PhD-FSTC-2016-24

The Faculty of Sciences, Technology and Communication

DISSERTATION

Defence held on 29/06/2016 in Luxembourg

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG

EN INFORMATIQUE

by

Xin SUN

Born on 3 November 1985 in Linhai (Zhejiang, P.R. China)

LOGIC AND GAMES OF NORMS: A COMPUTATIONAL PERSPECTIVE

Dissertation defence committee

Dr Leon van der Torre, dissertation supervisor

Professor, Université du Luxembourg

Dr Jan Broersen

Associate Professor, University of Utrecht

Dr Pierre Kelsen, Chairman

Professor, Université du Luxembourg

Dr Christian Straßer

Ruhr-University Bochum

Dr Xavier Parent, Vice Chairman

Université du Luxembourg

Acknowledgements

First of all , I would like to express my gratitude to my supervisor Leon van der Torre and Xavier Parent. Thanks for the advice, encourangement and criticism. Secondly, I thank Jan Broersen for all the attention that he has paid to me during these years. Thirdly, I would like to thank Christian Straßer and Pierre Kelsen for their willingness to assess my dissertation. Finally, I must thank all my colleagues and friends in Luxembourg for their help during my PhD.

Abstract

This thesis studies how to generate, represent and reason about norms and how to use norms to construct artificial ethical agents.

We investigate two types of norm generation: norm creation and norm emergence. In the study of norm creation, we consider norms as normative rules which are created using correlated equilibrium in games. In the study of norm emergence, we propose a model that supports the emergence of norms via multiagent learning in social networks.

We then investigate the formal representation of norms. The derivation systems of input/output logic are axiomatic representations of norms. We analyze various derivation rules of input/output logic in isolation and define the corresponding semantics. Theory of joining-systems is an algebraic approach to normative systems. We develop two variants of theory of joining-systems: Boolean joining-systems and Heyting joining-systems. Those two variants algebraically characterize unconstrained input/output logic in the sense that a norm is axiomatically derivable from a set of norms if and only if it is in the space of norms algebraically generated this set of norms.

We study how to reason about norms from a computational perspective. We show that input/output logic is coNP-hard and in the 2nd level of the polynomial hierarchy. We further show prioritized input/output logic out_1^p, as well as prioritized imperative logic, is complete for the 2ed level of the polynomial hierarchy while deontic default logic is located in the 3ed level of the polynomial hierarchy.

We use norms-based deontic logic and games to build ethical agents. Norms-based deontic logic are used to reason about norms and Boolean games are used to represent the interaction of agents. We use norms to assess the normative status of strategies. Then agents' preferences are changed by the normative status of strategies. We study some complexity issues related to normative reasoning/status and agents' preference change.

Contents

Chapter 1

Introduction: Logic and Games of Norms

In this introductory chapter we describe the background, objective and methodology of this thesis.

1.1 Norms in multiagent systems

Norms are found everywhere in our daily life. For example, the following norms can be easily encountered:

- You **should** drive on the right side.
- Murder is **forbidden**.
- A PhD student in China is **permitted** to take no holidays.
- A PhD student in Luxembourg is **obliged** to take all his holidays before the end of his contract.

Syntactically, norms are expressed using deontic modalities such as *obliged*, *must*, *permitted*, *may* and *forbidden*. The literature is populated with various definitions of the term *norm*. The Webster online dictionary* provides three definitions for the term *norm*:

1. an authoritative standard;
2. a principle of right action binding upon the members of a group and serving to guide, control, or regulate proper and acceptable behavior;
3. average:
   (a) as a set standard of development or achievement usually derived from the average or median achievement of a large group;
   (b) as a pattern or trait taken to be typical in the behavior of a social group;
   (c) as a widespread or usual practice, procedure, or custom.

These definitions are representatives of the term and it is used in many areas of research, including deontic logic, legal theory, sociology and game theory.

1. In deontic logic, norms are usually represented as (conditional) obligations, permissions, prohibitions that an agent has to a larger social system (Gabbay et al., 2013, 2016).
2. In legal theory, norms are rules of behavior imposed by an authorized body and enforced via applying sanctions (Posner, 2002).

