UNIVERSITÀ DEGLI STUDI DI URBINO CARLO BO

Dipartimento di Scienze Pure e Applicate
Dottorato in Scienze di Base e Applicazioni
Curriculum Scienza della Complessità

XXXI Ciclo

A LOGICAL LANGUAGE FOR COMPUTATIONAL TRUST

Relatore:
Prof. Alessandro Aldini

Candidato:
Mirko Tagliaferri

Anno Accademico 2018-2019

A Giorgia,
insostituibile compagna di vita.

Contents

List of Figures

Chapter 1

Introduction

1.1 Summary

Trust is an ambiguous concept. Most of this ambiguity is produced by the fact that trust (such as, e.g., love) is a naïve concept: every human being possesses a rough and pre-theoretical notion of trust and, to apply it, they often rely on their *guts feelings*, rather than objective evaluations. For instance, even though not advisable, there seems to be no particular problem with the sentence: “I don’t know why, but I feel I can trust him.”, uttered by someone placing his trust in a complete stranger. This fact produces the unpleasant effect of not having a general meaning of trust on which everyone agrees, since each individual attaches a specific meaning to the concept based on their personal experiences and, more importantly, on their moral and ethical status and upbringing. A further source of ambiguity is generated by the fact that trust is a *wide concept*, applicable in different contexts with meanings that can vary greatly. When someone utters: “I trust you are enjoying your summer.”, she has a quite different meaning in mind compared to when she utters: “I trust you will complete the task in time.”; the former sentence expressing hope, while the latter expressing a form of belief on the abilities of the other agent.

Coupling those two facts together (trust naïve and multi-purpose nature), it comes with no surprise that in everyday life there is a plethora of different interpretations surrounding the concept of trust. *Per se*, this doesn’t produce any specific problems: in natural languages there are many concepts which do not have a precise meaning and, after all, the word “trust” is employed on a daily basis by many agents without particular difficulties. This is due to the fact that in ordinary interactions, context identification and physical cues can help disambiguating the various meanings of the concepts employed and, when they don’t, further enquiries and repeated interactions make it possible for groups of agents to agree on common meanings. However, the multifaceted nature of trust becomes problematic when the plethora of interpretations is carried over to formal settings. If two political economists discuss some national policy which involves trust, but they do not realize that they are not sharing a common meaning for the notion, then it is very unlikely that they will settle on the correct way of implementing the policy. This is indeed what happens with the concept of trust. Different disciplines conceptualize trust differently and, often, inside the same discipline, it is possible to find highly different conceptualizations. The most straightforward example is that of economy, where:

> “ [T]rust is defined by some as a characteristic of a particular class of action, while others identify it as a precondition for any such action to take place. At the same time, some discuss trust with reference to governments and organisations, while others examine trust between individuals or people in particular roles. ” [39]

Another example is that of (the security fragment of) computer science: with the gradual transition of social interactions from face-to-face to digital environments, the importance of having a digital counterpart of trust contributed to the emergence of numerous theoretical analyses of the notion, which, instead of fostering a unified account, produced a *potpourri* of definitions with different domains of application and different levels of abstraction. This is a pressing issue for computer scientists who desire to build general frameworks for digital systems which include a trust component: different definitions of trust might have drastically different effects on the frameworks and thus, might produce incompatible digital systems. This calls for a unified formal account of the notion of trust and a formal framework which embody this formal notion. With such structures in hand, a computer scientist would be able to construct general frameworks which can reproduce and imitate social environments better and, thus, promote interactions of a higher quality on the web.

The focus of this thesis is, therefore, that of forming a set of core features of socio-economical notions of trust. From there, the aim is to build a computational counterpart of the notion, which does justice to previous attempts of formalising trust in computer science. This computational counterpart is then employed to build gradually more powerful formal languages which allow reasoning about trust. The main properties of those formal languages are analysed and, finally, comparisons are made with other models which are employed to model trust in computer science.

1.2 Scope

This thesis doesn’t belong to a single discipline. In line with the definition of complexity science (which characterizes the curriculum under which this thesis was produced), i.e., “an emerging approach to research, complexity science is a study of a system. It is not a single theory, but a collection of theories and conceptual tools from an array of disciplines." [129], the thesis employs different techniques and explores different disciplines in order to achieve the result of providing a good formal definition of trust.

In the first part, the concept of trust is inspected through a philosophically-oriented conceptual analysis based on literature data from the disciplines of biological evolution, economy and the social sciences. This part is therefore a work in the field of philosophy of science.

In the second part, the emphasis shifts to computer science models of trust, which are classified following both a qualitative analysis of the core features attributed to trust in each model, and a quantitative analysis of the relevance of the models in the literature, based on number of citations. Theoretical attention is also paid to paradigms in the computational trust literature, which define the main characteristics of computational environments in which trust is implemented. This part is therefore a work in the field of theoretical computer science.

In the third part, after a brief introduction to modal logic, a proper logical language is built based on the results of the previous chapters and theorems about the decidability of the computational problems of the language are given. This part if therefore a work in the fields of formal methods and computational logic.

In the fourth and final part, the logical language presented in the third part is compared to other models employed to represent trust and uncertainty (placing emphasis on models suited for applications in security systems). Similarities and differences are pinpointed and fusions between them are proposed. This part is therefore a work in the field of computer science and formal modelling.

It is important to note, that the intended scope of the thesis is here explicitly presented just to give the reader a general idea of what were the approaches followed during the various analyses made. This is fully compatible (if not even desirable) with the fact that most results are open to interpretation and are applicable to other fields, other than the ones imagined by the author. In this spirit, the wider scope of the thesis should be taken to be that of all fields which can benefit from implementations of formal structures for trust.

1.3 Objectives and Methodology

As was mentioned at the end of the previous section (see section 1.2), the wider aim of the thesis is that of providing a unified formal definition of trust employable in all fields which can benefit from formal implementations of the notion. In this sense, there are three main objectives in the thesis: 1) To provide a thorough conceptual analysis of the concept of trust, comprising reflections from all major fields which focused (at least part of) their attention on it; 2) to employ such analysis to build a logical language that can help to describe and reason about trust and its relationship with other cognitive notions (e.g., beliefs and knowledge); 3) to produce an account of trust in the computer science literature and compare existing models for trust with the semantical structure for the logical language presented in the thesis.

Objective one is a philosophical one and it is carried out both through a meta-analysis of existing literature on trust and *armchair reflection* on such meta-analysis. First, a possible explanation for the origin of trust is proposed, starting from studies on reciprocal altruism [126] in biological systems and extending those with socio-biological studies. Second, seminal papers on trust in economy and sociology are examined, in order to extract some common features of trust recognized by both communities. Those features, it is argued, form a solid core for a general definition of trust. Third, the conceptual analysis is extended to the field of computer science: classical models of computational trust are surveyed and a second set of features is obtained. Those two set of features are then compared, in order to determine whether computer scientists are building reliable models for trust. Once the affinities and differences between the economical/sociological and the computational conceptions of trust are highlighted, conclusions are drawn and improvements are proposed.

Objective two is a logical one and it is carried out through a logical construction of an interpretative semantical structure for a novel logical language for trust. First, previous attempts to formalize trust are introduced, with emphasis on their advantages and disadvantages. Second, gradually more powerful logical languages are introduced, with their respective truth-theoretical semantical structures. The final language (i.e. a context-sensitive, single-agent logical trust language), which is the most expressive between those presented, is thoroughly analysed. The analysis will consist in proofs of the computational decidability problems for the language.

Objective three is a computer science one and it is carried out through comparison between existing models for the representation of trust and uncertainty, and the semantical resources introduced for the logical language at the centre of this thesis: in particular, bridges are built with Subjective Logic [59] and Dempster-Shafer Theory of Evidence [29, 116]. It is then shown how the logical language proposed can complement those formal models for trust and which benefits each bring to the other. This might help to obtain fruitful trust models employable in computational environments.

1.4 Contributions

The contributions of this thesis are classified according to the list of objectives presented in the previous section (see section 1.3).

For objective one, the results obtained are:

1. A proposal for the origins of trust, based on biological and anthropological reflections.

2. A conceptual analysis of the concept of trust in economy and sociology, based on a general survey of seminal economical and sociological papers on trust.

3. The construction of a solid core of features for trust, based on socio-economic considerations.

4. A meta-analysis of existing computational trust models.

5. The construction of a taxonomy for computational trust models, useful for the classification of different formal models of trust.

6. A thorough comparison between the core features of a socio-economical conception and a computational conception of trust.

For objective two, the results obtained are:

1. The introduction of two different logical languages, of increasing expressivity, for describing and reasoning about trust. Syntax and semantical structures are proposed for each of the languages.

2. The proofs for the decidability problems for the second language.

For objective three, the results obtained are:

1. The presentation of bridge theorems between the semantical structures introduce in the thesis and existing formal structures for the representation of trust and uncertainty.

2. Some reflections on the possibility of merging different formal structures together to obtain more powerful structures.

1.5 Organization

The thesis is organized in chapters and sections as follows.

For chapter two (2):

- In section one (2.1), an initial classification of the dimensions characterizing trust is introduced.

- In section two (2.2), a possible explanation for the origins of trust is presented.

- In section three (2.3), an economical/sociological analysis of the concept of trust is presented.

- In section four (2.4), conclusions are drawn for the previous sections and a unified set of core features of trust is proposed.

For chapter three (3):

- In section one (3.1), a justification for the need of computational notions of trust in the computer science community is provided.
- In section two (3.2), a meta-analysis of existing surveys about different models for computational trust is made. A taxonomy is built for classifying computational trust models.
- In section three (3.3), representative classical computational trust models are selected. Those models are analysed and core features of a computational conception of trust are determined.
- In section four (3.4), conclusions are drawn and the features of the socio-economic conception and the computational conception of trust are compared.
- In section five (3.5), some theoretical paradigms used to set assumptions on the environments in which computational trust models are implemented are analysed.

For chapter four (4):

- In section one (4.1), prerequisite notions needed for the construction of the logical languages are introduced: in particular modal logic based on neighborhood structures is introduced.
- In section two (4.2), the syntax and semantics of a context-free, single-agent logical language for trust are given.
- In section three (4.3), the syntax and semantics of a context-sensitive, single-agent logical language for trust are given.
- In section four (4.4), decidability results for the language presented in section 4.4 are proved.

For chapter five (5):

- In section one (5.1), Subjective Logic is analysed and bridge theorems are given between the logical languages of chapter 4 and Subjective Logic.
- In section two (5.2), Dempster-Shafer Theory of Evidence is analysed and bridge theorems between it and the logical languages of chapter 4 are given and proved.

General conclusions follow.

Chapter 2

Conceptual analysis of trust

In this chapter, seminal papers on the concept of trust belonging to different disciplines are selected and then analysed. The aim is that of obtaining a set of features common to all those definitions provided in the different disciplines. This common ground is thought to be a useful starting point for further reflections on trust and can provide valuable insights on necessary conditions for the presence of trust in any kind of system. The chapter starts by presenting different dimensions which can characterize trust. Those dimensions are inspired by multi-disciplinary reports about trust [110, 111, 117] and the relevant bibliographical elements of those reports [106, 130, 131]. After the conceptual map of trust is built, a possible explanation for the evolution of trust in biological and anthropological systems is proposed: this should be thought of as a purely rational (as opposed to empirical) *hypothesis* on the way trust originated. Understanding where trust might have come from can provide important insights on how trust can be formed in contemporary complex systems and what to expect from behaviours generating from trustful relationships. The chapter then proceeds with economical and sociological analyses of trust. The main reason for choosing those two disciplines is the importance of trust for them. In economy, trust is mainly thought to be an enabling factor in exchanges [6, 105], improving the quantity and quality of interactions between agents. In sociology, trust is seen as an element permitting the actual existence of society and of social relationships [82, 119]. Given the centrality of the notion of exchange/interaction in economy and that of society in sociology, it is evident why having a clear definition of trust can improve the quality of research in those fields. Those reasons justify the existence of this chapter, which attempts at giving a unified conceptual treatment of trust. Indeed, given the facts expressed above (the fundamental importance of trust in sociology and economy) it should be expected that general treatments of the concept of trust already exist. However, as Gambetta notes:

> “ [T]he importance of trust is often acknowledged but seldom examined, and scholars tend to mention it in passing, to allude to it as

> a fundamental ingredient or lubricant, an unavoidable dimension of social interaction, only to move on to deal with less intractable matters. " [40]

This fact, true at the end of the '80, when Gambetta edited his volume, swiftly changed in the '90, when various disciplines started to focus their attention on trust, which moved from being an elusive notion to becoming a key issue to address. Yet, despite the common understanding of the importance of the notion, the close attention paid to the concept of trust produced more confusion than clarity. Suddenly, a proliferation of definitions flooded the different disciplines, making it progressively implausible to find common grounds on trust in those disciplines (if not even intra-disciplines). Therefore, in order to achieve the aim of the chapter (i.e., that of obtaining an unified set of features for trust), attention will be paid on pioneering works on trust and contemporary literature, extracting the features which defined the concept of trust employed in those researches.

In section one, a conceptual map of trust is given, with the aim of providing a guide for the conceptualizations of different analyses of trust. In section two, a tentative proposal on the origins of trust is developed; data is taken from biology and ethology, and the method followed is the one typical of conceptual analysis. In section three, trust is analysed from the point-of-view of economy and sociology. Finally, in section four, a tentative set of common features of trust is given; such set is formed based on the reflections of all previous sections.

2.1 Conceptual map of trust

Navigating through the various definitions of trust given in the different disciplines can be a burdensome task. First of all, disciplines as diverse as sociology [9, 23, 40, 82], economy [28, 36, 114, 133], political science [53, 54, 78] and evolutionary biology [10, 126, 127] dedicated some of their attention to trust, obviously prioritizing their specific needs and using their typical examination techniques. This produced many theoretical definitions of trust which diverge on the technical language employed to express the definitions and the principal features that are highlighted by those. This section is aimed at producing a conceptual map which can help the novice reader in his navigation. The map (which can be seen in figure 2.1) is constructed around three dimensions which characterize trust and it is claimed that all definitions of trust (already existing or future ones) eventually fall under a specific quadrant of the map. The map is taken from [110] and its dimensions are discussed with reference to the original authors who introduced them. Specifically, the three dimensions regard *the nature of*: i) the actual trust relation; ii) the agents' trusted; iii) the context in which to trust.

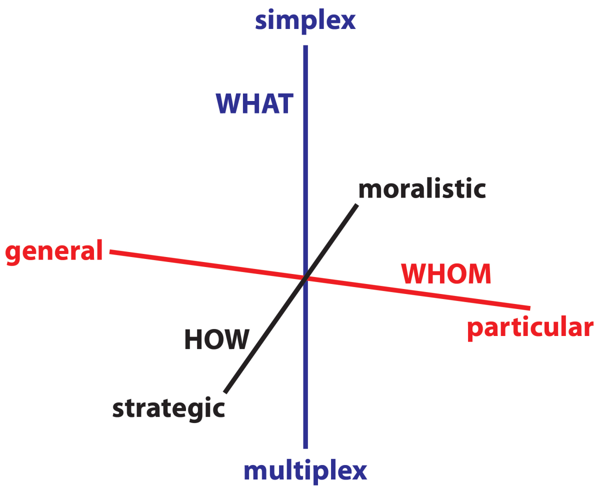


Figure 2.1: Conceptual map of trust dimensions.

The first dimension, indicated in [110] as the *how* dimension, distinguishes between trust definitions that are strategic and those that are moralistic. A *strategic* definition [23, 53, 54] of trust identifies the phenomenon of trusting as one depending on explicit knowledge and explicit computations about the interacting party's trustworthiness, intentions and capacities. On the other hand, a *moralistic* definition [85, 130, 131] of trust identifies the phenomenon of trusting as a by-product of an agent's moral and ethical upbringing and consequently it depends on his psychological predispositions as defined by social norms and the values of the agent's culture. Where strategic trust can be described by the motto: *Agent A trusts agent B to do X*; moralistic trust is simply described by saying that: *agent A trusts.* This dimension of trust is absolutely important to discussions concerning the notion, inasfar as strategic definitions of trust presuppose that, for agent A to trust agent B, repeated encounters between the agents are necessary and, moreover, agent A must possesses the computational powers to compute trustworthiness values. Even though plausible, those assumptions are suited only for small communities and apply to a small number of interactions and thus, strategic trust can't account for *all* the transactions and collaborations that occur in ordinary life. Moralistic versions of trust are designed to overcome this downside of strategic trust. If trust is produced as a moral commandment (similar in spirit to Kant's *categorical imperative* [68]), then even complete strangers might initiate a trust relationship. In the case of moralistic trust, it is the culture of the trustor that determines whether or not he will trust someone else and past experiences with the trustee are neither required nor important.

The second dimension, indicated in [110] as the *whom* dimension, distinguishes between trust definitions that are particular and those that are general.

A *particular* definition of trust identifies the phenomenon of trusting as a one-to-one relation, where trust can only be placed on specific individuals. In particular, the individuals that are considered to be trust bearers are those on whom the trustor has a fair amount of information, such as, e.g., family members, friends or colleagues. On the other hand, a *general* definition of trust identifies the phenomenon of trusting as a one-to-many relation, where trust can be placed also on anonymous individuals or strangers and such that there is no specific task or context of evaluation. In such a case, it might be said that trust is considered as an omnicomprehensive attitude towards a specific group of individuals (often those attitudes are determined by stereotypical categories). This dimension has an obvious relation with the first one: moralistic trust seem to lend well to general trust, while strategic trust is strictly tied to particular trust. However, those links are not absolute, leaving open the possibility for strategic general trust and moralistic particular trust. The former case is typical of views in which trust is seen as a stereotype, while the latter identifies views for which agents are morally inclined to cooperate (and therefore trust) close relatives and known others.

The third, and final, dimension, indicated in [110] as the *what* dimension, distinguishes between trust definitions that are simplex and those that are multiplex. A *simplex* definition of trust identifies the phenomenon of trusting as being context-specific, where trust is granted according to a specific task. On the other hand, a *multiplex* definition of trust identifies the phenomenon of trusting as being context-free, where agents trust other agents without considering any specific issue or scenario of evaluation. In the former case, trust is evaluated on a case-by-case basis and the same two agents might trust each other in specific contexts and refuse to do so in different situations. Given the variety of different scenarios that might happen in the real world, the assumption that trust is context-specific seem to be a suitable one for a good definition of trust. However, it is important to note that there are times in which an agent trusts others blindly, independently from all context considerations. For instance, a child trusts his parents blindly[1]. Moreover, even admitting that pure multiplex phenomenon of trust are impossible (independently from how much you trust someone, that someone might not be able to perform given actions, e.g., piloting a plane and thus he shall not be trusted in such contexts), it is still plausible that mild-versions of multiplex trust exist, where trust is granted with respect to a set of contexts sharing some core features, rather than a single ones. It is, however, open to debate whether those kinds of multiplex trust are genuinely multiplex or just a combination of multiple simplex trust evaluations.

Given the three dimensions introduced, it is possible to allocate trust definitions into eight different categories (in figure 2.1 each quadrant represents

[1] Note that some authors might claim that the child isn't actually trusting the parents, since he has no choice other than relying on them.

a category). Each category correspond to a given idea of what is trust. In particular, the categories are the following:

1. **Strategic particular simplex trust**: trust is seen as a specific belief about another person's reliability on a specific issue.
2. **Strategic general simplex trust**: trust is seen as a specific expectation about strangers' reliability on a specific issue.
3. **Strategic particular multiplex trust**: trust is seen as a specific belief about another person's reliability in general.
4. **Strategic general multiplex trust**: trust is seen as a specific expectation about strangers' reliability in general.
5. **Moralistic particular simplext trust**: trust is seen as a general trusting attitude towards specific individuals in specific circumstances.
6. **Moralistic general simplex trust**: trust is seen as a general trusting attitude towards strangers in specific circumstances.
7. **Moralistic particular multiplex trust**: trust is seen as a general trusting attitude towards specific individuals.
8. **Moralistic general multiplex trust**: trust is seen as a general trusting attitude towards strangers.

This conceptual map will help all further discussion on trust, by allowing the indication of a specific class to which definitions that will be introduced from now on belong to.

2.2 The origins of trust: a tentative proposal

In this section, a tentative proposal on the origins of trust is developed. The section is structured as follows: i) first, a proposal on the biological evolution of trust is formulated based on biological researches [10, 126, 127]; ii) second, some experimental data on what actions and situations foster or reduce trust are reviewed [25, 42, 89, 90]; iii) finally, the proposal is confronted with the experimental data, in order to establish its plausibility.

The proposal presented can be summarized as follows: *Reciprocal altruism led to the development of trust relationships.* Note that this proposal is not a completely original one[2]; as noted by Bateson:

[2] A similar idea to the one presented here can be found in [74] and [99].

> “ . . . questions about the evolution of cooperation do not bear directly on the issue of trust, though they may give pause to anyone who supposes that trust is required for effective cooperation . . . Even though the study of cooperation in animals seems irrelevant to an understanding of trust in humans, careful analysis of the conditions in which cooperative behaviour is expressed suggests that many animals are exquisitely sensitive to the behaviour of others. This observation suggests an explanation for the evolution of the mental state that we recognize as trust in ourselves. ” [10]

Bateman suggests that the existence and evolution of cooperative behaviours can provide valuable insights into the development of the mental state of trust in human beings. The idea presented in this thesis is similar in spirit, even though the focus is placed on reciprocal altruism specifically instead of cooperative behaviour in general: the former defining sets of actions where who performs the action incurs in a loss, while the recipient of the action benefits, while the latter defining sets of actions where both the actor and the recipient benefit from the actions performed [47].

It is held that, once it is established that reciprocal altruism is compatible with and favoured by natural selection, trust can be seen as a psychological trait developed to sustain human reciprocal altruism behaviour, i.e., trust helps in perpetuating the advantageous behaviour over less optimal options (e.g., cheating behaviours). Thus, *trust is a specific mechanism developed in order to categorize others according to their propensity towards reciprocal altruism behaviours.*

2.2.1 Reciprocal altruism

In Trivers [126], a model which explains the evolution of reciprocal altruism is presented. Notably, the model shows how altruistic behaviours in different species can be evolutionarily selected for even in cases where kin selection [47] can be ruled out (i.e., when close relationships are not present between the agents involved in the actions). To achieve his goal, Trivers presents some relevant conditions that explain the possibility of selecting altruistic behaviours on the basis of reciprocation. The conditions identified are three: 1) presence of many altruistic situations in the lifetime of the altruists; 2) repeated interactions between a given altruist and the same small set of individuals; 3) symmetrical expositions to altruistic situations for each couple of altruists. The claim is that if those conditions are fulfilled, an agent will dispense altruism based on the tendencies of the recipient agent, rather than on kinship or randomly. Formally, this is equivalent to the claim that, when the conditions hold, the following inequality is true:

$$(\frac{1}{p^2})(\sum_k^1 b_k - \sum_j^1 c_j) > (\frac{1}{q^2})(\sum_m^1 b_m) \tag{2.1}$$

Where $\frac{1}{p^2}$ and $\frac{1}{q^2}$ can be interpreted respectively as the frequency in the population of altruistically behaving and of non-altruistically behaving individuals; b_k is the benefit for the altruist of the k*th* altruistic act performed towards him; c_j is the cost for the altruist of the j*th* altruistic act he performs and b_m is the benefit for the non-altruist (an agent who won't reciprocate) of the m*th* altruistic act performed towards him. Two further assumptions are required in order for the conditions to influence the truth of the inequality: i) the altruist stops behaving altruistically with the non-altruist once he finds out that the latter doesn't reciprocate[3]; ii) the cost of performing an altruistic action for the actor is lower than the benefit gained by the recipient of the action, thus producing a net gain for the whole system. After individuating the three conditions necessary for the validity of the inequation, Trivers proceeds to elaborate such conditions into a set of relevant biological traits which affect the selection of altruism over non-altruism. He identifies, in particular, six traits: a) length of lifetime; b) dispersal rate; c) degree of mutual dependence; d) parental care; e) dominance hierarchy; f) aid in combat. Since trait (d) can be seen as a special case of trait (c), and trait (f) is specific only of situations in which combative scenarios are plausible, only four of the six traits are analysed.

a) *Length of lifetime*: The longer individuals of a species live, the higher the chance that any two individuals of such species will encounter occasions in which altruistic acts should be performed.

b) *Dispersal rate*: The lower the dispersal rate of the individuals of a given species, the higher the chance that any individual will interact, repeatedly, with the same set of agents.

c) *Degree of mutual dependence*: The more dependent from each other are individuals of a given group, the higher the chance that individuals in the group will encounter occasions in which altruistic acts are required to achieve collective goals. This trait is closely connected to dispersal rate, inasfar as high degree of dependence impose low dispersal rates.

e) *Dominance hierarchy*: The existence of a dominance relation between two individuals, decreases the chances of altruistic behaviour. The reason is that the more dominant individual can often force the less dominant one to perform the altruistic act against his will and without the necessary requirement of reciprocating the act in the future.

It is easy to check that individuals in human societies possess such traits.

[3] Put in game-theoretical terms, this is equivalent to the claim that the agent follows a *tit for tat* strategy.

First, the life span of human beings is long enough to guarantee the occurrence of numerous situations where altruistic behaviours are required; second, humans tend to be sedentary, thus spending most of their life inside the same community, increasing the chances of interacting with the same agents; third, the variability in talents and abilities of human beings favours high levels of mutual dependencies between them; finally, few exceptions made (e.g., workplaces), human societies are void of dominance relations between individuals, whereas law and social norms determine the boundaries of what an agent can forcefully demand from another agent. Those facts lead to the conclusion that reciprocal altruism should be selected over non-altruism in human societies. However, such a scenario is still compatible with the adaptiveness of subtle forms of cheating. A subtle cheater shall be distinguished from a gross cheater according to the following definitions: a *gross cheater* is an individual who fails to reciprocate at all, producing a scenario where the altruist individual's costs to perform altruist actions outweighs the benefits he receives from the actions performed by the cheater. Formally, this is expressed by the following inequality:

$$\sum_{i}^{1} c_i > \sum_{j}^{1} b_j \tag{2.2}$$

Where c_i is the cost for the altruist of the i*th* altruistic action performed by him, while b_j is the benefit he gains from the j*th* altruistic action performed by the cheater. On the other hand, a *subtle cheater* is an individual who reciprocates subpar, i.e., producing actions that generate less benefits for the others with respect to the benefits that would follow if those agent were performing actions in his place. Formally, this is expressed by the two inequalities, assuming that $\sum_i b_{a,i} > \sum_j c_{a,j}$:

$$\sum_{i,j} (b_{sc,j} - c_{sc,i}) > x \tag{2.3}$$

$$x > \sum_{i,j} (b_{a,i} - c_{a,j}) \tag{2.4}$$

Where the assumption says that the sum of the benefits for the altruist deriving from all the actions performed by the cheater is superior to the sum of the costs for the altruist of all the actions he performs towards the cheater, i.e., even if the altruist is interacting with a cheater, he has a net gain from the relationship. Inequality 2.3 states that the subtle cheater has a net gain *superior* to the net gain (x) he would get from of an equitable relationship, while inequality 2.4 states that the altruist has a net gain *inferior* to the net gain (x) he would get from an equitable relationship.

The main difference between a gross cheater and a subtle cheater is that when dealing with the former, an altruist will incur in losses, while, when dealing with the latter, he will benefit from the relationship, even though not optimally. It is therefore easy for the altruist to recognize gross cheaters, but quite difficult to recognize subtle cheaters, since it is never clear if the behaviour of the other interacting party is due to unavoidable limitations or cheating intentions; this ignorance can be expressed by the fact that the altruist doesn't have access to the value x of equitable relationships, but he can only compute his benefits and his costs. Furthermore, the difficulty of recognizing subtle cheaters increases when those cheaters develop the ability to mimic some traits designed to foster altruistic behaviours in others. Examples of those traits are: friendship, guilt, sympathy, gratitude and moralistic aggression[4]. In such cases, subtle cheaters might induce altruists to perform altruistic behaviours towards them even in situations where such behaviours aren't strictly necessary.

Trust, it is proposed, evolved as an effective mechanism to improve the ability of agents of detecting subtle cheaters in their social neighbours. Specifically, trust is employed as a intention-selecting mechanism and thus as a tool to categorize individuals into altruists and cheaters:

> “ Selection may favour distrusting those who perform altruistic acts without the emotional basis of generosity or guilt because the altruistic tendencies of such individuals may be less reliable in the future. ” [126]

Given the computational difficulties of recognizing the behaviours of subtle cheaters over the ones of genuine altruistic agents, trust might have evolved in order to classify individuals according to their intentions based on the scarce information available in the environment. In general, genuine altruism encourages further altruism, while altruism induced by sheer utilitaristic computations is less likely to be reciprocated [56, 71, 75]: trust, therefore, aids an agent in distinguishing between those two typologies of altruistic behaviours, decreasing the chances of incurring in subtle cheating and increasing the overall net gain of the social system of which the agent is part of. There is evidence [25] that, in interacting settings, when an agent believes that the other party is genuinely willing to participate in a positive collaboration, his choices will be driven by trust. There are also strong indications [18] that trust is tied to reciprocity in altruistic behaviours, such that communities in which reciprocity norms are most active display strong tendencies towards trusting behaviours. Moreover, this proposal on the origins of trust is compatible with classical sociological theories of trust, which define the concept as a complexity-reducing tool[5]. All those facts contribute to the plausibility of the proposal advanced in this section. In the next subsection the proposal is going to be put to the test, judging its status with respect to experimental data on trust.

[4] See [126] for a thorough analysis of all those traits.

[5] See section 2.3.

Note that no references have been made to which kind of information an agent seeks in order to determine whether to trust or not another agent. In fact, there is no fixed set of information typologies on which trust evaluations are based. The main reason is that trust is a highly subjective concept and the kind of information on which trust evaluations are based might vary greatly from agent to agent. A possible explanation is that an agent is able to assess whether the other party is or isn't a subtle cheater only according to his past interactions with other subtle cheaters; thus, the agent will tend to focus on information typologies which helped him to recognize the cheating behaviour in the past. Nonetheless, general elements of evaluation are recognized in the literature as the basic starting points for each new relation which must be established; among those, the most prominent is *reputation*, i.e., the public trustworthiness evaluation of an agent made by the community he lives in. Yet, finding a general set of pieces of information necessary for the arising of trust (over and above those few often mentioned) is an implausible task. This should not, though, refrain discussion on the way trust is established and how it evolves through time; after all, determining such set of pieces of information, and explaining the origins and evolution of trust are two separate (orthogonal) tasks. In this thesis (and in this chapter in particular) the focus is placed on the second task rather than the first, i.e., attention is focused on experimental data pointing at the establishment and dynamics of trust, without claiming that the features that will be highlighted are the uniquely necessary feature required to have trust. In this sense, the data presented are thought of only as supporting elements that increase the plausibility of the proposal made in this section and not as conclusive elements which prove the correctness of it.

One final caveat to highlight is that the proposal makes no assumptions on the actual status of trust (see section 2.1 for a general discussion on what trust might be). It doesn't say anything about what trust actually is, but only on how it might have come to exist. The view is compatible with different interpretations on what trust actually is (e.g., a belief, an expectancy or an unconscious psycological attitude). To test such a claim, the different dimensions of trust introduced in the previous section will be assumed one at a time and it will be shown that those can all be seen as categorizing mechanisms.

Assume that trust is a strategic phenomenon. In such a case, it was argued, trust depends on the trustor's previous encounters with and knowledge about the other interacting party. This dimension is compatible with trust being a cheater selecting mechanism. In fact, previous encounters with and knowledge about the other party can be seen as means to an end, where the end is that of determining if the other party is willing to cheat or act altruistically. Assume now that trust is a moralistic phenomenon. Also in this case, moral and ethical values of an agent can be seen as automatic mechanisms that foster altruism. In particular, moral dictatums are employed as rules of thumbs and help in producing altruistic behaviour in situations where lack of information would cause stagnation. This exhausts the *how* dimension.

Assume that trust is a particular phenomenon. This is obviously in line with the hypothesis that trust is a selecting mechanism. The fact that only specific close-connected agents might be trusted is completely compatible with the fact that those specific agents, by being closer to the trustor, are less likely to indulge in cheating behaviours. On the other hand, if trust is assumed to be a general phenomenon, this could be explained by the fact that, once stereotypes are taken into account, under uncertainty conditions, it is always preferable for an agent to trust rather than not trusting, since the net gain of reciprocating with a subtle cheater is still positive, while gross cheaters can be quickly identified after a small number of interactions and thus the loss incurred would be limited. This exhausts the *whom* dimension.

Assume that trust is a simplex phenomenon. This would mean that agents trust others only according to specific contexts of evaluation. This is compatible with view of trust as an intention-selecting mechanism since other agent might have different intentions based on contexts. Thus, an agent categorizing the other interacting parties might produce different categorizations based on the context of evaluation. On top of that, generic categorizations might be produced, in order to facilitate the case-by-case selection and reduce the computational effort required to make the decision on whether to collaborate or not. Such a possibility, grants that the proposal is compatible also with multiplex trust. This exhausts the *what* dimension.

All dimensions have been covered and it has been shown that the current proposal is compatible with different definitions of trust.

Even though no actual assumption has been made, up to now, on the nature of trust, some assumptions are going to be made in the subsections that follow and in later chapters. Those assumptions will deeply depend on the subject under analysis, as trust is seen as a different object in different disciplines; this won't, however, cause any harm to this proposal on the origins of trust, since, as it has just been argued, the proposal is compatible with different ontological views on trust.

2.2.2 Experimental data on trust

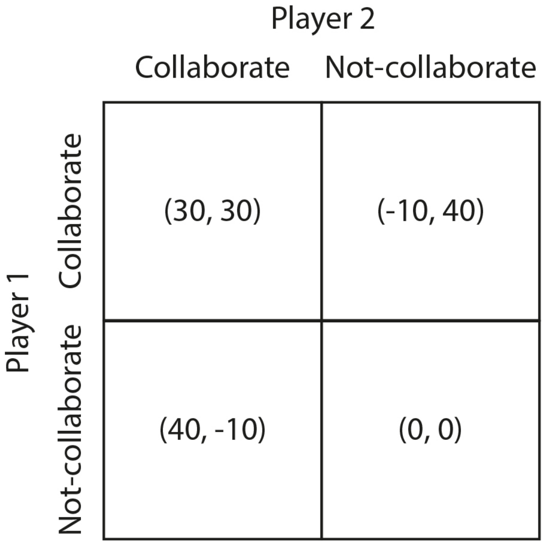
The experimental data that is going to be discussed in this subsection are taken both from the field of economy and that of sociology. In particular, attention is placed on laboratory experiments and the conclusions derived from those. No field studies analyses are considered, mainly because it is thought that the data gathered through those studies are influenced by many uncontrolled (and uncontrollable) factors (especially cultural ones) and thus might only be useful in building culturally specific conception of trust and not general ones. It is recognized that also experimental settings have their issues and the choice of relying only on them for validation of a thesis might be highly problematic. It is believed, though, that the results obtained from the works reported are reliable enough to draw some important conclusions with respect to the proposal made on the origins of trust. The sources of the results presented in the following subsection are taken from [25, 42, 89, 90] and all relevant works reported in those papers. The selection of this material was determined by the desire of having diversity of disciplines and approaches to the question of analysing trust and its dynamics, thus allowing results from sociology, behavioural psychology, and economy [6].

In [42], the author examines four facts about trust. Each fact presented is examined with respect to the increase of cooperation it elicits. The facts analysed are: i) relations and exchanges that last over longer periods of time; ii) the lack of threat potential; iii) small initial investments (with small gradual increases); iv) high amount of communication between agents.

The first and fourth facts are derived from analyses of various instances of the Prisoner's Dilemma (PD) [73]. PD is an experimental matrix game where two or more agents are given a payoff matrix and they must decide which row or column of the matrix to choose based on their intention to cooperate or non-cooperate with the other agent. In figure 2.2, it is possible to observe a two-players example of a game matrix: the best result in terms of both players is achievable when they both collaborate, while the worst result for a player is obtained when he collaborates, but the other player defects (and viceversa for the best result for a player).

