathbb{B}\lambda_k, \mathbb{F} \rangle$$

where $j,n,m,k \in \mathbb{N}$ are indexes identifying different agents (i.e., $j \neq n \neq m \neq k$).

As a more concrete example, we consider the architecture in Figure 6.3.4 where a sender and a receiver (whose behavior is depicted as a transition system) are communicating through a mono-direction channel *sender* → *receiver*. The sender senses the data of a physical component. The relation between the beliefs of the sender (i.e., what the sender compute) and the set of facts (i.e., $\mathbb{F}$) determines the relation between the real physical status of the physical component and what the sender senses. For example, if the physical component is a tank containing 50l of water, the sensor may sense the level of the water inside the tank. The sender outputs (to the port indicated with O in the figure) the data on the $A_{s,r}$ channel and the receiver only receives input from the sender (from the port indicated with I in the figure). The rest of the relation are intuitive and depicted in Figure 6.3.4:

- $(\mathbb{A}_{S,R}, \mathbb{F})$, expresses the relation between the information exchanged in the channel and the true fact of the physical components of the system.
- $(\mathbb{A}_{S,R}, \mathbb{B}_S)$, expresses the relation between the information exchanged over the channel and the ones computed by the sender (i.e., the sensed data).

- $(\mathbb{A}_{S,R}, \mathbb{B}_R)$, since there is no other source of information for the receiver, this relation expresses the difference between what the receiver receives and produces by computation.
- $(\mathbb{B}_S, \mathbb{B}_R)$, expresses the difference between the computed data by S and R. In general, agents (i.e., components) that implements different behaviors, generates different beliefs even if the initial set of beliefs is the same. However, there exist specific configurations (e.g., components implementing the same logic/behavior) where this relation can be interesting to analyze. For example, in a redundant setting the relation between beliefs of the two agents (or systems of agents) can be analyzed to understand if any misalignment occurred.
- $(\mathbb{B}_R, \mathbb{F})$, expresses the difference between the data computed by R and the true fact of the physical components of the system.

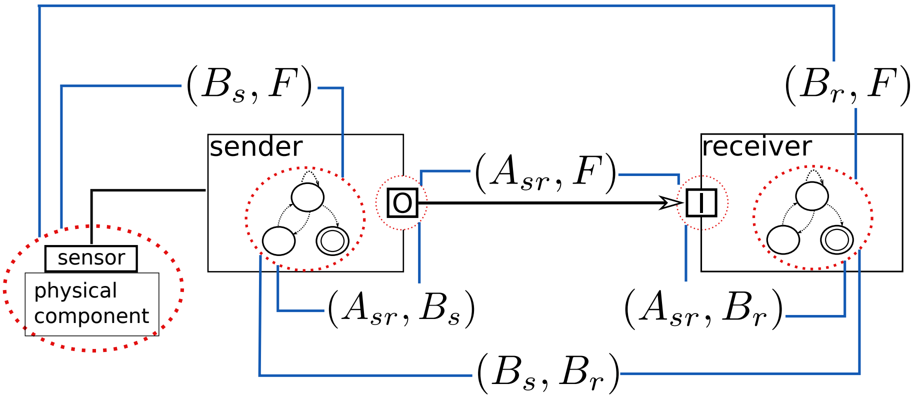


Fig. 6.3. The architecture of $\lambda_S \rightarrow \lambda_R$.

For the definition of bi-directional channels we rely on the definition of mono-directional channel, and a bi-directional channel is defined as the union of two (opposite) mono-directional channels.

Definition 3 Bi-directional channel *A bi-directional channel $\lambda_i \leftrightarrow \lambda_j$ between the two agents λ_i, λ_j is defined as the two mono-directional channels $\lambda_0 \rightarrow \lambda_1$, $\lambda_1 \rightarrow \lambda_0$.*

6.3.5 Channel properties

We define different security properties over a mono-directional channel (the extension to a bi-directional channel is straightforward given that a bi-directional channel is defined as two mono-directional channels).

We first recall the informal definition of CIA (Confidentiality, Integrity, Availability) security properties:

- Confidentiality of a channel guarantees that no agent but the intended recipient will be able to understand the information exchanged over the channel.

- Availability of information/service over a channel guarantees that whenever an agent want to retrieve an information/service, the information/service will be provided.
- Integrity of a channel guarantees that if the information exchanged over that channel are modified, these modification are detected.

The RCC5 calculus allow us to reason on the security configuration of a mono-directional channel as we describe in Table 6.6. In particular, the relation between the assertions sent to the mono-directional channel (i.e., $\mathbb{A}_{\lambda_S,\lambda_R}$) and the ones received (i.e., $\mathbb{B}_{\lambda_S,\lambda_R}$).