*http://www.merriam-webster.com/dictionary/norm

3. In sociology, norms are rules or restrictions of behavior that are socially enforced and considered valid by the majority of a social group (Bendor and Swistak, 2001).

4. In game theory, a norm is a pattern of behavior that has been adopted by the majority of a social group and is considered successful (Gintis, 2010).

1.1.1 Normative multiagent system

In human societies, norms play a crucial role in regulating the behavior of the individuals. In multiagent systems (Shoham and Leyton-Brown, 2009; Wooldridge, 2009; Weiss, 2013), artificial agents are constructed as possessing characteristics that are similar to human beings. Multiagent system researchers have been interested in norms because norms are helpful in maintaining social order (Conte and Dellarocas, 2001) and facilitating cooperation and coordination (Shoham and Tennenholtz, 1992, 1997; Axelrod, 1997). Norms can also reduce the computational costs required by agents since they do not have to search their state space of possible actions if they decide to follow norms. The behavior of agents should also be more predictable than when norms are absent (Epstein, 2001).

Normative multiagent systems (NorMAS) are multiagent systems in which agents' behavior are regulated by norms (Shoham and Tennenholtz, 1992, 1997; Boman, 1999; Conte et al., 1999; Dignum, 1999; Boella et al., 2008b; Andrighetto et al., 2013). The first definition of a normative multiagent system emerged after two days of discussion at the first workshop on normative multiagent systems held in 2005 as a symposium of the Artificial Intelligence and Simulation of Behavior Convention in Hatfield, United Kingdom:

The norm change definition. "A normative multiagent system is a multiagent system together with normative systems in which agents on the one hand can decide whether to follow the explicitly represented norms, and on the other the normative systems specify how and in which extent the agents can modify the norms." (Boella et al., 2006).

The second definition of a normative multiagent system emerged at the second workshop on normative multiagent systems held as Dagstuhl Seminar 07122 in 2007. After four days of discussions, the participants agreed to the following consensus definition:

The mechanism design definition. "A normative multiagent system is a multiagent system organized by means of mechanisms to represent, communicate, distribute, detect, create, modify, and enforce norms, and mechanisms to deliberate about norms and detect norm violation and fulfillment." (Boella et al., 2008b)

According to Savarimuthu and Cranefield (2011), research on NorMAS can be categorized into two branches. The first branch focuses on normative system architectures, norm representations, norm adherence and the associated punitive or incentive measures. The second branch is concerned with the generation of norms.

Several architectures have been proposed for the study of norms (López y López and Marquez, 2004; Boella and van der Torre, 2006; Fonseca dos Santos Neto et al., 2012). Some researchers have used deontic logic to define and represent norms (Makinson and van der Torre, 2000; Boella and van der Torre, 2006; Governatori and Rotolo, 2010). Other works have investigated mechanisms for norm compliance and enforcement (Axelrod, 1986; López y López et al., 2002; Aldewereld et al., 2006).

While the first branch studies how norms are formalized and represented, it does not address the question of where norms come from. Two general approaches to the generation of norms have been investigated in the literature: online approaches (Shoham and Tennenholtz, 1997; Sen and Airiau, 2007; Morales et al., 2011; Airiau et al., 2014; Morales et al., 2015) and offline approaches (Shoham and Tennenholtz, 1996; van der Hoek et al., 2007; Ågotnes and Wooldridge, 2010; Ågotnes et al., 2012). Online approaches aim to establish agents with the ability to dynamically coordinate their activities. In contrast, offline approaches aim at developing a coordination device at design-time, and build this regulation into a system for use at run-time.

1.1.2 Deontic logic and games in NorMAS

Deontic logic and games are two important intellectual sources for NorMAS. Deontic logic is used for the representation and reasoning about norms. It is also applicable in the construction of ethical agents. Game theory is popular in the study of norm generation as well as the interaction of agents governed by norms.

What is deontic logic?

Deontic logic can be defined as the formal study of normative reasoning (Gabbay et al., 2013, 2016). Generally speaking, one might define logic as the study of the principles of correct reasoning. It tells us whether certain conclusions follow from a number of given assumptions. Propositional logic looks at the logical relationships amongst statements which assert whether something can be judged as true or false. Such a sentence is usually called a declarative sentence. By contrast, deontic logic is the study of logical relationships among propositions which assert that certain actions or states of affairs are obligatory, forbidden or permitted.