The hypothesis, developed in [104] and reported in [42], is that if subjects view the interchange from a long-term perspective (i.e., they employ long-term thinking in their decision making), they will display cooperation-inclined behaviours, often showing trust in the other subjects in the experiment. The hypothesis is built from experimental results involving (and comparing) *short* and *long* instances of Prisoner's Dilemma scenarios. Short and long instances of PD are never directly quantified, but results suggest a switch between short and long term thinking in the range of 30 to 60 trials of a game [108]. The proposed explanation is that, on the one hand, when there are only few instances of the game to be played, players will tend to try to maximize their welfare, by either exploiting collaborative players or by defending their-selves from defecting players; this can explain the seemingly paradoxical result studied in the PD literature for which often both players decide to defect and they therefore end up in a much worse situation compared to the one they could have achieved by cooperating. On the other hand, when the game is played repeatedly, players tend to favour collaborative behaviours, in order to improve the overall quality of the outcomes and the general net gain. For the fourth fact, results are drawn from experiments involving four different instances of PD, each one consisting of increasing levels of communication between the

[6] Even if this thesis is driven by a spirit of completeness and broadness, it is an unlikely task that of reporting all existing experimental data on trust. The reader is directed to the following papers for further results about trust: [3, 24, 33, 52, 87, 134]. The results explicitly reported in this thesis are thought to be exemplary reports on laboratory experiments and this is the reason they were selected with respect to the other works cited in this footnote.



		Player 2	
		Collaborate	Not-collaborate
Player 1	Collaborate	(30, 30)	(-10, 40)
	Not-collaborate	(40, -10)	(0, 0)

Figure 2.2: Example of a prisoners' dilemma matrix.

players [132]. In the first instance, players were completely isolated; in the second, they could see but not hear each other; in the third, they could hear, but not see each other; in the fourth and final one, they could both see and hear each other. The results show that increasing levels of communication (especially visually-oriented) produce increasing levels of cooperation. This is explained by the formation of strong beliefs of the other agent's intentions based on verbal and non-verbal communication. This increased knowledge about the other agent decreases the uncertainty about possible defections and therefore foster collaboration.

The second and third facts are derived from analyses of instances of the Trucking Game (TG) [30]. TD is a game in which two players represent manufacturers who must deliver their goods. In the game there are three roads which can be used by the players, two which are respectively exclusive to each player and one in common; the common road (employable by only one player at a time) is the shortest between each player's producing facility and delivery point. In the game, time is a resource, therefore employing the shortest road benefits both players. In figure 2.3, an example of the game is displayed. In [30], among others, two important variations of the game are explored: in the first variation each manufacturer has control over a specific gate placed on the short road, allowing them to eventually block the other manufacturer from using the road; in the second variation, no gates are present and the only way for a manufacturer to block the other is by leaving a truck on the road (therefore blocking also its-own deliveries).

The experimental results show that, when there are no gates (i.e., there is

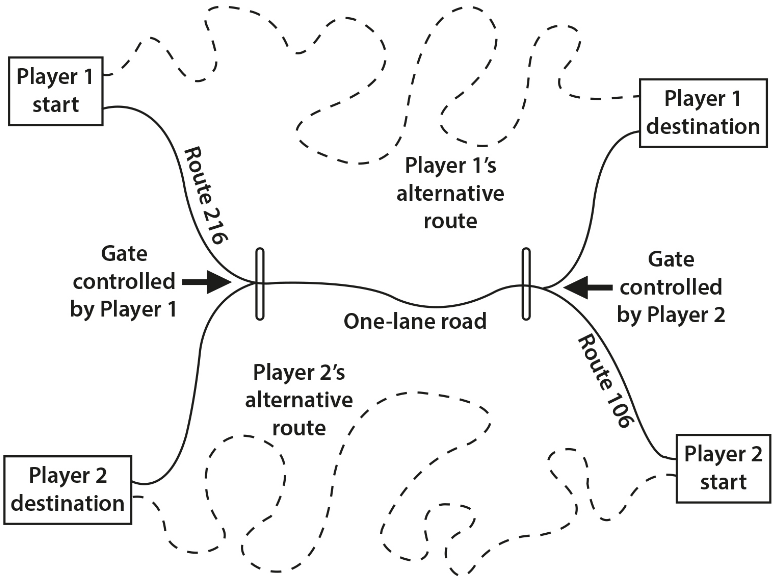

Player 1 start
Route 216
Player 1's alternative route
Player 1 destination
Gate controlled by Player 1
One-lane road
Gate controlled by Player 2
Route 106
Player 2's alternative route
Player 2 destination
Player 2 start

Figure 2.3: Example of a trucking game map.

no perceived threat of blocking), collaboration has a higher chance of developing. An hypothesis which explains such evidence is given in [42]: from the study of [112], it is possible to conclude that individuals assign much higher subjective probabilities to situation arising when they are given a cause to think in detail about the situation, therefore exhibiting a form of cognitive inertia (they rather assign a higher probability to the event highlighted instead of computing actual probabilities); given this fact, players in the gate-version of TD might concentrate their attention on the gates and therefore come to expect, with a subjective probability higher than the objective one, that the other player will, in fact, implement the threat. Experimental results [31] also show that when rewards and costs are low (and only gradually increased during each interaction), players tend to collaborate more. This is again explained in [42], making use of the so called *set effect*: individuals tend to be bias towards the preservation of a theory, neglecting negative data and interpreting ambiguous information in line with their thesis[7]. The set effect might explain the behaviour of agents in TD in the following way: agents are willing to risk small sums in the initial phases of the game in order to evaluate the intentions of the other agent; once initial collaboration is achieved, through the set effect, further instances of interactions are interpreted according to the theory each agent developed during previous interactions and therefore confirm collaborative behaviours.

[7] The existence of such phenomenon is examined and proved in [81].

In [25], the authors examine the following fact about trust: intention attribution increments the chances of trusting behaviour.

The fact is explored analysing results of three separate games: one being an instance of the Moonlighting Game (MG) [1] and the other two distinct versions of the Dictator Game (DG) [66]. In MG, two players interact by taking resources from/giving resources to each other. The game is played in turn by the players, therefore, the first player, when giving money to the second player, often exhibits trust-driven behaviours. In DG, the setting is similar to MG, but one of the players is not able to make his choice, therefore limiting the outcomes of the game to the choices made the other, without any chance of retaliation. The experimental results show that in MG players going first make more trust-involving decisions compared to the same players playing DG. The authors suggest that a possible explanation of such findings is that agents which attribute intentional behaviour to other agents, tend to be more inclined towards trust compared to players that face other players who are coerced to make a choice.

In [89], the authors examine the following fact about trust: the amount of trust placed in other agents, strictly depend on past experiences.

The fact is explored analysing results of two consecutive instances of the Trust Game (TG) [14]. TG replicates the general setting of MG; however, players are only allowed to give resources, but not to take them from the other player. The experimental results show that agents take into great consideration past experiences and, moreover, this consideration is emphasized when an agent interacts with members of the same group. The authors suggest that the concept of trust in a collective entity is different, even though related, to the one of trust in an individual. The general result, nonetheless, is that past experiences influence trust in future occasions and group reputation has a stronger impact than individual considerations.

Finally, in [90], the authors examine the following fact about trust: trust is a target-specific phenomenon and not a general disposition of agents.

The fact is explored analysing results from two games: the first being the Distrust Game (DisG) and the second being an instance of DG. In DisG, two players are assumed to produce a given amount of resources, which can then be divided between the two players; player two is the one deciding how the resources have to be split, while player one has the option of investing a certain amount of resources to prevent player two from being able to make that decision, therefore splitting the resources evenly. The amount of resources player one is willing to invest in this warranty is taken to represent how much he distrusts player two. In the study, one instance of DG was employed to disclose the general attitudes of the agents and then various instances of DisG were employed to measure the distrust levels agents exhibit towards different demographics. The experimental results show that agents tend to employ their general information about a certain demographic to determine their distrust in other agents. The authors claim that those finding also highlight the fact that distrust is not part of the general attitudes of agents, but are target-specific, in the sense that the level of distrust strictly depends on who is the other interacting agent in a relationship.

To sum up, in this subsection, seven experimental results on trust have been presented. Those results all seem to have an impact on trust:

- Length of relations.
- Lack of potential threats.
- Small initial investments and small increments in such investments.
- High amount of communication.
- Intention attribution.
- Past experiences with the same agents.
- Group information.

2.2.3 Assessing the origins of trust

In this subsection, the proposal on the origins of trust given in subsection 2.2.1 is assessed with respect to the experimental results given in subsection 2.2.2. Recall that the proposal advanced is that trust evolved as an intention-selecting mechanism which has the purpose of detecting subtle cheaters in the agent's social neighborhood. It is claimed that all the experimental results improve the plausibility of such a proposal.

The proposal is consistent with result one: length of relations and long-term thinking increment agents' trust. The longer the relations, the higher the amount of information an agent has about the other interacting agent. This reduces the chances that the agents mimic altruism eliciting behaviors, since mimicking such behaviors for longer periods of time would be counter-intuitive for a cheater. Therefore, as expected, relations that last for longer periods of time should increase the propensity of agents to engage in reciprocal altruistic relationships.

The proposal is consistent with result two: lack of potential threats increment agents' trust. The non-existence of threats (or, at least, the lack of perception of potential threats) should improve the positive perception an agent has of the other interacting agents. The reason is that when attributing intentions to other agents, the existence of a potential threat that can be made effective could elicit the attribution of the emotion of temptation. Therefore, as expected, the existence of threats that can punish an agent fosters in that agent a sense of uncertainty and a related decrease in trust; conversely, the lack of those threats, increase the amount of trust.

The proposal is consistent with result three: small investments and small increments in such investments increment agents' trust. Since trust should favour reciprocal altruism strategies (given their positive impact on the general net gain of the interacting party), losing small investments can be considered an acceptable risk which might produce, in case of success, positive altruistic relations in the future; on the other hand, small increments with respect to previous investments can be considered acceptable in light of the possibility of strengthening past successful relations.

The proposal is consistent with result four: high amount of communication increment agents' trust. The more two subjects communicate, the harder it gets for a subtle cheater to hide his true intentions. This is due to the fact that the subtle cheater must create false information to justify, in the eyes of the altruistic agent, his altruistic intentions (which aren't, in fact, altruistic). The altruist has, therefore, better chances of individuating inconsistencies in what the cheater tells. This higher level of assurance produces a higher level of trust.

The proposal is consistent with result five: intention attribution is relevant for trust. This follows directly from the proposal, since trust is actually seen as a mechanism to select intentions. It is no surprise that if the choices of other agents are driven by genuine intentions, rather than mere calculation or coercion, then trust increases.

The proposal is consistent with result six: past experiences with other agents influence the levels of trust. Again, as in the previous case, this comes directly from the proposal. It should be expected that the ability to categorize agents between cheaters and altruists, affect the way trust is placed.

The proposal is consistent with result seven: group information influences individual trust. This is consistent with the proposal if it is assumed that group of agents share common features with respect to their intentions. This assumption seems a reasonable one, inasmuch as a given social group shall, at least, share a generally accepted minimum level of altruism. If this were not the case, i.e., the group contains members that do not share this minimum level of altruistic intentions, then there would be a contradiction, since it is expected from the theory of reciprocal altruism that the members of the group lacking this level of minimal altruistic intentions, should be ostracized. Once this assumption is accepted, it follows directly that having information about a social group automatically brings information about single individuals inside such group.

To conclude, the proposal seem to hold in the face of experimental results and therefore should be, at least, treated as a plausible explanation on the origins of trust.

2.3 Trust in Economy and Sociology

In this section, a conceptual analysis of the concept of trust is presented. The section is structured as follows: i) first, trust is analysed from the point-of-view of sociology [9, 23, 32, 82, 83, 119, 120]; ii) second, trust is analysed from the point-of-view of economy [28, 36, 40, 44, 114, 133]. The material from which the analysis starts are seminal papers in the economical and sociological literature. Attention is directed at peculiar features of trust and how they characterize the concept. Some level of contradiction is expected, since the works analysed come from different disciplines and, even when analysing trust inside the same discipline, the papers might differ for the theoretical framework in which they perform their reflections. Those contradictions shall not be considered a problem, since in the last section of this chapter a further refinements will be made and the consistent features of trust will be selected. It can be anticipated that all features are compatible with the various dimensions of trust introduced in section 2.1. Where possible, the definitions analysed in this section will be described in terms of those dimensions.

2.3.1 Trust in sociology

Both in his book *Trust and Power* and his paper *Familiarity, Confidence, Trust: Problems and Alternatives*, Luhmann tries to clarify what is the function of trust in society, comparing the concept with other closely related notions: the aim is to build a precise enough concept of trust to be employed in theoretical models of society. To achieve his goal, he tries to provide a conceptual distinction between trust, familiarity and confidence, focusing on their interrelationship and on the different impact that each one of these concepts has on society broadly conceived. For what concerns this thesis, the concept of familiarity will not be taken into consideration. The concept of confidence, however, can help to better understand Luhmann's ideas about trust.

"[T]rust is a solution for specific problems of risk" [83]. The preceding quote best summarizes Luhmann's proposal about trust. Luhmann interprets trust as an attitude towards positive expectations in situations of risk, where personal decisions determine the possibility of the situation of risk to occur. Thus, trust is present when an agent can take voluntarily a specific course of action and this course of action might have uncertain outcomes, where the losses of the bad outcomes outweigh the gains of the good outcomes. The latter condition is important, otherwise trusting "...would simply be a question of rational calculation and you would choose your action anyway, because the risks remain within acceptable limits" [83]. The decisions an agent can make are fundamental to the distinction between trust and confidence. Confidence arises in all occasions in which no alternatives are considered and it is felt that that outcomes are imposed on the agent. Trust becomes a matter of subjective evaluation and is dependent, for its status, on the agent's perceptions and attributions: “If you choose one action in preference to others in spite of the possibility of being disappointed by the action of others, you define the situation as one of trust” [83]. Confidence copes with danger, trust with risk, where the latter notion presupposes agency over passive considerations. Thus, the two concepts are often present at the same time, with confidence being a prerequisite for participating in society and trust being a condition for active action. If an agent lacks confidence, he will be alienated from society at large and will retreat to smaller social groups, where dangers are reduced. However, a lack of trust limits the possibility of rational action, thus “Through lack of trust a system may lose size; it may even shrink below a critical threshold necessary for its own reproduction at a certain level of development” [83]. Those considerations should be enough to understand that both confidence and trust are necessary components for society and that they must complement each other to produce action and participation; they help in reducing the complexities of evaluation about the world and its evolution. You form mechanisms of confidence, when obtaining full information about a system and therefore avoid all dangers is impossible; you form mechanisms of trust, when computing the outcomes of your actions and the actions of others is implausible.

From Luhmann’s ideas it is possible to extract some features of trust. Trust requires: a) a decision to be made; b) the possibility of incurring in a loss whose absolute value is higher than the possible gains of the trusting decision; c) there must be alternatives to the decision, allowing the agent to avoid the choice, if he is willing to do so; d) the decisions available are a matter of subjective interpretation and not objective possibilities (the agent must believe he can make the decisions, whether or not they are actually possible). Moreover, answering Luhmann’s original question on the function of trust: trust helps in reducing the complexities of computations on the effects of different courses of action.

With a similar aim, but a different approach, Barber, in his work *The Logic and Limits of Trust*, tried to produce a functional account of trust, determining the role of the concept in modern societies. Specifically, for Barber, trust provides “cognitive ad moral expectational maps for actors and systems as they continuously interact.” [9]. To fulfil its role, trust is split into three distinct attitudes towards others. A general feeling of trust involves expectations of the future, where what is expected is the persistence and fulfilment of a natural and moral social order. Two more specific trust phenomena are related to the expectation of expertise and competent behaviour on the part of the trustees and of commitment to carry out fiduciary obligations and responsibilities. In such a sense, for Barber, trust is a phenomenon whose *how* dimension is halfway through moralistic and strategic. It is neither purely moralistic, since expectations involve some conscious computations about the willingness of others to fulfil their moral obligations and maintain the moral social order and those computations often involve previous acquaintance with the interacting party; nor is it purely strategic, since the value of comparison is a moral social order which is culturally influenced and is based on values that depend on society at large. From Barber's account, it is possible to identify some important features of trust. Trust is basically a set of expectations, therefore it requires high amount of knowledge about the interacting agents. Those expectations help the trustors to navigate society, thus, they fulfil a role similar to the one presented by Luhmann, where trust is employed to reduce the complexities of modern societies. Finally, those maps are determined by the general moral values of the trustor, who assumes that in society there is a fixed, morally inclined, natural and social order. Moreover, the maps are determined by the perceived expertise and the moral status of the trustee, where the ability and willingness to fulfil obligations determine whether he is trusted or not.

A different route is followed, in sociology, by the analyses of Durkheim and Simmel. Both of those authors defend an idea for which trust is a by-product of the increasing division of labour and the functional specialization of societies. Their idea is to conflate small family-based microsocieties typical of medieval times with the complex multi-faceted macrosocieties of modern times. In the former type of societies, frequency and intimacy of relations produced scenarios where each agent knew almost everything about the other agents and where kinship bonds produced automatic collaborative behaviours. In such cases, trust was not required and thus non-existent. Perfect knowledge substituted trust. On the other hand, in the latter type of societies, interactions become more scarce and impersonal. The requirements of hyper specializations produce a *secret society*, where all agents are strangers to one another. This shift towards imperfect knowledge about the others, generates the necessity of building trust relationships. In Simmel words:

> " Without the general trust that people have in each other, society itself would disintegrate, for very few relationships are based entirely upon what is known with certainty about another person, and very few relationships would endure if trust were as strong as, or stronger than, rational proof or personal observation " [119]

This point, i.e., the importance of trust for the existence of modern societies, is highlighted even more in Durkheim treatment of the division of labour and of hyper specializations. According to Durkheim, the transition from premodern societies to modern ones, produced an equivalent transition from social solidarity based on likeness and similarity (mechanical solidarity) to social solidarity based on interdependence and trust (organic solidarity). Being dependent on one another for even simple tasks, require agents to form bonds. Those bonds can either be based on law and contracts or on trust and faith: the former typology often present in structured interactions and the latter typical of small-scale unstructured interactions. This division strictly depends on the high costs of establishing sovra-personal structures; incurring in this cost is justified only when transactions are repeated over time and they can produce some value for the entity establishing the structure. When interactions are scarce and they produce low added value for the participants, trust must take the place of the sovra-personal structures. In such a sense, trust is again an expectation of fulfilment and thus Durkheim's thesis could be partially conflated with the one of Barber. However, this conflation is not a complete one, since the expectation is only on the upholds of previously established agreements or of performing the actions prescribed by someone's role in society.

From Simmel and Durkheim's analyses it is possible to extract the following features of trust: i) trust is present when no perfect information is available; ii) where no familiar bonds are present, the existence of dependencies require trust to be present.

Summarizing the results, from the sociological point-of-view trust ought to be an expectation about future events:

1. Involving a collaborative decision to be made, where the trustor must consider possible alternative decisions.
2. Where the possible net costs for the trustor are higher than the possible net gains.
3. Where high amount of knowledge is required on the part of the trustor, who, however, is not omniscient about the trustee.
4. Where the moral status of the trustor, the trustee and the moral social order of society is taken into account.
5. Where a requirement of dependence is present.

Moreover, the function of trust is that of enabling action, by both permitting non-familiar bonds to be created and by reducing the complexity of making decisions.

2.3.2 Trust in economy

When discussing matters of trust in economics, it is very important to identify the economical paradigm in which the discussion takes place. The main reason is that in some economical paradigms, there is no space for social considerations and/or concepts. For example, in the neoclassical paradigm, agents are seen as perfectly rational and utility maximising actors, therefore, they never consider intentions and moral facts when taking decisions, relying solely on concrete computations and strategic thinking. Moreover, in a neoclassical environments, information is fully available and misbehaviour is forbidden.

However, the idealization of the market in such a way has received much criticism [34], especially concerning the fact that most assumptions of neoclassical theory are ill suited to represent the real world market. Agents are not omniscient oracles that can perform perfect computations about their utilities and welfares, and living in a benevolent environment; they are rather irrational beings, which make inconsistent choices and they live in a deceitful world. Those criticisms and the desire to produce analyses of the market closer to reality, produced new paradigms. The choice of analysing trust in such novel paradigms rather than the classical ones is made because computational agents are limited in a similar fashion to the agent of economical markets. Thus, concentrating on assumptions which consider boundedly rational agents will help extending this economic analysis to computational environments.

Some of those new paradigms (e.g., transaction cost theory [133]) include social considerations in their reflections about exchanges and interactions in economical systems. The most predominant factors that are taken into account in those alternative paradigms are opportunistic behaviours and uncertainty about the state of the general system. Opportunistic behaviour could be interpreted, in this thesis, as the behaviour a subtle cheater might hold in an interaction[8]. Uncertainty about the state of a system should be interpreted as limited information in the hand of the interacting agents, which makes them prone to incur in deceitful behaviour on the part of the other agent with whom they are interacting. The possibility of uncertainty and possible opportunistic behaviour are used in such theoretical frameworks to explain the existence of hierarchical organisations (e.g., law and legally binding contracts) as control mechanism. However, establishing such organisation is not always possible, because it is both complicated and costly to establish them and often the effort required is not worth the gain of the transaction. Trust, as a social mechanism, has the effect of diminishing or removing the cost of establishing those transaction-facilitating organisations.

In this subsection, three different positions of how trust is interpreted in economic environment are analysed. The first is typical of the paradigm of transaction-cost theory and is advanced in [39]. The second is taken from [44], where the social structures in which economic action take place are thoroughly analysed. Finally, the third position is taken from [36], where trust is discussed with respect to its impact on economical transactions.

Concerning the first position, it has been argued that trust serves as a vehicle to diminish the cost of transaction in economical systems; therefore trust is seen as an enabling factor in exchanges. However, the way this goal of trust is achieved can differ when different conceptions of trust are taken into consideration. In its simplest form, trust is seen as a calculative process, favouring aseptic numerical evaluations rather than careful examination. Under this interpretation, someone trusts another agent when his evaluation of

[8] See subsection 2.2.1.

the possible costs of malevolent behaviour is outweighed by the potential gains of the transaction. While justifiable in an economical framework, note how this interpretation is in direct contrast with Luhmann's view, for whom sheer computations are representing confidence behaviours, rather than trusting ones. This aseptic view of trust is partially mitigated when subjective evaluations are taken into account. For instance, in Gambetta's interpretation, trust is a subjective probability:

> " [T]rust... is a particular level of subjective probability with which one agent assesses that another agent or group of agents will perform a particular action, both before he can monitor such action (or independently of his capacity to monitor it) and in a context in which it affects his own action... When we say we trust someone or that someone is trustworthy, we implicitly mean that the probability that he will perform an action that is beneficial or at least not detrimental to us is high enough for us to consider engaging in some form of cooperation with him. " [40]

When trust is seen as a subjective probability, rather than an objective calculation based on possessed information, the attitudes and intentions of the agents re-enter the picture. While still numerical in value, the numbers are taken according to personal, and often biased, evaluations. Those evaluations, in turn, are based on general dispositions of the agents and their social upbringing, which determine conventional thresholds for cooperation and trustworthy behaviour: some agents will be more prone to trusting behaviours and this will show up in how the agents compute their subjective probabilities. Moreover, social norms will guide the perceptions of agents, thus determining how they assess each other's intentions and dispositions.

> " [T]he need for transaction-specific safeguards (governance) varies systematically with the institutional environment within which transactions are located. Changes in the condition of the environment are therefore factored in - by adjusting transaction-specific governance in cost effective ways. In effect, institutional environments that provide general purpose safeguards relieve the need for added transaction-specific supports. Accordingly, transactions that are viable in an institutional environment that provides strong safeguards may be nonviable in institutional environments that are weak - because it is not cost-effective for the parties to craft transaction-specific governance in the latter circumstances... [S]ocietal culture, politics, regulation, professionalization, networks, and corporate culture. Each can be thought of as institutional trust of a hyphenated king: 'societal-trust', ' political trust', and so forth. " [133]

As expected, in transaction-cost theory trust is seen as a formal and numerical measure, requiring computations in order to be applied. This doesn't come as a surprise, given the fact that economical models are often built with the aspiration of being completely formal. However, it has been argued, strict paradigms, which make strong assumptions about real-world systems, are ill suited to both describe and understand those complex systems and alternative loose paradigms must be assumed. In those paradigms trust is still seen as a numerical value, but whose computation take into account social and personal evaluations (thus moving from a purely strategic view of trust towards a milder position between strategic and moralistic trust).

From those reflections it is possible to extract some features of trust. Trust requires: a) a certain level of uncertainty with respect to the outcomes of a given interaction; b) a limit in the computational capacities of agents; c) the possibility of quantifying the cost and gains of actions. Moreover, trust has the purpose of diminishing the structural costs of building infrastructures needed for exchanges to take place risk-free.

Concerning the second position, in [44] Granovetter presents a thorough analysis of how economical actions are embedded in sociological structures (which, then, foster trust between agents in exchanges). In particular, he notes that previous attempts at describing economical actions have suffered from an undersocialized or an oversocialized treatment. In particular, he argues that:

> “ [T]he level of embeddedness of economic behaviour is lower in nonmarket societies than is claimed by substantivists and development theorists, and it has changed less with “modernization” than they believe; but . . . this level has always been and continues to be more substantial than is allowed for by formalists and economists. ” [44]

In this quote, the substantivist view of economic behaviour argues that premodern societies where heavily reliant on social constructs, while modern societies gradually moved towards independent approaches to exchanges. On the other hand, formalist views argue that neoclassical treatment of agents is not only useful in describing modern societies, but also premodern ones, for which reliance on social constructs has always been overestimated. Thus, Granovetter defends an idea that the level of social embeddedness of economical action is in between a undersocialized view (typical of neoclassicism) and a oversocialized view (typical of sociological studies, in which economy is just a by-product of social interaction). His analysis starts by realizing that while under/oversocialized view of the embeddedness might provide justified explanations for the behaviours observed at a macrolevel of economy, those are no longer appropriate when the level of analysis if that of micro-level imperfectly competitive markets. In such markets, the impacts of large numbers of competitors in the form of sellers and buyers is not present and thus, novel explanations on why the market doesn't collapse under the pressure of mistrust and malfeasance must be provided. Appeals to "generalized morality" are often quoted as plausible explanations (noteworthy is the connection between those explanations and moralistic views of trust). Those explanations rely on the existence of a set of values commonly shared by all members of a given society, where imperatives produce collaboration rather than deceit. While those shared values might explain some behaviours, they hardly explain all behaviours, especially when it is realized that communities are not stable entities, but evolve through time, allowing entrance and exit of members which might have different moral values. Thus, Granovetter, proposes to substitute reference to general morality with reliance on concrete personal relations and social networks. Thus trust emerges as a result of repeated interactions and knowledge about the others intentions and values. What trust depends on is not general reputations, but in personal expectations that the other will behave honestly: two thieves might trust each other during a robbery, even though their reputation is obviously bad. The expectations about the possible behaviours of others stand, thus, at the base of economical actions. Someone trusts when his expectations are positive, while he doesn't, when those expectations are negative. For instance, when a group of strangers flee from a fire, it is highly likely that stampeding phenomena will take place, diminishing the possibility of everyone escaping unharmed. On the other hand, when a family escapes from their burning house, it is highly unlikely that the members of the family will rush at the door in an unorganized way. This can be explained by the fact that, when dealing with strangers, there is no guarantee that everyone will behave appropriately. In economy, the same argument applies. In large-scale economical systems, infrastructures are required and trust is less present, since the chances of having misbehaviours is higher. On the other hand, on the smaller scales, personal knowledge and relations can foster well-informed expectations on the behaviours of others, thus allowing trusting decisions to be made. In this sense, economic analyses of trust are close to the ones pursued in sociology. This should not come as a surprise, since Granovetter's aim is explicitly that of taking into account sociological structures in which economical actions are embedded.

Finally, concerning the third position, in [36], Fehr advances a proposal on the nature of trust taking into account various aspects typical of risk analysis. In particular, he shows how trust involves an important aspect of betrayal aversion, where betrayal aversion is distinguished from the close concept of risk aversion. From [15], it is possible to derive the conclusion that agents are more willing to take a risk involving a specific probability of bad luck than to enter a relation with an equivalent probability of being cheated. This fact shows that social factors are considered more than sheer probability computations. The definition Fehr provides is a behavioural one, where trusting is an act for which a trustor voluntarily places some of his resources at the disposal of the trustee, without any legal (or otherwise binding) commitment from the latter. This definition, similar in spirit to the one given in [23] by Coleman, is then conflated with experimental and survey-based results on trust, confirming that possible action is a component of trust. While in this thesis the main idea behind Fehr proposal is accepted, it is argued that the trusting behaviour and trust *per se* are distinct phenomena, even though tightly related. Undoubtedly, there are some actions which are performed only under trusting assumption, however, those trusting assumptions are action enabling phenomena and not the action in itself. Conflating the two is equivalent to committing a category mistake, where the condition that allow a given course of action is confused with the actual action. Thus, what should actually follow from Fehr reflections is that trust is a willingness (based on a subjective expectation) to enter in a risk-involving interaction with another party, based on possessed knowledge, the beliefs and the mental states of the trustor; specifically, betrayal aversion is predominant among the mental states and thus influenced trust more than other components. One might argue that the betrayal aversion aspect is external to trust evaluations, determining only a comparison threshold with which to compare someone's trust beliefs. However, it is highly unlikely that purely belief-based definitions of trust are sufficient to capture trusting phenomena. This is because agents sharing the same beliefs might still behave differently and explaining this difference is explained better by diversity in the trust values placed on the trustees, rather than assuming that the threshold for collaboration is different, but the trust values are the same. From Fehr analysis, it follows that trust involves action and that this action is dependent, among other things, on betrayal aversion (thus, more generally, on uncertainty).

Summarizing the results, from the economical point-of-view trust ought to be an explicit computation about future events:

1. Embedded in an environment lacking other control mechanisms such as sovra-personal infrastructures or general moralistic values.

2. Whose outcomes are uncertain and exposed to risk.

3. Where the agents are unable to explicitly compute exact values for all outcomes.

4. Whose outcome is numerable, inasmuch as it is possible to attribute clear values to losses and gains.

5. Where the internal mental states of agent play an important role.

The socio-economical features of trust will now be merged in the next section.

2.4 Trust in a nutshell

In this section, the results obtained through the analyses made in this chapter are summarized and the core features of trust are presented. First of all, trust is an *action-enabling* concept: this can be derived from both the proposal on the origins of trust and the socio-economic studies presented. The environments in which such decisions must be taken must always involve a form of risk; moreover, in traditional analyses, the risk must be consciously perceived as being present, and it must be possible to compute the actual effects of the negative outcomes happening. The second point is that trust is an inherently subjective phenomenon and, therefore, all qualitative and quantitative evaluations made by an agent about his trust must take into account his personal mental states. Third, trust must involve uncertainty; assuming a world of perfect information and unlimited rationality, where the intentions of others, the state of the world and the outcomes of all possible actions are known, and there is no limit to the derivations agents can make, there is no room for trusting decisions. Finally, trust involves interactions, inasfar as decisions and actions acquire their full meaning in interactive scenarios.

We can thus derive the following conclusion: *Trust is a action-enabling concept produced by subjective evaluations in risk-involving interactions placed in an uncertainty-heavy environment.* This *pseudo*definition of trust is the starting point of all further discussions in the thesis.

Chapter 3

Trust in Digital Environments

In this chapter, trust is analysed with respect to digital environments. The aim is to examine the current state of the art in computer science on trust and then extract from the literature the core features of a computational notion of trust; a notion which can then be employed in digital environments (e.g., a digital market) to improve the quality of interactions in it. Particular attention is paid to existing computational trust models and on how they implement computational versions of the socio-economical concept of trust introduced in the previous chapter (see chapter 2). In section one, the need for trust in digital environments is justified and it is explained how such a notion can improve those environments; in section two, surveys on computational trust models are analysed and a novel taxonomy useful in classifying computational trust models is built; in section three, classical computational trust models are briefly analysed and core features of their respective concepts of trust are extracted; in section four, the features extracted in section three are conflated and a unified conception of computational trust is formed. This unified conception is then merged with the socio-economical definition of trust provided in the previous chapter (see section 2.4). Finally, a further discussion on paradigms employed to construct assumption on how to formalize trust is presented.

3.1 Justifying Trust in Digital Environments

In this section, a justification for the importance of trust in digital environments is provided, placing emphasis on the role trust might play in computational systems and the future development of digital societies. The main proposal is that trust can play two distinct roles: the first role is that of a soft-security mechanism that fosters benevolent behaviours and limits malevolent ones; the second is that of aiding tool, employed to guide agents in their choices in environments where their usual physical cues are lacking and they do not know how to properly make use of their instinct, which is tailored specifically for the recognition of relevant biological traits.

Society and economy are rapidly changing and moving towards digital environments, putting an increasing distance between them and the old-fashioned physical world. This is to be expected. The internet era makes it easier and faster to achieve results. Communications and exchanges of information have become instant, while a growing amount of services allow users to get access to resources they never even thought existed (nor they thought they needed). Companies as AirBnb, StackOverflow, BlaBlaCar and Amazon revolutionized the market and the way human beings interpret their environment [16, 17]. This shift from the physical world to the digital world brought with it advantages and disadvantages. On the one side, many new possibilities arose that enhanced relationships and economy. Once impossible, today is straightforward to entertain friendship relations with people all around the world. Academic research once relied on intensive manual search in libraries, while in contemporary times, a researcher could plausibly spend the whole time in his office and, nonetheless, have access to the most recent results in his field. Economy has also been reshaped. Most services and goods are purchasable online, with no spatial and/or temporal limits: food delivery services, airplane and hotel companies offering self-booking opportunities and content sharing platforms (among others) made it possible for people to vary their consumption habits and access information faster and better, which, in turn, allow better-informed decisions to be made. On the other side, technological advances also brought some disadvantages. One over all is that living in a digital world, all the biological mechanisms that allowed human beings to cope with their environment and which have evolved along the centuries are no longer able to provide reliable guidance. Importantly, there are no indications that things will get better in the future. The growing economic interest in all activities that are performed over the web attract benevolent and malevolent behaving agents alike. Having lost the natural defences developed through evolution with uncountable amounts of trials and errors put users in danger of becoming victims during transactions. Moreover, also the initial wave of enthusiasm and willingness to collaborate which characterized web interactions at its origin is no longer present. As soon as the digital world moved from an information exchange environment to a proper economical platform, malevolent users started to infiltrate it. Those concerns require computer scientists to design and develop systems which can mimic and substitute the natural defence mechanisms of human beings. Furthermore, the issue becomes even more pressing, when it is realized that digital environments are evolving also in the direction of allowing human-machine and machine-machine interactions [37]. Digital worlds are not exclusively a vehicle for human-to-human interactions, but have gradually transitioned to human-to-machine interactions and, with the development of IoT [84], it has now become also a platform allowing machine-to-machine interactions. In this sense, not only it is important to defend human beings from other human beings when interacting in the digital world, but it is also important to protect humans from malicious use of programs (which are often assumed to be honest by human agents, since they lack direct interests or intentions) and, often, also shelter digital agents from other digital agents, which might totally lack a sense of risk.

In this setting, trust plays a valuable role, both by being able to foster positive behaviour and as a defensive mechanism that shelters from malevolent behaviours [105]. Moreover, producing trust-aware digital agents might also contribute to the increase of confidence of those agents (as defined by Luhmann; see subsection 2.2.2), in a virtuous circle of self-enhancing attitudes.

This role of trust, and more in general of social norms implemented in digital environments, should be seen as that of a soft-security mechanism [109]. To understand what exactly a soft-security mechanism is, an analogy might prove to be useful. A digital system could be seen as a fortress. The goal of such fortress is to protect the information which is contained inside, allowing access to it only to certain persons with the right authorizations. The defenders of such fortress must obviously constantly aware of all possible weaknesses of the fortress and, moreover, they must check that their hard security mechanisms are always in place and working fine. Hard security mechanisms could comprehend, for example, identity authentication mechanisms for control access or barriers such as firewalls. This burden on the defenders is enormous and expensive in terms of resources and energy. On the other side of the fence, there are the attackers. The goal of the attackers is to access the information, even though they are not authorized in doing so. In order to do so, they must find an access to the fortress and thus they must exploit some vulnerability of it. What is important to note is that one vulnerability might be sufficient for the attackers to achieve their goal. Moreover, they can communicate with other attackers and form coalitions, which can spread the intel one attacker has obtained, making their attacking chances higher. In this spirit, the cost for the attackers in terms of resources and energy is way lower compared to those of the defenders. Constant and possibly complete knowledge for the defenders versus highly specific and partial knowledge for the attackers creates an important imbalance. Soft-security mechanisms enter the picture when, instead of focusing only on hard-security, the defenders note that part of their resources could be invested in forming a society that punish, by, e.g., ostracization, every agent who is found dwelling on the possibility of attacking the fortress. Moreover, they might also note that part of the investment could be diverted towards other agents that are told to try and construct attacks towards the fortress, letting know the defenders of all the possible exploitations they might find during their fake attack. In this sense, the soft-security actions fostered benevolent behaviours (agents tell the defenders of possible issues) and punished malevolent behaviours (real attackers are ostracized.