Definition	Relation	Representation
Ideal (no attacks)	EQ($\mathbb{A}_{\lambda_S,\lambda_R}$,$\mathbb{B}_{\lambda_S,\lambda_R}$)	$\lambda_S \longrightarrow \lambda_R$
Availability violation (e.g., Denial of service attack)	DR ($\mathbb{A}_{\lambda_S,\lambda_R}$,$\mathbb{B}_{\lambda_S,\lambda_R}$)	$\lambda_S / \longrightarrow \lambda_R$
Confidentiality & Integrity violation (e.g., Man-in-the-middle attack)	PO($\mathbb{A}_{\lambda_S,\lambda_R}$,$\mathbb{B}_{\lambda_S,\lambda_R}$)	$\begin{tikzcd} & {} \\ {} \arrow[ur] & \\ \lambda_S \arrow[r] & \lambda_R \\ {} \arrow[ur] & \end{tikzcd}$
Confidentiality violation: the adversary intercepts messages before they reach their destination	PPi($\mathbb{A}_{\lambda_S,\lambda_R}$,$\mathbb{B}_{\lambda_S,\lambda_R}$)	$\begin{tikzcd} & {} \\ {} \arrow[ur] & \\ \lambda_S \arrow[r] & \lambda_R \end{tikzcd}$
Integrity violation: someone fakes to be the sender	PP($\mathbb{A}_{SR}$,$\mathbb{B}_R$)	$\begin{tikzcd} \lambda_S \arrow[r] & \lambda_R \\ {} \arrow[ur] & \end{tikzcd}$

Table 6.6. Channel Properties

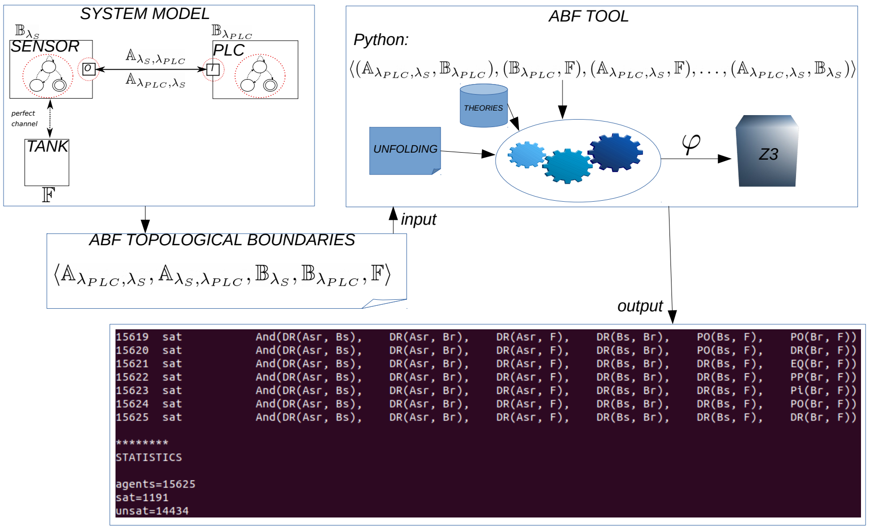
Table 6.7. RCC8 composition table over 3 sets. The results shows that there exist 193 possible relation. *T*(*X*,*Z*)={*DR*(*X*, *Z*),*PO*(*X*,*Z*),*EQ*(*X*,*Z*),PP(*X*,*Z*),*PPi*(*X*,*Z*)}

	DR(*X*,*Y*)	*EC*(*X*,*Y*)	*PO*(*X*,*Y*)	*TPP*(*X*,*Y*)	*NTTP*(*X*,*Y*)	*TPPi*(*X*,*Y*)	*NTPPi*(*X*,*Y*)	*EQ*(*X*,*Y*)
DR(*Y*,*Z*)	*T*	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*DR*(*X*,*Z*)	*DR*(*X*,*Z*)	*DR*(*X*,*Z*)
EC(*Y*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) TPPI *NTPPi*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *TPPi*(*X*,*Z*) *EQ*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*PO*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*)	*DR*(*X*,*Z*)	*EC*(*X*,*Z*)
PO(*Y*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*T*	*PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*PO*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*DR*(*X*,*Z*)	*PO*(*X*,*Z*)
TPP(*Y*,*Z*)	*DR*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*NTPP*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *TPPi*(*X*,*Z*) *EQ*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*TPP*(*X*,*Z*)
NTTP(*Y*,*Z*)	*DR*(*X*,*Z*)	*DR*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*NTPP*(*X*,*Z*)	*NTPP*(*X*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*T*	*NTPP*(*X*,*Z*)
TPPi(*Y*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPP*(*X*,*Z*) *TPPi*(*X*,*Z*) *EQ*(*X*,*Z*)	*PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*)	*TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*NTPPi*(*X*,*Z*)	*TPPi*(*X*,*Z*)
NTPPi(*Y*,*Z*)	*DR*(*X*,*Z*) *EC*(*X*,*Z*) *PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*)	*PO*(*X*,*Z*) *TPP*(*X*,*Z*) *NTPP*(*X*,*Z*) *TPPi*(*X*,*Z*) *NTPPi*(*X*,*Z*) *EQ*(*X*,*Z*)	*NTPPi*(*X*,*Z*)	*NTPPi*(*X*,*Z*)	*NTPPi*(*X*,*Z*)
EQ(*Y*,*Z*)	*DR*(*X*,*Z*)	*EC*(*X*,*Z*)	*PO*(*X*,*Z*)	*TPP*(*X*,*Z*)	*NTPP*(*X*,*Z*)	*TPPi*(*X*,*Z*)	*NTPPi*(*X*,*Z*)	*EQ*(*X*,*Z*)

7

ABF Tool

In this section, we describe the implementation and the necessary theoretical background of the ABF tool. The tool is general enough to be used as a basis for any implementation of provers (e.g., theorem provers) on topological boundaries in a topological space. In particular, we created a tool to identify all (or a selection of) the possible configurations of a system of agents defined satisfiable in the ABF theory. We implemented the RCC5 theory in the Z3 SMT solver (in Python) and defined ABF agents as regions in a topological space. This allow us to quantitatively reasoning on systems of communicating agents, studying the possible secure and non-secure configurations of a system of communicating agents. The full implementation of the tool is reported in Appendix A. The basic idea is to encode in the SMT-LIB language (input language of most SMT solver) the concept of topological boundary (a subset of a topological space), the RCC5 calculus, and then to encode agents as regions.