In 1951, the philosopher and logician Georg von Wright wrote a paper called "Deontic Logic" (von Wright, 1951), which subsequently became the name of the research area. The term deontic is derived from the ancient Greek *déon*, meaning that which is binding or proper. The basis of his formal system was an observed relation between obligation and permission. For example, he defined the obligation to tell the truth by interpreting that it is good to tell the truth, and therefore it is bad to lie. If it is bad to lie then it is forbidden to lie, and therefore it is not permitted to lie. Summarizing, something is obligatory when its absence is not permitted. This logical relation is based on the binary distinction between good and bad, as illustrated by its possible world semantics distinguishing between good and bad worlds. In fact, von Wright's deontic logic is exactly the same as the monadic modal logic KD defined as the valid formulas on the class of serial frames. Such logic is later called standard deontic logic (SDL).

Given a set $\mathbb{P}$ of propositional atoms, the language of standard deontic logic is represented by the following BNF. For p ranges over $\mathbb{P}$,

$$x := p \mid \neg x \mid x \wedge x \mid \bigcirc x$$

The intended reading of $\bigcirc x$ is "x is obligatory". The semantics of SDL is constructed using relational models.

Definition 1.1 (Relational model (Blackburn et al., 2001))**.** *A relational model $M = (W, R, V)$ is a tuple where:*

- *W is a (non-empty) set of possible worlds: $w, w', \ldots$*
- *$R \subseteq W \times W$ is a binary relation over W.*
- *$V : \mathbb{P} \mapsto 2^W$ is a valuation function for propositional atoms such that $V(p) \subseteq W$.*

Definition 1.2 (Satisfaction (Blackburn et al., 2001))**.** *Given a relational model $M = (W, R, V)$ and a world $w \in W$, the satisfaction relation $M, w \vDash x$ (read as "world w satisfies x in model M") is defined by induction on the structure of x using the following clauses*

- *$M, w \vDash p$ iff $w \in V(p)$.*
- *$M, w \vDash \neg x$ iff $M, w \nvDash x$.*
- *$M, w \vDash x \wedge y$ iff $M, w \vDash x$ and $M, w \vDash y$.*
- *$M, w \vDash \bigcirc x$ iff for all $w' \in W$, if $(w, w') \in R$ then $M, w' \vDash x$.*

An SDL formula x is valid, denoted as $\vDash x$, if for all relational models $M = (W, R, V)$ and all worlds $w \in W$, $M, w \vDash x$ holds. Sahlqvist theorem gives an Hilbert style axiomatization of SDL made up with:

- all tautologies of classical propositional logic.
- $\bigcirc(x \to y) \to (\bigcirc x \to \bigcirc y)$.
- $\bigcirc x \to \neg \bigcirc \neg x$.
- Modus ponens rule.
- Necessitation rule: from $\vdash x$ infer $\vdash \bigcirc x$.

Modern deontic logic started with a paper by Bengt Hansson in 1969, called "An Analysis of Some Deontic Logics" (Hansson, 1969). In that paper he introduced a semantics based on a preference relation for conditional obligations. Hansson's preference-based deontic logic is further developed by several researchers (Åqvist, 1986; Boutilier, 1994; van der Torre, 1997; Parent, 2008; van Benthem et al., 2014). With the work of Meyer (1988), deontic logic became a part of computer science. Meyer (1988) led to the creation of the DEON conference series (http://deonticlogic.org/).

Different approaches of deontic logic have been studied in the past 6 decades including imperative logic (van Fraassen, 1973; Alchourrón and Bulygin, 1981; Hansen, 2008), deontic action logic (Segerberg, 1982; Castro and Maibaum, 2009; Trypuz and Kulicki, 2015), dynamic deontic logic (Meyer, 1988; van der Meyden, 1996; Broersen, 2003), deontic STIT logic (Horty, 2001; Kooi and Tamminga, 2008; Sun, 2011; Sun and Baniasadi, 2014), input/output logic (Makinson and van der Torre, 2000, 2001, 2003; Stolpe, 2008b; Parent, 2011; Parent and van der Torre, 2014a; Straßer et al., 2016), deontic default logic (Horty, 2003, 2007, 2012, 2014), deontic defeasible logic (Antoniou et al., 2007, 2009; Governatori et al., 2013), adaptive deontic logic (Putte and Straßer, 2013; Beirlaen et al., 2013) and categorical deontic logic (Peterson, 2014, 2015). Those results are summarized in the two-volume handbook of deontic logic and normative systems (Gabbay et al., 2013, 2016).