It is argued in this thesis, that trust can play exactly this role of soft-security mechanism in digital societies. It does so by punishing malevolent agents with untrust and distrust, either reducing the amount of interactions they can have or eliminating this possibility in total. On the other hand, trust can increase the quality and number of interactions a benevolent and honest agent has, by directing other benevolent agents towards him. Moreover, once trust starts to spread, it can create nets in which positive behaviours produce further positive behaviours. This, as shown in chapter 2, can contribute to the increase in reciprocal altruism, which in turn, generates an increased net gain for the whole system at no extra cost in investments.

One question remains: why isn't trust simply generated as it is in normal face-to-face interactions? This question is interesting, because the main concern in investing time researching social phenomena and trying to implement them into system which are inherently social (even though digital), is that this investment is futile. The expectancy is that, as they developed in ordinary social systems, they will also develop in digital social systems. However, this is not actually the case. While some milder forms of social notions might transfer to digital communities, many of the cues humans are evolutionarily programmed to discern are lacking. In addition, there is an abundance of other cues (up to the point of having too many) which might confuse an agent trying to establish the social status of other agents. Another concern is that, as noted above, not all interactions in the digital world are between humans; often those interactions are between humans and machines, or between machines. Excluding the obvious fact that machines completely lack evolutionary tools to recognize malevolent interactors and therefore must be given formal tools to implement social concepts, a human being, when facing a machine (either through a computer screen or when interacting with a robot/android), is completely lost in determining which factors to take into consideration [5].

Those reason are the ones justifying research on trust, especially on trust in digital communities. In the rest of the chapter, general considerations on a computational notion of trust are made, leaving to the next chapter the goal of producing a suitable formalism for trust.

3.2 A Taxonomy of Computational Trust Models

In this section, a taxonomy for general computational trust models is presented. This taxonomy is built starting from a meta-analysis of existing surveys on computational trust and computational trust models.

Computational trust is the digital counterpart of trust as applied in ordinary social communities and computational trust models are mechanisms that implement the notion of trust in digital environments to increase the quantity and quality of interactions. Computational trust models are typically composed of two parts: a trust computing part and a trust manipulation part.

Note that this is not a distinction which is made explicitly in the literature on computational trust models. Generally speaking, authors tend to label their models as “computational trust models”, omitting whether they are specifically computing models or simply manipulation models. The main explanation for this phenomenon is that, from a practical perspective, the main focus for a computer scientist is to specify how a model works in the context in which it has to be applied. Moreover, often smaller models are merged together to obtain larger ones that can perform highly complex tasks: when those merged models are analysed is then hard to distinguish between the trust computing and the trust manipulation component. Nonetheless, given the theoretical focus of this thesis, it is important to draw a clear distinction between the two aspects of computational trust models; the reason is that doing so can help in understanding better the important aspects of trust that are relevant in computer science and in formal model building in general.

In the trust computing component, basic trust values are computed using other typologies of information gathered by the system (usually reputation scores based on previous interactions); basically, a trust computing model defines trust directly, by explicitly stating what are the components of trust and how all of those components come together to form trusting behaviours. In the trust manipulation component, new trust values are computed by manipulating already existing trust values using various operators[1]; in a trust manipulating model, less attention is paid at how trust comes about and the focus shifts on how trust can be spread in a community or how different trust evaluations are combined to form new ones. The better a model implements both components, the higher the quality of the model. However, existing models tend to specialize in one of the two tasks: those models either implement sophisticated notions of trust, focusing on repeated computations of new trust values, without ever combining those values together (e.g. [86]), or they implement sophisticated sets of operators employed to combine really simple representations of trust (e.g. [59, 135]). Moreover, the literature is abundant of different models which are highly specialized in the respective tasks for which they have been developed, but there are very few general models that can help in theoretical discussions about trust. Thus, there is a urge for and an omnicomprehensive formal language that allows an abstract discussion about trust *tout court*; this will indeed be the focus of the fourth chapter of this thesis 4. For now, the focus is going to be on traditional computational trust models.

The section is structured as follows: i) first, some terminological remarks are made about the conception of trust employed during all subsequent analyses; ii) second, the taxonomy is given and the methodology employed to build it is explained.

[1] See [59] for a wide range of examples.

3.2.1 Terminological remarks

Before moving to the meta-analysis of computational trust surveys, some terminological remarks are required. Those terminological remarks serve the purpose of identifying computational trust models that implement a conception of trust which is in line with the one provided in chapter 2. However, given the different nature of the discipline, some aspects of the socio-economical conception of trust provided must be adapted or partially ignored. The remarks which follow are thought to provide such adaptations and further clarify how trust is conceived in digital environments.

The first terminological remark is needed to distinguish the concept of trust and that of reputation. In particular, the remark is required to highlight that those are two distinct concepts. In computer science, they are often conflated and it's common to find reputation models presented as trust models. Moreover, initial trust values computed in various trust models depend solely on reputation, showing a strong dependence of the former concept on the latter[2]. However, since it is perfectly reasonable and natural to trust someone with a bad reputation or distrust highly reputable individuals, it is important to have two distinct conceptions. The following distinction between trust systems and reputation systems will be made:

> “ Trust systems produce a score that reflects the relying party's subjective view of an entity's trustworthiness, whereas reputation systems produce and entity's (public) reputation score as seen by the whole community. ” [64]

Reputation, as is given in the above definition, indicates how a given individual is perceived in a given community. Often such values are dependent on implicit biases or general past interactions between the individual and the community. For example, a doctor might have a good reputation inasfar as the academic titles he obtained and the way he interacts with his patients made him highly esteemed. Moreover, what distinguishes trust from reputation is that the reputation score of an individual is seldom personally formed by an agent, but is rather given directly by the community. On the other hand, trust must be formed by each agent, requiring, thus, a minimum level of interaction between the trusting parties. This said, in this thesis it is assumed that the reputation score of an agent can be obtained by joining all the interactions the agent had, thus representing a sort of “common trust value”. On the other hand, trust values will be considered personal evaluations, where a given agent must have access to at least some information about the other agent he shall trust: this way of defining trust, leaves open the possibility of having personal trust values entirely dependent on the reputation of a given agent, when such reputation is the only available source of information.

[2] Indeed, in some contexts, the dependency can be made even stronger by thinking about reputation as the publicly perceived trustworthiness of a person.

The second terminological remark is required by the fact that the term "trust" is applied differently in different context and therefore the same name might stand for different concepts (see the reflections made in section 1.1). The following working definition for trust will be employed:

> " . . . [T]rust implies a decision. Trust can be seen as a process of practical reasoning that leads to the decision to interact with somebody. " [102]

This definition highlights the fact that trust is involved in decision-making processes. In particular, trust is the process of evaluating whether it is valuable to collaborate with another agent or not. This definition is similar in spirit to what Audun Jøsang dubs decision trust:

> " [Decision] trust is the extent to which a given party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible. " [60]

Decision trust must be distinguished from other conceptions of trust such as *reliability trust*, for which trust indicates only the subjective assessment a trustor makes about the probability that a trustee will perform a given action on which the welfare of the trustor depends; reliability trust has a meaning similar to the pure calculative trust conception of economic theory (see subsection 2.2.2). The main difference between decision trust and reliability trust is that the former is more specific and considers more information for its evaluation: imagine there is a worn rope which has a high probability of breaking if used; that rope would never be trusted during a fire drill, but it might be during a real fire, if no other escape options are available to the trustor. In this simple example, the rope has the same reliability trust, however, since the negative consequences of not trusting the rope in the two scenarios are different, so is the decision trust value. From now on, all appearances of the term "trust" indicate "decision trust"[3]. Note that even though it is assumed that trust is always decision trust, no further assumption are made on its nature. However, this simple assumption can greatly influence the nature of trust. For example, even though trust can also be seen as a second-order relation (e.g., as a property of communications), such accounts count as examples of reliability trust, rather than decision trust. This is due to the fact that in order for a decision to be made, the trusting entities must have some sort of agency. Trust must therefore be always a first-order relation between two (or more) agents who can have the possibility of making a decision in a specific

[3] Note that this implies that the approach assumed in this chapter is different from the one present in some classical models for the assessment of trust [4, 59, 62, 67]. This is because the interest in this thesis is for a notion of trust that can be directly employed in digital environments and therefore it must be suited for dealing with decision making.

situation. This is an important point, since in chapter four of this thesis, trust will be presented as an operator on formulas, rather than a relation between agents. Anticipating what will be said later, even in the case in which trust is presented as an operator on formulas, when practical considerations are taken into account, such operators will always be applied to formulas expressing interacting situations which must be evaluated.

The final working definition of trust that will be employed in subsequent sections and chapters is thus the following:

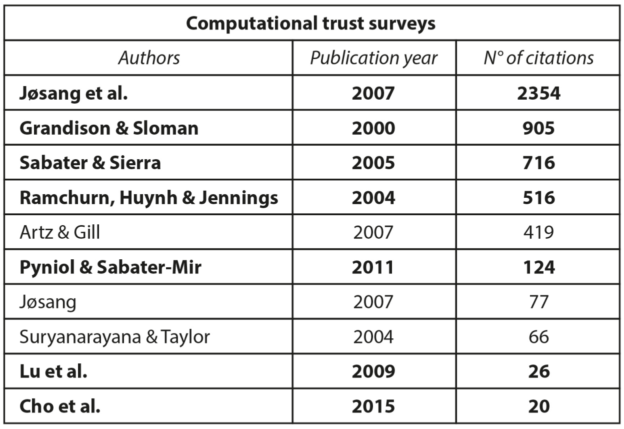
> “ [Trust is] *a belief about another person’s trustworthiness with respect to a particular matter at hand that emerges under conditions of unknown outcomes* [emphasis in original]. ” [110]

On top of such definition, it is assumed that trust, as reported, is decision trust. Therefore, the belief formed about the other person’s trustworthiness shall be employed to decide whether to collaborate with such agent or not.

3.2.2 The taxonomy

To build the taxonomy, a meta-analysis of existing surveys on computational trust models has been made. To select the surveys to analyse, a search on Research Gate was made. The combination of key words employed were: i) *Survey, Trust, Computation, Model*; ii) *Review, Trust, Computation, Model*; iii) *Survey, Trust, Computation*; iv) *Review, Trust, Computation*; v) *Survey, Trust, Computer Science*; vi) *Review, Trust, Computer Science.* Of all the survey obtained through such research pattern, ten were selected according to number of citations[4]. The ten surveys which were selected are [7, 22, 43, 60, 64, 80, 102, 107, 113, 121]. In order to avoid redundancies in the analysis, two further selection criteria were applied to the list: first, an author-based selection criterion was applied, thus excluding works which came from the exact same author; second, a temporal criterion was applied, excluding surveys published in the same year. The second criterion was applied because it is thought, by the author of this thesis, that works published in the same period might display cross influences and therefore biasing the results. Given those two added selection criteria, the survey analysed were brought down to seven. It is believed that those surveys represent a good and authoritative sample of all the general surveys existing in the computer science literature on computational trust. In figure 3.1 the list of surveys is displayed: boldface is employed to indicate which were the surveys that underwent the meta-analysis. It is important to note that many recent surveys and reviews on computational trust models were omitted, mostly due to the low number of citations. Since this parameter was the main tool for the selection of the surveys to employ in

[4] The number of citations was taken on 22/01/2018 from ResearchGate: https://www.researchgate.net/



Computational trust surveys		
Authors	*Publication year*	*N° of citations*
Jøsang et al.	**2007**	**2354**
Grandison & Sloman	**2000**	**905**
Sabater & Sierra	**2005**	**716**
Ramchurn, Huynh & Jennings	**2004**	**516**
Artz & Gill	2007	419
Pyniol & Sabater-Mir	**2011**	**124**
Jøsang	2007	77
Suryanarayana & Taylor	2004	66
Lu et al.	**2009**	**26**
Cho et al.	**2015**	**20**

Figure 3.1: Computational Trust Surveys.

the meta-analysis, recent works were penalized. However, it should be pointed out, in defence of the selection, that the surveys and reviews selected cover different aspects of trust models in great detail and the overlapping features identified can be considered, with a high level of confidence, as a comprehensive set of features shared by most computational trust models[5].

Cross-referencing the features highlighted in each survey, some interesting characteristics of trust models were extracted and four distinctive aspects of the possible way of building models were highlighted. Those distinctive aspects are the elements which help in building the classificatory taxonomy which is presented in this subsection. The aspects are:

- The typology of the trust models.
- The applicability criterion of the trust models.
- The typology of information accepted by the trust models.
- The way such information is acquired in the trust models.

Different combinations of elements characterizing the four aspect generate different computational trust models.

According to the *typology of the models feature*, a computational trust model can be either a *socio-cognitive model* or a *game-theoretical model.* A first clear characterization along this dimension of computational trust models can be found in [107] and it is then fully developed in [113]. This aspect of models determine how trust is interpreted inside the models. Game-theoretical models are typical of conceptions of trust based on economy (see subsection 2.2.2) and in those, trust is considered a probability measure, which is assessed based

[5] For completeness, some recent surveys are reported. For recent analyses of computational trust models or computational trust in general, see [2, 98, 128].

on interactions and strategic decisions. Those models reduce the whole process of trusting to computations based on precise values. One aspect to note is that, even though in economy there has been a shift towards subjective probabilities, rather than objective computations, in game-theoretical models, the probability representing trust is often computed aseptically based on the available data, with no reference to the subjective evaluation of agents. Although problematic from a conceptual point-of-view, such a choice allows for easy implementations of the models in actual computing systems. On the other side of game-theoretical models, there are socio-cognitive models. In those models, mental states and evaluations of an agent are highlighted (e.g., beliefs, desires and intentions). A thorough analysis of all the possible mental aspects on which trust might be based can be found in [22]. The major insights for the building of those models come from sociology and philosophy (see subsection 2.2.1) and trust values to be employed are computed using qualitative assessments rather than quantitative ones. Such a practice makes it hard (though not impossible) to have practical implementations of such models, since highly complicated mechanisms of representation and manipulation for the mental states are required from the computational systems. Moreover, mental notions often have an opaque meaning and lead to vague definitions, therefore making even harder understanding whether a computational system is actually implementing the correct notions.

According to the *applicability criterion of the models feature*, a computational trust model can be either a *general-purpose model* or an *application-specific model.* This distinction is firstly formalized in [102] and it characterizes the versatility of the models. In general-purpose models, the trust values computed can be employed in different contexts, allowing the transfer of computations from one scenario to the other. This allows those models to make effective use of the same computational resources in cross-domains. General-purpose models are the most versatile ones, but suffer from lack of precision, often relying on rough values for trust which might not be accurate for highly specific tasks. A remark to be made is that the versatility is not tied to the possibility of adapting the model in order to apply it in different contexts, but that the model it-self is designed to employ the same resources (i.e, the information received) to compute general trust values employable in different (and possibly all) scenarios. Those models are often just theoretical ones, with no actual implementation. When building such models, emphasis is put on ideas, rather than specific implementation concerns. However, obtaining a (possibly) precise general-purpose trust model is an important desideratum in computer science. Contrary to general-purpose models, application-specific models, as the name suggests, are models designed to compute trust values in specific scenarios or for specific purposes. Those are the most common models and literature abound with examples. The downside of building a model in an application-specific sense is that the model hardly applies in contexts which are different from the original one for which the model was built. The reason is that the architecture of those models is designed specifically to obtain a precise result in given situations.

According to the *typology of information accepted by the models feature*, a computational trust model can be either a *interaction-only based model* or a *cognitive-information based model.* This distinction is introduced in [113], expanded in [22] and it characterized the nature of the information employed to compute trust values. This level of distinction is highly dependent on the typology of the model distinction. In fact, socio-cognitive models obviously will employ cognitive information in their computations and game-theoretical models will only employ interaction results; however, some game-theoretical models allow the use of partially-cognitive information to affect their computations (when, e.g., utility functions are constructed starting from the desires of an agent). Interaction-only based models rely, for their computations, only on the outcomes of past interactions between the agents in the models. While typically related to reputation rather than trust models, the fact that some of the latter models heavily depend on reputation scores, makes is possible to construct computational trust models solely relying on the outcomes of past interactions. Those models are the easiest to build and then implement; the reason is that data is easily gathered, represented and then analysed. On the other hand, cognitive-information models consider various typologies of information (e.g., intentions of agents or aesthetical appearances of websites) to compute trust values. As with socio-cognitive models, computational models based on cognitive information are hard to implement and to build, mostly because cognitive concepts are hard to formalize. This limits their analysis to theoretical environments and often they are employed as descriptive models rather than being thought as practical devices.

Finally, according to the *way of acquisition of information by the model feature*, a computational trust model can be either a *direct-experience based model* or a *referral-based model.* In direct-experiences models, the information is acquired by direct experience and only first hand information is allowed. Such models are useful in environments in which different agents evaluate situations differently and therefore, might assign same values with different meanings. However, actual implementations of such models are difficult, mainly given the scarcity of information available. Digital communities are larger than traditional ones and repeated interactions between the same agents hardly happen. Without abundant raw data to analyse, trust models have a hard time in computing reliable trust values and therefore aid an agent in taking his decisions in the given environment. On the other side, there are referral-based models. Those models allow the use of various information from different sources, both directly acquired or reported by other agents. Thus, not only the past history of the evaluating agent is important, but all the histories of all agents from whom the evaluator can obtain data. Those histories are often collected in huge databases, from which someone can extract the information he needs and then compute initial trust values. The obvious advantage of those models lies in the fact that they allow trust computations also in contexts in which the trustor never interacted before with the trustee. However, this opens up the possible issue of having inconsistencies in the evaluations. Not all agents judge in similar ways: what might be exceptionally good for an agent, can be completely normal for another. Nonetheless, those computational trust models are the most common.

This exhausts the taxonomy. In the next section, classical computational trust models are going to be analysed in order to extract some core features necessary for a computational conception of trust.

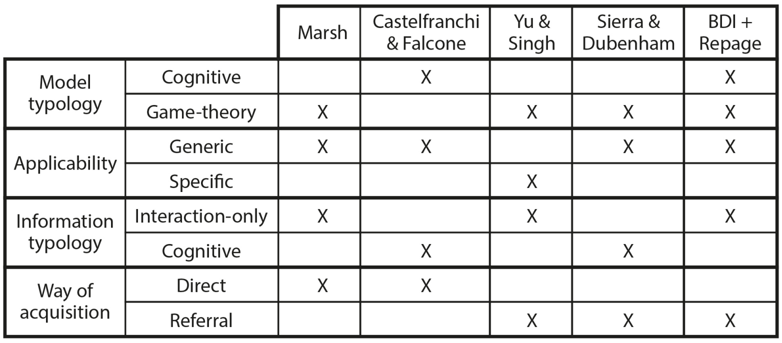
3.3 Computational Trust Models: an Analysis

In this section, some classical computational trust models are going to be analysed, in order to obtain some core features of a computational conception of trust. Those features will then be compared in the final section with the ones obtained in the previous chapter. It should be remembered that the choice of the models is partially influenced by the terminological remarks made in subsection 3.2.1. Therefore, it could be thought that the comparison between the features obtained in this section and the one obtained at the end of the previous chapter is futile, since the latter deeply influenced the former. However, it is claimed that such a comparison could still prove to be useful, inasmuch as understanding how socio-economical concepts are manipulated to obtain implementable version has great theoretical importance and can aid in future research on similar topics.

The models analysed were selected from a citation analysis made on the surveys employed for the construction of the taxonomy. On top of this first layer of selection, further refinements were made to increase the diversity of the models analysed. To obtain such a result, models were selected by taking different combinations of the aspects highlighted by the taxonomy. Those selection criteria individuated five classical computational trust models. The five models which were selected are [20, 86, 101, 118, 135]. In figure 3.2 it is possible to see the way such models are sorted according to the taxonomy.

3.3.1 Marsh's Trust Model

Marsh's computational trust model [86] is one of the first appeared in the computer science literature. Developed in his Ph.D. thesis, the model proposes a formal framework for trust, with the idea of applying it to distributed artificial intelligence and multi-agent systems. In his model, it is possible to identify three different forms of trust: basic trust, general trust and situational trust. *Basic trust* represents the general attitude of an agent, when all his experiences in life are considered; *general trust* is the overall trust a trustor has in a trustee; finally, *situational trust* is the specific trust a trustor has in a trustee when a specific collaborative task should take place. Given



		Marsh	Castelfranchi & Falcone	Yu & Singh	Sierra & Dubenham	BDI + Repage
Model typology	Cognitive		X			X
	Game-theory	X		X	X	X
Applicability	Generic	X	X		X	X
	Specific			X		
Information typology	Interaction-only	X		X		X
	Cognitive		X		X	
Way of acquisition	Direct	X	X			
	Referral			X	X	X

Figure 3.2: Classical examples of computational trust models.

the working definition of trust given in subsection 3.2.1, only situational trust qualifies as proper trust for this thesis and therefore that form of trust is the only one that will be analysed.

Situational trust represents the attitude an agent has in another agent. When a decision has to be made, different agents assess different qualities of the trustees and then decide whether to collaborate or not. In this sense, another characteristic of situational trust is its measurability. Situational trust is always represented with a value between -1 (complete distrust) and +1 (complete trust). The third characteristic is related to the fact that trust depends on the given situation, therefore it is context-dependant. Finally, the fourth and last characteristic is that situation trust is agent-dependant, i.e., each agent computes his personal value and two agents might return different results even when they possess the same information and evaluate in the same situation.

3.3.2 Castelfranchi and Falcone's Trust Model

In [20], the authors offer one of the first cognitive approaches to the formalisation of trust. In their model, trust is taken as a reducible concept, where all its parts are mental states. First of all, trust is again a relation between two agents, evaluated towards the achievement of a goal, therefore depending on the context. On top of that, there are the mental states of the trustor. The trustor must: have a specific goal in mind; believe that the trustee has the competence to perform some positive actions towards the goal; believe that the trustee is willing to perform the given action towards the goal if he is given the chance to; believe that he indeed needs the trustee to achieve the goal. All the above mental states contribute to the evaluation the trustor makes and, as a result, it is possible, in the model, to obtain a binary trust value (trust or not-trust), which is then employed in decision-making.

3.3.3 Yu and Singh's Trust Model

In Yu and Singh's [135] trust model, trust is completely based on reputation, which is based both on previous interactions between the trustee and the trustor and on reported interaction with the trustee by other agents in the social network of the trustor, which are registered in the formalism as values called Quality of Service. It is important to note that the two sets of data are always kept separate and an agent can report only direct interaction he had with the trustee and no other agents' ratings; this is to prevent that biased or unfair ratings are propagated to other agents. This model also includes a trust managing component, where new trust vales can be computed from old ones using Dempster-Shaffer's theory of belief revision. In this case, trust is a specific value attributed to each trustee by each trustor. The main source of information to compute trust are interactions, either direct or indirect. This leads to the fact that two different agents, with the same social network and the same past experiences, will attribute the same trust value to the trustee. Finally, in the formalism the past interactions used to compute trust should be similar to the interaction for which a decision must be made, providing a context-dependant value for trust.

3.3.4 Sierra and Debenham's Trust Model

In Sierra and Debenham's model [118], trust is a "...measure of expected deviations of behaviour..." and is represented as a conditional probability. While in this model trust is only a component and not the focus of attention, it is still possible to extract some core characteristics of the concept. First of all, trust represents a relation between what is expected from a signed contract and what the contract actually says; in this sense, if the trustee is expected to act in complete disagreement compared to what the contract says, then there will be complete distrust. Another characteristic of trust is that it is context-dependant, insofar as the expectancy might be different for different contracts (someone might behave properly for low-valued contracts and maliciously for high-valued contracts). Being a conditional probability, trust has a definite and continuous numerical value. Finally, the value is subjective, given that different agents might have different access to pieces of information that can generate different expectancies (an agent might know that the trustee has breached the terms of conditions of other contracts in the past).

3.3.5 BDI and Repage

The model BDI and Repage [101] integrates a previous model, Repage, with a BDI (Belief, Desire, Intention) component. This model is a logical framework with which to reason about the reputation of the trustee, considering the mental attitudes of the trustor. It should be noted that in this model there is no direct reference to trust, however, for the way the model works, the concept of reputation employed falls under the working definition of trust given in subsection 3.2.1. In the model there are six components, each one contributing to a final decision. There are three components representing the three mental states, two components representing communication and planning and one representing the Repage model. The Repage model should be considered the principal component representing trust. In such component, trust is evaluated using two items: the image and reputation of the trustee. The image stands for personal opinions about the trustee while reputation is a general and public score. Both scores are context-dependant and they might differ, in line with the distinction made between trust and reputation. In Repage, social evaluations are coded as first-order formulas and are organised as hierarchies of different levels of abstraction, with images and reputation predicates holding the top positions. The value of each evaluation is indicated with a quintuple, where an additivity criterion is required; this value of the image/reputation predicate indicates the trust the trustor has in the trustee. On top of the Repage component, the three BDI components complement the model, by suitably manipulating the first-order sentences. In this model, trust is a relation between two agents depending on a context. Moreover, the value is completely subjective in nature, given the reference to the mental states of the trustor and his image. Finally, trust has a definite value, given in terms of tuples in the Repage component and as Booleans in the final computations of the whole model (collaborate or not).

3.4 Computational Trust in a Nutshell

In this section, the core features of computational trust extracted in the previous section from classical computational trust models are summarized and then compared with the results obtained at the end of the previous chapter. This will help in building an alternative and novel formal trust language, employable in computer science as a guide to the implementation of the socio-economical notion of trust.

From the previous section, four characteristics of computational trust can be extracted. Those characteristics determine the nature of trust in computer science; trust is: a) relational; b) subjective; c) measurable; d) context dependent. It is argued, given the methodology followed to obtain them, that those characteristics form a necessary set of elements that all computational conceptions of trust ought to possess to qualify as such. All characteristics will be analysed in turn.

Computational trust has a *relational nature.* This means that there must always be someone/something trusting (trustor) and someone/something to be trusted (trustee). What might differ from model to model is the nature of the entities, their number and the properties of the relation itself. The entities can be humans or inanimate objects. They can be two entities, making the relation one-to-one; they can be groups, making the relation one-to-many, many-to-one or many-to-many. The properties of the relation can differ, being, e.g., reflexive (self-trust), symmetric (reciprocal trust) or transitive (referral trust). All those differences characterize different models and slightly different conceptions of trust, but they all share the fact that trust is represented as a relation.

Computational trust has a *subjective nature*. This means that trust depends for its value on the entity making the valuation. This characteristic is extremely important, because it might help in distinguishing between aseptic and impersonal computations from actual trusting evaluations. Even more important, the subjectivity component helps in drawing a better and clear distinction between what might be labelled as reputation and what can be labelled as proper trust. Trust is always something personal, while, on the other hand, reputation always expresses a general opinion shared by everyone.

> “ ... [T]rust ultimately is a personal and subjective phenomenon that is based on various factors or evidence, and that some of those carry more weight than others. Personal experience typically carries more weight than second hand trust referrals or reputation... ” [64]

What might differ from model to model is the way this subjectivity is captured: the subjective nature of trust might depend, e.g., on the difference in past interactions between entities or on the different social networks agents have; sometimes, this subjectivity can be represented using agent-specific general attitudes or by specifying an agent’s peculiar mental states.

Computational trust has a *measurable nature*. This characteristic is what marks the difference between computational conceptions of trust and traditional ones. Computational trust is often designed with the idea of implementing the concept in computational systems; being this the case, it is a prerequisite that the concept is somewhat measurable. On the other hand, even though it still might be a measurable concept, in traditional models, this is not a strict requirement. For example, in Luhmann’s works, no direct reference to measure is ever made, even if some can argue that the presence or absence of trust is itself a form of measurement (specifically a Boolean form of measurement). What might differ from model to model is the nature of the measurability and the number of values trust has: the measures can either be qualitative (e.g., using words) or quantitative (e.g., using natural numbers); moreover, there might be any number of values in play, from two (e.g., 0/1) to a continuous number of them (e.g., using real numbers). It should be noted that it is always possible to move from qualitative to quantitative valuations and vice-versa. Likewise, it is possible to move from valuations with more values to valuations with less values, but the contrary is not always possible.

Finally, computational trust has a *context-dependent nature*. This characteristic is controversial, since it depends for its necessity on the definition of trust that is employed. To understand this point, it should be noted that in Marsh's model there were three different concepts of trust and only the latter (situational trust) depended for its value on the context of application. Even though controversial, this characteristic of trust is still valuable, especially considering that computational trust models should be viewed (and are viewed in this paper) as soft-security mechanisms that favour cooperation and facilitate transactions on the web. If trust models are viewed as soft-security mechanisms, different scenarios call for different computations of trust: rewarding a vendor for the fact that he/she is good at selling books by granting him/her trust, doesn't mean the vendor should also be trusted when he/she sells insurances. Furthermore, even admitting different definitions for trust, it might still be possible to provide a relevant and related definition of context that maintains the characteristic intact. For example, using Marsh's model again, it is possible to define a notion of general context, i.e., a context that encompasses all other contexts, for the respective notion of general trust.

3.5 Computational Trust: Variations on the Theme

In this section, two paradigms important for building models for computational trust are presented and examined (see [61] for an analysis which taken into consideration other models and proposes another paradigm for computational trust model constructions). The focus is on abstract evaluations, rather than on specific models. No specific examples of models will be given, if not strictly necessary and, even then, only for exemplative purposes. Specifically, the two paradigms that are examined are the Beta Paradigm [95] and the EigenTrust Approach [67]. Peculiar features of those paradigms are identified and the general methodologies they generate are explicated. Stress in put on ideas, rather than concrete approaches. The aim is to identify possibilities for improvements, maintaining their good features and modifying the bad ones. Under those considerations, the section is developed in a descriptive fashion, trying to limit the use of formulas and mathematical devices to the minimum necessary. The section in structured as follows: i) first, the Beta Paradigm is discussed; ii) second, the EigenTrust Approach is explained and iii) finally, general conclusions will be drawn from the discussion.

3.5.1 Beta Paradigm

> "The Beta model is central to the Beta paradigm. " [95]

The Beta paradigm is a set of assumptions made on how trust (and related notions) should be interpreted [63]. This paradigm is principally based on the so called *Beta function.* A Beta function is a statistical device employed to properly interpret binary events, whose outcomes are either successes or failures. Specifically, a Beta function is classified as a continuous function indexed by two parameters α and β.

Definition 1 (Beta Function)**.** *The Beta distribution, denoted* $f(p \mid \alpha, \beta)$ *can be expressed using the gamma function* Γ *as follows, where* $0 \leq p \leq 1$*,* $\alpha > 0$*,* $\beta > 0$*:*

$$f(p \mid \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1} \tag{3.1}$$

Starting from the Beta function, it is possible to compute expected values for the occurrence of future events. The expectancy formula is the following:

$$E(p) = \frac{\alpha}{\alpha+\beta} \tag{3.2}$$

The expected value represent the most likely value of the success of a future event (which, in this case, are transactions between agents), while the Beta function indicates the distribution of likelihood of all values (i.e., it indicates, for all possible values of x, what is the probability that a future interaction will succeed with likelihood x).

Give this instrument, the Beta paradigm makes three major assumptions on the nature of trust and how the results of applying the Beta function should be interpreted:

1. Trust is a interaction-only oriented phenomenon, thus ignoring any specific cognitive aspects of the interacting agents (nobody is strictly necessary for an exchange to take place, because the goal of the interaction is all that matters, not the participants); moreover, interactions are all equal in nature or, at least, similar enough to be grouped under a single category.

2. All interactions are fully analysable to the point of clearly determining their goal; on top this, it is objectively possible to identify whether the goal was achieved or not, univocally and surely determining success and failure.

3. There is not enough information in the environment which might allow an agent to determine the other agents mental/cognitive states, therefore expectations are purely based on computations based on environmental evaluations.

Note that those assumptions identify a specific class of models with respect to the taxonomy given in subsection 3.2.2: computational trust models based on the beta paradigm are *game-theoretic, specific, interaction-only* based computational trust models.

The first assumption has the purpose of letting a modeler specify clearly a meaning for trust opinions, which are simple estimations that future interactions are successful, based on reflections on similar interactions in the past. Moreover, agency is thought to have very little impact on the success of the interaction, since cognitive phenomena are neglected under the Beta paradigm. The second assumption has the purpose of creating a standard of evaluation. On one hand, no interpretation of results is possible; they are either positive (success) or negative (failure). Moreover, those evaluations are completely objective, making it possible to treat second-hand information just as first-hand one. After all, in combination with assumption one, no cognitive aspects can influence an opinion and therefore there is no space for personal judgement and taste. The third assumption has the purpose of eliminating disturbing factors from computations of possible future success. If nothing about the other agent can be known, expect for how many successful interactions he had, to compute expectations an agent will employ only such data and ignore the rest. In the Beta paradigm, the value representing the likelihood behaviour of an agent is indicated with the word *integrity.* A correct computation about the integrity of agents is a central tenet of the Beta paradigm. As quoted in the opening of the subsection, another tenet of the Beta paradigm is the *Beta model.* A Beta model is a formal model based on the Beta distribution function. In turn, the Beta distribution function is a well-known statistical device employed in binary systems to compute conditional expected values given Booleans observations as inputs and the Gamma function as operator on those inputs [19]. Being a probability distribution function, its role is that of indicating what is the probability of different integrities of agent, which, in turn, determine the likelihood that the agent will succeed in a future interaction. Being a function which is constantly updated, it might be said that the Beta function falls under a Bayesian interpretation of probability. In addition, by determining the likelihood that an agent has a give integrity, the function can be said to provide a measure of how much an agent should trust the other agent in concluding the interaction. In such a sense, the function is also compatible with the idea of attributing high likelihood to a low integrity of the other agent, thus expressing the idea that it is possible to trust someone not doing something appropriate for the trustor.

As previously pointed out, this kind of paradigm offers a valuable environment for specific classes of computational trust models. Moreover, it is absolutely ideal for reputation systems, where general trustworthiness values must be computed, ignoring subtleties of single agents intentions and cognitive states. However, note also that the class it identifies, i.e., *game-theoretic, specific, interaction-only* based computational trust models, is highly restricted and makes assumptions on trust which are in contrast with many experimental data on what trust is and how it is fostered (see subsection **??**. The reason is still accepted and extensively employed is that the characteristics of this class of models are the best available to produce practical computational implementations, therefore they are all desirable from the point-of-view of applied computer science. Nonetheless, researchers shouldn't content theirselves with those results, mainly because having a conception of computational trust much different from the socio-economical one might have also impacts on the positive effects trust fosters, possibly diminishing them to be point of being futile to have such computational implementations. That said, the Beta paradigm accomplishes to goal of providing good assumptions that any modeler wishing to have practical implementations of his formal systems must keep in mind and this is what is going to be done in future sections of this thesis[6].

3.5.2 EigenTrust

EigenTrust is an algorithm employed to manage reputation in P2P networks [67]. Thus, technically speaking, EigenTrust is neither a paradigm nor it refers to trust. Instead it is a specific algorithm to represent reputation in digital environments. Nonetheless, it is included in this section because the assumptions made when building the algorithm can be seen as a set of features denoting a specific paradigm for computational trust. In particular, those assumptions seem to mimic a typical environment where trust might have developed (see chapter 2). The assumption are that the system in which EigenTrust should be applied ought to be:

1. Self-Policing, where there is a shared ethic and the enforcement of such ethic is promoted by the users themselves.
2. Anonymous, where there is no way of individuating directly a user, but only his digital profile.
3. Profitless for newcomers, where entering the system for the first time, doesn't grant any immediate benefit.
4. Minimal in terms of overhead, where the resources required for the system to perform well are minimal.
5. Robust, inasfar as it has to be able to survive attacks from malicious collectives who collaborate.

As said, those assumptions seem to mimic an ideal situation for trust to emerge: cognitive phenomena are at the centre and are self-regulated by the agents and not the environment; true intentions and personalities are opaque, inaccessible to most agents; trust is earned through interactions and

[6] It is important to be aware that the most prominent model based on the Beta paradigm is Subjective Logic. The model is not included in this chapter because it will be analysed thoroughly later in the thesis, where parallels will be built between this formalism and the one proper of this thesis.

not granted freely; the task of evaluating trust require minimal effort, thus, reducing the complexity an agent has to face in his environment; the system is intentionally designed to overcome cheating behaviours. In this scenario, EigenTrust provides efficient ways of producing scores, which represent the willingness of a given agent to collaborate or not with the agent being evaluated. Specifically, agents stock information about interactions in a simple way, dividing them between satisfactory and unsatisfactory transactions. This information form local trust values, from which it is possible to compute an expected value for possible future transactions. However, those transactions might be rare on the web, where the amount of users and the size of the market, make it implausible that the exact two agents interact often, thus limiting oneself to only local trust values might be senseless. For this reason, the main contribution of EigenTrust is that of providing efficient ways of aggregating local trust values into general trustworthiness values for a given agent. Therefore, EigenTrust identifies a class of models quite different from the one individuated by the Beta paradigm. Specifically, it identifies *cognitive, generic and referral* based computational trust models. Note that no specific reference has been made to the typology of information that is taken into consideration by EigenTrust. This is because, even though the ethics of the agents participating in the system is important to determine the way the algorithm will spread some data over other, no measure that captures cognitive information directly is included in the algorithm. Therefore, it seems that the cognitive information is valuable only as a meta-variable and not as a specific feature of the computations.