SYSTEM MODEL
$\mathbb{B}_{\lambda_S}$
SENSOR
$\mathbb{B}_{\lambda_{PLC}}$
PLC
O
I
$\mathbb{A}_{\lambda_S,\lambda_{PLC}}$
$\mathbb{A}_{\lambda_{PLC},\lambda_S}$
perfect channel
TANK
$\mathbb{F}$
ABF TOOL
Python:
$\langle(\mathbb{A}_{\lambda_{PLC},\lambda_S},\mathbb{B}_{\lambda_{PLC}}),(\mathbb{B}_{\lambda_{PLC}},\mathbb{F}),(\mathbb{A}_{\lambda_{PLC},\lambda_S},\mathbb{F}),\ldots,(\mathbb{A}_{\lambda_{PLC},\lambda_S},\mathbb{B}_{\lambda_S})\rangle$
THEORIES
UNFOLDING
φ
Z3
input
ABF TOPOLOGICAL BOUNDARIES
$\langle\mathbb{A}_{\lambda_{PLC},\lambda_S},\mathbb{A}_{\lambda_S,\lambda_{PLC}},\mathbb{B}_{\lambda_S},\mathbb{B}_{\lambda_{PLC}},\mathbb{F}\rangle$
output

```
15619  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    PO(Bs, F),    PO(Br, F))
15620  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    PO(Bs, F),    DR(Br, F))
15621  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    DR(Bs, F),    EQ(Br, F))
15622  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    DR(Bs, F),    PP(Br, F))
15623  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    DR(Bs, F),    Pi(Br, F))
15624  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    DR(Bs, F),    PO(Br, F))
15625  sat          And(DR(Asr, Bs),    DR(Asr, Br),    DR(Asr, F),    DR(Bs, Br),    DR(Bs, F),    DR(Br, F))

********
STATISTICS

agents=15625
sat=1191
unsat=14434
```

Fig. 7.1. The ABF tool.

7.1 Implementation of the Region Connection Calculus

In Section 6, we defined agents as a tuple of relations between three topological boundaries A,B,F. We define a topological boundary a `Sort` in Z3 in order to define the theory of a topological space over the domain (i.e., the sort) of topological boundary. We do not refer to topology or topological space explicitly in the implementation of the tool and the notion of boundary could be abstracted into the more high-level concept of region. However, this would not change the detail of the results but would have slightly complicated the implementation.

In Listing 7.1 we define the topological boundary as a sort and three generic topological boundaries `X,Y,X` that will be used afterward in this section to define the RCC5 calculus.

Listing 7.1. Definition of topological boundary as a sort

```
Boundary = DeclareSort('Boundary')

X = Const('X', Boundary)
Y = Const('Y', Boundary)
Z = Const('Z', Boundary)
```

The RCC5 is defined by the five relations: equality, discrete from, partial overlap, proper part, and the inverse of proper part. Those five relations are, in turn, defined over the concept of parts (mereological representation) or from the more basic concept of connected boundaries (topology). We define here the RCC5 over the concept of connected spaces but, for performance purposes, our results are based on the mereological implementation. In Listing 7.2, we define the terms that identify the RCC5 relations as Boolean (uninterpreted) functions $R(X,X) \rightarrow \{True, False\}$ that takes as input two topological boundaries X, where R can be any of the following relations: connects with (`C`), part of (`P`), overlaps (`O`), equal to (`EQ`), discrete from (`DR`), partial overlap (`PO`), proper part of (`PP`), inverse of proper part of (`PPi`).

Listing 7.2. RCC5 relations

```
C   = Function('C', Boundary, Boundary, BoolSort())
P   = Function('P', Boundary, Boundary, BoolSort())
O   = Function('O', Boundary, Boundary, BoolSort())
EQ  = Function('EQ', Boundary, Boundary, BoolSort())
DR  = Function('DR', Boundary, Boundary, BoolSort())
PO  = Function('PO', Boundary, Boundary, BoolSort())
PP  = Function('PP', Boundary, Boundary, BoolSort())
PPi= Function('PPi', Boundary, Boundary, BoolSort())
```

We define in Listing 7.3, the semantics of the various relations, where X,Y,Z are topological boundaries, as follows:

- the parthood relation connects with (primitive binary inclusion relation) as a reflexive (i.e., $\forall X, C(X,X)$ where X is a topological boundary) and symmetric (i.e., $\forall X,Y\ C(X,Y) \rightarrow C(Y,X)$) relations
- Part of (P): $\forall X,Y\ P(X,Y) \leftrightarrow \forall Z C(Z,X) \rightarrow C(Z,Y)$
- Overlaps (O): $\forall X,Y\ O(X,Y) \leftrightarrow \exists Z P(Z,X) \wedge P(Z,Y)$
- Equal to (EQ): $\forall X,Y\ EQ(X,Y) \leftrightarrow P(X,Y) \wedge P(Y,X)$