In input/output logic, imperative logic, deontic default logic and deontic defeasible logic, norms are explicitly represented. The truth value of deontic propositions in those logics are explained not by some set of possible worlds among which some are ideal, but with reference to a set of given norms. Such a non-possible world semantics is calldc norm-based semantics in Hansen (2014). We then use norm-based deontic logic as a general term to refer input/output logic, imperative logic, deontic default logic and deontic defeasible logic and use deontic modal logic to refer those approaches which adopt possible world semantics. Norm-based deontic logic will be extensively studied in this thesis.

Why is normative reasoning relevant for NorMAS?

In multiagent systems, artificial and human agents interact with each other. Since the use of norms is a key element of human social intelligence, norms may be essential too for artificial agents that collaborate with humans. By integrating norms and individual intelligence, normative multiagent systems provide a promising model for human and artificial agent cooperation and coordination (Keogh and Sonenberg, 2013), multi-agent organizations (Haynes et al., 2013; Testerink et al., 2013), electronic institutions (Aldewereld, 2009; Frantz et al., 2013), etc.

Advantages of formal methods

Deontic logic and other formalisms for normative reasoning are examples of formal methods. Formal methods may be helpful when employed as a modeling language, such as during the design of multiagent systems, explaining their structure to other designers and for reasoning behind the system. Formal methods provide a mathematically rigorous framework for modeling normative multiagent systems. The modeling language is given a formal semantics, which constrains the intuitive characterization of the normative notions being used. The language is equipped with a complete axiomatic characterization. On the one hand, the meaning of the deontic concepts is given by the axioms governing their use. On the other hand, a corollary to completeness is consistency. In this manner, there is a guarantee that the framework is consistent. Without such a guarantee, the move to the implementation level would be pointless: an inconsistent framework could be as easily implemented as a consistent one, but it would be useless (Broersen et al., 2013).

Game theory in NorMAS

In the past 50 years, game theory has become rather popular in the study of norms (Lewis, 1969; Taylor, 1976; Axelrod, 1986; Sugden, 1989; Shoham and Tennenholtz, 1997; Binmore, 2005; Bicchieri, 2006; Alexander, 2007; Boella and van der Torre, 2007a; Sen and Airiau, 2007; Sen and Sen, 2009; Savarimuthu and Cranefield, 2011; Skyrms, 2014). In general, the game theoretical analysis of norms considers norms as Nash equilibria of games played by rational agents. The insight underlying all these contributions is that if agents play a game with several Nash equilibria, norms can serve to choose a unique equilibrium among these equilibria. Most work on the emergence of norms are from a game theoretical perspective (Shoham and Tennenholtz, 1997; Alexander, 2007; Sen and Airiau, 2007; Sen and Sen, 2009; Savarimuthu and Cranefield, 2011). Evolutionary game theory and learning in games are also useful tools in the study of norm emergence. Some special games are introduced in the study of norms, for example "norms game" is introduced in Axelrod

(1986) and "violation game" is introduced in van der Torre (2010b). Norm negotiation in online multi-player games is introduced in Boella and van der Torre (2007b) and Boella et al. (2009a).

In this thesis, we study norms from the perspective of deontic logic and game theory. We consider **norms as normative rules**. A norm is a rule in the sense that it contains both a premise and a consequence where the premise describes in what situation the norm is triggered and the consequence describes the prescription or the demand of the rule. Norms are normative in the sense that they prescribe deontic consequences, classifying what is obligatory, permitted or forbidden. Imperatives are a simple type of norms which are obligatory statements expressing commands. Social norms and conventions are special types of norms which are publicly accepted by a group of agents such that every agent in this group conforms to those norms and expects others to conform as well. We start our investigation of norms by explaining the objective and methodology of this thesis. Following this, we review the current states of deontic logic and game theoretical analysis of norms.