In the algorithm, starting from primitive values (determined by a restricted set of agents who are assumed to be trustworthy), each agent is associated with a generic score, obtained by aggregating all the local trust values of other agents who have had previous interactions with such said agent. The way data is aggregated allows to address important practical issues. First of all, it provides a way to compute pre-interaction trust values. It does so by combining the evaluations of agents with special-status, indicated initially in the design of the system and seldom updated; thus, this set of agent is solid with respect to malicious attempts to make the system unstable from the base. Second, the system can deal with inactive users, where someone is inactive if he doesn't interact with other agent for prolonged periods of time. In such a case, the algorithm sets the value for such agent to that indicated by the pre-trust evaluators. Finally, and most importantly, the algorithm is able to deal with malicious collectives (it was actually this issue which encouraged the authors to develop EigenTrust). This is achieved by including several mechanism which lower the possibility for an agent to assign high values to close peers in the system and low values to all the distant peers. By adjusting values according to a general average and selecting probabilistically the evaluators from which to take the data, EigenTrust can isolate malicious pees and greatly reduce their potential to harm the system. Moreover, freeriders (comparable to subtle cheaters) are incentivized to become honest dealers. The principal reason is that, in EigenTrust, higher levels of reputations (and, thus, trust) allow agents to have access to same-reputation cycles of users. Thus, by reciprocating less, the freerider would be inserted in a cycle where it is likely that the other agents are themselves freeriders. This would likely cause the freerider to benefit less than he could otherwise, therefore increasing his willingness to reciprocate more and have access to a higher quality cycle.

As with the Beta paradigm, the main advantage of EigenTrust is that it is easy to implement in computational systems. Moreover, it highlights aspects of trust which seem to resemble the features typical of a primitive notion of trust. However, as it has already been noted, those features are partially relegated to the meta-analysis level, where it becomes apparent that they are not implemented computationally in a direct way, but are produced by human beings through the computational system. The difference between having the features in the system, rather than being used upon the system, is an important one, since in the latter case, the algorithm would be useless in a machine-machine or a human-machine interaction. It is therefore important to recognize that the assumptions made by EigenTrust are also relevant for computational system and not only biological and sociological systems, but that those assumptions shouldn't just be a by-product of the way humans use the system, but specific features of the system itself.

3.5.3 Learning from the Variations

A close analysis of the two approaches presented for the implementation of trust in digital environments, made it possible to recognize that there is no unique and correct way to formalize trust computationally. Specifically, what is interesting to note, is that the two approaches differ greatly in how they set the assumptions. The former (Beta paradigm) sees agents as standardized users, which always evaluate interactions of the same kind; the latter (EigenTrust) is designed with the idea that different agents might act with different purposes and therefore evaluate situations differently (up to the point of intentionally misjudging).

What should be highlighted, however, is that both system display a great success in being able to practically implement their respective notions of trust. They achieve this goal by employing precise and well-designed assumptions that permit to identify the scope of the models, without giving up genera spirit. It is therefore necessary, for any suitable system that aspires at a good representation of computational trust, to mimic the precision with which those approaches introduced their assumptions and try to recognize which of those assumptions make the formalisms adapt for practical implementations. In particular, for the scope of this thesis, and partially inspired by the assumption shown in the previous models, the assumptions which will be taken into consideration and will be implemented in the design of the logical language for trust are:

1. Information is fully analysable, to the point of clearly determining its content, i.e., the content of the information is always clear to an agent and, in case of ambiguities or fuzziness, the agent has the necessary resources (let them be time or computational power) to clarify the ambiguities and sort out the fuzziness. It is, thus, always possible to determine if an information is relevant to an interacting situation and it is possible, furthermore, to establish whether the information is possessed or not by the agent which must entertain the interaction. This assumes that the language employed is itself free from fuzzy or ambiguous concepts.

2. Information in the environment is scarce, thus, most evaluations are based on personal opinions about this information and not the actual information *per se*. This is equivalent to the fact that the most relevant aspects employed in making decisions are beliefs, rather than actual knowledge.

3. Not only will new agents lack a trust value, but they will also not be aware of the trust values of others. In this sense, each new agent must form his own knowledge base independently, thus achieving his personal trust values without having to rely of the opinions of others.

4. Even though interactions can be of the same type, they might depend on different aspects, depending on the agents who are evaluating the interaction. Therefore no general, universally recognized value, can be computed based on past interactions, because the success or failure of a past interaction might depend on contingent elements that where not factored in the simple annotation of success or failure.

Assumptions 1. - 4. are going to be the starting point for the proposal of next chapter.

Chapter 4

A Language for Trust

In this chapter, the goal is that of providing a formal structure in which to reason about trust and its relation with information and knowledge. To achieve this goal, modal logic will be employed to formalize knowledge, while a special structure for trust is designed and implemented into the language. There are four reasons to choose logic, over other formalisms for trust (see chapter 3).

The first reason is that logic has a great descriptive power: once a logical language is built for a given context of application, it is rather easy to translate really complicated statements within such context with simple logical formulas; this allows a modeler to avoid ambiguities and to make intelligible statements that, otherwise, would be hard to comprehend and manipulate.

The second reason is that logic has a great normative power: once general rules for derivations and proofs are established, it is easy to determine good practices of reasoning from bad ones, thus allowing a modeler to check straightforwardly which facts follow from which other. In the case of trust, this aspect is even more important, because it might be highly detrimental for an agent to accept a specific fact as trustworthy based on incorrect inferences.

The third reason is that logic is closely connected to computer programming: excluding the obvious relation between declarative programming and logical languages (see, e.g., [70]), the standards of precision and clarity of logic and the way semantical structures are built resemble that of computational thinking and programming. This allows a modeler to move from specific manipulations on the semantical structures to the design of algorithms.

The fourth reason is that logical formalisms are set at a highly abstract level: it makes few assumptions on the way information is represented and how to implement the actual manipulations on it. Starting from such a high level of abstraction, it is possible to move top-down towards more concrete implementations (see chapter 5 for two possibilities of this sort). Achieving the highest possible abstract description of trust relationships and their interaction with knowledge should be a desirable goal of a unifying formalism for trust. In fact, if the goal is achieved, then all instances of trust would be special cases of the abstract notions described in the formalism; having such a dependence between an abstract concept of trust and the concrete examples in which trust manifests itself can greatly enhance our comprehension of latter instances and the way various circumstances can elicit feelings of trust.

The standard rigour of logic, coupled with its descriptive and normative power, makes it a suitable choice for the formalization of complex concepts, especially of those concepts which play important roles in our lives. The choice becomes even more desirable, if the formalization is thought to be a first step towards a computational implementation of the target concepts. Thus, for this thesis, there was little doubt that the direction to take was that of building a logical language for trust, rather than develop other existing theories and models of trust (see chapter 3 for some examples).

As said, the basic language employed in the thesis to formalize trust is that of modal logic. The semantical structure that will provide meaning to the formulas of this language is that of augmented neighbourhood semantics and the part referring to trust depends on a weight assignment to formulas, where the weights will determine the relevance of those formulas to trust. The reasons to choose neighbourhood structures[1], over the more common relational structures (otherwise known as Kripke structures) is mainly practical: since in neighbourhood structures all the operations are performed over set of states, rather than on formulas, and since those operations are principally algebraic or set-theoretic in nature, it is straightforward to move from the logical language to computational implementations. Moreover, being neighbourhood structures a proper generalization of relational structures, they permit more freedom in the choice of axioms and properties of the semantical structures. This flexibility is, again, a desirable feature for a language that is thought as the theoretical base for practical implementations of a concept of trust: being able to implement slightly different notions of trust in computational systems, allows the modeler to adapt his systems to the specific scenarios he is building such systems for.

The chapter is structured in the following way. In section one, an introduction to the basic modal language and to neighbourhood semantics is made; focus will be put on theorems and properties of those particular semantical structures for modal logic. The section is advisable also for expert reader, inasfar as they can benefit from acquiring some familiarity with the formalism and the various symbols employed. In section two, a first approach to add trust evaluations to a formal language is explored; in particular, a context-free, single agent language for trust is introduced and examined. In section three, a second approach is explored; contexts are added to the semantical structure in order to properly reproduce the main features of trust (see sections **??** and **??**

[1] This typology of semantical structure has been fairly neglected in the logical literature in the past forty-five years. However, they received more attention recently: see for example [27, 41, 96, 97, 100].

to obtain a list of such features). Finally, in section four decidability theorems will be proved and complexity results will be given for the language introduces in section three.

4.1 Modal Logic: an Introduction

In this section, a basic introduction to the formal structures employed to build a logical language for trust is made. The aim is to introduce all the necessary notions and theorems which will allow the reader to better understand the reasons and ideas that guide the proposed language given in section 4.3. The structure of the section is the following. First, a brief introduction to modal logic and its scope is made. Second, modal logic based on neighborhood semantics is introduced: specifically, definitions for the syntax and semantics are given. Third, invariance and decidability results for modal logic based on neighbourhood semantics are shown (proofs included). Finally, a short analysis on what might be needed in a logical language to implement trust measures is made.

4.1.1 Brief Introduction to Modal Logic

The following introduction is based on [12, 21, 48, 100].

Modal logic is a particular branch of logic in which the modeler can qualify the truth of propositions of the language. In this sense, modal languages are extremely useful in dealing with reasonings in intensional contexts, that is, contexts in which the *principle of extension* does not hold, i.e. it is not possible to always determine the truth value of composite sentences solely based on the truth values of their components.

To better understand the difference between extension and intension take the following example. Take the sentences:

1. Hesperus (the morning star) is Hesperus.
2. Phosphorus (the evening star) is Phosphorus.
3. Hesperus is Phosphorus.

Since both Hesperus and Phosphorus are, in fact, the planet Venus, all previous sentences say the same thing (convey the same meaning), that is, Venus is Venus. However, when reading the sentences, sentences 1. and 2. seem to be uninformative, while 3. provides useful insight about the morning and the evening stars, i.e. that they are the same object. Those sensations depend on the fact that words have both a reference/extension (the object designated by the word) and a sense/intension (the way such word designates the object). An extensional context is a context in which the reference of words is the only thing that matters, while an intensional context is one in which the sense of words is also relevant.

The extensionality principle is based on this distinction between extension and intension of a word (sentence)[2]. The contexts in which the extensionality principle hold are those contexts in which it is possible to substitute in propositions expressions with the same extension without modifying the truth value of the proposition (commonly known as substitution *salva veritate*). An example of purely extensional context is mathematics. For example, given the two sentences:

1. $3 + 2 = 5$.
2. $5 = \sqrt{25}$.

It is possible to substitute in the second part of sentence 1. an expression that is extensionally equivalent to it, i.e., the second part of sentence 2., and obtain a new sentence with the same truth value, i.e., $3 + 2 = \sqrt{25}$. In an intensional context, on the other hand, this kind of substitutions are not guaranteed to produce equal truth values. Take, as an example, a modified version of the previous example:

1. I know that $3 + 2 = 5$.
2. $5 = \sqrt{25}$.

Now, produce the same substitution made in the previous example, generating the sentence: I know that $3 + 2 = \sqrt{25}$. Assuming that the first two sentences are true, there is no guarantee that also the third is. This leads to the conclusion that knowledge contexts (often labelled epistemic contexts) are intensional. What matters is how an object is designated, not the designated object *per se*.

Given its close relation to intensional contexts, modal logic proved to be incredibly useful to analyses in various fields. From linguistics [91, 92, 93], to computer science [103], passing through economy [8], modal logic provided valuable tools to improve the comprehension of different phenomena in different fields. In the following sections, it will be shown that this fragment of logic can provide further benefits to the comprehension of complex phenomena, such as trust.

4.1.2 Modal Logic: Syntax and Semantics

A language, formal or not, is always defined through syntax and a semantics. Therefore, in order to define the basic modal language at the base of the main proposal of this thesis, syntax and semantics will now be presented. Note that

[2] See [38] for an exposition of the whole theory of sense and reference.

modal logic can be developed at different levels of generality, but in this thesis there will be a focus only on the propositional part of modal logic, leaving aside predicate considerations.

Definition 2 (Basic Modal Language)**.** *Given a (finite or countable) set of basic unanalysed propositions At, the set of well-formed formulas generated from At, called $\mathcal{L}$, is the smallest set of formulas defined by the following BNF grammar:*

$$\phi := p \mid \neg\phi \mid \phi \wedge \phi \mid \Box(\phi) \mid \Diamond(\phi)$$

where $p \in At$.

In the following language connectives are interpreted as in standard propositional logic, while the modalities are given by the two modal operators $\Box$ (called box) and $\Diamond$ (called diamond). Those (unary) operators are duals, which means that it is possible to specify one in terms of the other, employing negations, i.e. $\Box(\phi) = \neg\Diamond(\neg(\phi))$. The way those modalities are read can vary depending on the use of the language. They might have, for example, an alethic or an epistemic reading, where in the former the box modality is read as "it is necessary that" and in the latter it is read as "it is known that"; other readings are obviously possible. To formally interpret this language, a semantical structure is necessary. The semantical structure introduced in this thesis is the one based on neighbourhood semantics. Neighbourhood semantics is a generalization of standard Kripke semantics for modal logic and is based on the notion of a neighbourhood frame:

Definition 3 (Neighbourhood Frame)**.** *Let S be a non-empty set of states. A function $N : S \to \wp(\wp(S))$ is called a* ***neighbourhood function****. A pair $\langle S, N \rangle$ is called a* ***neighbourhood frame*** *if S is a non-empty set and N is a neighbourhood function.*

On a neighbourhood frame it is possible to assume some properties. Those properties are implemented by imposing some constraints on the behaviour of the N function. Some interesting properties are the following:

Definition 4 (Possible Properties of N)**.** *Let S be a non-empty set of states and $\mathcal{U} \subseteq \wp(S)$. Then basic properties of $\mathcal{U}$ are:*

- *$\mathcal{U}$ is* ***closed under intersection****, if, for any collection of sets $\{X_i\}_{i\in I}$ s.t. $\forall i \in I$, $X_i \in \mathcal{U}$, $\bigcap_{i\in I} X_i \in \mathcal{U}$. With $\mid I \mid= 2$, then it becomes* ***closure under binary intersections****. With I finite, it becomes* ***closure under finite intersections****.*

- *$\mathcal{U}$ is* ***closed under unions****, if, for any collection of sets $\{X_i\}_{i\in I}$ s.t. $\forall i \in I$, $X_i \in \mathcal{U}$, $\bigcup_{i\in I} X_i \in \mathcal{U}$. The same considerations as above can be applied.*

- *$\mathcal{U}$ is **closed under complements**, if, for each $X \subseteq S$, if $X \in \mathcal{U}$, then $\overline{X} \in \mathcal{U}$, where $\overline{X} = \{s \mid s \in S, s \notin X\}$ is the complement of X.*

- *$\mathcal{U}$ is **monotonic**, if, for each $X \subseteq S$, if $X \in \mathcal{U}$ and $X \subseteq Y \subseteq S$, then $Y \in \mathcal{U}$. Other ways of defining this property are: being **supplemented** or being **closed under superset**.*

- *$\mathcal{U}$ is a **clutter** if $\emptyset \notin \mathcal{U}$ and, moreover, $\nexists X, \nexists Y$ s.t. $X, Y \in \mathcal{U}$ and $X \subset Y$.*

- *$\mathcal{U}$ **contains the unit**, if $S \in \mathcal{U}$; similarly for the empty set.*

- *$\mathcal{U}$ **contains its core**, if $\bigcap(\mathcal{U}) \in \mathcal{U}$, where the set $\bigcap(\mathcal{U})$ is called the core of $\mathcal{U}$.*

- *$\mathcal{U}$ is **proper**, if $X \in \mathcal{U}$ implies $\overline{X} \notin \mathcal{U}$.*

- *$\mathcal{U}$ is **consistent**, if $\emptyset \notin \mathcal{U}$.*

- *$\mathcal{U}$ is **non-trivial**, if $\mathcal{U} \neq \emptyset$.*

Composite properties of $\mathcal{U}$ are:

- *$\mathcal{U}$ is a **filter**, if it contains the unit, is closed under binary intersections, and is monotonic.*

- *$\mathcal{U}$ is a **topology**, if it contains the unit, the empty set, is closed under finite intersections, is closed under arbitrary unions.*

- *$\mathcal{U}$ is **augmented**, if is contains its core and is monotonic.*

It is possible, in the language, to extend properties of the function N to frames. Therefore, it is possible to say that a neighbourhood frame $\langle S, N \rangle$ is augmented, if $\forall s \in S$, $N(s)$ is augmented.

The definition of neighbourhood model follows directly from that of neighbourhood frame:

Definition 5 (Neighbourhood Model)**.** *Given a neighbourhood frame $\mathcal{F} = \langle S, N \rangle$, a **model** based on $\mathcal{F}$ is a tuple $\langle S, N, \pi \rangle$, where $\pi : At \to \wp(S)$ is a valuating function (assigning set of states to each unanalysed proposition $p \in At$).*

Before giving the truth theoretical definition of the language, a further function is introduced, which is then employed to define the notion of truth-set of a formula. Note that a neighborhood function N can induce a map m_N, which is a function that associates to each element $X \in \wp(S)$ another element $Y \in \wp(S)$, according to the neighborhood function N, i.e. given

$N : S \to \wp(\wp(S))$, we have a $m_N : \wp(S) \to \wp(S)$. The function m_N is defined formally as follows:

$$m_N(X) = \{s \mid X \in N(s)\} \tag{4.1}$$

Definition 6 (Truth set)**.** *Given a neighbourhood model $\mathcal{M} = \langle S, N, \pi\rangle$, then the truth set of a formula $\pi^{ext}_{\mathcal{M}}(\phi)$ is defined recursively as follows:*

$$\begin{gathered}\pi^{ext}_{\mathcal{M}}(p) = \pi(p).\\ \pi^{ext}_{\mathcal{M}}(\neg(\phi)) = S - \pi^{ext}_{\mathcal{M}}(\phi).\\ \pi^{ext}_{\mathcal{M}}(\phi \wedge \psi) = \pi^{ext}_{\mathcal{M}}(\phi) \cap \pi^{ext}_{\mathcal{M}}(\psi).\\ \pi^{ext}_{\mathcal{M}}(\Box(\phi)) = m_N(\pi^{ext}_{\mathcal{M}}(\phi)).\\ \pi^{ext}_{\mathcal{M}}(\Diamond(\phi)) = S - m_N(S - \pi^{ext}_{\mathcal{M}}(\phi)).\end{gathered}$$

The definition of truth in a pointed-model $(\mathcal{M}, s)$ follows easily:

Definition 7 (Truth)**.** *Given a neighbourhood model $\mathcal{M} = \langle S, N, \pi\rangle$ with $s \in S$. Truth of a formula $\phi \in \mathcal{L}$ at s is defined recursively as follows:*

$$\begin{gathered}(\mathcal{M}, s) \models p \text{ iff } s \in \pi(p);\\ (\mathcal{M}, s) \models \neg\phi \text{ iff } s \in \pi^{ext}(\neg\phi);\\ (\mathcal{M}, s) \models \phi \wedge \psi \text{ iff } s \in \pi^{ext}(\phi \wedge \psi);\\ (\mathcal{M}, s) \models \Box(\phi) \text{ iff } s \in \pi^{ext}(\Box\phi));\\ (\mathcal{M}, s) \models \Diamond(\phi) \text{ iff } s \in \pi^{ext}(\Diamond\phi))\end{gathered}$$

A formula ϕ is **satisfiable** if there is some model $\mathcal{M} = \langle S, N, \pi\rangle$ and state $s \in S$ such that $(\mathcal{M}, s) \models \phi$. Similarly, a set of formulas Γ is satisfiable if $\forall\phi \in \Gamma, (\mathcal{M}, s) \models \phi$. From the notion of satisfiability, it follows the definition of validity:

Definition 8 (Validity)**.** *Given a neighbourhood frame $\mathcal{F} = \langle S, N\rangle$ and a neighbourhood model $\mathcal{M} = \langle S', N', \pi'\rangle$, then: a formula $\phi \in \mathcal{L}$ is **valid on** $\mathcal{M}$, denoted $\mathcal{M} \models \phi$, when $\forall s' \in S', (\mathcal{M}, s') \models \phi$; a formula ϕ is **valid at** s **in** $\mathcal{F}$, denoted $(\mathcal{F}, s) \models \phi$, provided that $\forall\mathcal{M}$ based on $\mathcal{F}, (\mathcal{M}, s) \models \phi$; a formula ϕ is **valid on** $\mathcal{F}$, denoted $\mathcal{F} \models \phi$, provided that $\forall s \in S, (\mathcal{F}, s) \models \phi$. Suppose G is a class of frames. A formula ϕ is **valid on** G, denoted $\models_G \phi$, provided that $\forall\mathcal{F} \in G, \mathcal{F} \models \phi$.*

The validity definition exhausts the semantical structure.

4.1.3 Modal Logic: Invariance

Invariance results for the basic modal language will now be presented. Such results are required to obtain logical equivalences of structures based on affinities of the structures themselves. In particular, for modal logic, the best known concept of equivalence between structures is bisimulation. In this subsection, restrictions will be made on neighbourhood models. The main reason is that the language presented later will be specified in such restricted structures and therefore it proves useful to discuss the results obtainable for such restricted structures, rather than the general ones: thus, results about augmented neighbourhood structures will become the focus now.

The definitions are therefore refined as required by the new class of models that is analysed. The starting point is the definition of the non-monotonic core of a collection of sets:

Definition 9 (Non-Monotonic Core)**.** *Suppose that $\mathcal{U}$ is a monotonic collection of subsets of S. The **non-monotonic core**, denoted $\mathcal{U}^{nmc}$ is a subset of $\mathcal{U}$ defined as: $\mathcal{U}^{nmc} = \{X \mid X \in \mathcal{U} \text{ and } \forall X' \subseteq S, if\ X' \subset X, then\ X' \notin \mathcal{U}\}$.*

The non-monotonic core of a collection of sets is the set of minimal elements of the collection under the subset relation. Interestingly, when the set of states S from which $\mathcal{U}$ is formed is finite (as it will be the case for the logical trust language introduced later), then the non-monotonic core of $\mathcal{U}$ is guaranteed to be non-empty. Moreover, this non-monotonic core completely determines the elements of $\mathcal{U}$.

From definition 9, it follows that of core-completeness:

Definition 10 (Core Completeness)**.** *A monotonic collection of sets $\mathcal{U}$ is **core complete** provided that $\forall X \in \mathcal{U}, \exists Y \in \mathcal{U}^{nmc}$ s.t. $Y \subseteq X$.*

Starting from those definitions (i.e., definitions 9 and 10), five invariance results can be proven for augmented neighbourhood models. Recall that a neighbourhood model is said to be augmented if $\forall s \in S$, $N(s)$ is augmented (i.e., it is monotonic and contains its core). The invariance results which will be proven are:

1. Invariance under **disjoint unions**.
2. Invariance under **bounded core morphism**.
3. Invariance under **core bisimulation**.
4. Invariance under **generated submodels**.
5. Invariance under **unravelling**.

All the results are taken from [48]. In particular, all results which concern the core of a model are exclusively due to [48], while most other results can be found in various introductions to modal logic. In particular, historically, it seems like those results are due to [21], where they are presented without proofs (which are left as exercises for the reader). Those invariance results are important for one main reason: it allows the modeler t check the expressivity of a language. If two models satisfy the same class of formulas, then it can rightly be assumed that they can say the same things. Once this result is extended to all the structures which resemble the canonical model for the semantics (intuitively: the model capturing all the validities of the semantical structure and leaving out all the non-theorems), then an expressivity measure for all the language is identified. Obviously, the more transformations between models are possible, the better is the control over the expressivity of the language. Finally, unravelling will prove to be useful in proving the tree model property, which, in turn, will be useful in proving that the satisfiability problem for the language is decidable.

Before proceeding with the definitions and proofs of the invariance results, it will be given a clear definition of modal equivalence and it will be shown that indeed augmented neighbourhood models are core-complete, thus justifying the choice of taking invariance results tied to core-complete models, rather than the general ones.

The definition of modal equivalence is the following.

Definition 11 (Modal equivalence)**.** *Given a pointed neighbourhood model* $(\mathcal{M}, s)$*, indicate with* $Th_{\mathcal{L}}(\mathcal{M}, s)$ *the* ***theory*** *of* $(\mathcal{M}, s)$*, i.e. the set of modal formulas true at* s *in* $\mathcal{M}$*:* $Th_{\mathcal{L}}(\mathcal{M}, s) = \{\phi \in \mathcal{L} \mid (\mathcal{M}, s) \models \phi\}$*.*

Two pointed models $(\mathcal{M}, s)$ *and* $(\mathcal{M}', s')$ *are said to be* ***modally-equivalent****, denoted* $(\mathcal{M}, s) \equiv_{\mathcal{L}} (\mathcal{M}', s')$*, when* $Th_{\mathcal{L}}(\mathcal{M}, s) = Th_{\mathcal{L}}(\mathcal{M}', s')$*.*

For the core-completeness of augmented neighbourhood models.

Lemma 1. *Augmented neighbourhood models are core-complete models.*

Proof. *Assume* $\mathcal{M} = \langle S, N, \pi\rangle$ *is an augmented neighbourhood model. Then, by definition* $\forall s \in S, \bigcap N(s) \in N(s)$*. Take an arbitrary* s*. Now take an arbitrary* $X \in N(s)$*, show that* $\exists Y \in N^{nmc}(s)$ *s.t.* $Y \subseteq X$*. First note that* $\bigcap N(s) \in N^{nmc}(s)$*. Now take* $Y = \bigcap N(s)$*. Given the definition of* $\bigcap N(s)$ *and the assumption that* $X \in N(s)$*, it follows directly that* $\bigcap N(s) \subseteq X$*. Therefore,* $Y \in N^{nmc}(s)$ *and* $Y \subseteq X$*. Concluding,* $\mathcal{M} = \langle S, N, \pi\rangle$ *is also core-complete.* □

It is now possible to proceed with the invariance results. The first truth-invariant transformation procedure is that of disjoint union.

Definition 12 (Disjoint Unions)**.** *Given a collection of disjoint models* $\{\mathcal{M}_i = (S_i, N_i, \pi_i)\}_{i\in I}$*, their disjoint union, labelled* $\biguplus(\mathcal{M}_i)_{i\in I} = (S, N, \pi)$*, where* $S = \bigcup_{i\in I}(S_i)$*,* $\pi(p) = \bigcup_{i\in I}(\pi_i(p))$*, and for* $X \subseteq S$ *and* $s \in S_i$*:*

$$X \in N(s) \text{ iff } X \cap S_i \in N_i(s)$$

Theorem 1 (Invariance Under Disjoint Unions)**.** *Given a collection of disjoint models* $\{\mathcal{M}_i = (S_i, N_i, \pi_i)\}_{i\in I}$ *and their disjoint union* $\biguplus(\mathcal{M}_i)_{i\in I} = (S, N, \pi)$*, then for each formula* $\phi \in \mathcal{L}$*, for each* $i \in I$ *and for each* $s \in S_i$*, it holds that:*

$$(\mathcal{M}_i, s) \equiv_{\mathcal{L}} (\biguplus(\mathcal{M}_i), s).$$

Proof (Proof of Theorem 1)**.** *Start by rephrasing the thesis as follows:* $\forall \phi \in \mathcal{L}, \pi^{ext}(\phi) = \bigcup_{i \in I}(\pi_i^{ext}(\phi))$*. The proof is given by induction on the structure of* ϕ*. Given an arbitrary* $i \in I$ *and* $s \in S_i$*.*

Base Case.

ϕ *is an unanalysed propositions* $p \in At$*. This follows directly from the definition of disjoint union, i.e.* $s \in \pi(p)$ *iff* $s \in \bigcup_{i \in I}(\pi_i(p))$*.*

Inductive Hypothesis (IH).

$\pi^{ext}(\phi) = \bigcup_{i \in I}(\pi_i^{ext}(\phi))$*.*

Inductive Steps.

Negation: Show that $s \in \pi^{ext}(\neg\phi)$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\neg\phi))$*.*

- $s \in \pi^{ext}(\neg\phi)$ *iff* $s \in (S - \pi^{ext}(\phi))$*. [Definition of* $\pi^{ext}(\neg\phi)$*]*
- $s \in (S - \pi^{ext}(\phi))$ *iff* $s \in (S - \bigcup_{i \in I}(\pi_i^{ext}(\phi))$*. [IH]*
- $s \in (S - \bigcup_{i \in I}(\pi_i^{ext}(\phi))$ *iff* $s \in S$ *and* $s \notin \bigcup_{i \in I}(\pi_i^{ext}(\phi))$*. [Definition of set difference]*
- $s \in S$ *iff* $s \in \bigcup_{i \in I}(S_i)$*. [Defition of disjoint union for S]*
- $s \in \bigcup_{i \in I}(S_i)$ *and* $s \notin \bigcup_{i \in I}(\pi_i^{ext}(\phi))$ *iff* $s \in \bigcup_{i \in I}(S_i - \pi_i^{ext}(\phi))$*. [Definition of set difference]*
- $s \in \bigcup_{i \in I}(S_i - \pi_i^{ext}(\phi))$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\neg\phi))$*. [Definition of* $\pi^{ext}(\neg\phi)$*]*

Conjunction: Show that $s \in \pi^{ext}(\phi \wedge \psi)$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi \wedge \psi)$*.*

- $s \in \pi^{ext}(\phi \wedge \psi)$ *iff* $s \in \pi^{ext}(\phi) \cap \pi^{ext}(\psi)$*. [Definition of* $\pi^{ext}(\phi \wedge \psi)$*]*
- $s \in \pi^{ext}(\phi) \cap \pi^{ext}(\psi)$ *iff* $s \in \pi^{ext}(\phi)$ *and* $s \in \pi^{ext}(\psi)$*. [Definition of intersection]*
- $s \in \pi^{ext}(\phi)$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi))$*. [IH]*
- $s \in \pi^{ext}(\psi)$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\psi))$*. [IH]*
- $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi))$ *and* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\psi))$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi) \cap \pi_i^{ext}(\psi)$*. [Definition of intersection]*
- $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi) \cap \pi_i^{ext}(\psi)$ *iff* $s \in \bigcup_{i \in I}(\pi_i^{ext}(\phi \wedge \psi)$*. [Definitio of* $\pi^{ext}(\phi \wedge \psi)$*]*

Box modality: Show that $\pi_i^{ext}(\phi) \in N_i(s)$ *iff* $\pi^{ext}(\phi) \in N(s)$*.*

Left-to-right. Assume that $\pi_i^{ext}(\phi) \in N_i(s)$*. From IH, it is possible to derive that* $\pi_i^{ext}(\phi) \subseteq \pi^{ext}(\phi)$*. Therefore,* $\pi^{ext}(\phi) \cap S_i \in N_i(s)$ *(note that* $\pi^{ext}(\phi) \cap S_i = \pi_i^{ext}(\phi)$*). By definition of disjoint unions for* N*,* $\pi^{ext}(\phi) \in N(s)$*.*

Right-to-left. Assume $\pi^{ext}(\phi) \in N(s)$. By definition of disjoint unions for N, $\pi^{ext}(\phi) \cap S_i \in N_i(s)$. By IH, $\pi^{ext}(\phi) \cap S_i = \pi_i^{ext}(\phi)$. Thus, $\pi_i^{ext}(\phi) \in N_i(s)$.

Diamond modality: The proof is obtained by combining the proofs for negation and the box modality. □

The second truth-invariant transformation procedure is that of bounded core morphism.

Definition 13 (Bounded Core Morphism)**.** *Given two core-complete, monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. A function $f : S_1 \to S_2$ is a* ***bounded core morphism*** *from $\mathcal{M}_1$ to $\mathcal{M}_2$ if*

1. *s and $f(s)$ satisfy the same unanalysed propositions $p \in At$.*
2. *If $X \in N_1^{nmc}(s)$, then $f[X] \in N_2^{nmc}(f(s))$.*
3. *If $Y \in N_2^{nmc}(f(s))$, then $\exists X \subseteq S_1$ s.t. $f[X] = Y$ and $X \in N_1^{nmc}(s)$.*

Before introducing the concept of invariance under bounded core morphisms, it is useful to introduce the notion of bounded morphism and then prove the invariance result for such notion. The invariance under bounded core morphism will then follow easily.

Definition 14 (Bounded Morphism)**.** *Given two monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. A function $f : S_1 \to S_2$ is a* ***bounded morphism*** *from $\mathcal{M}_1$ to $\mathcal{M}_2$ if*

1. *s and $f(s)$ satisfy the same unanalysed propositions $p \in At$.*
2. *If $X \in N_1(s)$, then $f[X] \in N_2(f(s))$.*
3. *If $Y \in N_2(f(s))$, then $\exists X \subseteq S_1$ s.t. $f[X] \subseteq Y$ and $X \in N_1(s)$.*

Lemma 2. *Conditions 2. and 3. of definition 14 taken together are equivalent to the following condition:*

$$f^{-1}[Y] \in N_1(s) \textit{ iff } Y \in N_2(f(s))$$

Proof (Proof of Lemma 2)**.** *Assume conditions 2. and 3. of definition 14 hold. Show that $f^{-1}[Y] \in N_1(s)$ iff $Y \in N_2(f(s))$.*

Assume that $f^{-1}[Y] \in N_1(s)$. Given condition 2. of definition 14, $f[f^{-1}[Y]] \in N_2(f(s))$. By the fact that $Y \supseteq f[f^{-1}[Y]]$ and by monotonicity of N_2, $Y \in N_2(f(s))$.

Assume that $Y \in N_2(f(s))$. Given condition 3. of definition 14, $\exists X \subseteq S_1$ s.t. $f[X] \subseteq Y$ and $X \in N_1(s)$. From $f[X] \subseteq Y$ it follows that $X \subseteq f^{-1}[f[X]] \subseteq f^{-1}[Y]$. By monotonicity of N_1, $f^{-1}[Y] \in N_1(s)$.

Now, assume $f^{-1}[Y] \in N_1(s)$ iff $Y \in N_2(f(s))$ (call it α). Show that conditions 2. and 3. of definition 14 hold.

For condition 2., assume $X \in N_1(s)$. By monotonicity of N_1, $f^{-1}[f[X]] \in N_1(s)$ and, by α, $f[X] \in N_2(f(s))$.

For condition 3., assume $Y \in N_2(f(s))$. By α, $f^{-1}[Y] \in N_1(s)$. Given $f[f^{-1}[Y]] \subseteq Y$, the required X can be taken to be $f^{-1}[Y]$. □

The theorem of invariance under bounded morphism follows:

Theorem 2 (Invariance Under Bounded Morphism)**.** *Given two monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. If $f : S_1 \to S_2$ is a bounded morphism from $\mathcal{M}_1$ to $\mathcal{M}_2$, then $\forall \phi \in \mathcal{L}$ and each $s \in S_1$:*

$$(\mathcal{M}_1, s) \equiv_{\mathcal{L}} (\mathcal{M}_2, f(s))$$

Proof (Proof of Theorem 2)**.** *Start by rephrasing the thesis as follows: $f^{-1}[\pi_2^{ext}(\phi)] = \pi_1^{ext}(\phi)$, where $f^{-1}(X) = \{s \in S_1 \mid f(s) \in X\}$. The proof is given by induction on the structure of ϕ.*

Base Case.

ϕ is an unanalysed propositions $p \in At$. This follows directly from condition 1. of definition 14, i.e. $s \in f^{-1}[\pi_2^{ext}(p)]$ iff $s \in \pi_1^{ext}(p)$

Inductive Hypothesis (IH).

$f^{-1}[\pi_2^{ext}(\phi)] = \pi_1^{ext}(\phi)$

Inductive Steps.

Negation: Show that $s \in f^{-1}[\pi_2^{ext}(\neg\phi)]$ iff $s \in \pi_1^{ext}(\neg\phi)$.