- Discrete from (FR): $\forall X,Y\ DR(X,Y) \leftrightarrow \neg O(X,Y)$
- Partial overlap (PO): $\forall X,Y\ PO(X,Y) \leftrightarrow O(X,Y) \wedge \neg P(X,Y) \wedge \neg P(Y,X)$
- Proper part of (PP): $\forall X,YPP(X,Y) \leftrightarrow P(X,Y) \wedge \neg P(Y,X)$
- Inverse of proper part of (PPi) $\forall X,YPPi(X,Y) \leftrightarrow P(Y,X) \leftrightarrow \neg P(X,Y)$

Listing 7.3. Implementation of the semantics of RCC5

```
reflexivity=ForAll(X, C(X, X))
symmetry=ForAll([X,Y], Implies(C(X, Y), C(Y,X)))
part_of=ForAll([X,Y], P(X,Y) == ForAll(Z, Implies(C(Z,X),
    C(Z,Y))))
overlaps=ForAll([X,Y], O(X,Y) == Exists(Z, And(P(Z,X),
    P(Z,Y))))
equal_to=ForAll([X,Y], EQ(X,Y) == And(P(X,Y), P(Y,X)))
discrete_from=ForAll([X,Y], DR(X,Y) == Not(O(X,Y)))
partial_overlap=ForAll([X,Y], PO(X,Y) == And(O(X,Y),
    Not(P(X,Y)), Not(P(Y,X))))
proper_part=ForAll([X,Y], PP(X,Y) == And(P(X,Y),
    Not(P(Y,X))))
proper_part_i=ForAll([X,Y], PPi(X,Y) == And(P(Y,X),
    Not(P(X,Y))))
```

In Listing 7.4 we create (lines 1 and 2) a new Z3 solver variable s that contains a conjunction of all the formulas defining RCC5 (i.e., the theory of RCC5). In lines 4–6 we create three constants of type Boundary representing the three sets A, B, and F. We then create a list (a Python dictionary) of the five RCC5 relations in line 8 so that, in lines 15, 17, and 19 we can loop over all the possible combinations of RCC5 relations on the three sets A, B, and F (i.e., R(A,B), R(B,F), R(A,F) where R is any RCC5 relation). For each combination we create an agent (line 22) that we add to the solver (line 24, 25). We then check the satisfiability of the agent (i.e., of the combination of RCC5 relations over the three sets A, B, F). The remaining lines of code (from line 27 on) prints the output and keeps counters on how many configurations are satisfiable and how many are not.

Listing 7.4. Implementation of Agents

```
1  s = Solver()
2  s.add(And(reflexivity, symmetry, part_of, proper_part,
       proper_part_i, partial_overlap, discrete_from,
       equal_to, overlaps))
3
4  A = Const('A', Boundary)
5  B = Const('B', Boundary)
6  F = Const('F', Boundary)
7
8  rcc5={'eq':EQ, 'pp':PP, 'ppi':PPi, 'po':PO, 'dr':DR}
9
10 counter=1
11 counter_sat=0
12 counter_unsat=0
13 counter_unknown=0
14
```

```
for i in rcc5:
  r1=rcc5[i](A,B)
    for j in rcc5:
      r2=rcc5[j](B,F)
      for k in rcc5:
        r3=rcc5[k](A,F)

        agent=And(r1,r2,r3)

        s.push()
        s.add(agent)
        check=s.check()
        offset1=' '*(5-len(str(counter)))
        offset2=' '*(12-len(str(check)))
        print("%d %s %s %s %s"%(counter, offset1, check,
            offset2, agent))

        if(check == unsat):
            counter_unsat+=1
        if(check == unknown):
        counter_unknown+=1
        if(check == sat):
        counter_sat+=1

        s.pop()
        counter+=1

print("\n********\nSTATISTICS\n\nagents=%d
    \nsat=%d\nunsat=%d\nunknown=%d"%(counter-1,
    counter_sat,counter_unsat,counter_unknown))
```

As mentioned before in this section we also implemented a slightly modified version in which we used the PO relation as the basic building block of RCC5 (i.e., mereotopological representation) instead of starting from the notion of “connected with”. The PO relation has been implemented as shown in Listing 7.5. The two variants do not differ in terms of results.

Listing 7.5. Alternative implementation of PO relation

```
reflexivity=ForAll(X, P(X,X))
antisymmetry=ForAll([X,Y], Implies(And(P(X,Y),P(Y,X)),
    X==Y))
transitivity=ForAll([X,Y,Z], Implies(And(P(X,Y),P(Y,Z)),
    P(X,Z)))
```

7.2 Experiment and Results

We first used our tool to understand how many different configurations of systems of communicating agents with respect to our theory. The tool, as developed in Section 7 do not terminate even for a system with 2 agents with unidirectional communication.

SM	Sets	Agents	Permut.	Config. N.	Sat
λ_S	$\mathbb{A},\mathbb{B},\mathbb{F}$	1	5	5^3=125	54
$\lambda_S \rightarrow \lambda_R$	$\mathbb{A}_{\lambda_S \lambda_R},\mathbb{B}_{\lambda_S},\mathbb{B}_{\lambda_R},\mathbb{F}$	2	6	5^6=15.625	1.191
$\lambda_S \leftrightarrow \lambda_R$	$\mathbb{A}_{\lambda_S \lambda_R},\mathbb{A}_{\lambda_R \lambda_S},\mathbb{B}_{\lambda_S},\mathbb{B}_{\lambda_R},\mathbb{F}$	2	10	5^{10}=9.765.625	51.345

Table 7.1. ABF tool System Model configurations

We then modified the code in order to unfold the quantifiers over the finite number of sets defining the system of communicating agents. With this tuning, we managed to enumerate the configurations in Table 7.1. Showing how rapidly the number of configurations grows even if we just introduce one (unidirectional) channel, i.e. from around 1200 to more than 51000 configurations. In the first row (with 1 agent), we recall that one agent has 54 different configurations.

7.2.1 Two agents - unidirectional channel

	Categorization $Agents_S$			$Channel_{SR}$
	Collaboration	Competence	Honesty	Constraint
	$(\mathbb{A}_{SR},\mathbb{B}_S)$	$(\mathbb{B}_S,\mathbb{F})$	$(\mathbb{A}_{SR},\mathbb{F})$	$(\mathbb{A}_{SR},\mathbb{B}_{S,R})$
PP	Sincere	Competent	Honest	No Authentication
PPi	Collaborative	Omniscient	Oracle	Interception
EQ	Fair	Wise	Right	Ideal
PO	Saboteur	Incompetent	Incorrect	Vulnerable
DR	Braggart	Ignorant	False	No communication

Table 7.2. ABF – Properties of agents

An unidirectional channel can only have five different configurations in RCC5. In Table 7.2, we report the different configurations of the channel compared to the collaboration, competence, and honesty of the agent. Of the 1191 different configurations we investigated those where a Man-in-the-middle attack (MITM) is present, i.e., adding the constraint that Only some of the informations reaches the intended recipient: $PO(Asr,Br)$. The ABF tool identified 318 different configurations of the system. These configurations represent the different MITM specializations. For example, one specialization of the MITM is the selective forwarding attack where an attacker drops some of the packets traveling over the network. In order to study the selective forwarding attack we introduced the following constraints.

- (MITM) Only some of the informations reaches the intended recipient: $PO(\mathbb{A}_{S,R},\mathbb{B}_{S,R})$
- Anything the sender believes is fact: $PP(\mathbb{B}_S,\mathbb{F})$
- Anything the receiver believes is a fact: $PP(\mathbb{B}_{S,R},\mathbb{F})$
- Anything the sender announce is a fact: $PP(\mathbb{A}_{S,R},\mathbb{F})$

Since there is no believes that do not correspond to a fact and there is no communication from the sender to the receiver that is not a fact but not all the information reaches the intended recipient, it means that part of the communication was dropped

1	EQ(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PO(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
2	PP(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	Pi(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
2	PP(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PO(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
4	Pi(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PP(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
5	Pi(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PO(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
6	Pi(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	DR(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
7	PO(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	EQ(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
8	PO(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PP(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
9	PO(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	Pi(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
10	PO(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PO(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
11	PO(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	DR(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
12	DR(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PP(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
13	DR(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	PO(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)
14	DR(Asr, Bs)	PO(Asr, Bsr)	PP(Asr, F)	DR(Bs, Bsr)	PP(Bs, F)	PP(Bsr, F)

Table 7.3. Selective forwards attack states

(i.e., selective forwarding). Our tool returns the 14 configurations summarized in Table 7.3 where the four contraints for the selective forwards are fixed and the relations `Asr,Bs` and `Bs,Bsr` changes. The relation `Bs, Bsr` are meaningful only when the sender and receiver implement the same behavior, for example when an attacker performs traffic analysis and compares the behavior of one of the nodes he controls with another node he does not control. The relation between `Asr` and `Bs` tells us how much of the messages sent by `S` reaches the receiver, as follows.

- `EQ`, all the messages sent on the channel reaches the receiver and the receiver only receives messages from the sender.
- `PP`, all the message sent on the channel reaches the receiver but the receiver receives messages from other sources.
- `PPi`, only some of the messages sent on the channel reaches the receiver (some are lost) but the receiver does not receive messages from other sources.
- `PO`, part of the messages sent by the sender are lost, part are received, and the receiver receives messages from other sources.
- `DR`, the receiver only receives messages from other sources.

Part IV

State of the art

There is a rather long research stream on the problem of aligning beliefs and announcements, that starts from Dynamic Epistemic Logic. See [31], for a general framework analysis in the early stage of Public Announcement Logics (PAL), and recent development in [2, 3, 28]. Although these scholars have deeply dealt with the problem of announcements, they have made a very strong, and clearly oversimplified assumption: agents are always truthful and sincere. As a consequence, an observer on a communication channel always trusts the agents making announcements on that channel. In recent developments on PAL, the possibility that an announcement is made producing changes in beliefs of agents is provided in the form of belief revision operators: whenever an agent belief contrasts with what announced, he revises his knowledge base [3].

On the other hand, there have been meany scholars who concentrated their attention upon the ways in which agents communicate false announcements, as recently analysed in [32].