- *$s \in f^{-1}[\pi_2^{ext}(\neg\phi)]$ iff $f(s) \in \pi_2^{ext}(\neg\phi)$. [Definition of f^{-1}]*
- *$f(s) \in \pi_2^{ext}(\neg\phi)$ iff $f(s) \in (S_2 - \pi_2^{ext}(\phi))$. [Definition of π^{ext}]*
- *$f(s) \in (S_2 - \pi_2^{ext}(\phi))$ iff $f(s) \in S_2$ and $f(s) \notin \pi_2^{ext}(\phi)$. [Definition of set difference]*
- *$f(s) \in S_2$. [Hypothesis]*
- *$f(s) \notin \pi_2^{ext}(\phi)$ iff $s \notin \pi_1^{ext}(\phi)$. [IH]*
- *$s \in S_1$ and $s \notin \pi_1^{ext}(\phi)$ iff $s \in (S_1 - \pi_1^{ext}(\phi))$. [Definition of set difference]*
- *$s \in (S_1 - \pi_1^{ext}(\phi))$ iff $s \in \pi_1^{ext}(\neg\phi)$. [Definition of π^{ext}]*

Conjunction: Show that $s \in f^{-1}[\pi_2^{ext}(\phi \wedge \psi)]$ iff $s \in \pi_1^{ext}(\phi \wedge \psi)$.

- *$s \in f^{-1}[\pi_2^{ext}(\phi \wedge \psi)]$ iff $f(s) \in \pi_2^{ext}(\phi \wedge \psi)$. [Definition of f^{-1}]*
- *$f(s) \in \pi_2^{ext}(\phi \wedge \psi)$ iff $f(s) \in \pi_2^{ext}(\phi)$ and $f(s) \in \pi_2^{ext}(\psi)$. [Definition of π^{ext}]*

- *$f(s) \in \pi_2^{ext}(\phi)$ iff $s \in \pi_1^{ext}(\phi)$. [IH]*
- *$f(s) \in \pi_2^{ext}(\psi)$ iff $s \in \pi_1^{ext}(\psi)$. [IH]*
- *$s \in \pi_1^{ext}(\phi)$ and $s \in \pi_1^{ext}(\psi)$ iff $s \in \pi_1^{ext}(\phi \wedge \psi)$. [Definition of π^{ext}]*

Box modality: Show that $s \in f^{-1}[\pi_2^{ext}(\Box(\phi))]$ iff $s \in \pi_1^{ext}(\Box(\phi))$

- *$s \in f^{-1}[\pi_2^{ext}(\Box(\phi))]$ iff $f(s) \in \pi_2^{ext}(\Box(\phi))$. [Definition of f^{-1}]*
- *$f(s) \in \pi_2^{ext}(\Box(\phi))$ iff $f(s) \in m_N(\pi_2^{ext}(\phi))$. [Definition of π^{ext}]*
- *$f(s) \in m_N(\pi_2^{ext}(\phi))$ iff $\pi_2^{ext}(\phi) \in N_2(f(s))$. [Definition of m_N]*
- *$\pi_2^{ext}(\phi) \in N_2(f(s))$ iff $f^{-1}[\pi_2^{ext}(\phi)] \in N_1(s)$. [Lemma 2]*
- *$f^{-1}[\pi_2^{ext}(\phi)] \in N_1(s)$ iff $\pi_1^{ext}(\phi) \in N_1(s)$. [IH]*
- *$\pi_1^{ext}(\phi) \in N_1(s)$ iff $s \in m_N(\pi_1^{ext}(\phi)])$. [Definition of m_N]*
- *$s \in m_N(\pi_1^{ext}(\phi)])$ iff $s \in \pi_1^{ext}(\Box(\phi))$. [Definition of π^{ext}]*

Diamond modality: The proof is obtained by combining the proofs for negation and the box modality. □

It is possible now to prove that augmented neighbourhood models are truth invariant under bounded core morphisms.

Theorem 3. *Every bounded core morphism is a bounded morphism.*

Proof (Proof of Theorem 3)**.** *First note, that to have a bounded core morphism, the models have to be core-complete.*

Show that if f is a bounded core morphism, then it is also a bounded morphism. Assume that f is a bounded core morphism. Show that conditions 1-3 of definition 14 hold.

For condition 1., show that: s and $f(s)$ satisfy the same unanalysed propositions $p \in At$. This follows directly from condition 1. of definition 13.

For condition 2., show that: If $X \in N_1(s)$, then $f[X] \in N_2(f(s))$. Assume $X \in N_1(s)$. By core-completeness (see definition 10), $\exists Y \in N_1^{nmc}(s)$ s.t. $Y \subseteq X$. By condition 2. of definition 13, if $Y \in N_1^{nmc}(s)$, then $f[Y] \in N_2^{nmc}(f(s))$. By the definition of non-monotonic core (see definition 9), if $f[Y] \in N_2^{nmc}(f(s))$, then $f[Y] \in N_2(f(s))$. Given $f[Y] \subseteq f[X]$ and monotonicity of N_2, $f[X] \in N_2(f(s))$.

For condition 3., show that: If $Y \in N_2(f(s))$, then $\exists X \subseteq S_1$ s.t. $f[X] \subseteq Y$ and $X \in N_1(s)$. Assume $Y \in N_2(f(s))$. By core-completeness, $\exists Z \in N_2^{nmc}(f(s))$ s.t. $Z \subseteq Y$. By condition 3. of definition 13, if $Z \in N_2^{nmc}(f(s))$, then $\exists W \subseteq S_1$ s.t. $f[W] = Z$ and $W \in N_1^{nmc}(s)$. By the definition of non-monotonic core, if $W \in N_1^{nmc}(s)$, then $W \in N_1(s)$. Given the fact that $f[W] = Z \subseteq Y$, W is the set we are looking for. □

By theorem 3, it follows that for core-complete neighbourhood models, bounded-core morphisms preserve truth of formulas.

The third truth-invariant transformation procedure is that of core bisimulation.

Definition 15 (Core Bisimulation)**.** *Given two augmented neighbourhood models $\mathcal{M}_1 = \langle S_1, N_1, \pi_1\rangle$ and $\mathcal{M}_2 = \langle S_2, N_2, \pi_2\rangle$, a non-empty binary relation $E \subseteq S_1 \times S_2$ is a* ***core bisimulation between*** *$\mathcal{M}_1$ and $\mathcal{M}_2$ (denoted by $E : \mathcal{M}_1 \underline{\leftrightarrow}_c \mathcal{M}_2$), if*

- *(Local Harmony) If s_1Es_2, then s_1 and s_2 satisfy the same unanalysed propositions, i.e. $\forall p \in At, s_1 \in \pi_1(p)\ iff\ s_2 \in \pi_2(p)$.*
- *$(zig)_c$ If s_1Es_2 and $X \in N_1^{nmc}(s_1)$, then $\exists X_2 \subseteq S_2$ s.t. $X_2 \in N_2^{nmc}(s_2)$, and $\forall t_2 \in X_2$, $\exists t_1 \in X_1$ s.t. t_1Et_2.*
- *$(zag)_c$ If s_1Es_2 and $X_2 \in N_2^{nmc}(s_2)$, then $\exists X_1 \subseteq S_1$ s.t. $X_1 \in N_1^{nmc}(s_1)$, and $\forall t_1 \in X_1$, $\exists t_2 \in X_2$ s.t. t_1Et_2.*

As with bounded-core morphisms, it is useful to first introduce the notion of standard bisimulation for monotonic neighbourhood models, prove the invariance result for such transformation and then show that core bisimulations over core-complete neighbourhood models are also bisimulations over those models.

Definition 16 (Bisimulation)**.** *Given two monotonic neighbourhood models $\mathcal{M}_1 = \langle S_1, N_1, \pi_1\rangle$ and $\mathcal{M}_2 = \langle S_2, N_2, \pi_2\rangle$, a non-empty binary relation $E \subseteq S_1 \times S_2$ is a* ***bisimulation between*** *$\mathcal{M}_1$ and $\mathcal{M}_2$ (denoted by $E : \mathcal{M}_1 \underline{\leftrightarrow} \mathcal{M}_2$), if*

- *(Local Harmony) If s_1Es_2, then s_1 and s_2 satisfy the same unanalysed propositions, i.e. $\forall p \in At, s_1 \in \pi_1(p)\ iff\ s_2 \in \pi_2(p)$.*
- *(zig) If s_1Es_2 and $X_1 \in N_1(s_1)$, then $\exists X_2 \subseteq S_2$ s.t. $X_2 \in N_2(s_2)$, and $\forall t_2 \in X_2$, $\exists t_1 \in X_1$ s.t. t_1Et_2.*
- *(zag) If s_1Es_2 and $X_2 \in N_2(s_2)$, then $\exists X_1 \subseteq S_1$ s.t. $X_1 \in N_1(s_1)$, and $\forall t_1 \in X_1$, $\exists t_2 \in X_2$ s.t. t_1Et_2.*

The theorem of invariance under bisimulation follows:

Theorem 4 (Invariance Under Bisimulation)**.** *Given two monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. If $E \subseteq S_1 \times S_2$ is a bisimulation between $\mathcal{M}_1$ and $\mathcal{M}_2$, then $\forall \phi \in \mathcal{L}$, $s_1 \in S_1$ and $s_2 \in S_2$ s.t. s_1Es_2, it holds that:*

$$(\mathcal{M}_1, s_1) \equiv_{\mathcal{L}} (\mathcal{M}_2, s_2)$$

Proof (Proof of Theorem 4)**.** *Rephrase the thesis as follows: $s_1 \in \pi_1^{ext}(\phi)$ iff $s_2 \in \pi_2^{ext}(\phi)$. The proof is given by induction on the structure of ϕ. Assume that $s_1 E s_2$.*

Base Case*.*

ϕ is an unanalysed propositions $p \in At$. This follows directly from local harmony.

Inductive Hypothesis (IH)*.*

$s_1 \in \pi_1^{ext}(\phi)$ iff $s_2 \in \pi_2^{ext}(\phi)$.

Inductive Steps*.*

Negation: Show that $s_1 \in \pi_1^{ext}(\neg\phi)$ iff $s_2 \in \pi_2^{ext}(\neg\phi)$.

- *$s_1 \in \pi_1^{ext}(\neg\phi)$ iff $s_1 \in (S_1 - \pi_1^{ext}(\phi)$. [Definition of π^{ext}]*
- *$s_1 \in (S_1 - \pi_1^{ext}(\phi)$ iff $s_1 \in S_1$ and $s_1 \notin \pi_1^{ext}(\phi)$. [Definition of set difference]*
- *$s_2 \in S_2$. [Hypothesis]*
- *$s_1 \notin \pi_1^{ext}(\phi)$ iff $s_2 \notin \pi_2^{ext}(\phi)$. [IH]*
- *$s_2 \in S_2$ and $s_2 \notin \pi_2^{ext}(\phi)$ iff $s_2 \in (S_2 - \pi_2^{ext}(\phi)$. [Definition of set difference]*
- *$s_2 \in (S_2 - \pi_2^{ext}(\phi)$ iff $s_2 \in \pi_2^{ext}(\neg\phi)$. [Definition of π^{ext}]*

Conjunction: Show that $s_1 \in \pi_1^{ext}(\phi \wedge \psi)$ iff $s_2 \in \pi_2^{ext}(\phi \wedge \psi)$.

- *$s_1 \in \pi_1^{ext}(\phi \wedge \psi)$ iff $s_1 \in \pi_1^{ext}(\phi)$ and $s_1 \in \pi_1^{ext}(\psi)$. [Definition of π^{ext}]*
- *$s_1 \in \pi_1^{ext}(\phi)$ iff $s_2 \in \pi_2^{ext}(\phi)$. [IH]*
- *$s_1 \in \pi_1^{ext}(\psi)$ iff $s_2 \in \pi_2^{ext}(\psi)$. [IH]*
- *$s_2 \in \pi_2^{ext}(\phi)$ and $s_2 \in \pi_2^{ext}(\psi)$ iff $s_2 \in \pi_2^{ext}(\phi \wedge \psi)$. [Definition of π^{ext}]*

Box modality: To prove the theorem for the box modality, go back to the original statement. Therefore, show that:

$$(\mathcal{M}_1, s_1) \models \Box(\phi) \text{ iff } (\mathcal{M}_2, s_2) \models \Box(\phi)$$

Left-to-right:

Assume $(\mathcal{M}_1, s_1) \models \Box(\phi)$. By the definition of π^{ext} and that of m_N, it follows that $\pi_1^{ext}(\phi) \in N_1(s_1)$. By the (zig) condition, there is a $Y \subseteq S_2$ s.t. $Y \in N_2(s_2)$ and $\forall t_2 \in Y$, $\exists t \in \pi_1^{ext}(\phi)$ s.t. $t_1 E t_2$. By (IH), it follows that $Y \subseteq \pi_2^{ext}(\phi)$. By monotonicity, $\pi_2^{ext}(\phi) \in N_2(s_2)$. By the definition of π^{ext} and that of m_N, it follows that $(\mathcal{M}_2, s_2) \models \Box(\phi)$.

Right-to-left:

Symmetrical to the left-to-right direction, employing the (zag) condition instead of the (zig).

Diamond modality: The proof is obtained by combining the proofs for negation and the box modality. □

By proving that a core bisimulation is also a bisimulation, the truth-invariance theorem will apply also to core bisimulations[3]

Theorem 5. *Given two core-complete, monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. If $E \subseteq S_1 \times S_2$ is a core bisimulation between $\mathcal{M}_1$ and $\mathcal{M}_2$, then it is also a bisimulation between them.*

Proof (Proof of Theorem 5)**.** *Assume that E is a core bisimulation between $\mathcal{M}_1$ and $\mathcal{M}_2$. Show that E satisfies all conditions of bisimulations.*

The local harmony condition is straightforward, since the two conditions state the same fact.

For the (zig) condition. Assume $s_1 E s_2$ and $X \in N_1(s_1)$. By core-completeness, $\exists Y_1 \in N_1^{nmc}(s_1)$ s.t. $Y_1 \subseteq X$. By the $(zig)_c$ condition of core bisimulation, $\exists Y_2 \subseteq S_2$ s.t. $Y_2 \in N_2^{nmc}(s_2)$ and $\forall t_2 \in Y_2$, $\exists t_1 \in Y_1$ s.t. $t_1 E t_2$. Since $Y_1 \subseteq X$, $\forall t_2 \in Y_2$, $\exists t_1 \in X$ s.t. $t_1 E t_2$. By the definition of non-monotonic core, it also follows that $Y_2 \in N_2(s_2)$. Therefore, the (zig) condition is satisfied.

For the (zag) condition. Analogous to the (zig) condition, employing the $(zag)_c$ condition instead of the $(zig)_c$.

The fourth truth-invariant transformation procedure is that of generated submodels.

The starting point of the procedure of generated submodels, is the definition of submodel:

Definition 17 (Submodel)**.** *Given a monotonic neighbourhood model $\mathcal{M} = (S, N, \pi)$. Then $\mathcal{M}' = (S', N', \pi')$ is a **submodel** of $\mathcal{M}$ if*

- $S' \subseteq S$.
- $\forall p \in At$, $\pi'(p) = \pi \cap S'$.
- $N' = N \cap (S' \times \wp(S'))$, *i.e.* $\forall s \in S' : N'(s) = \{X \subseteq S' \mid X \in N(s)\}$.

Given a monotonic model $\mathcal{M} = (S, N, \pi)$ and any subset S' of S, it is possible to construct the related submodel, denoted by $\mathcal{M} \restriction_{S'} = (S', N \restriction_{S'}, \pi \restriction_{S'})$.

Starting from the notion of submodel, it is possible to define a notion of generated submodel (i.e., a notion of submodel that guarantees truth-invariance).

[3] It is indeed possible to prove that the two concepts are equivalent, where every bisimulation over core-complete, monotonic models, is also a core-bisimulation.

Definition 18 (Generated Submodel)**.** *Given a monotonic neighbourhood model $\mathcal{M} = (S, N, \pi)$ and a submodel of $\mathcal{M}$, $\mathcal{M}' = (S', N', \pi')$. $\mathcal{M}'$ is a **generated submodel** of $\mathcal{M}$, if the identity map $i : S' \to S$ is a bounded morphism, i.e., $\forall s' \in S'$ and $\forall X \subseteq S$,*

$$(i^{-1}[X] = X \cap S') \in N'(s') \textit{ iff } X \in N(s').$$

With the definition of generated submodel for monotonic neighbourhood models, the truth invariance result for generated submodels follows directly from the proof of truth-invariance for bounded morphism.

Given a monotonic model $\mathcal{M} = (S, N, \pi)$ and a subset X of S, the **submodel generated by X in $\mathcal{M}$** is defined as the submodel $\mathcal{M} \restriction_{S'}$, where $S' = \bigcap_Y X \subseteq Y$ with $\mathcal{M} \restriction_Y$ being a generated submodel of $\mathcal{M}$.

Moreover, for a core-complete, monotonic model $\mathcal{M} = (S, N, \pi)$ and $X \subseteq S$, the submodel generated by X in $\mathcal{M}$ can also be obtained by restricting $\mathcal{M}$ to the subset $W_\omega(X)$, whose definition is:

Definition 19 (Rooted Model)**.** *Given a core-complete, monotonic model $\mathcal{M} = (S, N, \pi)$. For $X \subseteq S$, $N^{nmc}_\omega(X)$ and $W_\omega(X)$ are defined recursively as follows:*

For $N^{nmc}_\omega(X)$:

- $N^{nmc}_0(X) = \bigcup x \in XN^{nmc}(x)$
- $N^{nmc}_{n+1}(X) = \bigcup x \in W_{n+1}(X)N^{nmc}(x)$
- $N^{nmc}_\omega(X) = \bigcup x \in W_\omega(X)N^{nmc}(x)$

For $W_\omega(X)$:

- $W_0(X) = X$
- $W_{n+1}(X) = \bigcup_{Y \in N^{nmc}_n(X)} Y$
- $W_\omega(X) = \bigcup_{n \in \omega} W_n(X)$

*If $W_\omega(\{s\}) = S$, then $\mathcal{M}$ is called a **rooted** model with root s.*

Those definitions will prove to be useful in the definition of the next truth-invariant transformation procedure, which is unravelling.

Definition 20 (Tree-like Models)**.** *Given a core-complete, monotonic model $\mathcal{M} = (S, N, \pi)$ and a **root** element of S, i.e., $root \in S$, a model $\mathcal{M}_{root}$ is said to be a **tree-like monotonic model** if the following hold:*

- $S = W_\omega(\{root\})$.

- $\forall s \in S$, $s \notin \bigcup_{n>0} W_n(\{s\})$.
- $\forall s_1, s_2, t \in S$ *and* $\forall X_1, X_2 \subseteq S$, *if* $v \in X_1 \in N^{nmc}(s_1)$ *and* $v \in X_2 \in N^{nmc}(s_2)$, *then* $X_1 = X_2$ *and* $s_1 = s_2$.

Intuitively, this definition says that a core-complete, monotonic neighbourhood model is tree-like, if all states of the model are *reachable* from the state, indicated as the root of the model, through a sequence of core neighbourhoods. Moreover, all such neighbourhoods must be disjoint and unique. Finally, the path traced must not contain cycles, i.e., no neighbourhoods might contain the root of the model. Such models are interesting because it can be shown that if a formula is satisfiable in a core-complete neighbourhood model, then it is satisfiable in a tree-like model, where the formula is true at the root of the model. The important step, therefore, is that of obtaining a truth invariant transformation of core-complete, neighbourhood models which turns them into tree-like models. Unravelling is such transformation.

Before introducing the unravelling transformation, some formal elements must be defined.

Definition 21. *Given a monotonic model* $\mathcal{M} = (S, N, \pi)$ *and a state* $s_1 \in S$, *the set* $\vec{S_{s_1}}$ *is defined as follows:* $\vec{S_{s_1}} = \{(s_1X_2s_2X_3s_3 \dots X_ns_n) \mid n \geq 0$ *and* $\forall i \in \{1, \dots, n\}, X_i \in N(s_{i-1})$ *and* $s_i \in X_i\}$

From each $(s_1X_2s_2X_3s_3 \dots X_ns_n) \in \vec{S_{s_1}}$, it is possible to define two maps:

1. *Pre*: $(s_1X_2s_2X_3s_3 \dots X_ns_n) \mapsto (s_1X_2s_2X_3s_3 \dots X_{n-1}s_{n-1}X_n)$
2. *Last*: $(s_1X_2s_2X_3s_3 \dots X_ns_n) \mapsto s_n$

Definition 22. *Given a monotonic model* $\mathcal{M} = (S, N, \pi)$, $\bar{x} \in \vec{S_{s_1}}$ *and* $Y \subseteq \vec{S_{s_1}}$, *define a neighbourhood function* $\xi : \vec{S_{s_1}} \to \wp(\wp(\vec{S_{s_1}}))$ *as follows:* $Y \in \xi(\bar{x})$ *iff* $\forall \bar{y} \in Y(Pre(\bar{y}) = \bar{x}X)$ *and* $Last[Y] = X \in N(Last(\bar{x})$, *for some* $X \in \wp(S)$.

Definition 23. *Given a monotonic model* $\mathcal{M} = (S, N, \pi)$, *the neighbourhood function* $\vec{N_{s_1}}$ *is defined taking* ξ *and closing it under superset, i.e.,* $Y \in \vec{N_{s_1}}(\bar{x})$ *iff* $\exists Y' \in \xi(\bar{x})$ *s.t.* $Y' \subseteq Y$.

Finally, a definition for the valuation functions:

Definition 24. *Given a monotonic model* $\mathcal{M} = (S, N, \pi)$, *the valuation function* $\vec{\pi_{s_1}}$ *is defined as follow:* $\bar{x} \in \vec{\pi_{s_1}}(p)$ *iff* $Last(\bar{x}) \in \pi(p)$.

From those definitions it is possible to define the unravelling of a model.

Definition 25 (Unravelling of Models)**.** *Given a monotonic model* $\mathcal{M} = (S, N, \pi)$ *and a state* $s_1 \in S$, *the* ***unravelling of*** $\mathcal{M}$ ***from*** s_1 *is defined as the model* $\vec{\mathcal{M}_{s_1}} = (\vec{S_{s_1}}, \vec{N_{s_1}}, \vec{\pi_{s_1}})$

Given the fact that the map *Last* : $\vec{\mathcal{M}}_{s_1} \to \mathcal{M}$ is a bounded morphism, it holds that an unravelled model is truth-invariant compared to the starting model.

One interesting aspect of unravelling is that it is possible to prove, for monotonic modal logic based on neighbourhood semantics, that the tree model property holds. Specifically, the property states that if a formula is satisfied in a model, then it is satisfied also at the root of some tree-like model.

Theorem 6 (Tree Model Property). *Given a modal formula $\phi \in \mathcal{L}$, if ϕ is satisfiable in some monotonic model, then ϕ is satisfiable at the root of some tree-like monotonic model.*

Proof (Proof of Theorem 6). *Given the monotonic model $\mathcal{M} = (S, N, \pi)$ which satisfies ϕ at state $s \in S$, $\mathcal{M}$ can be unravelled from s to produce an unravelled version of the model $\vec{\mathcal{M}}_s = (\vec{S}_s, \vec{N}_s, \vec{\pi}_s)$ as defined in definition 25. Given the invariance result for such transformation, it is true that $(\mathcal{M}, s) \models \phi$ iff $(\vec{\mathcal{M}}_s, (s)) \models \phi$. Given the assumption, the formula ϕ is satisfied at the root of the tree-like model.*

For what concerns this thesis, this last invariance result concludes the discussion over truth-invariant transformations of models. However, before moving to the reflections about trust in logical languages, an interesting result should be mentioned.

Specifically, it is interesting to see that augmented neighbourhood models are in a one-to-one correspondence with standard Kripke models. This is interesting because those standard models are often employed to model knowledge in computational systems. The proof of the one-to-one correspondence follows.

The definition of a standard Kripke model is the following:

Definition 26 (Kripke/Relational Models). *A **Kripke frame** is a tuple $\langle S, R \rangle$, where S is a non-empty set of states and $R \subseteq S \times S$ is a relation on S. A **Kripke model** is a triple $\mathfrak{M} = \langle S, R, V \rangle$, where $\langle S, R \rangle$ is a Kripke frame and $V : At \to \wp(S)$ is a **valuation function** assigning sets of states to atomic propositions.*

In a relational model, truth of a formula at a pointed model is defined in the following way:

Definition 27 (Truth in a Relational Structure). *Given a model $\mathfrak{M} = \langle S, R, V \rangle$ and a formula ϕ, the formula is true at a pointed model $(\mathfrak{M}, s)$ in the following sense:*

$$(\mathfrak{M}, s) \models p \text{ iff } s \in \pi(p);$$
$$(\mathfrak{M}, s) \models \neg\phi \text{ iff } (\mathfrak{M}, s) \not\models \phi;$$
$$(\mathfrak{M}, s) \models \phi \wedge \psi \text{ iff } (\mathfrak{M}, s) \models \phi \text{ and } (\mathfrak{M}, s) \models \psi;$$
$$(\mathfrak{M}, s) \models \Box(\phi) \text{ iff } \forall t \text{ s.t. } sRt, (\mathfrak{M}, t) \models \phi.$$

Note now that, given a relation R on a set S, it is possible to define two corresponding functions:

- $R^{\rightarrow} : S \rightarrow \wp(S)$. For each $s \in S$, let $R^{\rightarrow}(s) = \{t \mid sRt\}$.
- $R^{\leftarrow} : \wp(S) \rightarrow \wp(S)$. For each $X \subseteq S$, let $R^{\leftarrow}(X) = \{s \mid \exists t \in X \ s.t. \ sRt\}$.

From those definition it is possible to define the notion of $R - necessity$.

Definition 28 (R-Necessity)**.** *Given a relation R on a set S and a state $s \in S$, a set $X \subseteq S$ is R-**necessary at** s if $R^{\rightarrow}(s) \subseteq X$. Define $\mathcal{N}_s^R$ to be the set of sets that are R-necessary at s, i.e., $\mathcal{N}_s^R = \{X \mid R^{\rightarrow}(s) \subseteq X\}$. The R superscript will be omitted when R is clear from context.*

One nice property of $R - necessary$ sets is the following:

Lemma 3. *Given a relation R on a set S, then $\forall s \in S, \mathcal{N}_s^R$ is augmented.*

Proof (Proof of Lemma 3)**.** *Take an arbitrary R on S and an arbitrary $s \in S$. Show that $\mathcal{N}_s^R$ is augmented, i.e., that $\mathcal{N}_s^R$ is monotonic and that it contains its core. For monotonicity: Assume that $X \in \mathcal{N}_s$, show that if $X \subseteq Y \subseteq S$, then $Y \in \mathcal{N}_s^R$. Assume that $X \subseteq Y \subseteq S$. Since $X \in \mathcal{N}_s^R$, then $R^{\rightarrow}(s) \subseteq X$. Given the assumption that $X \subseteq Y$, then, by transitivity of the subset relation, $R^{\rightarrow}(s) \subseteq Y$. By definition of $\mathcal{N}_s^R$, $Y \in \mathcal{N}_s^R$. For core containment: Show that $\bigcap(\mathcal{N}_s^R) \in \mathcal{N}_s^R$. By the definition of $\mathcal{N}_s^R$, $\forall X \in \mathcal{N}_s^R, R^{\rightarrow}(s) \subseteq X$, therefore $R^{\rightarrow}(s)$ is the smallest set common to all subsets of $\mathcal{N}_s^R$. Therefore $\bigcap(\mathcal{N}_s^R) = R^{\rightarrow}(s)$. By the equality condition of the subset relation, $R^{\rightarrow}(s) \subseteq R^{\rightarrow}(s)$. By the definition of $\mathcal{N}_s^R$, $R^{\rightarrow}(s) \in \mathcal{N}_s^R$. By substitution, $\bigcap(\mathcal{N}_s^R) \in \mathcal{N}_s^R$.* □

Given the two definitions of augmented neighbourhood models and of Kripke models, it is now possible to prove the modal equivalence between the class of relational models and that of augmented neighbourhood models.

First define a notion of equivalence between relational and neighbourhood frames.

Definition 29 (Point-wise equivalence)**.** *Given a non-empty set S, a neighbourhood frame $\mathcal{F} = \langle S, N \rangle$ and a relational frame $\mathfrak{F} = \langle S, R \rangle$, $\mathcal{F}$ and $\mathfrak{F}$ are said to be **point-wise-equivalent** if $\forall X \subseteq S$, $X \in N(s)$ iff $X \in \mathcal{N}_s^R$.*

Before providing the proof of the equivalence between the two classes of models, two lemmas are necessary.

Lemma 4. *Given a relational frame $\mathfrak{F} = \langle S, R \rangle$, there is a modally equivalent augmented neighbourhood frame.*

Proof (Proof of Lemma 4)**.** *Take an arbitrary relational frame $\mathfrak{F} = \langle S, R \rangle$. Build a neighbourhood frame $\mathcal{F} = \langle S', N \rangle$ in the following way: for S', just take the set S of the relational frame. For N, set each $N(s) = \mathcal{N}_s^R$, where the relation R is that of the relational frame. By lemma 3, it is guaranteed that those $\mathcal{N}_s^R$ are augmented, therefore the frame $\mathcal{F}$ is also augmented.*

□

Lemma 5. *Given an augmented neighbourhood frame $\mathcal{F} = \langle S, N \rangle$, there is a modally equivalent relational frame.*

Proof (Proof of Lemma 5)**.** *Take an arbitrary augmented neighbourhood frame $\mathcal{F} = \langle S, N \rangle$. Build a relational frame $\mathfrak{F} = \langle S', R \rangle$ in the following way: for S', just take the set S of the augumented neighbourhood frame. For R, set that $s_1 R s_2$ only if $s_2 \in \cap N(s_1)$. It is now necessary to show that for each $s \in S$, $\mathcal{N}_s^R = N(s)$. Assume $s \in S$ and $X \subseteq S$.*

Assume that $X \in \mathcal{N}_s^R$, then $R^{\rightarrow}(s) \subseteq X$. Since $R^{\rightarrow}(s) = \cap N(s)$ and N contains its core, $R^{\rightarrow}(s) \in N(s)$. Moreover, given the fact that N is monotonic, $X \in N(s)$.

Now, assume that $X \in N(s)$, then $\cap N(s) \subseteq X$. Therefore, $X \in \mathcal{N}_s^R$.

□

Theorem 7 (Equivalence between Relational and Augmented Neighbourhood Models)**.** *Given an augmented neighbourhood frame $\mathcal{F} = \langle S, N \rangle$ and a relational frame $\mathfrak{F} = \langle S, R \rangle$, if $\mathcal{F}$ and $\mathfrak{F}$ are point-wise equivalent, then, for any π and $\forall s \in S$, if $\mathcal{M} = \langle \mathcal{F}, \pi \rangle$ and $\mathfrak{M} = \langle \mathfrak{F}, \pi \rangle$, then $(\mathcal{M}, s) \equiv_{\mathcal{L}} (\mathfrak{M}, s)$.*

Proof (Proof of Theorem 7)**.** *Assume that $\mathcal{F} = \langle S, N \rangle$ and $\mathfrak{F} = \langle S, R \rangle$ are point-wise equivalent. Take an arbitrary π and an arbitrary $s \in S$. Show that $(\mathcal{M}, s) \equiv_{\mathcal{L}} (\mathfrak{M}, s)$, with $\mathcal{M} = \langle \mathcal{F}, \pi \rangle$ and $\mathfrak{M} = \langle \mathfrak{F}, \pi \rangle$. The proof is by induction on the structure of the formula ϕ.*

Base Case*.*

ϕ is an unanalysed propositions $p \in At$. This follows directly from the fact that the valuation π and the set S is the same on both structures.

Inductive Hypothesis (IH)*.*

The $\pi^{ext}(\phi)$ is the same in both structures.

Inductive Steps*.*

All Boolean cases follow directly from arguments similar to the one given for the base case.

Box modality: Show that $(\mathcal{M}, s) \models \Box(\phi)$ iff $(\mathfrak{M}, s) \models \Box(\phi)$.

Left-to-right.

Assume $(\mathcal{M}, s) \models \Box(\phi)$. Then $s \in \pi^{ext}(\Box(\phi))$, which means that $s \in m_N(\pi^{ext}(\phi)$. By definition, this means that $\pi^{ext}(\phi) \in N(s)$. By the definition of point-wise equivalence, it follows that $\pi^{ext}(\phi) \in \mathcal{N}_s^R$. By definition of $\mathcal{N}_s^R$, it follows that $R^{\rightarrow}(s) \subseteq \pi^{ext}(\phi)$, therefore, ϕ holds in all states accessible from s. Thus, $(\mathfrak{M}, s) \models \Box(\phi)$.

Right-to-left.

Assume $(\mathfrak{M}, s) \models \Box(\phi)$. This means that ϕ holds in every state accessible from s. Thus $R^{\rightarrow}(s) \subseteq \pi^{ext}(\phi)$. By definition of $\mathcal{N}_s^R$, it follows that $\pi^{ext}(\phi) \in \mathcal{N}_s^R$. By the definition of point-wise equivalence, it holds that $\pi^{ext}(\phi) \in N(s)$. By the definition of m_N and of truth in a neighbourhood model, it follows that $(\mathcal{M}, s) \models \Box(\phi)$.

Diamond modality: The proof is obtained by combining the proofs for negation and the box modality. □

Putting together theorem 7 with lemmas 4 and 5, it holds that the two classes of models are modally equivalent. Therefore, general results concering one of the semantical structures, also hold in the other.

4.1.4 Reasoning About Trust

Adding trust to the semantical structure of a logical language is no easy task. What is required is an element, inside the structure, that can evaluate trust formulas according to a proper definition of computational trust. As argued in previous chapters of this thesis (see chapters 2 and 3), such a notion must account for the fact that trust is a relation, that trust is measurable, that this measure is subjective and that it can vary depending on different contexts of evaluation. Thus, what the structure should contain is, first of all, a set of evaluation contexts in which to assess trust. Moreover, in each of those contexts, all else equal, different agents must be allowed to attribute different values to trust. Those measures must depend on various elements known by the agent. As it has been shown, which elements are relevant can vary greatly depending on how computational trust models are built, starting from just reputation scores, up to complex data such as the mental states of other agents. Regardless of how many and which ones should be taken into account, a general model for computational trust, must allow agent to attribute relevance values to those elements, in order to transform those relevance *for trust* values into *actual* trust values. Finally, the trickiest part is that of representing trust as a relation. This might be straightforward in a setting where predicate logic is employed; however, in this thesis, the focus is on the modal fragment of propositional logic, therefore, there is no direct space for relations in the language. To overcome this difficulty, the best way is to translate relations into propositions and then assess trust over those propositions. This simple solution can then generalize the concept of trust, allowing it to be assessed for all formulas of the language. In this sense, trusting a formula becomes equal to trusting that the formula under analysis is true. The fact that trust, so intended, becomes a modality over truth is interesting, because its treatment can become quite similar to that of other modal operators for which properties and general rules are studied. Despite this nice parallel, formalising trust in this way in a logical language expresses a quite strong view on trust. This view, although reasonable, is far from being innocent., therefore, the approach should be seen as one among many others, which might turn out to be more intuitive. On the other hand, the approach presented in this thesis seem to be a good one under many aspects. For instance, the semantical structure which will be proposed possesses many desirable properties a semantical structure ought to possess (e.g., being decidable with respect to most decision problems); in addition, the tools which characterize the structure can be employed to draw parallels with other formalisms, allowing a strengthening of those formalisms; finally the formalism is simple and straightforward and its internal procedures permit easy implementations in computational systems.

In the next sections, two versions of the language will be proposed: the first one will be contextless, where it should be assumed that the context of evaluation is set beforehand by the modeller and thus making the language application-specific; the second language will include the contexts, making it a more versatile version of the previous one. At this stage, contexts are still seen as primitives. This is an issue which should be explored more deeply and which is, unfortunately, relegated to future works. Another important aspect is that both languages are single-agent and thus, do not allow for interactions between trust and knowledge of different agents inside the same language. However, differently from the analysis required by contexts, building multi-agent languages should not be troublesome. Unfortunately, this thesis does not contain any detailed analysis of such possibilities. One final extension which might prove to be interesting is that of a dynamic language, in which trust values and knowledge could be transferred from agent to agent. Again, this is a proposal for the future and thus, not included in the thesis.

4.2 A Context-Free Language for Trust

It will now be introduced the syntax and semantics of a formal language that will allow to reason about knowledge and trust and, furthermore, to provide a trust computing mechanism that can produce values to be fed into other models' manipulation component. The language will be called Modal Logic for Trust (MLT) and, specifically, Context-free Modal Logic for Trust (CF-MLT) and Contextual Modal Logic for Trust (C-MLT).