Back to early stage investigations on Public Announcements, the idea that someone could forecast others' lies is incorporated within the logic itself [4]. A more recent study focused on communication, and introduces, in a different way with respect to the approach we adopted here, a notion of channel [26]. The framework is Dynamic Epistemic Logic. Authors consider only truthful communications. A message can be sent only if the sender knows that the message is true. Agents cannot lie. Agents communicate by sending messages to a group of other agents. This is distinct from passing on channels where: (i.) The agent can choose on which channels say what, (ii.) the agent has not control on who is looking at those channels, he thus does not have control on who can access his announcement. The authors deal with updating the knowledge of each agent after a message has been sent.

In [7], authors dealt with the problem of how to express a semantics for Agent Communication Logic in order to make non-monotonic inferences on the ground of speech acts. The above mentioned researches have exploited the flaws we referred to in this thesis. Some attempts to solve these flaws have been proposed in [5, 6, 10, 11, 19, 27, 33]. Although interesting perspectives, the focus of those papers and their aims share little with the purpose of this work.

The most comprehensive investigation about lying agents, from the viewpoint of agent communication logical framework is [32].

First things first, what author considers a lie? You lie to me that p, if you believe that p is false while you say that p, and with the intention that I believe p.

This is the first strong difference between the two approaches. An agent's beliefs are typically private, as such another agent can only guess if an agent believes something or not. Therefore, lying in our framework is based on just what an agent announces; an agent thus lies if he announces p as well as $\neg p$ on (possibly) different channels.

The author claims that a lying agent considers that p is false when he announces it. We value this viewpoint not exhaustive. If an agent has not any knowledge regarding p nor $\neg p$ but he announces p on a channel while $\neg p$ on (possibly) another channel, we say that even in this situation that agent is lying. (In [32], the author calls it *bluffing*, whereas we have named it incompetent.)

Another strong difference is that a lie is successful if the recipient of the lie ends up believing in its truthfulness (provided the type of this recipient agent, that is credulous, sceptical, or revising). Even if we can understand the author aim, we do not agree on it. A lie is a lie, independently on whether, after it has been told, some agents end up in believing in it. If you announce that the authors of this work are females, we shall not believe it, but we know that you are a liar (or a joker in a more sympathetic scenario).

An agent observing the communication channels is not really interested in what the beliefs of another agent are, but rather on *what is said* on the channels under his observation. The information he uses by his deduction process can therefore rely exclusively on those announcements, his knowledge and his own beliefs, and he can combine all such information to make his own prejudices about the other agents.

To complete this short analysis of the reference literature, an important investigation regards complexity of reasoning in PAL. In [18], the author proves two interesting results: Satisfiability in single-agent PAL is NP-complete and in multi-agent PAL is PSpace-complete.

Part V

Conclusion and Future Work

In this thesis, we dealt with the problem of combining beliefs and announcements in a framework that also allows to provide prejudices about agent communication attitudes. The basic results we obtained are: (i.) a formalisation of the modal logic MCL which allows to express facts, beliefs and announcements, (ii.) the analysis of a semantics for this logic, (iii.) we proposed a topological categorization of agents for MCL using RCC5, (iv.) we defined an upper bound on the number of different agents in a MAS, and (v.) we applied our results to the security of CPS. Moreover, we showed that our results can be used to address the problem of defining attack states for CPS. We then defined a system of communicating agents in MCL along with mono- and bi-directional channels. We provided the details of the ABF tool that accepts a configuration of a system of communicating agents (defined in MCL) as input and returns the possible configurations of the system.

As stated in Chapter 3, we have based our work on the stream of extensions to Dynamic Epistemic Logic, in particular referring to PAL, which was originally proposed by Plaza in [20]. The basis of our approach has been to quit the oversimplified assumption of truthfulness of agents. Issues about truthfulness of agents have often been dealt with in PAL and other agent-based logic approaches, as in [34].

There are several ways in which this research can be taken further. We are looking at extensions to the logical framework to cover *partially observable channels* that include temporal aspects and access permissions. Agent tags presented in Section 4.1 might be refined. For instance, an agent can be considered insincere only when announcing the opposite of a belief of hers, while an agent making a statement on which he has no knowledge of truthfulness might be classified as braggart. Furthermore, we plan to study and implement the RCC3 and RCC8 calculi. RCC3 would provide less precision but also less complexity and then higher performances of the ABF Tool. On the contrary, RCC8 would provide higher precision but also higher complexity. In order to apply the tool to more extensive case studies and to detail the security properties as intentions/intents of the agents, correlating our work with the BDI (Belief, Desire, Intent) framework.

References

[1] P. Balbiani and P. Seban. Reasoning about permitted announcements. *Journal of Philosophical Logic*, 40(4):445–472, 2011.

[2] Philippe Balbiani, Nadine Guiraud, Andreas Herzig, and Emiliano Lorini. Agents that speak: modelling communicative plans and information sources in a logic of announcements. In *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3*, pages 1207–1208.

[3] Philippe Balbiani and Pablo Seban. Reasoning about permitted announcements. *J. Philosophical Logic*, 40(4):445–472, 2011.