The leading idea of the framework comes from the assumption made at the end of chapter 3, i.e., that a system in which a computational concept of trust must be implemented should have the following features:

1. Information is fully analysable to the point of clearly determining its content. It is, thus, always possible to determine if an information is relevant to an interacting situation and it is possible, furthermore, to establish whether the information is possessed or not by the agent which must entertain the interaction.

2. Information in the environment is scarce, thus, most evaluations are based on personal opinions about this information and not the actual information *per se*. This is equivalent to the fact that the most relevant aspects employed in making decisions are beliefs, rather than actual knowledge.

3. Not only will new agents lack a trust value, but they will also not be aware of the trust values of others. In this sense, each new agent must

form his own knowledge base independently, thus achieving his personal trust values without having to rely of the opinions of others.

4. Even though interactions can be of the same type, they might depend on different aspects, depending on the agents who are evaluating the interaction. Therefore no general, universally recognized value, can be computed based on past interactions, because the success or failure of a past interaction might depend on contingent elements that where not factored in the simple annotation of success or failure.

Another important aspect which determined specific decisions concerning the language come from an insight found in [64]:

> " ...[T]rust ultimately is a personal and subjective phenomenon that is based on various factors or evidence, and that some of those carry more weight than others. Personal experience typically carries more weight than second hand trust referrals or reputation ... "

The idea is therefore that of using the expressive power of a formal language to describe the information possessed by an agent and then transform this knowledge into a trust value about a given proposition.

This idea of linking possessed knowledge and trust is not a novel one. In the sociological literature, one of the authors expressing the idea of the importance of this link is Russell Hardin:

> " ...rational subjects must choose in the light of what knowledge they have, and that knowledge determines their capacities for trust. ...It is commonly argued that trust is inherently embedded in iterated, thick relationships. But such relationships are merely one source of relevant knowledge in a street-level epistemological account. Early experience may heavily influence later capacity for trust. For example, bad experiences may lead to lower levels of trust and therefore fewer opportunities for mutual gain. " [54]

It is therefore advisable to construct a language that uses knowledge as an enabling factor for trust. This insight, coupled with the fact that modal logic is one of the best available formal languages for modelling knowledge [58], inspired the idea of designing a language for trust embedded in modal logic.

The language which derives from those choices is similar to an evidence-based logic (where evidence is represented as possessed knowledge[4]), such that different pieces of evidence are assigned weights that determine whether an agent trusts a given proposition or not. Basically, the language is a modal

[4] See [13] for another example on how evidence can be represented.

language augmented with a trust operator, interpreted in an augmented neighborhood semantical structure[5].

4.2.1 Syntax

In the language (CF-MLT) $\mathcal{L}(At)$ (for short $\mathcal{L}$) of logic formulas (which are ranged over by $\phi, \psi, \dots$), the starting point is the finite set At of atomic propositions representing basic pieces of information. Given $p \in At$ the language is defined by the following BNF grammar:

$$\phi := p \mid \neg\phi \mid \phi \wedge \phi \mid K(\phi) \mid T(\phi)$$

All other connectives are defined in the standard way and are included (as abbreviations) a dual operator for knowledge and one for trust (expressing possible knowledge and possible trust):

1. $\phi \vee \psi := \neg(\neg\phi \wedge \psi)$;
2. $\phi \rightarrow \psi := \neg\phi \vee \psi$;
3. $\phi \leftrightarrow \psi := (\phi \rightarrow \psi) \wedge (\psi \rightarrow \phi)$;
4. $\widehat{K}(\phi) := \neg K(\neg\phi)$;
5. $\widehat{T}(\phi) := \neg T(\neg\phi)$

Formula $K(\phi)$ should be intuitively read as "formula ϕ is known"; such formulas are called *knowledge formulas*. Formula $T(\phi)$ should be intuitively read as "formula ϕ is trusted"; such formulas are called *trust formulas*. The degree to which a formula can be trusted goes from 0, complete distrust, to 1, complete trust; the point of transition from distrust to trust will strictly depend on the semantical structure interpreting the language.

4.2.2 Semantics

The semantics provided in this section is in truth theoretical form and depends on a structure that is a combination of an augmented neighborhood structure for modal logics [100] and an added component to assign weights to formulas. This added component is fundamental to interpret trust formulas. Even though it has been stressed their importance for trust evaluations, in this first version of the language, contexts are not included. This little caveat implies that the notion of trust captured by this logical language is one that is *multiplex* in nature, as defined in chapter two. On the other hand, both

[5] See [21, 48, 72, 12] for a general introduction to modal logics and monotonic neighborhood structures. Moreover, see [124] for an approach that interprets the same language in a standard relational structure.

the *how* and the *whom* dimensions are fully captured by the laguage, since the information chosen by the evaluating agent might be both fully strategic or moralistic, while the object of trust can be both single agents (when the formulas evaluated concern only single entities) or whole groups (when the formulas evaluated express information about a whole group). Note, therefore, that the language allows a high level of freedom to the modeler, who can freely choose to describe or reason about many different situations.

The language is interpreted in the following structure:

Definition 30 (Context-free Trust Model). *A* ***context-free trust model*** *is a tuple $M = (S, \pi, N, \mathcal{T}, \theta)$, where*

- *S is a finite set of possible states of the system which is modeled $s, s', \ldots$*[6].
- *π is a valuation function, assigning set of states to atomic propositions.*
- *N is an augmented neighborhood function.*
- *$\mathcal{T} = \{\langle \omega, \mu_\phi \rangle \mid \phi \in \mathcal{L}\}$ is a trust relevance structure.*
- *θ is a trustworthiness threshold function.*

A context-free trust model in which there is no valuation function π is called a context-free trust frame.

Intuitively, a possible state $s \in S$ represents a way in which the system (either the real world or the states of a computing device) can be specified[7]; hence, two states differ from one another by what propositions hold in such states. It is assumed that states are *maximally consistent* descriptions of the system. They are maximal inasfar as the truth value of each proposition is specified. They are consistent inasfar as a proposition and its negation can't both be trusted in the same state.

Function π is a valuation function that assigns to each proposition $p \in At$ a set of states, i.e., $\pi : At \to \wp(S)$; a state is included in the set if, and only if, the proposition holds in the given state[8]. Starting from π it is possible

[6] Often, in logic, those states are also called possible worlds, without any reference to the systems being modeled. On the other hand, it is customary in computer science to refer to systems which are being modelled and possible states as possible configurations (e.g., distribution of values to the variables in the description of the system) of the system.

[7] Here the reference is made to systems whose states can be determined clearly. The language chosen is typical of computer science, where the objects that are modeled are often computational systems and where the specifics are the values of the variables defining the system.

[8] This is an important point in the language, because most functions employed work on states (sets of states) rather than on propositions and/or formulas. However, note that it is always possible to move from propositions to sets of states in which such propositions hold, and viceversa. Therefore, it is reasonable and intuitive to assume that all functions working on sets of states, actually work on propositions instead. The same holds when generalizing to formulas in place of atomic propositions.

to define a further labeling function $L : S \to \wp(At)$: L is a function that associates each state with the subset of atomic propositions that are true in that state. The labeling function L is introduced to simplify the discussion in the sections that will follow, however, this function is not strictly needed, thus it is not part of the semantical structure for the language.

Function N is an augmented neighborhood function that assigns to each state $s \in S$ a set of subsets of S, i.e., $N : S \to \wp(\wp(S))$; the set of subsets obtained by applying N is closed under superset, i.e., for each $X \subseteq S$ and each $s \in S$, if $X \in N(s)$ and $X \subseteq Y \subseteq S$, then $Y \in N(s)$. Moreover, N contains its core, i.e., $\cap N(s) \in N(s)$. Intuitively, function N assigns to each state the sets of states *corresponding to the known* propositions in such state [9]. The neighborhood function is employed to interpret the knowledge operators of the language. Note that using neighborhood functions knowledge is defined directly[10]: thus, once the informative content of a proposition is determined (in the specific case by applying function π), the function N assigns to each state of the system a set containing all those contents corresponding to the known propositions. The closure under superset condition expresses the intuitive idea that when something is known, weakened pieces of information derived from the knowledge possessed are also known [11]. The closure under core, on the other hand, indicates that an agent is always aware of the conjunction of the information he possesses.

$\mathcal{T}$ is a trust relevance structure, where for each $\phi \in \mathcal{L}$, there is an ordered couple $\langle \omega, \mu_\phi \rangle$. ω is a function that assigns to each formula $\phi \in \mathcal{L}$ a consistent[12] set of subsets of S, i.e., $\omega : \mathcal{L} \to \wp(\wp(S) - \emptyset)$ (this condition expresses the informal idea that contradictions might not be considered relevant for trust formulas). This consistent set, which we call Ω_ϕ, contains the sets of states corresponding to the formulas relevant for trust in ϕ. μ_ϕ is a trust weight function, assigning to elements in Ω_ϕ rational numbers in the range [0, 1] according to their relevance for trust in the formula, i.e. $\mu_\phi : \Omega_\phi \to [0, 1] \in \mathbb{Q}$. 0 represents no trust relevance and 1 represents full trust relevance. It is assumed that the weights assigned are subadditive to 1, i.e., $\sum_{X \in \Omega_\phi} \mu_\phi(X) \leq 1$, guaranteeing that it is never possible to exceed full trust (i.e. the value 1). Intuitively, the functions μ_ϕ assign to the trust relevant formulas a specific weight for trust, with respect to a given formula ϕ, which is evaluated for

[9] To make the exposition simpler during the course of the paper, elements of $\wp(S)$ will be indicated with letters from the end of the alphabet capitalized and with eventual superscripts and subscripts, i.e., $X, X_2, Y, X', X'_2, Y' \ldots$.

[10] This is a completely different approach from standard relational structures, where the modally relevant operators are defined in terms of truth in the structure. However, note that there is a one-to-one correspondence between relational structures and augmented neighbourhood structures

[11] For instance, if a proposition p is known at a state s, i.e., $\pi(p) \in N(s)$, then also $p \vee q$ is known at s, i.e., $\pi(p \vee q) \in N(s)$.

[12] Recall that the definition of consistency for a collection of sets is the following: $\mathcal{U}$ is **consistent**, if $\emptyset \notin \mathcal{U}$.

trust. The notion of relevance employed is an intuitive one: an information related to a formula is relevant for trust, if knowing such information would modify the trust assessment made towards that formula. Obviously, having no trust relevance means that whether or not the information is known, the trust assessment would be the same; on the other hand, full trust relevance means that knowing the information is the only way it is possible to modify the trust assessment.

Finally, θ is a trustworthiness threshold function that assigns to each formula $\phi \in \mathcal{L}$ a rational number between 0 and 1[13], i.e. $\theta : \mathcal{L}(At) \to [0,1] \in \mathbb{Q}$. This rational number indicates the minimum threshold needed to trust the formula.

It is important to notice that the values attributed in $\mathcal{T}$ are arbitrary and will depend mostly on the specific application of the model. Theoretically, it can be thought that $\mathcal{T}$ contains all the infinite possible attribution of values that are consistent with the assumptions made on the model. Although this holds in theory, on the practical side of the thing, different applications will require different attribution of values. The general idea is that each agent participating in a digital community will have a specific set of functions representing his attitudes.

Before providing the truth definition for a formula in a model, some further functions must be added; those functions will help in defining the truth of knowledge and trust formulas. Some of the functions have already been introduced in subsection 4.1.2; they are reintroduced just for clarity.

First note that a neighborhood function N can induce a map m_N, which is a function that associates to each element $X \in \wp(S)$ another element $Y \in \wp(S)$, according to the neighborhood function N, i.e. given $N : S \to \wp(\wp(S))$, there is a map $m_N : \wp(S) \to \wp(S)$. The function m_N is defined formally as follows:

$$m_N(X) = \{s \mid X \in N(s)\} \tag{4.2}$$

Intuitively, m_N returns, for each set of states corresponding to a formula (i.e. the formulas informative content), a set of states such that a state is in the set if, and only if, the formula is known in the state. Therefore, if a formula ϕ corresponds to a set of states X, then $s \in m_N(X)$ iff $K(\phi)$ holds in s.

The relation between N and m_N is similar in spirit to the one between the valuation function π and the labelling function L. The function m_N will help in defining the truth of knowledge formulas.

A second and important derived element of the semantical structure is the family of functions $\Lambda = \{\tau_\phi \mid \phi \in \mathcal{L}(At)\}$, which contains functions that assign ideal trust values to states of the system. Intuitively, a function τ_ϕ

[13] Real numbers could have been employed. However, it is believed that density is sufficient to capture the different grades of trust and continuity is not required. For this reason, it follows the choice to use rational numbers.

($\tau_\phi : S \rightarrow [0,1] \in \mathbb{Q}$) indicates how much trust an agent has in the formula ϕ (representing the parameter of μ_ϕ) in the given state denoting the argument of τ_ϕ, provided that the agent is aware, in such a state, of all the relevant basic information related to ϕ, i.e., the agent knows all the relevant propositions which are true in that state. Another way to put it is the following: if an agent knows exactly which one is the current state of the system, then τ_ϕ will specify the amount of trust the agent has towards ϕ. Thus, τ_ϕ represents an ideal measurement of trust. Note that, even though ideal, this is a trust measure indicating how much an agent trusts the proposition ϕ in the given state and is therefore a subjective measurement.

Functions τ_ϕ[14] are defined as follows:

$$\tau_\phi(s) = \sum_{X \in \Omega_\phi : s \in X} \mu_\phi(X) \tag{4.3}$$

It is assumed that if in (4.3) there is no X such that $s \in X$ then $\tau_\phi(s) = 0$. Moreover, the subadditivity criterion on μ_ϕ guarantees that τ_ϕ itself never exceeds 1 (this is to be expected, since trust, even in an ideal setting might never exceed the maximum value of 1, i.e., full trust). Note that it is possible that $\tau_\phi(s) = 0$ and $\tau_{\neg\phi}(s) < 1$, thus the functions do not complement each other. This is perfectly reasonable, given the fact that trust, especially in ideal settings, might not be closed under complementation. In fact, it is perfectly acceptable that an agent doesn't trust a given proposition at all and, at the same time, he does not fully trust the negation of such proposition.

Given the family of functions τ_ϕ, it is possible to define a trust value for each $X \in \wp(S)$. The functions performing such task will be defined as τ_ϕ^{ext}. Such functions are defined as follows:

$$\tau_\phi^{ext}(X) = min_{s \in X}\{\tau_\phi(s)\} \tag{4.4}$$

Intuitively, the function τ_ϕ^{ext} looks at all states in the set X under analysis and selects the worst-case scenario, i.e., that in which the trust value is the lowest. This choice models the behaviour of a cautious agent, which will only consider the information he possesses to make an evaluation on trust and won't therefore make any other assumption on the trustworthiness of the formula under analysis. However, other possibilities for the definition of τ_ϕ^{ext} are possible, such as taking the maximum (which would model the behaviour of an optimistic agent) or the average value between all the $\tau_\phi(s)$ (which would model the behaviour of an agent which is neither cautious nor optimistic).

Specifically, such definition would be formalized as follows.

For the maximum (optimistic agent):

$$\tau_\phi^{ext}(X) = max_{s \in X}\{\tau_\phi(s)\} \tag{4.5}$$

[14] Again, one for each $\phi \in \mathcal{L}$.

For the average (neutral agent):

$$\tau_{\phi}^{ext}(X) = \frac{\sum_{s \in X}\{\tau_{\phi}(s)\}}{\mid X \mid} \tag{4.6}$$

Where $\mid X \mid$ stands for the cardinality of X, i.e., it represents the number of states in X.

A further remark needs to be done about the value of τ_{ϕ}^{ext}, which can never exceed 1, meaning that for no set of states can trust exceed its maximum value. It is interesting to observe that if the formula is applied to a singleton set containing only a single state s (i.e., $X = \{s\}$), then the value of the function $\tau_{\phi}^{ext}(X)$ is equal to the value of $\tau_{\phi}(s)$. This proves that τ_{ϕ}^{ext} is a proper extension of τ_{ϕ}.

Put concisely, the way trust formulas will be evaluated is the following:

1. Determine the minimal set of states compatible with all the information known by an agent in a state.

2. Compute the trust relevance weight of such minimal set.

3. The value obtained indicates how much an agent trusts the formula indicated by the set.

To improve the readability of the truth theoretical definition for the formulas, a definition of truth set is given for each formula of the language.

Definition 31 (Extension of the Valuation Function)**.** *Given a context-free trust model $M = (S, \pi, N, \mathcal{T}, \theta)$, then the truth set of a formula, denoted π_M^{ext}, is defined recursively as follows:*

- $\pi_M^{ext}(p) = \pi(p)$ *for all* $p \in At$;
- $\pi_M^{ext}(\neg\phi) = S - \pi_M^{ext}(\phi)$;
- $\pi_M^{ext}(\phi \wedge \psi) = \pi_M^{ext}(\phi) \cap \pi_M^{ext}(\psi)$;
- $\pi_M^{ext}(K(\phi)) = m_N(\pi_M^{ext}(\phi))$;
- $\pi_M^{ext}(T(\phi)) = \{s \mid \tau_{\phi}^{ext}(\bigcap_{X \in N(s)} X) \geq \theta(\phi)\}$.

Two things that characterize the truth sets of trust formulas are: first, $\bigcap_{X \in N(s)} X$, which can also be indicated with $\bigcap N(s)$, is the core of $N(s)$ and indicates the minimal set of states which are compatible with all the knowledge of the agent; second, to compute the π^{ext} of $T(\phi)$, it must be checked whether in a given state the trust value of the core of N in such state is greater than or equal to the trustworthiness threshold for the formula.

Now that all the elements of the semantical structure have been introduced, it is possible to provide the truth definition of a formula ϕ:

Definition 32 (Truth). *Given a context-free trust model $M = (S, \pi, N, \mathcal{T}, \theta)$, and a state $s \in S$, a formula ϕ is true in a context-free pointed model, indicated with $(M, s) \models \phi$, according to the following conditions:*

- $(M, s) \models p$ *iff* $s \in \pi(p)$;
- $(M, s) \models \neg\phi$ *iff* $s \in \pi^{ext}(\neg\phi)$;
- $(M, s) \models \phi \wedge \psi$ *iff* $s \in \pi^{ext}(\phi \wedge \psi)$;
- $(M, s) \models K(\phi)$ *iff* $s \in \pi^{ext}(K(\phi))$;
- $(M, s) \models T(\phi)$ *iff* $s \in \pi^{ext}(T(\phi))$.

The above structure is sufficient to reason about knowledge and trust and their inter-relationship in a single-agent, context-free environment. Furthermore, the language provides effective tools to compute pre-trust values and therefore can be also employed as a trust computing model.

In the next section, an extension of this language will be proposed.

4.3 A Language for Trust

In this section, an extension of the language presented in the previous section is proposed. In this language, a specific component for contexts is added at the semantical level. Such contexts will allow to produce semantical reflections in line with all previous reasonings about trust. Specifically, trust values will depend both on the formula being evaluated and the context in which it is evaluated. Those contexts of evaluations will allow for a more fine-grained description of trust: where in the previous language a formula was either trusted or not, in this contextual version of the language, a formula might be trusted in some contexts, but not in others. Moreover, having contexts will also allow to define different notions of validity, which, it will be shown, correspond to versions of trust commonly found in the literature.

4.3.1 A Language for Trust: Syntax

The syntax for the language is the same as the one presented in the previous section. Therefore, the language (C-MLT) $\mathcal{L}(At)$ (for short $\mathcal{L}$) of logic formulas (which are ranged over by $\phi, \psi, \dots$), the starting point is the finite set At of atomic propositions representing basic pieces of information. Given $p \in At$ the language is defined by the following BNF grammar:

$$\phi := p \mid \neg\phi \mid \phi \wedge \phi \mid K(\phi) \mid T(\phi)$$

All other connectives are defined in the standard way and are included (as abbreviations) a dual operator for knowledge and one for trust (expressing possible knowledge and possible trust)

4.3.2 A Language for Trust: Semantics

In the contextual version of the language for trust, logical formulas are interpreted in the following structure.

Definition 33 (Contextual Trust Model)**.** *A* ***contextual trust model*** *is a tuple $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$, where*

- *S is a finite set of possible states of the system $s, s', \ldots$.*
- *C is a finite set of primitive evaluation scenarios $c, c', \ldots$. Such scenarios can be considered as points of evaluation, where an agent must determine whether he trusts or not a given formula. In this sense, in each possible state of the system, the same set of primitive contexts is attributed and then, for each point in such set, a different evaluation of trust formulas is given in the state. Note that it is theoretically possible to evaluate formulas according to subsets of the set of contexts, thus assessing trust in scenarios which include different evaluation criteria. However, for the scope of this thesis, only evaluations considering single contexts will be taken into consideration*[15]*.*
- *π is a valuation function, assigning set of states to atomic propositions.*
- *N is an augmented neighborhood function.*
- *$\mathcal{T} = \{\langle \omega_c, \mu_{c,\phi} \rangle \mid c \in C$ and $\phi \in \mathcal{L}\}$ is a trust relevance structure.*
- *θ_c is a trustworthiness threshold function.*

As it is easy to notice, the only difference between a context-free trust model and a contextual trust model is given by the presence of the set of contexts C and by the influence this set of contexts has on the trust related part of the model. Intuitively, the set C is a finite set of primitive scenarios, where a scenario is a situation in which trust must be assessed. For instance, someone might trust his mechanic when it comes to fixing cars, but might not trust him for financial advices. In the example, “fixing cars" and “giving financial advices" are to be considered two separate contexts of evaluation, thus two elements of the set of contexts. Note that those two contexts might

[15] It is possible to imagine that the semantics provided in the following section is two-dimensional, in the sense that formulas are evaluated according to two distinct dimensions. The main dimension is the state of evaluation. Such dimension determines the facts that are true for the system and what is known by the evaluator. The second dimension is the context of evaluation. Such dimension determines the reason why some formula is being evaluated and thus what is actually relevant for trust in such formula. The first dimension shall be considered the one characterizing the main interpretation tool for formulas, while the second dimension is the one characterized by the contexts in which such evaluation shall take place.

be considered in every possible state of the system and thus, it might also be assumed that there are equivalent sets of contexts for each state of the system.

All other elements of the structure behave exactly as in context-free structures. The only slight difference is that the trust components of the model are always indexed with respect to a context. Similarly, all functions derived from those components are indexed according to the same context of evaluation. While a small addition from a formal perspective, having contexts in the language greatly enhance the expressivity of the language. In particular, *simplex* notions of trust can now be modeled without problems, hence covering the whole space of trust definitions.

We thus get:

$$\tau_{c,\phi}(s) = \sum_{X \in \Omega_{c,\phi} : s \in X} \mu_{c,\phi}(X) \tag{4.7}$$

And:

$$\tau^{ext}_{c,\phi}(X) = min_{s \in X}\{\tau_{c,\phi}(s)\} \tag{4.8}$$

This influences the truth set of trust formulas in the following way:

$$\pi^{ext}(T(\phi)) = \{s \mid \tau^{ext}_{c,\phi}(\textstyle\bigcap_{X \in N(s)} X) \geq \theta(c, \phi)\}$$

Those changes produce a change also in the truth definition for the semantics, which is given with respect to contextual pointed models, rather than simple pointed models:

Definition 34 (Contextual Pointed Model)**.** *A **contextual pointed model** is a triple (M_C, s, c), where M_C is the contextual trust model, $s \in S$ is a state of the system and, finally, $c \in C$ is a context of evaluation.*

The definition of truth follows:

Definition 35 (Truth)**.** *Given a contextual trust model $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$, a state $s \in S$, and a context $c \in C$, a formula ϕ is true in a contextual pointed model, indicated with $(M, s, c) \models \phi$, according to the following conditions:*

- *$(M, s, c) \models p$ iff $s \in \pi(p)$;*
- *$(M, s, c) \models \neg\phi$ iff $s \in \pi^{ext}(\neg\phi)$;*
- *$(M, s, c) \models \phi \wedge \psi$ iff $s \in \pi^{ext}(\phi \wedge \psi)$;*
- *$(M, s, c) \models K(\phi)$ iff $s \in \pi^{ext}(K(\phi))$;*
- *$(M, s, c) \models T(\phi)$ iff $s \in \pi^{ext}(T(\phi))$.*

In general, it seems like context do not add much to the structure, if not subtle nuances. It might be argued that it is sufficient to construct a different context-free trust model for each context and the same results that can be achieved in the contextual models are achieved in the multiple context-free models. While this critique is partially true, the strength of contextual models come from their definition of validity. In fact, for contextual models, four different notions of validity might be defined, each one corresponding to different notions of trust, showing that the semantical structure proposed has the impressive capacity of capturing slightly different conceptions of trust, employing the same tools. In particular, the notions of validity that can be defined in contextual models are:

Definition 36 (Validity)**.** *Given a contextual trust model $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$:*

*A formula ϕ is **context-valid** with respect to M_C if:*

$$\exists c \in C \text{ s.t. } \forall s \in S : (M, s, c) \models \phi \tag{4.9}$$

*A formula ϕ is **state-valid** with respect to M_C if:*

$$\exists s \in S \text{ s.t. } \forall c \in C : (M, s, c) \models \phi \tag{4.10}$$

*A formula ϕ is **model-valid** with respect to M_C if:*

$$\forall s \in S \ \forall c \in C : (M, s, c) \models \phi \tag{4.11}$$

*Finally, a formula ϕ is **valid** ($\models \phi$) if it is model-valid for every model M.*

When assessing trust formulas according to those validity principles, nice considerations about trust might be derived.

If a trust formula is context-valid, then the notion of trust analysed is one for which it exists a context in which what is known is irrelevant for the attribution of trust. Thus, whatever the state of the system is, in that context trust will be granted. This kind of trust is typical of situations in which there is little choice other than trusting and no matter what is the level of knowledge, trust is always the best decision. An example could be a situation where the cost of not trusting and therefore not collaborating can be so high that even if the other agent then defects the collaboration, the loss is still less or equal to the cost of not trusting. Take, again, the example of the worn rope which an agent must choose whether to use or not to escape his house while it is burning. Assuming that the cost of not using the rope is death for the agent, no matter what he knows about the rope, he will trust it and use it to try and escape. This is because, even if the rope breaks (defects the trusting relationship), the worse that can happen to the agent is that he fractures his leg falling. Trust formulas that are context-valid seem to capture profoundly the idea behind moralistic versions of trust. In moralistic trust, education and upbringing determine instinctive attributions of trust. It the ethical and moral status of the trustor to determine whether he trusts or not and information specific to the trustee is often irrelevant. On the other hand, strategic versions of trust can hardly be context-valid, since high amount of specific information is required to determine whether to trust or not the trustee.

If a trust formula is state-valid, then the notion of trust analysed is similar to what Marsh [86] calls, general trust. Such a notion of trust describes an omnicomprehensive and general attitude of an agent towards a proposition in a given state of the system, independently from the context of evaluation. This means that the knowledge he possesses, in the given state, is sufficient to have trust in the proposition independently on what is actually relevant for it. This might be the case when an agent evaluates some general factors as relevant for trust independently from the contexts. For example, he might believe that, independently from the situation, a buddhist monk wouldn't never fail to collaborate or maintain his word, therefore, knowing that someone is a buddhist monk is sufficient to trust him, no matter the context. Trust formulas that are state-valid seem to capture the idea behind multiplex versions of trust. In its multiplex version, trust doesn't depend on the specific context of evaluation and it therefore represents a general feeling towards the trustee. Those conception of trust is clearly captured by state-valid trust formulas.

If a trust formula is model-valid, then the notion of trust analysed is that of blind trust. Independently from the knowledge of the agent and the context of evaluation, the agent simply trusts someone else. This happens often with parental relationships. Children trust their parents instinctively, independently from what they know (they often know very little) and what is the context of evaluation (they do not trust them **to**, but just trust them). Trust formulas that are model-valid seem to capture an idea of trust that is close to prejudice. When agents assess trust through prejudices, they often rely very little on actual information and they do not consider what is the task to be performed. Comparably, model-valid trust formulas are those that are true independently from knowledge and context and thus capture this idea of trust based on prejudice quite well.

Finally, if a trust formula is valid, then the notion of trust described in one where independently from what is modelled, there is blind trust in the proposition. A trust of this kind might be so rare, than in fact it might also be that is doesn't exist. As it will be shown, the language proposed can predict this, since it can be proved that there are no valid trust formulas in the language.

This concludes reflections on the semantical structure. In the next section, decidability results for the language are given.

4.4 A Language for Trust: Decidability Results

In this subsection, the proofs of the decision procedures for some interesting problems in the setting of the language presented will be given. In particular, given any logical language, there are three natural computational problems that arise:

- *Model checking problem*: This is the problem of deciding, given a model and a formula, whether the model satisfies the formula.
- *Model equivalence problem*: This is the problem of deciding, given two models, whether the models satisfy the same formulas. The model equivalence issue helps in determining the expressive power of a language, since, once it can be determined which models are equivalent and which are different, it is also possible to determine which classes of models are equal and which are different, thus determining if some class of model can express more or less things (in the form of logical formulas) compared to other classes.
- *Satisfiability problem*: This is the problem of deciding, given a formula, whether that formula is satisfiable by some model. The importance of the satisfiability problem is related to the validity problem. Once it is determined whether a formula is satisfiable or not, it is also possible to determine through an effective procedure which are the valid formulas of the language. Given the fact that sets of valid formulas semantical identify classes of models, through model equivalence, the solution to the satisfiability problem helps in determining those classes of models.

It will be proved that the last language introduced in this thesis is decidable w.r.t. the three decision problems indicated above.

Theorem 8 (Model Checking for C-MLT.)**.** *The model checking problem for Contextual Modal Logic for Trust is decidable, i.e., given a contextual model $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$, a state $s \in S$ and a formula $\phi \in \mathcal{L}$, it is possible to decide whether $(M, s, c) \models \phi$ or $(M, s, c) \not\models \phi$.*

Proof (Proof of Theorem 8)**.** *Take an arbitrary formula $\phi \in \mathcal{L}(At)$, an arbitrary contextual model $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$, an arbitrary state $s \in S$ and an arbitrary context $c \in C$. Compute $\pi^{extT}(\phi)$ by following the procedure given in the previous subsection. At this point, check if the state s is a member of the set $\pi^{extT}(\phi)$, i.e., check whether $s \in \pi^{extT}(\phi)$.*

If yes, then the formula is satisfied by the model in the state, otherwise it is not. In both cases, you obtain an answer to the model checking problem. □

In order to provide the proof of the model equivalence problem, a modified version of bisimulation will be introduced (see definitions 15 and 16 for the standard definition of bisimulation for neighbourhood semantics):

Definition 37 (Trust bisimulation). *Let $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$ and $M'_{C'} = (S', C', \pi', N', \mathcal{T}', \theta')$ be contextual trust models. A non-empty binary relation $Z \subseteq S \times S'$ is a **trust bisimulation** between M_C and $M'_{C'}$ (in symbols $Z : M_C \underline{\leftrightarrow} M'_{C'}$) if, with $s \in S$, $s' \in S'$, $C = C'$, and $\theta = \theta'$[16]:*

- *(prop) If sZs', then for all $p \in At$, $s \in \pi(p)$ iff $s' \in \pi'(p)$.*
- *(zig) If sZs' and $X \in N(s)$, then there is an $X' \subseteq S'$ s.t. $X' \in N'(s')$ and, for all $t' \in X'$, there exists a $t \in X$ s.t. tZt'.*
- *(zag) If sZs' and $X' \in N'(s')$, then there is an $X \subseteq S$ s.t. $X \in N(s)$ and, for all $t \in X$, there exists a $t' \in X'$ s.t. tZt'.*
- *(trust zig) If sZs', then for all the sets $\Omega_{c,\phi}$ generated by the ω_c functions, if $X \in \Omega_{c,\phi}$ and $s \in X$, then there exists a **unique** $X' \subseteq S'$ s.t. $X' \in \Omega'_{c,\phi}$, $s' \in X'$, and for all other $t' \in X'$, there exists a $t \in X$ s.t. tZt'. Moreover, $\mu_{c,\phi}(X) = \mu'_{c,\phi}(X')$.*
- *(trust zag) If sZs', then for all the sets $\Omega'_{c,\phi}$ generated by the ω'_c functions, if $X' \in \Omega'_{c,\phi}$ and $s' \in X'$, then there exists a **unique** $X \subseteq S$ s.t. $X \in \Omega_{c,\phi}$, $s \in X$, and for all other $t \in X$, there exists a $t' \in X'$ s.t. tZt'. Moreover, $\mu_{c,\phi}(X) = \mu'_{c,\phi}(X')$.*

Intuitively, the (prop) condition is needed to preserve local harmony at the atomic level. The two (zig) and (zag) conditions are needed to preserve equivalence of knowledge formulas between models. Finally, the two (trust zig) and (trust zag) conditions are needed to preserve equivalence of trust formulas between models. The uniqueness condition inside the (trust zig) and (trust zag) conditions simply state numerical conditions on how many sets might be in the trust bisimilarity relation. If more than one element exists inside the sets being compared, then the two sets must be label as not being trust bisimilar. Those conditions are mandatory in order to obtain the model equivalence results that will follow.

The following lemma will help us to prove model equivalence between trust bisimilar models.

Lemma 6. *Let $Z \subseteq S \times S'$ be a trust bisimulation between two contextual trust models M_C and $M'_{C'}$, then the following holds, with $s \in S$, $s' \in S'$, $C = C'$, and $\theta = \theta'$:*

$$\textit{If } sZs', \textit{ then } \tau_{c,\phi}(s) = \tau'_{c,\phi}(s'), \textit{ for all couples } (c, \phi).$$

[16] Each time it is claimed that the set of contexts are equal, what is argued is actually that the two sets contain exactly the same elements. Note that this would mean that both sets could be indicated with the same symbol. The equivalence of the teeta functions is similar in spirit. When it is claimed that they are equivalent, what is said is that they return the same values when given the same arguments.

Proof. Assume sZs' and take an arbitrary couple (c, ϕ). Rephrase the thesis, using the definition of τ, in the following way:

$$\textstyle\sum_{X \in \Omega_{c,\phi} : s \in X} \mu_{c,\phi}(X) = \sum_{X' \in \Omega'_{c,\phi} : s' \in X'} \mu'_{c,\phi}(X')$$

Expand both sums in the following way (with all the $X \in \Omega_{c,\phi}$ s.t. $s \in X$ and all the $X' \in \Omega'_{c,\phi}$ s.t. $s' \in X'$):

$$\mu_{c,\phi}(X_1) + \cdots + \mu_{c,\phi}(X_n) = \mu'_{c,\phi}(X'_1) + \cdots + \mu'_{c,\phi}(X'_m)$$

First, note that by the (trust zig) and (trust zag) conditions, it must be the case that $n = m$. If this were not the case, i.e., $n \neq m$, then either the (trust zig) or the (trust zag) unique existence condition would not be fulfilled and there would be a contradiction with the assumption that sZs'. For instance, suppose $n > m$, then it would be impossible to fulfil the (trust zag) condition, because it would be impossible to find a unique $A \in \Omega_{c,\phi}$ for each one of the $A' \in \Omega'_{c,\phi}$.

Now the proof is straightforward. Taking the (trust zig) and (trust zag) conditions together it is possible to create a one-to-one correspondence between the elements of the two summations. Moreover, the two conditions also assure that the $\mu_{c,\phi}$ and the $\mu'_{c,\phi}$ values of corresponding sets are themselves equal. Therefore the whole summations will return the same results. which means that $\tau_{c,\phi}(s) = \tau'_{c,\phi}(s')$.

□

Another important result which is needed is that of the bisimilarity of the two cores of two bisimilar models. Note that this is not the same thing as core bisimulation. In a core bisimulation, it is just require that the non-monotonic cores of each model are bisimilar, where the non-monotonic cores might contain more than one set inside them. On the other hand, *the core* of a set is a single set. Specifically, it is the set whose elements are common to all sets in the collection under analysis. What is important, therefore, is not that the non-monotonic cores are bisimilar (this would follow from the inverse of theorem 5), but that the actual cores are. The result will be proven for core-complete, monotonic models. Given that the part concerning knowledge of contextual trust models is based on augmented neighbourhood models and given the fact that those models are in fact core-complete and monotonic, then the result proven applies also to contextual trust models. In order to introduce the theorem, it is also necessary to extend the definition of bisimulation from models (and states) to sets.

Definition 38 (Set Bisimulation)**.** *Given two monotonic models $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ and $\mathcal{M}_2 = (S_2, N_2, \pi_2)$. If there is a bisimulation E between them, i.e., $\mathcal{M}_1 \underline{\leftrightarrow} \mathcal{M}_2$, then, given two subsets $X \subseteq S_1$ and $Y \subseteq S_2$, X and Y are said to be* ***set bisimilar****, indicated with $X \underline{\leftrightarrow} Y$ if:*

- *(Set zig)* $\forall x \in X, \exists y \in Y$ *s.t.* xEy.
- *(Set zag)* $\forall y \in Y, \exists x \in X$ *s.t.* xEy.

Intuitively, two sets are bisimilar, if a bisimulation relation can be built among all of the elements of the sets.

Theorem 9. *Given two core-complete, monotonic models* $\mathcal{M}_1 = (S_1, N_1, \pi_1)$ *and* $\mathcal{M}_2 = (S_2, N_2, \pi_2)$*. If there is a bisimulation* E *between* $s_1 \in S_1$ *and* $s_2 \in S_2$*, then, the two cores* $\bigcap N_1(s_1)$ *and* $\bigcap N_2(s_2)$ *are set bisimilar.*

Proof (Proof of Theorem 9)**.** *Assume* $s_1 E s_2$*. Show that* $\bigcap N_1(s_1) \underline{\leftrightarrow} \bigcap N_2(s_2)$*.*

The proof is by contradiction. Assume that $\bigcap N_1(s_1)$ *and* $\bigcap N_2(s_2)$ *are not set bisimilar. Then, either (a)* $\exists v \in \bigcap N_1(s_1)$ *s.t. for no* $v' \in \bigcap N_2(s_2)$, vEv'*, or (b)* $\exists r' \in \bigcap N_2(s_2)$ *s.t. for no* $r \in \bigcap N_1(s_1)$, rEr'*. Both cases produce a contradiction.*

Case (a): $\exists v \in \bigcap N_1(s_1)$ *s.t. for no* $v' \in \bigcap N_2(s_2)$, vEv'*. This is in contradiction with the (zag) condition of bisimulation, i.e., if* $s_1 Z s_2$ *and* $X_2 \in N_2(s_2)$*, then there is an* $X_1 \subseteq S_1$ *s.t.* $X_1 \in N_1(s_1)$ *and, for all* $t \in X_1$*, there exists a* $t' \in X_2$ *s.t.* tZt'*. To make the contradiction explicit, assume that* v_1 *is the actual element of* $\bigcap N_1(s_1)$ *for which there is not* $v' \in \bigcap N_2(s_2)$ *s.t.* $v_1 E v'$*. By the fact that* $v_1 \in \bigcap N_1(s_1)$*, then* $\forall X \in N_1(s_1)$, $v_1 \in X$*. Now, take* $\bigcap N_2(s_2)$*, by the (zag) condition of bisimulation,* $\exists X \subseteq S_1$, $X \in N_1(s_1)$ *and, for all* $v \in X_1$*, there exists a* $v' \in \bigcap N_2(s_2)$ *s.t.* vZv'*. Since* $v_1 \in X$*, then* $\exists v' \in \bigcap N_2(s_2)$ *s.t.* $v_1 E v'$*. This contradicts the assumption that there is no such* v'*.*

Case (b): analogous, but with a contradiction with the (zig) condition.

Both cases generate a contradiction, therefore $\bigcap N_1(s_1)$ *and* $\bigcap N_2(s_2)$ *must be set bisimilar.* □

We can now prove that trust bisimilar models satisfy the same formulas.

Theorem 10 (Invariance Under Trust Bisimulation)**.** *Let* $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$ *and* $M'_{C'} = (S', C', \pi', N', \mathcal{T}', \theta')$ *be contextual trust models. If* $Z \subseteq S \times S'$ *is a trust bisimulation between* M_C *and* $M'_{C'}$*, then, for each formula* ϕ *and* $s \in S$*,* $s' \in S'$ *s.t.* sZs'*, with* $C = C'$*, and* $\theta = \theta'$*, we have:*

$$(M, s, c) \models \phi \text{ } \textit{iff} \text{ } (M', s', c) \models \phi.$$

Proof. Both directions by induction on the structure of the formula.

Base case. $\phi = p$, for $p \in At$. Assume $(M, s, c) \models p$, then, by the (prop) condition of trust bisimulation, $(M', s', c) \models p$. The proof is symmetrical in the other direction.

Inductive hypothesis (IH)

$$(M, s, c) \models \phi \text{ iff } (M', s', c) \models \phi.$$

Inductive steps:

Negation: Assume $(M, s, c) \models \neg\phi$. By the truth definition of negation, $s \in \pi^{ext}(\neg\phi)$. By the definition of truth set, $s \in (S - \pi^{ext}(\phi))$. By the definition of set difference, $s \in S$ and $s \notin \pi^{ext}(\phi)$. By IH, $s' \notin \pi'^{ext}(\phi)$. By assumption, $s' \in S'$. By the definition of set difference, $s' \in (S' - \pi'^{ext}(\phi))$. By the definition of truth set, $s' \in (S' - \pi'^{ext}(\phi))$. By the truth definition of negation, $s' \in \pi'^{ext}(\neg\phi)$. Therefore $(M', s', c) \models \neg\phi$. The proof is symmetrical in the other direction.

Conjunction: Assume $(M, s, c) \models \phi \wedge \psi$. By the truth definition of conjunction $s \in \pi^{ext}(\phi \wedge \psi)$. By the definition of truth set, $s \in \pi^{ext}(\phi) \cap \pi^{ext}(\psi)$. By intersection definition, $s \in \pi^{ext}(\phi)$ and $s \in \pi^{ext}(\psi)$. By IH, $s' \in \pi'^{ext}(\phi)$ and $s' \in \pi'^{ext}(\psi)$. By intersection definition $s' \in \pi'^{ext}(\phi) \cap s' \in \pi'^{ext}(\psi)$. By definition of truth set, $s' \in \pi'^{ext}(\phi \wedge \psi)$. Therefore $(M', s', c) \models \phi \wedge \psi$. The proof is symmetrical in the other direction.

Knowledge Formulas: (Left to right) Assume $(M, s, c) \models K(\phi)$. By the truth definition of knowledge formulas $s \in \pi^{ext}(K(\phi))$. By the definition of π^{ext}, $s \in m_N(\pi^{ext}(\phi))$. By the definition of m_N, $\pi^{ext}(\phi) \in N(s)$. Call $\pi^{ext}(\phi)$, A. By the (zig) condition of trust bisimulation, there exists a $A' \in N'(s')$ and, for all $t' \in A'$, there exists a $t \in A$ s.t. tZt'. Note now that **IH** could be rephrased has:

$$s \in \pi^{ext}(\phi) \text{ iff } s' \in \pi'^{ext}(\phi).$$

It therefore follows that, by **IH**, $A' \subseteq \pi'^{ext}(\phi)$. Since $A' \in N'(s')$ and N' is monotonic, $\pi'^{ext}(\phi) \in N'(s')$. By the definition of m'_N, $s' \in m'_N(\pi'^{ext}(\phi))$. By the definition of π'^{ext}, $s' \in \pi'^{ext}(K(\phi))$. By the truth definition of knowledge formulas, $(M', s', c) \models K(\phi)$.

(Right to left) The proof is symmetrical, but employing the (zag) condition instead of the (zig) condition.

Trust Formulas: (Left to right) Assume $(M, s, c) \models T(\phi)$. By the truth definition of trust formulas $\tau^{ext}_{c,\phi}(\bigcap_{X \in N(s)} X) \geq \theta(c, \phi)$. Recall that we assumed that, when evaluating trust formulas in two models, the context of evaluation and the trustworthiness threshold is the same for both evaluations, therefore $\theta(c, \phi) = \theta'(c', \phi)$. This said, we must prove that, with sZs':

$$\tau^{ext}_{c,\phi}(\textstyle\bigcap_{X \in N(s)} X) = \tau'^{ext}_{c,\phi}(\textstyle\bigcap_{X' \in N'(s')} X')^{17}$$

Recall that the definition of $\tau^{ext}_{c,\phi}(X)$ is the following: $min_{t \in X}\{\tau_{c,\phi}(t)\}$.

Thus what must be proven is that the minimum value of $\tau_{c,\phi}(t)$ among all the $t \in \bigcap N(s)$ is equal to the minimum value of $\tau'_{c,\phi}(t')$ among all the $t' \in \bigcap N'(s')$. By theorem 9, $\bigcap N(s)$ and $\bigcap N'(s')$ are set bisimilar. Now take the $t \in \bigcap N(s)$ s.t. $\tau_{c,\phi}(t)$ is the minimum. By set bisimilarity, there is a

[17] The latest statement "being equal to" could be substituted with "being less than or equal to". Nonetheless, by proving the equivalence, there is an implicit proof of the right-to-left direction inclueded in the left-to-right direction.

$t' \in \bigcap N'(s')$ s.t. tEt'. By lemma 6, $\tau_{c,\phi}(t) = \tau'_{c,\phi}(t')$. Moreover, $\tau'_{c,\phi}(t')$ is the minimum value of the set $\bigcap N'(s')$. Suppose it is not. Then there is an element in $\bigcap N'(s')$ with a lower value and which is, by a reverse reasoning, in a bisimilarity relation with a element in $\bigcap N(s)$ whose $\tau_{c,\phi}$ are equal. Thus, this would contradict the fact that $\tau_{c,\phi}(t)$ is the minimum value. □

□

We can finally give the proof of the model equivalence decidability problem.

Theorem 11 (Model Equivalence for C-MLT)**.** *The model equivalence problem for Contextual Modal Logic for Trust is decidable.*

Proof. Given two models $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$ and $M'_{C'} = (S', C', \pi', N', \mathcal{T}', \theta')$ check the following:

- (Context equivalence) $C = C'$.
- (Trustworthiness threshold equivalence) $\theta = \theta'$.
- (Trust bisimulation existence) Whether it exists a trust bisimulation between M_C and $M'_{C'}$

If all points provide positive answers, then, by theorem 10, the two models satisfy the same formulas and are therefore equivalent. If any of the previous points fails to hold, then the two models are distinct. In both cases, you obtain an answer to the model equivalence problem.

□

Finally, the satisfiability decision problem. One important thing to notice is that trust formulas are always satisfiable, therefore, they can't be valid in any class of models. This should be expected, given the highly subjective nature of trust. In fact, whatever the proposition to be trusted might be, it is always possible to construct a suitable model that satisfies the trust formula containing, under its scope, the proposition to be trusted.

Lemma 7. *Every trust formula is satisfied by at least one model.*

Proof. Take an arbitrary trust formula $T(\phi)$. Now construct a model in the following way:

- Take an arbitrary set S of states.
- Take an arbitrary set C of contexts.
- Take an arbitrary function π.
- Take a function N s.t. $\exists s \in S$ s.t. $\bigcap_{X \in N(s)} X \neq \emptyset$.
- Take $\mathcal{T}$ s.t. $\omega_c(\phi) = \{S\}$ and $\mu_{c,\phi}(S) = 1$.

- Take an arbitrary θ.

In such a model, the trust formula $T(\phi)$ is satisfied in all states s.t. $\bigcap_{X \in N(s)} X \neq \emptyset$. This is because the τ of each state is 1 and once we have a non-empty set of states X on which to take the minimum, we will always end up with the value 1. Therefore, we always end up with full trust in the formula ϕ, satisfying the formula no matter what the value of the trustworthiness threshold is.

□

Given lemma 7, the satisfiability decision problem for the language depends only on whether it is possible to decide, given a formula of the other typologies, if they are satisfiable by some model. In this sense, note that in (contextual) trust logic all models are finite, therefore, if a formula is satisfiable in the language, it must be satisfied by a finite model. This property is the so called finite model property, which, in the language here presented, is forced on the formulas. This property, even if imposed, is of extreme importance when dealing with the satisfiability decision problem. However, this property alone is not sufficient, because a check on an infinite number of finite models might still be needed. It is necessary, thus, to provide an effective upper bound to the size of the models that might satisfy a given formula. A proof that such a bound exists for monotonic neighborhood models can be found in [100] and its bound can also be applied to augmented neighbourhood models, given the fact that those models are indeed monotonic. However, the path followed in this thesis is slightly different and will rely on the fact that there is strict correspondence between relational structures and augmented neighbourhood models. Therefore, the proof of the effective bound on the size of a model satisfying a formula of the language is given in terms of relational models. Note that determining satisfiability will become the task of finding the adequate relational model satisfying the formula and then translate such model into an augmented neighbourhood model through the procedure described in lemma 4.

The theorem that can help with satisfiability is that of the effective finite model property, where this property states that a satisfiable formula is satisfied in a finite model of a certain size (dependent on the size of the formula under analysis).

Theorem 12 (Effective Finite Model Property)**.** *Modal Logic has the effective finite model property.*

Before producing the proof of the theorem, some important variations on definitions that were provided for neighbourhood structures will now be provided for relational structures.

Definition 39 (Relational Bisimulation)**.** *Given two relational models $\mathfrak{M} = \langle S, R, V \rangle$ and $\mathfrak{M}' = \langle S', R', V' \rangle$ a binary relation $E \subseteq S \times S'$ is a bisimulation between two states $s \in S$ and $s' \in S'$, indicated with sEs', if:*

1. *s and s' satisfy the same unanalysed propositions $p \in At$.*
2. *If sRv in $\mathfrak{M}$, then $\exists v' \in \mathfrak{M}'$ s.t. $s'Rv'$ and vEv'.*
3. *If $s'Rv'$ in $\mathfrak{M}'$, then $\exists v \in \mathfrak{M}$ s.t. sRv and vEv'.*

Definition 40 (Relational Tree Unravelling)**.** *Every relational model $\mathfrak{M} = \langle S, R, V\rangle$ has a bisimulation with a rooted tree-like model. The tree-like model is constructed as follows. The set of states of the model are all finite path of states $s \in S$, starting with a specific root root and passing only to R-successors at each step. The relation R holds between two paths if the second is one step longer than the first. Valuations are equal in both models.*

Proof (Proof of Theorem 12)**.** *Take and arbitrary non-trust formula ϕ which can be satisfied in a relational model $\mathfrak{M} = \langle S, R, V\rangle$. Unravel $\mathfrak{M}$ through relational tree unravelling, s.t. ϕ holds at root. To prove the effective finite model property, it will be shown that the evaluation of the formula only requires finite path depth and finite branching width.*

To prove it, transform, through equivalences, the formula into a Boolean combination of unanalysed propositions and modal formulas defined through $\widehat{K}$. Unanalysed propositions only depend on the valuation function and therefore can be established right away. For the modal part, for every true $\widehat{K}$, choose a verifying successor state in the model for such $\widehat{K}$. The total number of successors to be chosen is bounded by the number of elements in the Boolean transformation of the formula ϕ, therefore there is a finite branching width. For false $\widehat{K}$, no successor needs to be chosen. Note that, at each step, a level of modal operators is lost, therefore, the depth of the path to follow is restricted to the maximum among all the numbers of operators elements of the Boolean combination equal to ϕ have. This sets a finite path depth to the model. Combining the finite width with the finite depth, it is possible to obtain a bound on the size of the model satisfying the formula. □

From theorem 12 and lemma 7, the satisfiability result follows.

Theorem 13 (Satisfiability Problem for MLfT)**.** *The satisfiability problem for Modal Logic for Trust is decidable.*

Proof (Proof ot Theorem 13)**.** *By lemma 7 it is known that every trust formula is satisfiable in at least one model. By theorem 12 it is known that, if a non-trust formula is satisfied in a relational model, then it is satisfied in a effectively bounded finite relational model. Check all models (up to bisimulation) of such size, if it is found, then transform such model into an augmented neighbourhood model through the procedure described in lemma 4. The resulting augmented neighbourhood model also satisfies the formula and thus is answer to the satisfiability problem is positive. If there is no such relational model, then there is no augmented neighbourhood model that satisfies the formula. Assume there was, then, by the procedure described in lemma 5, it would* *be possible to construct a relational model in which the formula would be satisfied, contradicting the assumption. Therefore, in such a case the answer to the satisfiability problem is negative. In either case an answer is obtained.*

□

This chapter is concluded with a brief reference to possible proof theories for trust logics. A specific proof theory for the language here presented is lacking and is part of a work in progress. It should be noted, however, that, as already mentioned earlier in the text, there are no trust formulas valid in the semantics just proposed. This is due to the highly subjective nature of trust and by the fact that it is hardly possible to find truisms about such concept. Therefore, any proof theory that captures the validities concerning the modal part of the language will suffice as a proof theory for the current language, where trust formulas will appear only as substitutions inside tautologies. In this sense, the most appropriate axiomatic system for the language here presented is the one containing the distributivity axiom over the modal operator. Nonetheless, it is important to mention that there are many axiomatic systems employed to formalize formal languages that contain operators for trust [11, 26, 77]. While sound and complete and useful in reasoning about trust in general, such system are, nevertheless, faulty isasfar as they model trust directly, without making use of the insight that trust often depends on other information.

Chapter 5

Frameworks for Trust Reasoning

In this chapter, other theoretical model employed to represent trust and uncertainty are proposed. Differently from chapter 3, where the focus was on practical models for trust, here the focus will be on omnicomprehensive theoretical frameworks for trust. The idea is to check how the logical language for trust presented in chapter 4 behaves in comparison to those other theoretical frameworks. Specifically, the frameworks analysed will be Subjective Logic [59] and Dempster-Shafer Theory of Evidence [29, 116]. Those frameworks and characterizing elements of those will be introduced. Weaknesses and strengths of each of them are going to be highlighted and theorems about correspondences between their formal structures and the semantical structure of C-MLT will be proven. Once the correspondences are built, it is argued that those formalisms might benefit by considering procedures operated inside C-MLT. The chapter is structured as follows: in section one, Subjective Logic is discussed and a correspondence between it and C-MLT is built; in section two, Dempster-Shafer Theory of Evidence is discussed and a correspondence between it and C-MLT is built.

5.1 Subjective Logic

In this section, Jøsang's Subjective Logic [59] (SL) will be introduced and then it will be shown how the semantical structure of C-MLT can be employed to compute trust values (in the form of opinions) to be fed into the models of SL. It will also be explained why this specific model was chosen and why it is believed that building a correspondence between contextual trust models and the models of SL is beneficial to both. As a side result in the analysis of the comparison between the two formalisms, the difference between trust computing and trust manipulation models for trust will be further clarified (recall that the distinction was introduced in chapter 3.

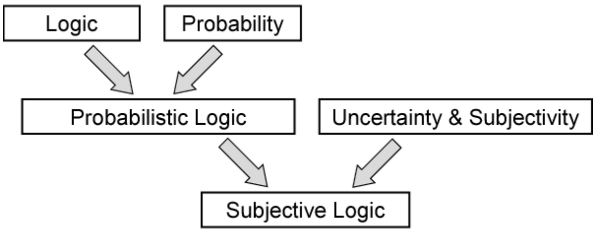


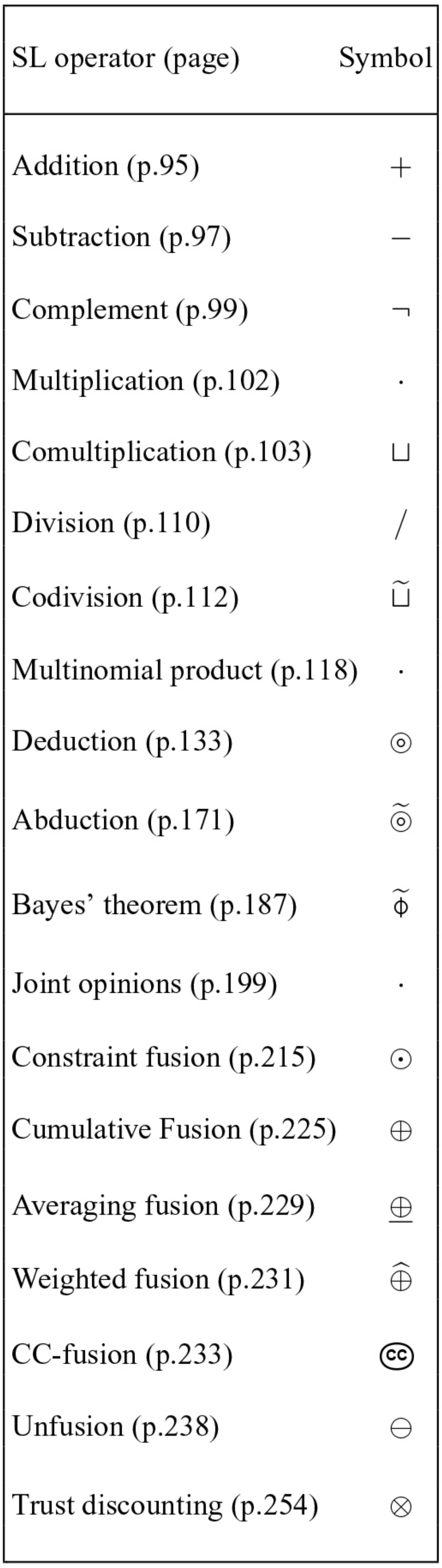
Figure 5.1: Genesis of Subjective Logic.

SL is a formal framework for artificial reasoning based on the Beta paradigm. In general, Subjective Logic can be thought of as an extension of probabilistic logic (see figure 5.2). The extension is obtained by introducing, in a standard probabilistic logic, two new elements: one of uncertainty and one of subjectivity. The former, which should be seen as a second-order probability measurement, helps in evaluating what is the likelihood on obtaining a certain level of probability for specific events. The latter, is employed to attribute beliefs (or estimates) to specific agents, rather than having anonymous evaluations valid for all. The fact that SL represents an extension of probabilistic logic is also proven formally, through the construction of an homorphism between a suitable version of the former (i.e., one in which only dogmatic multi-nomial opinions are considered) and the latter. To understand why uncertainty might play a role in real life examples, take the following example: two urns are placed in front of an agent, who has to decide from which urn he wants to extract a ball. Each urn contain 100 balls, which are either red or black. The agent will receive a prize if he extracts a red ball and nothing if he extracts a black one. He is told that in urn one, there are exactly 50 red ball and 50 black ones, therefore he knows that the probability of extracting a red ball is 1/2. He receives no information about the second urn. By the principle of indifference [69], the agent should also assign a 1/2 probability of extracting a red ball to urn number two. However, as it is evident, the two probabilities, while equal, represent two distinct forms of information. In the first case, the probability is assigned with certainty, while in the second one, the probability is assigned with maximum uncertainty. In a standard probabilistic language, this distinction can't be expressed, but in SL, the expressivity is sufficient to capture the subtleties behind this example.

Formally, SL consists of a belief model, whose elements are called opinions, and a set of algebraic operations defined on this model for combining opinions in different ways. Possible operations are addition, subtraction and fusion of beliefs[1].

Concerning computational trust, in SL trust is represented as the opinion

[1] In fact, there are at least nineteen different operators which have been well studied in SL. See figure 5.2 for the complete list, with reference to the pages where they are discussed in [59].



SL operator (page)	Symbol
Addition (p.95)	$+$
Subtraction (p.97)	$-$
Complement (p.99)	$\neg$
Multiplication (p.102)	$\cdot$
Comultiplication (p.103)	$\sqcup$
Division (p.110)	$/$
Codivision (p.112)	$\widetilde{\sqcup}$
Multinomial product (p.118)	$\cdot$
Deduction (p.133)	$\circledcirc$
Abduction (p.171)	$\widetilde{\circledcirc}$
Bayes' theorem (p.187)	$\widetilde{\phi}$
Joint opinions (p.199)	$\cdot$
Constraint fusion (p.215)	$\odot$
Cumulative Fusion (p.225)	$\oplus$
Averaging fusion (p.229)	$\underline{\oplus}$
Weighted fusion (p.231)	$\widehat{\oplus}$
CC-fusion (p.233)	🅭
Unfusion (p.238)	$\ominus$
Trust discounting (p.254)	$\otimes$

Figure 5.2: Subjective Logic Operators.

of an agent x about the truth of a given proposition p [65]. The propositions on which opinions range, express sentences which describe collaborative frameworks. For example, a proposition p might express the following sentence: "Agent y won't defect the partnership in the next month". Specifically, an opinion has three components, plus a fourth optional component, which, however, is fundamental to compute expected trust in the truth of the proposition. The three major components are, respectively, a belief component b, a disbelief component d and an uncertainty component u, while the fourth component (a) is defined as the base rate and indicates the prior probability associated with the truth of a proposition when no initial relevant information is available, i.e., the base rate represents a first estimate of the plausibility of the truth of the proposition. The belief, disbelief and uncertainty components are additive to one, leading to the fact that SL is effectively an extension of traditional probabilistic logics (if the uncertainty component is assumed to be equal to zero, SL becomes a traditional probability model). This also determines the fact that an opinion has two levels of freedom, where the third component is always determined by the values of the first two components defined. The additivity principle of the three major components allows also for a nice visualization of opinions through a triangle, which is called *opinion triangle*[2], see Fig. 5.3. It is possible to observe in the figure that an opinion ω is identified through the three major components of belief, disbelief and uncertainty and, after a generic opinion is obtained, it is possible to compute the expected trust value $E(\omega)$ using the base rate: the base rate determines the slope of the projection of the opinion on the base of the triangle and allows to compute an expected value when uncertainty is assumed to be zero.

Being based on the Beta paradigm, the core of the theory of SL is given in terms of the Beta function. Recall that the Beta function is employed to compute expected values for the likelihood of a given event of having a certain probability.

$$f(p \mid \alpha, \beta) = \tfrac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} p^{\alpha-1}(1-p)^{\beta-1}$$

In SL only a subset of the Beta functions are considered, i.e., probability certainty density functions, whose parameters are taken to be the successful interactions (r) and the failures (s), where the relation between the two parameters of the beta functions (α, β) and the two parameters (r, s) is the following:

$$\alpha = r + 1 \text{ s.t. } r \geq 0 \ \beta = s + 1 \text{ s.t. } s \geq 0$$

[2] This visualization works fine until we deal with trinomial opinions (which correspond, visually, to a tetrahedron). In this thesis, only binomial opinions are important, since it is assumed that propositions can only be either true or false and nothing inbetween those values. Therefore, the standard opinion triangle is sufficient as a visual aid of the opinion components. See [59] (Section 3.5) for a discussion on what a multinomial opinion represents.

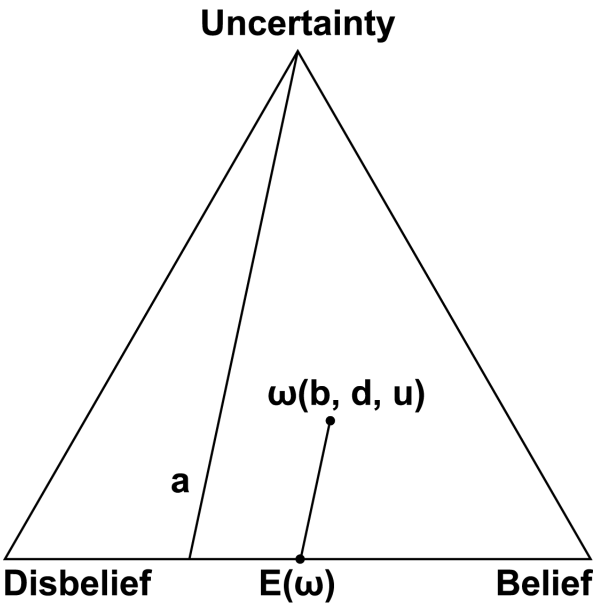


Figure 5.3: Opinion triangle

Through those equivalences and the Beta function results, the different components of an opinion are defined as follows:

$$b_p = \frac{r_p}{r_p+s_p+2};\ d_p = \frac{s_p}{r_p+s_p+2};\ u_p = \frac{1}{r_p+s_p+2}.$$

Where the denominators are employed to normalize the values obtained by counting the successful and failed interactions.

SL is a widely employed model to manipulate trust and the reason for this is that the many algebraic operators that are included in the model can describe numerous interacting situations and precisely indicate the dynamics of trust in all those situations. However, SL is rather ill-suited when it comes to compute initial trust values to be used as inputs to the model. The reason is that the only source of information that SL allows to compute trust values is a set of reputation scores based on past interactions, i.e., the success or failure of those interactions. Once it is noticed that different agents might evaluate interactions differently (therefore, that which is a success for someone, might be a failure for someone else and viceversa) and that reputation scores in one context are not easily transferable to another context (being successful in an interaction involving selling cars, might have no impact in the possibility of success when it comes to prepare meals for dinner), the fact that SL has no other means to compute initial trust values becomes a big drawback. This is also noted by Jøsang himself [65]:

> “The major difficulty with applying SL is to find a way to consistently determine opinions to be used as input parameters. People

> may find the opinion model unfamiliar, and different individuals may produce conflicting opinions when faced with the same evidence."

The aim of this section is specifically to show that contextual trust models can be employed as a trust computing component that can produce initial trust values which can then be plugged into SL, effectively improving the model. The leading idea is that, in order to have an effective formalization of the notion of trust, a trust model must accomplish two basic goals. The first goal is that of providing a reduction of the notion of trust to more basic notions that can be known by an agent. This is necessary to explain how trust is generated in different contexts. The second goal is that of explaining the dynamics of trust, i.e. how trust evolves under different circumstances. To each goal corresponds a different component of trust models. Specifically, it is possible to identify a *trust computing* and a *trust manipulation* component. The former serves the purpose of gathering relevant information and then use it to compute initial trust values; the latter takes the initial trust values as granted and manipulates them for specific purposes using different operators. As said, SL possesses an effective trust manipulation component, but lacks an effective trust computing component[3]. In the next following subsection, a mapping between contextual trust models and opinions in SL is built. What will become evident is that contextual trust models allow a modeller to specify initial trust values for SL which can be based on any kind of information (as long as it is expressible as a proposition in a logical language) both quantitative and qualitative.

5.1.1 Pre-Trust Computations for SL

The aim of a pre-trust computation is to obtain the three distinct components of SL opinions. Such components are, respectively, belief, disbelief, and uncertainty (base rate values are assumed to be known, as it happens in SL). It will now be shown that obtaining those three components is straightforward starting from contextual trust models, once all the semantical components of the models are explicitly given. The three opinion components are explicitly specified as: "agent i believes in proposition p" (symbolically $b_i(p)$) means that agent i, the trustor, believes, to a given degree, in the truth of proposition p; "agent i disbelieves in proposition p" (symbolically $d_i(p)$) means that agent i disbelieves, to a given degree, in the truth of proposition p; finally, "agent i is uncertain about proposition p" (symbolically $u_i(p)$) means that agent i does not possess any relevant information on whether to trust or not the proposition

[3] See [86] for an example of a computational trust model that has an effective trust computing component, but a poor trust manipulation component. Other examples of computational trust models where the two components are easily identifiable are [101] and [135]. See also [7, 22, 43, 101, 113, 94, 123] for surveys on computational trust models.

p. Note that the three components are agent-specific, while in the trust logical languages introduced in the previous chapter, there is no specific reference to different agents. This issue is easily avoidable by constructing an appropriate model for each agent (or by building a more powerful multi-agent language). Note also that in the language, the quantity of information is given by the cardinality of the set attributed by π^{extT} to formulas. Specifically, the smaller the cardinality (the fewer states are present in the set), the higher the amount of information possessed (i.e., more propositions are taken into consideration). Given those intuitions, it is possible to proceed with the computation of the components.

The first step is that of the selection of a state s used as vantage point. The N function is then checked for the state selected: this will specify what the agent knows, i.e., the information the agent possesses. The set of states compatible with the knowledge of the agent, i.e., the core of N, is then selected. At this point the only thing that is needed is to compute the $\tau_{c,\phi}(s)$, such that $s \in (\bigcap N(s))$. The three opinion components are then obtained by taking:

- for $b_i(p)$, the minimum among all the $\tau_{c,\phi}(s)$ (recall that this is the meaning of trust in C-MLT);
- for $d_i(p)$, 1 minus the maximum among all the $\tau_{c,\phi}(s)$;
- for $u_i(p)$, the difference between the maximum and the minimum.

Intuitively, the value $b_i(p)$ is obtained by looking at how much trust is present in the worse possible state compatible with the knowledge of the agent. The value $d_i(p)$ is obtained by looking how much trust is lost in the best possible state compatible with the knowledge of the agent. Finally, $u_i(p)$ is obtained by looking at how much information can still be achieved.

Formally, the mapping works in the following way:

Definition 41. *Given a contextual trust model $M_C = (S, C, \pi, N, \mathcal{T}, \theta)$, a state $s \in S$ and a context $c \in C$. An opinion $(b_i(p), d_i(p), u_i(p))$ of Subjective Logic, where the proposition p is evaluated at c and is expressible in C-MLT as formula ϕ is obtained in the following way:*

- $b_i(p) = min\{\tau_{c,\phi}(t) \mid t \in \bigcap N(s)\}$.
- $d_i(p) = 1 - max\{\tau_{c,\phi}(t) \mid t \in \bigcap N(s)\}$.
- $u_i(p) = max\{\tau_{c,\phi}(t) \mid t \in \bigcap N(s)\} - min\{\tau_{c,\phi}(t) \mid t \in \bigcap N(s)\}$.

It is important to note that there is no actual way in C-MLT to represent agency. Unfortunately, the language developed in this thesis is still single agent and therefore the mapping from trust values in C-MLT and opinions in SL is only partially defined. In fact, at this level of generality (i.e., before the language is expanded to cover multi-agent scenarios), one must build a specific contextual trust model for each agent that has to be described. However, since it is fairly reasonable to imagine a multi-agent extension of the language, this issue can be solved without much troubles. Once a multi-agent version of the language is provided, the mapping from trust values in C-MLT and opinions in SL can be fully established.

The advantage of moving from C-MLT to SL is that, given its nature, in C-MLT basic trust values can be computed starting from a wide variety of information. The only limit is the expressivity of such information in propositional terms. Provided that, for human beings, most knowledge is represented propositionally, it is clear that many typologies of information can be used in the language to obtain trust values. This allows for a great amount of freedom in the modelling of trust computing algorithms. On the other side, once the initial trust values are obtained, it is advisable to plot them in a trust manipulation model which is efficient in transforming and combining those trust values. Manipulating trust at the syntactical and/or semantical level in the language might turn out to be troublesome, while manipulations in SL are straightforward and easy to perform. This should explain why SL might need C-MLT and, on the reverse, why C-MLT can benefit from being able to translate its trust values into opinions in SL.

5.2 Dempster-Shafer Theory of Evidence

In this section, Dempster-Shafer Theory of Evidence [29, 116] (DSTE) will be briefly introduced and then it will be shown how to move from contextual trust models in C-MLT to Belief and Plausibility functions in DSTE. The treatment of DSTE is based on the presentation of the theory given in [45], where, however, most theorems are left as exercises. Those theorems will be proved explicitly and new definitions useful for C-MLT will be introduced. In particular, emphasis will be put on operations on the trust structures of contextual trust models. Those operations, which help in drawing the parallel with DSTE, can also allow reflections on possible properties to impose on trust structures to obtain determinate trust concepts in C-MLT.

DSTE is a logical framework designed to interpret and model the likelihood of events. The core idea of DSTE is to use two functions, namely a belief and a plausibility function, to express how evidence influences the likelihood of specific possible outcomes. The base on which DSTE works is a set of possible worlds (those should be thought of as the state of the system introduced for C-MLT, therefore, maximally consistent descriptions of possible ways the world might be) W. Events are then indicated with subsets of W; a set $U \subseteq W$ represents an event in the sense that in all worlds (states) included in U a give event takes place. For example, the event of Mirko Tagliaferri being blonde, identifies a subset U of W, specifically, the subset U s.t. $\forall w \in U$, Mirko

Tagliaferri is blonde in w. Informally, given an event U, the two functions $Bel(U)$ and $Plaus(U)$, represent, respectively, the amount of support someone has in favor of the hypothesis that event U is true (i.e., the real world is in U) and the amount of support someone has against the hypothesis that event U is true (i.e., the real world is not in U).

Formally, $Bel(U)$ is a function assigning numerical values to subsets of W, where the numbers that are assigned range over the set $[0,1]$ of real numbers, i.e. $Bel : \wp(W) \rightarrow [0,1] \in \mathbb{R}$. The properties that the function Bel must satisfy are the following:

1. $Bel(\emptyset) = 0$;
2. $Bel(W) = 1$;
3. $Bel(\bigcup_{i=1}^{n} U_i) \geq \sum_{i=1}^{n} \sum_{\{I \subseteq \{1,\ldots,n\} : |I| = i\}} (-1)^{i+1} Bel(\bigcap_{j \in I} U_j)$.

It can be said that belief functions provide a lower bound to the likelihood of events.