[4] Alexandru Batlag, Lawrence S. Moss, and Slawomir Solecki. The logic of public announcements and common knowledge and private suspicions. In Itzhak Gilboa, editor, *Proceedings of the 7th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-98), Evanston, IL, USA, July 22-24, 1998*, pages 43–56. Morgan Kaufmann, 1998.

[5] Guido Boella, Rossana Damiano, Joris Hulstijn, and Leendert van der Torre. A common ontology of agent communication languages: Modeling mental attitudes and social commitments using roles. *Applied Ontology*, 2(3-4):217–265, 2007.

[6] Guido Boella, Rossana Damiano, Joris Hulstijn, and Leendert W. N. van der Torre. Role-based semantics for agent communication: embedding of the 'mental attitudes' and 'social commitments' semantics. pages 688–690. ACM, 2006.

[7] Guido Boella, Guido Governatori, Joris Hulstijn, Régis Riveret, Antonino Rotolo, and Leendert van der Torre. Time and defeasibility in FIPA ACL semantics. *J. Applied Logic*, 9(4):274–288, 2011.

[8] Matteo Cristani, Elisa Burato, Katia Santacá, and Claudio Tomazzoli. The spider-man behavior protocol: Exploring both public and dark social networks for fake identity detection in terrorism informatics. In *KDWeb*, pages 77–88, 2015.

[9] Matteo Cristani, Francesco Olivieri, and Katia Santacà. A logical model of communication channels. In *Intelligent and Evolutionary Systems*, 2016.

[10] Benoit Gaudou, Andreas Herzig, and Dominique Longin. A logical framework for grounding-based dialogue analysis. *Electr. Notes Theor. Comput. Sci.*, 157(4):117–137, 2006.

[11] Benoit Gaudou, Andreas Herzig, Dominique Longin, and Matthias Nickles. A new semantics for the FIPA agent communication language based on social attitudes. In *ECAI 2006, 17th European Conference on Artificial Intelligence, August 29 - September 1, 2006, Riva del Garda, Italy*, pages 245–249, 2006.

[12] D. Grossi, L. Royakkers, and F. Dignum. Organizational structure and responsibility : An analysis in a dynamic logic of organized collective agency. *Artificial Intelligence and Law*, 15(3):223–249, 2007.

[13] Rolf Grütter, Thomas Scharrenbach, and Bettina Bauer-Messmer. Improving an rcc-derived geospatial approximation by OWL axioms. In *ISWC*, 2008.

[14] Eunsuk Kang, Sridhar Adepu, Daniel Jackson, and Aditya P. Mathur. Model-based security analysis of a water treatment system. In *SEsCPS*, 2016.

[15] S.K. Khaitan and J.D. McCalley. Design techniques and applications of cyber-physical systems: A survey. *IEEE Systems Journal*, 9(2):350–365, 2015.

[16] J. Lin, S. Sedigh, and A. Miller. Modeling Cyber-Physical Systems with Semantic Agents. In *COMPSACW*, 2010.

[17] Tsau Young Lin, Qing Liu, and Y. Y. Yao. Logics systems for approximate reasoning: Approximation via rough sets and topological spaces. In *ISMIS*, 1994.

[18] Carsten Lutz. Complexity and succinctness of public announcement logic. In *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006), Hakodate, Japan, May 8-12, 2006*, pages 137–143, 2006.

[19] Matthias Nickles, Felix A. Fischer, and Gerhard Weiß. Communication attitudes: A formal approach to ostensible intentions, and individual and group opinions. *Electr. Notes Theor. Comput. Sci.*, 157(4):95–115, 2006.

[20] Jan Plaza. Logics of public communications. *Synthese*, 158(2):165–179, 2007.

[21] D. Prawitz. On the idea of a general proof theory. *Synthese*, 27(1-2):63–77, 1974.

[22] Marco Rocchetto and Nils Ole Tippenhauer. CPDY: Extending the Dolev-Yao Attacker with Physical-Layer Interactions. In *ICFEM*, 2016.

[23] G. Sabaliauskaite and A. P. Mathur. Intelligent checkers to improve attack detection in cyber physical systems. In *CyberC*, 2013.

[24] T. Sanislav and L. Miclea. Cyber-physical systems - concept, challenges and research areas. *Control Engineering and Applied Informatics*, 14(2):28–33, 2012.

[25] Katia Santacà, Matteo Cristani, Marco Rocchetto, and Luca Viganò. A topological categorization of agents for the definition of attack states in multi-agent systems. In *Multi-Agent Systems and Agreement Technologies*, pages 261–276. Springer, 2016.

[26] Floor Sietsma and Jan van Eijck. Message passing in a dynamic epistemic logic setting. In Krzysztof R. Apt, editor, *Proceedings of the 13th Conference on Theoretical Aspects of Rationality and Knowledge (TARK-2011), Groningen, The Netherlands, July 12-14, 2011*, pages 212–220. ACM, 2011.

[27] Munindar P. Singh. A social semantics for agent communication languages. volume 1916 of *Lecture Notes in Computer Science*, pages 31–45. Springer, 2000.

[28] Liz Sonenberg, Peter Stone, Kagan Tumer, and Pinar Yolum, editors. *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3*. IFAAMAS, 2011.

[29] David Urbina, Jairo Giraldo, Alvaro A. Cardenas, Nils Ole Tippenhauer, Junia Valente, Mustafa Faisal, Justin Ruths, Richard Candell, and Henrik Sandberg. Limiting the impact of stealthy attacks on industrial control systems. In *CCS*, 2016.