On the other hand, $Plaus(U)$ is just like $Bel(U)$, in that it assigns numerical values to subsets of W, i.e. $Plaus : \wp(W) \rightarrow [0,1] \in \mathbb{R}$. However, $Plau$ is defined often defined in terms of Bel. Therefore, to compute $Plaus$, the following formula is employed:

$$Plaus(U) = 1 - Bel(\bar{U})$$

Given this relation, the properties defining $Plaus$ are:

1. $Plaus(\emptyset) = 0$;
2. $Plaus(W) = 1$;
3. $Bel(\bigcap_{i=1}^{n} U_i) \geq \sum_{i=1}^{n} \sum_{\{I \subseteq \{1,\ldots,n\} : |I| = i\}} (-1)^{i+1} Plaus(\bigcup_{j \in I} U_j)$.

It can be said that plausibility functions provide a upper bound to the likelihood of events.

Other ways of seeing Bel and $Plaus$ functions is to interpret them inside a theory of evidence, where evidence provides different degrees of support to different subsets of W. In this sense, a belief function is nothing more than the sum of all those evidences supporting a specific subset of W. This is expressed formally employing *mass functions*, where a mass function m is, again, a function assigning numerical values to subsets of W, i.e., $m : \wp(W) \rightarrow [0,1] \in \mathbb{R}$. The properties a mass function must satisfy are the following:

1. $m(\emptyset) = 0$;
2. $\sum_{U \subseteq W} m(U) = 1$.

Mass functions play an important role in DSTE and they are even more important for the purpose of this section, since they will be at the base of the correspondence between DSTE and C-MLT. In particular, it will be shown that specific classes of contextual trust models, based on precise trust structures, can be interpreted as being mass functions in DSTE.

In DSTE, it is possible to prove (see [116]) that beliefs and plausibility functions can be characterized in terms of mass functions and, moreover, that, for any belief (plausibility) function, there is a unique mass function characterizing it. To produce a belief function starting from a mass function, the following formula is employed:

$$Bel_m(U) = \sum_{[U':U' \subseteq U]} m(U') \tag{5.1}$$

As it is possible to notice, a belief function over U is obtained by summing up the probabilities of the evidence of sets which confirm U or more specific instances which are contained in U. To produce a plausibility function starting from a mass function, the following formula is employed:

$$Plaus_m(U) = \sum_{U':U' \cap U \neq \emptyset} m(U') \tag{5.2}$$

Plausibility functions over U are obtained by summing up the probabilities of the evidence of sets which are compatible with U.

With those definitions in hand, in DSTE there are different rules to manipulate belief, plausibility and mass functions. Among them, the most important is that of combination, which permits to combine different sets of evidence in order to obtain a unified set which indicates how likely is the event consistent with all the evidences obtained. No details will be given about the combination rule, but the reader should understand that part of the strength of DSTE is contained in the way evidence is represented and the possibility of combining it properly. This is what promoted DSTE at the top of the possible theories for the representation of uncertainty and beliefs.

5.2.1 From Trust Values to Beliefs

The aim is that of generating a belief function (and, accordingly, a plausibility function) starting from the resources of contextual trust models. By looking at the structure of mass functions, it is easy to notice that they are extremely similar in spirit to μ functions in contextual trust models and thus it is advisable to work on those functions to obtain the correspondence. One issue which prohibits to establish the correspondence directly is that μ functions are subadditive to 1, where mass functions are additive to one. To solve this problem, the definition of extensions of trust structures is required.

Definition 42. *A trust structure* $\mathcal{T}' = \{\langle \omega'_{c'}, \mu'_{c',\phi} \rangle \mid c' \in C'$ *and* $\phi \in \mathcal{L}\}$ *is* *said to be* ***an extension*** *of a trust structure* $\mathcal{T} = \{\langle \omega_c, \mu_{c,\phi} \rangle \mid c \in C$ *and* $\phi \in \mathcal{L}\}$ *if:*

- $c = c'$;
- $\omega_c(\phi) \subseteq \omega'_{c'}(\phi)$, $\forall \phi \in \mathcal{L}$;
- *If* $X \in \omega_c(\phi)$, *then* $\mu_{c,\phi}(X) = \mu'_{c',\phi}(X)$.

An important aspects of extended trust structures is that they are monotonic with respect to which trust formulas they allow to be satisfied. Therefore, if a trust structure allows a trust formula $T(\phi)$ to be satisfied in a specific contextual trust model, then, all extensions of such trust structure, will allow $T(\phi)$ to be satisfied in the same contextual trust model[4].

Even though useful for other reasons (e.g., studying possible limitations, based on extensions, to place on trust structures in order to obtain specific conceptions of trust), the concept of extension alone is not sufficient to derive mass functions. In order to obtain such derivation, a special class of trust structures extensions are required. This class is now defined:

Definition 43. *A trust structure* $\mathcal{T}' = \{\langle \omega'_{c'}, \mu'_{c',\phi} \rangle \mid c' \in C'$ *and* $\phi \in \mathcal{L}\}$ *is said to be* ***an additive extension*** *of a trust structure* $\mathcal{T} = \{\langle \omega_c, \mu_{c,\phi} \rangle \mid c \in C$ *and* $\phi \in \mathcal{L}\}$ *if:*

- $\mathcal{T}'$ *is an extension of* $\mathcal{T}$;
- $\sum_{X \in \omega_c(\phi)} \mu_{c,\phi}(X) = 1$, $\forall \phi \in \mathcal{L}$.

Thus, an additive extension of a trust structure is simply an extension for which the μ functions are additive to one. Note that, using the term improperly, additive extensions of trust structures are fixed points under the extension operation. This is because all extensions of an additive structure can only differ in the elements contained in $\omega_c(\phi)$, which, however, will receive a relevance weight of 0. This means that such extensions would not improve the amount of trust formulas satisfiable in the contextual trust model. The additive extensions of a trust structure are yet not sufficient to properly define mass functions; one last step to make is to set the values of all other $X \in \wp(S)$ s.t. $X \notin \omega_c(\phi)$ equal to 0. This is required because mass functions are defined over the whole powerset of the initial set of possible states and therefore, the same must hold for $\mu_{c,\phi}$ if the goal is to construct a bridge between the two. In particular, the $\mu_{c,\phi}$ for which this further condition has been imposed will be indicated with $\mu^{mass}_{c,\phi}$

[4] It is said "allow to satisfy", rather than simply "satisfy", because a formula is satisfied by a contextual pointed trust model and not by a trust structure *per se*. However, it is evident that trust structures play an important role in the possibility of satisfying a formula.

Theorem 14. *Given a trust structure $\mathcal{T} = \{\langle \omega_c, \mu_{c,\phi} \rangle \mid c \in C \text{ and } \phi \in \mathcal{L}\}$, a proposition ϕ, and a context of evaluation c, the function $\mu'^{mass}_{c,\phi}$ is a mass function in DSTE terms.*

Proof (Proof of Theorem 14)**.** *By the consistency definition of ω_c, the empty set is never contained in any extension (additive or not) of a trust structure. Therefore, by the definition of $\mu'^{mass}_{c,\phi}$, such set will receive the value 0. This proves the first condition on mass functions.*

By the fact that $\mu'^{mass}_{c,\phi}$ derives from an additive extension of a trust structure, $\sum_{X \in \omega_c(\phi)} \mu'^{mass}_{c,\phi}(X) = 1$, thus the second condition on mass functions is proved.

By the fact that for $\mu'^{mass}_{c,\phi}$ both conditions of mass functions hold, it is itself a mass function in DSTE terms. □

Theorem 14 allows to build a bridge between C-MLT and DSTE, by producing a mass function starting from trust structures. This can benefit DSTE because it allows the theory to build very peculiar mass functions. Normally, mass functions in DSTE take evidence as a general notion, one for which only the relevance for certain fact is assessed. The mass functions that originate from the procedure described above, on the other hand, describe a very specific typology of evidence, i.e., that of knowing that an agent trusts a given proposition. $\mu'^{mass}_{c,\phi}$ specifies how different sets of states might be influenced by the fact that $T(\phi)$ is satisfied in the model. Thus, those mass functions should be seen as a reverse inference on relevance, once it is established what is trusted. At that point, belief functions become ways of determining the belief of an agent that the real state of the system is one of a set of possible states. Given this reverse inference feature, this correspondence can also help C-MLT to design special algorithms that can support, once implemented in a digital environment, different kind of reasonings. Thus, the benefit is mutual.

Chapter 6

Conclusion

The aim of this thesis was that of building a formal language for trust, which could help reasoning about computational versions of trust. The plan was to construct such language taking into consideration both socio-economical and computational analyses of trust. This was due to the fact that it is believed, by the author, that a good conceptual comprehension of what trust might be is essential to a correct formalisation of the concept. Above this, understanding which phenomena might have contributed to the origins of trust might also provide benefits to new communities which hope to generate trusting attitudes in their specific environment. This is the case for computer science. The massive transition from face-to-face interactions to web-based interactions, created the need for new computational social attitudes which could mimic their biologically-dependent counterparts.

For all those reasons, after a brief introduction, in chapter 2, a proposal on the possible origins of trust was made, assessing it with respect to some experimental results and determining its plausibility from both a practical and a theoretical point-of-view. Trust was then analysed from a socio-economical perspective, in order to select the features that are considered fundamental by the respective communities. This helped in defining an initial set of characteristics of trust and to generate an initial working definition for it.

Given the scope of thesis, the next logical step was that of analysing the literature on computational trust, highlighting differences and similarities between the treatment trust receives in sociology/economy and the one it receives in computer science. In chapter 3 computational trust models where studied. A taxonomy useful in categorizing them was built and conceptions of trust of classical computational trust models were extracted and fused together to obtain a fundamental core for computational trust. The analysis embraced also some important theoretical frameworks, which helped in determining procedures of selection for assumptions to be made in digital environment in order to allow computational trust to be present. All this work made it possible to understand how computer scientists attempted to replicate trust in digital communities and gave interesting insights on some core features every formal definition of trust ought to possess to qualify as computational. Those insights constituted the base on which the logical language was built.

In chapter 4, the actual logical language for trust was built. After an introduction to modal logic, with emphasis on neighbourhood semantics, two versions of the language were proposed. One in which contexts were missing and the other where they were included. Decision procedure results were presented, thus showing characteristics which make the language appealing to computer scientists interested in practical implementations. The semantical structure used to interpret the language was thoroughly explored, highlighting all the philosophical reasons which produced the choices made.

Finally, in chapter 5, other theoretical frameworks employed to model trust in computer science and to reason about uncertainty were introduced. This allowed for the possibility of exploring formal frameworks which are already well-established and therefore, represent excellent examples of success of formalisation. After being introduced, pros and cons of those frameworks were explicitly shown. Correspondence theorems between the semantical structure for the language of chapter 4 and the formal frameworks were proven and it was discussed how such semantical structure can help to improve those frameworks. This showed that, even if still underdeveloped and novel in spirit, the logical language introduced in this thesis already possesses some interesting features, which can be employed, from the get-go, to implement a formal concept of trust and to understand the phenomenon of trusting better.

Along all the chapters, all the aims set forth in the introduction have been achieved.

6.1 Future Works

All the results presented in this thesis could be greatly improved. Among those, some interesting topics to explore are:

With respect to chapter 2.

1. Constructing some psychological experiment explicitly designed to test the proposal brought forwards about the origins of trust. Such experiments would revolve around reciprocal altruism and cheating, testing whether the total absence of cheating might induce trust to disappear.

2. Improve on the cross-disciplinary examination of trust. Currently, many research on trust is discipline-specific and most communities tend to ignore results obtained by the others. Few exception made [40, 110, 111], no multidisciplinary approach has been attempted with good success. This fact is unfortunate, because it is widely recognized that trust is a multi-faceted concept and therefore, it seem a prerequisite that any

good attempt at understanding trust, must involve an interdisciplinary analysis rather than an intradisciplinary one.

With respect to chapter 3.

1. Refine the proposed taxonomy, in order to include more aspects of trust models and thus, allowing a better categorization for them.

2. Create a repository of computational trust models, sorted according to their features. This would greatly improve the ability of computer scientists to navigate among all the possible implementations of computational trust. Moreover, having all the models sorted according to the features highlighted by the taxonomy, searching for a model suitable for highly specific purposes would be easy and fast.

3. Further refine the description of computational trust paradigms and provide an omnicomprehensive analysis of all the existing ones. This would enhance all future research on computational trust, allowing new researchers to identify the paradigm (and therefore the core assumptions) perfectly suited for the concept of trust they are trying to model.

With respect to chapter 4

1. Provide a proper axiomatic system for the language, identifying rules of inference for various notions of trust and see what has to be done at the semantical level to capture those slightly different notions of it. In the thesis, a possible mechanism to impose limitations on the trust structures has been introduced, however, no real results have been achieved employing it.

2. Provide the algebraic counterpart of the semantical structure. It is already known that the part relating to neighbourhood semantics is correspondent to coalgebraic structures, however, given the novelty of the trust structure part, no algebraic considerations have ever been made on those.

3. A better analysis of the complexity results for the decidability problems for the language. All decision procedures have been proven, but no real algorithms have been proposed for effectively carrying out such decision procedures. Therefore, lacking those algorithms, no clear complexity classes for those problems have been identified. Understanding how complex those procedures are can help understand how good is the design for the language.

4. Multi-agent and dynamic versions of the language should be designed and then thoroughly studied. In particular, multi-agent versions of

the language would allow to discuss notions such as distributed knowledge [46, 55, 57] and common knowledge [46, 76, 79, 88] and thus improving the expressivity of the language. Moreover, it would allow also to discuss group related notions of trust, such as social trust. On the other side, dynamic versions of the language, would allow to discuss the effects of public and private announcements and the impact those have on trust. For example, a public announcement might influence the knowledge of an agent at the point of making less states compatible with what he knows and, thus, this would influence how much he trusts a given proposition.

With respect to chapter 5

1. Explore other theoretical formalisms to reason about trust and/or uncertainty and then compare their formal structure with the semantical structure of the language here proposed. In particular, it would be nice to analyse the relationship between contextual trust models and probability theory. Another interesting issue to analyse is the relation between trust structures and algebras in measure theory.

Chapter 7

Acknowledgements

I'd like to thank, first of all, my girlfriend Giorgia. She undoubtedly helped me during those years to keep my mental health intact. She supported me uncountably many times and she is one of the best partners someone might wish to have close during happy and sad moments. I would also like to thank my parents, Dorothy and Enzo, who always taught me the importance of studying and learning. It is their endlessly passion and commitment put towards the task of helping me to achieve my goals that often gave me the strength to overcome tough moments. I thank my supervisor, Alessandro Aldini, for foolishly deciding to embark himself in the journey that led to the production of this thesis. It is never easy to deal with someone like me, therefore, his effort is especially appreciated. I also thank Pierluigi Graziani, for his enthusiasm in teaching logic to mathematically adverse philosophers. If it wasn't for his lectures, I would probably have left academia by now. Finally, I thank all the members of the Synergia Group in Urbino: Adriano, Vincenzo, Mario, Isabella, Gino, Stefano, Annamaria, Claudio and all the others. They made it possible to create an inspiring environment full of prompts even in a small university as that of Urbino.

Bibliography

[1] K. Abbink, I. Bernd, R. Elke, “The Moonlighting Game: an Empirical Study on Reciprocity and Retribution”, Journal of Economic Behavior and Organization 42, pp. 265–277, 2000.

[2] W. Abdelghani, C.A. Zayani, I. Amous, F. Sedes, “Trust management in Social Internet of Things: A Survey”, 15th IFIP Conference on e-Business, e-Services and e-Security, 2016.

[3] T.K. Ahn, E. Ostrom, D. Schmidt, J. Walker, “Trust in Two-Person Gmaes: Game Structures and Linkages”, in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 323–351, 2003.

[4] A. Aldini, “Design and Verification of Trusted Collective Adaptive Systems”, ACM Transactions on Modelling and Computer Simulation 28(2), 2018.

[5] A. Angelucci, M. Bastioni, P. Graziani, M.G. Rossi, “Philosophical Look at the Uncanny Valley”, in: J. Seibt, R. Hakli, M. Nørskov (eds.), Social Robots and the Future of Social Relations, pp. 165–170, 2014.

[6] K. Arrow, “Gifts and Exchanges”, Philosophy and Public Affairs 1(4), pp. 343–362, 1972.

[7] D. Artz, Y. Gil, “A Survey of Trust in Computer Science and the Semantic Web”, Web Semantics: Science, Services and Agents of the World Wide Web 5, pp. 58–71, 2007.

[8] R.J. Aumann, “Agreeing to Disagree”, The Annals of Statistics 4(6), pp. 1236–1239, 1976.

[9] B. Barber, “The Logic and Limits of Trust”, Rutgers University Press, 1983.

[10] P. Bateson, “The Biological Evolution of Cooperation and Trust”, in: D. Gambetta (ed.), Trust: Making and Breaking Cooperative Relations, Blackwell, pp. 31–48, 1988.

[11] M.Y. Becker, A. Russo, N. Sultana, "Foundations of Logic-Based Trust Management", IEEE Symposium on Security and Privacy, 2012.

[12] J. van Benthem, "Modal Logic for Open Minds", CSLI Publications, 2010.

[13] J. van Benthem, D. Fernández-Duque, E. Pacuit, "Evidence Logic: a New Look at Neighborhood Structures", Advances in Modal Logic 9, pp. 97-118, 2012.

[14] J. Berg, J. Dickhaunt, K. McCabe, "Trust, Reciprocity, and Social History", Games and Economic Behavior 10, pp. 122–142, 1995.

[15] I. Bohnet, F. Greig, B. Herrmann, R. Zeckhauser, "Betrayal Aversion: Evidence from Brazil, China, Oman, Switzerland, Turkey, and the United States", American Economic Review 98, pp. 294–310, 2008.

[16] R. Botsman, "What's Mine is Yours: How Collaborative Consumption is Changing the Way We Live", Collins, 2011.

[17] R. Botsman, "Who Can You Trust?", Portfolio/Penguin, 2017.

[18] M.R. Carter, M. Castillo, "The Economic Impact of Trust and Altruism: An Experimental Approach to Social Capital", Wisconsin-Madison Agricultural and Applied Economics Staff Papers 448, 2003.

[19] G. Casella, R.L. Berger, "Statistical Inference", Duxbury Press, 1990.

[20] C. Castelfranchi, R. Falcone, "Social Trust: a Cognitive Approach", Trust and Deception in Virtual Societies, pp. 55–90, 2001.

[21] B.L. Chellas, "Modal Logic: an Introduction", Cambridge University Press, 1980.

[22] J.H. Cho, K. Chan, S. Adali, "A Survey on Trust Modeling", ACM Computing Surveys 48(2), 2015.

[23] J. Coleman, "Foundations of Social Theory", Harvard University Press, 1990.

[24] K.S. Cook, R.M. Cooper, "Experimental Studies of Cooperation, Trust, and Social Exchange", in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 209–244, 2003.

[25] J.C. Cox, "Trust, Fear, Reciprocity, and Altruism: Theory and Experiment", Andrew Young School of Policy Studies Research Paper Series (07-16), 2006.

[26] M. Cramer, D.A. Ambrossio, P.V. Hertum, “A Logic of Trust for Reasoning about Delegation and Revocation”, Proceedings of the 20th ACM Symposium on Access Control Models and Technologies, pp. 173–184, 2015.

[27] T. Dalmonte, S. Negri, N. Olivetti, “Non-normal modal logics: bi-neighbourhood semantics and its labelled calculi”, Advances in Modal Logic, 2018.

[28] P. Dasgupta, “Trust as a Commodity”, in: D. Gambetta (ed.), Trust: Making and Breaking Cooperative Relations, Blackwell, pp. 49–72, 1988.

[29] A.P. Dempster, “Upper and Lower Probabilities Induced by a Multivalued Mapping”, The Annals of Mathematical Statistics 38(2), pp. 325–339, 1967.

[30] M. Deutsch, R.M. Krauss, “Studies of Interpersonal Bargaining”, Journal of Conclict Resolution 6, pp. 52–76, 1962.

[31] M. Deutsch, D. Canavan, J. Rubin, “The Effect of Size of Conflict and Sex of the Experimenter upon Interpersonal Bargaining”, Journal of Experimental Social Psychology 7, pp. 258–267, 1971.

[32] E. Durkheim, “The Division of Labor in Society”, MacMillan, 1893.

[33] C.C. Eckel, R.K. Wilson, “The Human Face of Game Theory: Trust and Reciprocity in Sequential Games”, in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 245–274, 2003.

[34] A. Eichner, “Why Economics is not yet a Science”, in: A. Eichner (ed.), Why Economics is not yet a Science, pp. 205–241, 1983.

[35] R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi, “Reasoning about Knowledge”, MIT Press, 2003.

[36] E. Fehr, “On the Economics and Biology of Trust”, Journal of the European Economic Association 7, pp. 235–266, 2009.

[37] L. Floridi, “The 4th Revolution: How the Infosphere is Reshaping Human Reality”, Oxford University Press, 2014.

[38] G. Frege, “´Uber Sinn und Bedeutung”, Zeitschrift f´ur Philosophie und Philosophische Kritik 100, pp. 25–50, 1892.

[39] D. Furlong, “The Conceptualization of ’Trust’ in Economic Thought”, IDS Working Paper 35, 1996.

[40] D. Gambetta, (Ed.), "Trust: Making and Breaking Cooperative Relations", Blackwell, 1988.

[41] M. Girlando, S. Negri, N. Olivetti, V. Risch, "Conditional Beliefs: from Neighbourhood Semantics to Sequent Calculus", to appear in The Review of Symbolic Logic, 2018.

[42] D. Good, "Individuals, Interpersonal Relations, and Trust", in: D. Gambetta (ed.), Trust: Making and Breaking Cooperative Relations, Blackwell, pp. 31–48, 1988.

[43] T. Grandison, M. Sloman, "A Survey of Trust in Internet Applications", IEEE Communications Surveys and Tutorials 3(4), pp. 2–16, 2000.

[44] M. Granovetter, "Economic Action and Social Structure: the Problem of Embeddedness", American Hournal of Sociology 91, pp. 481–510, 1985.

[45] J.Y. Halpern, "Reasoning about Uncertainty", MIT Press, 2017.

[46] J.Y. Halpern, Y. Moses, "Knowledge and Common Knowledge in a Distributed Environment", Journal of the ACM 37(3), pp. 549–587, 1990.

[47] W.D. Hamilton, "The Genetic Evolution of Social Behaviour", Journal of Theoretical Biology 7, pp. 1–52, 1964.

[48] H.H. Hansen, "Monotonic Modal Logic", Master's Thesis.

[49] H.H. Hansen, C. Kupke, "A Coalgebraic Perspective on Monotone Modal Logic", Procs. of the 7th Workshop on Coalgebraic Methods in Computer Science, pp. 121–143, 2004.

[50] H.H. Hansen, C. Kupke, E. Pacuit, "Bisimulation for Neighborhood Structures", Procs. of the 2nd Conference on Algebra and Coalgebra in Computer Science, pp. 279–293, 2007.

[51] H.H. Hansen, C. Kupke, E. Pacuit, "Neighborhood Structures: Bisimilarity and Basic Model Theory", Logical Methods in Computer Science 5(2), pp. 1–38, 2009.

[52] W.T. Harbaugh, K. Krause, S.G. Liday Jr., L. Vesterlund, "Trust in Children", in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 302–322, 2003.

[53] R. Hardin, "Trust and Trustworthiness", Russell Sage Foundation, 2002.

[54] R. Hardin, "The Street-Level Epistemology of Trust", Politics and Society 21, pp. 505–529, 1993.

[55] F. Hayek, “The Use of Knowledge in Society”, American Economic Review 35, pp. 519–530, 1945.

[56] F. Heider, “The Psychology of Interpresonal Relations”, Wiley, 1958.

[57] R. Hilpinen, “Remarks on Personal and Impersonal Knowledge”, Canadian Journal of Philosophy 7, pp. 1–9, 1977.

[58] J. Hintikka, “Knowledge and Belief: An Introduction to the Logic of the Two Notions”, Cornell University Press, 1962.

[59] A. Jøsang, “Subjective Logic”, Springer, 2016.

[60] A. Jøsang, “Trust and Reputation Systems”, in: A. Aldini, R. Gorrieri (eds.), Foundations of Security Analysis and Design IV, pp. 209–245, 2007.

[61] A. Jøsang, “Prospectives for Modelling Trust in Information Security”, Proceedings of the 2nd Australasian Conference on Information Security and Privacy, pp. 2–13, 1997.

[62] A. Jøsang, B. Touhid, X. Yue, C. Clive, “Combining Trust and Reputation Management for Web-Based Services”, Procs. of the 5th International Conference on Trust, Privacy and Security in Digital Businesses, pp. 90–99, 2008.

[63] A. Jøsang, R. Ismail, “The Beta Reputation System”, Proceedings of the 15th Bled Electronic Commerce Conference, e-Reality: Constructing the e-Economy, pp. 324–337, 2002.

[64] A. Jøsang, R. Ismail, C. Boyd, “A Survey of Trust and Reputation Systems for Online Service Provision”, Decision Support Systems 43(2), pp. 618–644, 2007.

[65] A. Jøsang, S.J. Knapskog, “A Metric for Trusted Systems”, Procs. of the 15th IFIP/SEC International Information Security Conference (IFIP), 1998.

[66] D. Kahneman, J.L. Knetsch, R. Thaler, “Fairness and the Assumptions of Economics”, Journal of Business 59, pp. 285–300, 1986.

[67] S.B. Kamvar, M.T. Schlosser, H. Garcia-Molina, “The EigenTrust Algorithm for Reputation Management in P2P Networks”, Procs. of the 12th International Conference on World Wide Web, pp. 640–651, 2003.

[68] I. Kant, “Groundwork of the Mataphysic of Morals”, 1785.

[69] J.M. Keynes, “Treatise on Probability”, Macmillan and Co., 1921.

[70] R.A. Kowalski, “The Early Years of Logic Programming”, Communications of the ACM 31(1), pp. 38–43, 1988.

[71] D. Krebs, “Altruism: an Examination of the Concept and a Review of the Literature”, Psychological Bullettin 73, pp. 258–302, 1970.

[72] S. Kripke, “Semantical Considerations on Modal Logic”, Acta Philosophica Fennica 16, pp. 83–94, 1963.

[73] S. Kuhn, “Prisoner’s Dilemma”, in: E.N. Zalta (ed.), The Stanford Encyclopedia of Philosophy, visited February 2018.

[74] R. Kurzban, “Biological Foundations of Reciprocity”, in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 105–127, 2003.

[75] R. Leeds, “Altruism and the Norm of Giving”, Merrill-Palmer Quarterly 9, pp. 229–240, 1963.

[76] D. Lehmann, “Knowledge, Common Knowledge, and Related Puzzles”, Proceedings of the 3rd ACM symposium on Principles of Distributed Computing, pp. 62–67.

[77] C. Leturc, G. Bonnet, “A Normal Modal Logic for Trust in the Sincerity”, Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, pp. 175–183, 2018.

[78] M. Levi, “A State of Trust”, in: V. Braithwaite, M. Levi, K.S. Cook, R. Hardin (eds.) Trust and Governance, Russell Sage Foundation, pp. 77–101, 1998.

[79] D. Lewis, “Convention, A Philosophical Study”, Harvard University Press, 1969.

[80] G. Lu, J. Lu, S. Yao, J. Yip, “A Review on Computational Trust Models for Multi-Agent Systems”, Proceedings of the International Conference on Internet Computing, pp. 325–331, 2007.

[81] A.S. Luchins, “Mechanization in Problem Solving”, Psychological Monographs 54, 1942.

[82] N. Luhmann, “Trust and Power”, John Wiley and Sons Inc, 1979.

[83] N. Luhmann, “Familiarity, Confidence, Trust: Problems and Alternatives”, in: D. Gambetta (ed.), Trust: Making and Breaking Cooperative Relations, Blackwell, pp. 94–108, 1988.

[84] S. Madakam, R. Ramaswamy, S. Tripathi, “Internet of Things (IoT): A Literature Review”, Journal of Computer and Communications 3(3), pp. 164–173, 2015.

[85] J. Mansbridge, “Altruistic Trust”, in: M.E. Warren (ed.), Democracy and Trust, Cambridge University Press, pp. 290–309, 1999.

[86] S. Marsh, “Formalising Trust as a Computational Concept”, Ph.D. Thesis, University of Stirling, 1994.

[87] K.A. McCabe, V.L. Smith, “Strategic Analysis in Games: What Information Do Players Use?”, in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 275–301, 2003.

[88] J. McCarthy, M. Sato, T. Hayashi, S. Igarishi, “On the Model Theory of Knowledge”, Technical Report STAN-CS-78-657, Stanford University, 1979.

[89] B. McEvily, R.A. Weber, C. Bicchieri, V.T. Ho, “Can Groups be Trusted? An Experimental Study of Trust in Collective Entities”, in: R. Bachmann, A. Zaheer (eds.), Handbook of Trust Research, pp. 52–67, 2006.

[90] B. McEvily, J.R. Radzevick, R.A. Weber, “Whom do you distrust and how much does it cost? An Experiment on the Measurement of Trust”, Games and Economic Behavior 74, pp. 285–298, 2012.

[91] R. Montague, “The Proper Treatment of Quantification in Ordinary English”, in: J. Hintikka, J. Moravcsik, P. Suppes (eds.), Approaches to Natural Language, pp. 221–242. 1973.

[92] R. Montague, “English as a Formal Language”, in: B. Visentini (ed.), Linguaggi nella SocietÃ e nella Tecnica, pp. 189–223. 1970.

[93] R. Montague, “Universal Grammar”, Theoria 36, pp. 373–398, 1970.

[94] L. Mui, A. Halberstadt, M. Mohtashemi, “Notions of Reputation in Multi-Agent Systems: a Review”, Procs. of the 1st Int. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS’02), ACM Press, pp. 280–287, 2002.

[95] V.C. Muller, “A Formal Analysis of Trust Operations”, Ph.D. Thesis, University of Amsterdam, 2013.

[96] S. Negri, “Proof Theory for Non-Normal Modal Logics: the Neighbourhood Formalism and Basic Results”, Journal of Logics and their Applications, Mints’ Memorial Issue, 4(4), pp. 1241–1286, 2017.

[97] S. Negri, “Non-Normal Modal Logics: a Challenge to Proof Theory”, in: P. Arazim, T. Lavicka (eds.), The Logica Yearbook, pp. 125–140, 2016.

[98] H. Nunoo-Mensah, K.O. Boateng, J.D. Gadze, "The Adoption of Socio and Bio-Inspired Algorithms for Trust Models in Wireless Sensor Networks: A Survey", International Journal of Communication Systems 31(4), 2017.

[99] E. Ostrom, "Towards a Behavioral Theory Linking Trust, Reciprocity, and Reputation" in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 19–79, 2003.

[100] E. Pacuit, "Neighborhood Semantics for Modal Logic", Springer, 2017.

[101] I. Pinyol, J. Sabater-Mir, P. Dellunde, M. Paolucci, "Reputation-based Decisions for Logic-based Cognitive Agents", Autonomous Agents and Multi-Agents Systems 24(1), Springer, pp. 175–216, 2012.

[102] I. Pinyol, J. Sabater-Mir, "Computational Trust and Reputation Models for Open Multi-Agent Systems: a Review", Artificial Intelligence, Review 40, pp. 1–25, 2013.

[103] V.R. Pratt, "Semantical Considerations on Floyd-Hoare Logic", 17th Annual Symposium on Foundations of Computer Science, pp. 109–121, 1976.

[104] D.G. Pruitt, M.J. Kimmel, "Twenty Years of Experimental Gaming: Critique, Synthesis, and Suggestions for the Future", Annual Review of Psychology 28, pp. 363–392, 1977.

[105] R. Putnam, "Making Democracy Work", Princeton University Press, 1993.

[106] L.M. PytlikZillig, C.D. Kimbrough, "Consensus on Conceptualizations and Definitions of Trust: Are We There Yet?", in: E. Shockley, T.M.S. Neal, L.M. PytlikZillig, B.H. Bornstein (eds.), Interdisciplinary Perspectives on Trust: Towards Theoretical and Methodological Integration, Springer, pp. 17–47, 2016.

[107] S.D. Ramchurn, D. Huynh, N.R. Jennings, "Trust in Multi-Agent Systems", The Knowledge Engineering Review 19(1), pp. 1–25, 2004.

[108] A. Rapoport, A.M. Chammah, "Prisoner's Dilemma: a Study in Conflict and Cooperation", Ann Arbor: University of Michigan Press, 1965.

[109] L. Rasmusson, S. Jansson, "Simulated Social Control for Secure Internet Commerce", NSPW Procs. of the 1996 Workshop on New Security Paradigms, pp. 18–25, 1996.

[110] B.G. Robbins, "What is Trust? A Multidisciplinary Review, Critique, and Synthesis", Sociology Compass 10(10), pp. 972–986, 2016.

[111] B.G. Robbins, “On The Origins of Trust”, Ph.D. Thesis, University of Washington, 2014.

[112] L. Ross, M.R. Lepper, F. Strack, J. Steinmetz, “Social Explanation and Social Expectation: Biase Attributional Processes in the Debriefing Paradigm”, Journal of Personality and Social Psychology 35, pp. 485–494, 1977.

[113] J. Sabater-Mir, C. Sierra, “Review on Computational Trust and Reputation Models”, Artificial Intelligence Review 24(1), Springer, pp. 33-60, 2005.

[114] T. Schelling, “The Strategy of Conflict”, Harvard University Press, 1960.

[115] D. Scott, “Advice on Modal Logic”, Philosophical Problems in Logic, pp. 143–173, 1970.

[116] G. Shafer, “A Mathematical Theory of Evidence”, Princeton University Press, 1976.

[117] E. Shockley, T.M.S. Neal, L.M. PytlikZillig, B.H. Bornstein (eds.), “Interdisciplinary Perspectives on Trust: Towards Theoretical and Methodological Integration”, Springer, 2016.

[118] C. Sierra, J. Debenham, “An Information-Based Model for Trust”, Proceedings of AAMAS ’05, pp. 497–504, 2005.

[119] G. Simmel, “Philosophy of Money”, Routlege, 1978.

[120] G. Simmel, “The Sociology of Georg Simmel”, New York Free Press, 1950.

[121] G. Suryanarayana, R.N. Taylor, “A Survey of Trust Management and Resource Discovery Technologies in Peer-to-Peer Applications”, ISR Technical Report UCI-ISR-04-6, University of California, 2005.

[122] M. Tagliaferri, A. Aldini, “A Logical Language for Trust Computations”, to be submitted to Journal of Logic and Computation.

[123] M. Tagliaferri, A. Aldini, “A Taxonomy of Computational Models for Trust Computing in Decision- Making Procedures”, Procs. of the 17th European Conference on Cyber Warfare and Security (ECCWS’18), to appear, 2018.

[124] M. Tagliaferri, A. Aldini, “From Knowledge to Trust: a Logical Framework for Pre-Trust Computations”, Procs. of the 12th IFIP International Conference on Trust Management (IFIPTM’18), to appear, 2018.

[125] M. Tagliaferri, A. Aldini, "A Trust Logic for Pre-Trust Computations", Procs. of the 21th International Conference on Information Fusion (Fusion'18), to appear, 2018.

[126] R.L. Trivers, "The Evolution of Reciprocal Altruism", The Quarterly Review of Biology 46(1), pp. 35–57, 1971.

[127] R.L. Trivers, "Natural Selection and Social Theory: Selected Papers of Robert Trivers", Oxford University Press, 2002.

[128] N.B. Truong, H. Lee, B. Askwith, G.M. Lee, "Toward a Trust Evaluation Mechanism in the Social Internet of Things", Sensors 17(6), pp.1346–1370, 2017.

[129] University of Victoria, "Complexity Science in brief", informative report, 2012.

[130] E.M. Uslaner, "Who Do You Trust?", in: E. Shockley, T.M.S. Neal, L.M. PytlikZillig, B.H. Bornstein (eds.), Interdisciplinary Perspectives on Trust: Towards Theoretical and Methodological Integration, Springer, pp. 71–83, 2016.

[131] E.M. Uslaner, "Varieties of Trust", FEEM Working Paper 69, 2005.

[132] H. Wichman, "Effects of Isolation and Communication on Cooperation in a Two-Person Game", Journal of Personality and Social Psychology 16, pp. 114–120, 1970.

[133] O. Williamson, "Calculativeness, Trust, and Economic Organization", Journal of Law and Economics 36(2), pp. 453–486, 1993.

[134] T. Yamagishi, "Cross-Societal Experimentation on Trust: A Comparison of the United States and Japan", in: E. Ostrom, J. Walker (eds.), Trust and Reciprocity: Interdisciplinary Lessons for Experimental Research, Russell Sage Foundation, pp. 352–370, 2003.

[135] B. Yu, M.P. Singh, "Detecting Deception in Reputation Management", Procs. of the 2nd Int. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'03), ACM Press, pp. 73–80, 2003.