[30] Johan Van Benthem, Jan Van Eijck, and Barteld Kooi. Logics of communication and change. *Information and computation*, 204(11):1620–1662, 2006.

[31] Johan van Benthem, Jan van Eijck, and Barteld P. Kooi. Logics of communication and change. *Inf. Comput.*, 204(11):1620–1662, 2006.

[32] Hans van Ditmarsch. Dynamics of lying. *Synthese*, 191(5):745–777, 2014.

[33] Mario Verdicchio and Marco Colombetti. From message exchanges to communicative acts to commitments. *Electr. Notes Theor. Comput. Sci.*, 157(4):75–94, 2006.

[34] Michael Wooldridge. Semantic issues in the verification of agent communication languages. *Autonomous Agents and Multi-Agent Systems*, 3(1):9–31, 2000.

A

Implementation of the ABF tool

```
from z3 import *

Boundary = DeclareSort(’Boundary’)

X = Const(’X’, Boundary)
Y = Const(’Y’, Boundary)
Z = Const(’Z’, Boundary)

C   = Function(’C’, Boundary, Boundary, BoolSort())
P   = Function(’P’, Boundary, Boundary, BoolSort())
O   = Function(’O’, Boundary, Boundary, BoolSort())
EQ = Function(’EQ’, Boundary, Boundary, BoolSort())
DR = Function(’DR’, Boundary, Boundary, BoolSort())
PO = Function(’PO’, Boundary, Boundary, BoolSort())
PP = Function(’PP’, Boundary, Boundary, BoolSort())
PPi= Function(’PPi’, Boundary, Boundary, BoolSort())

################################
# CONNECTS WITH
# define here the parthood relation C (primitive binary inclusion relation called connects with)
# which is reflexive and antisymmetric
################################

reflexivity=ForAll(X, C(X, X))
symmetry=ForAll([X,Y], Implies(C(X, Y), C(Y,X)) )
#extensionality= ForAll([X,Y], Implies(ForAll(Z, C(Z, X) == C(Z, Y)), X == Y))

################################
# PART OF
# P(X,Y) : forall Z C(Z,X) => C(Z,Y)
################################

part_of=ForAll([X,Y], P(X,Y) == ForAll(Z, Implies(C(Z,X), C(Z,Y))))
```

```

################################
# OVERLAPS
# O(X,Y) : exists Z P(Z, X) /\ P(Z, Y)
################################

overlaps=ForAll([X,Y], O(X,Y) == Exists(Z, And(P(Z,X),
    P(Z,Y))))

################################
# EQUAL TO
# E(X,Y) : P(X, Y ) /\ P(Y, X )
################################

equal_to=ForAll([X,Y], EQ(X,Y) == And(P(X,Y), P(Y,X)))

################################
# DISCRETE FROM
# DR(X,Y) : not O(X,Y)
################################

discrete_from=ForAll([X,Y], DR(X,Y) == Not(O(X,Y)))

################################
# PARTIAL OVERLAP
# PO(X,Y) : O(X, Y) /\ (not P(X, Y)) /\ (not P(Y, X))
################################

partial_overlap=ForAll([X,Y], PO(X,Y) == And(O(X,Y),
    Not(P(X,Y)), Not(P(Y,X))))

################################
# PROPER PART OF
# PP(X,Y) : P(X, Y) /\ (not P(Y, X))
################################

proper_part=ForAll([X,Y], PP(X,Y) == And(P(X,Y),
    Not(P(Y,X))))

################################
# INVERSE OF PROPER PART OF
# PPi(X,Y) : P(Y, X) /\ (not P(X, Y))
################################

proper_part_i=ForAll([X,Y], PPi(X,Y) == And(P(Y,X),
    Not(P(X,Y))))

################################
# AGENT
################################

```

```
s = Solver()
s.set(auto_config=False, mbqi=False)
s.add(And(reflexivity, symmetry, part_of, proper_part,
    proper_part_i, partial_overlap, discrete_from,
    equal_to, overlaps))


A = Const(’A’, Boundary)
B = Const(’B’, Boundary)
F = Const(’F’, Boundary)

rcc5={’eq’:EQ, ’pp’:PP, ’ppi’:PPi, ’po’:PO, ’dr’:DR}

counter=1
counter_sat=0
counter_unsat=0
counter_unknown=0

for i in rcc5:
r1=rcc5[i](A,B)
for j in rcc5:
r2=rcc5[j](B,F)
for k in rcc5:
r3=rcc5[k](A,F)

agent=And(r1,r2,r3)

s.push()
s.add(agent)
check=s.check()
offset1=’ ’*(5-len(str(counter)))
offset2=’ ’*(12-len(str(check)))
print(”%d %s %s %s %s”%(counter, offset1, check, offset2,
    agent))

if(check == unsat):
    counter_unsat+=1
    #print(”a %s”%s.unsat_core())
if(check == unknown):
counter_unknown+=1
if(check == sat):
counter_sat+=1

s.pop()
counter+=1

print(”\n********\nSTATISTICS\n\nagents=%d\nsat=%d\nunsat=%d\nunknown=%d”%
    (counter-1,counter_sat,counter_unsat,counter_unknown))
```