1.2 Background and Objectives

In this thesis, we study how to generate, represent and reason about norms, and how to use norms to construct artificial ethical agents. The background and objectives of this thesis are explained as follows.

1.2.1 On norm creation

Although the insight that norms are Nash equilibria applies to several important social situations, it does not apply to most. Gintis (2010) suggests a more general principle, according to which norms are considered as correlated equilibria, of which the function is like a "choreographer" who sends signals to agents to improve their coordination.

In game theory, correlated equilibrium is a solution concept that is more general than Nash equilibrium. Correlated equilibrium is first discussed in Aumann (1974). A correlated equilibrium is a probability distribution over agents' action profiles, which can be understood as a public randomized signal, such that each agent can choose its action according to the recommendation of the signal. If no agent would want to deviate from the recommended action (assuming the others don't deviate), the probability distribution is called a correlated equilibrium. Gintis (2010) argues that treating norms as correlated equilibria has two attractive properties that are lacking in the treating of norms as Nash equilibria.

> *"First, the conditions under which rational agents play Nash equilibria are generally complex and implausible, whereas rational agents in a very natural sense play correlated equilibria. Second, the social norms as Nash equilibria approach cannot explain why compliance with social norms is often based on other-regarding and moral preferences in which agents are willing to sacrifice on behalf of compliance with social norms. We can explain this association between norms and morality in terms of the incomplete information possessed by the choreographer."*(Gintis, 2010)

The first attractive property of correlated equilibrium is that the epistemic condition for correlated equilibrium is much more realistic than that of Nash Equilibrium. Aumann and Brandenburger (1995) show that in a Nash equilibrium, each agent's strategy is optimal given his belief about the other players strategies, and this belief is correct. Such epistemic conditions are extremely confining and cannot be expected to hold in general. By contrast, Aumann (1987) shows that in a correlated equilibrium an agent is required to make the best choice according to his belief about the other agents' strategies, but such belief is not assumed to be correct.

Gintis (2010) argues that the second attractive property of correlated equilibrium is that it explains why agents are sometimes willing to sacrifice themselves in order to comply to norms. Indeed, all Nash equilibria are correlated equilibria but not vice verse. In a non-Nash correlated equilibrium, it might happen that agents sacrifice themselves by following the suggestions produced by the choreographer.

Although Gintis' proposal is appealing, two things are missing in his account. The first is that Gintis does not study specific correlated equilibrium which are optimal in some senses. For example those correlated equilibrium which maximizes the sum of the expected utility of all agents can be considered as optimal. In addition to the study of creating norms from correlated equilibrium in general, we should also study how to create norms from optimal correlated equilibrium. The second missing thing is that Gintis does not provide algorithms to transform correlated equilibria to norms. The first objective of this thesis is to fill these gaps in Gintis' framework.

- **Objective 1**: Design methods to create norms for an arbitrary given normal-form game by first computing an optimal correlated equilibrium of the game, then transforming the correlated equilibrium to norms.

1.2.2 On norm emergence

In the literature of multiagent systems, the online approach of norm emergence (Shoham and Tennenholtz, 1997; Sen and Airiau, 2007; Morales et al., 2011, 2015) aims to establish agents with

the ability to dynamically coordinate their activities, for example by reasoning explicitly about coordination at run-time or by learning from the interaction with other agents. Norm emergence is also studied by philosophers. An evolutionary game theoretical approach of norm emergence can be found in Alexander (2007) and Skyrms (2014).

Games studied in Alexander (2007) includes the prisoner's dilemma, stag hunt, cake cutting and the ultimatum game. Alexander uses these games to analyze the emergence of norms of cooperation, trust, fair division and retaliation respectively. In this thesis, we study norm emergence in a game called Ali Baba and the Thief, which is not explored in Alexander (2007). We show that some norms prohibiting harmful behaviors, such as "you should not rob", can emerge after repeated play of Ali Baba and the Thief.

- **Objective 2**: Introduce the game Ali Baba and the Thief and use it to study the emergence of certain norms via repeated play of the game in some social networks.

1.2.3 On norm representation

The derivation systems in unconstrained input/output logic (Makinson and van der Torre, 2000; Stolpe, 2008b; Parent and van der Torre, 2014a) provide axiomatic representations of norms. One feature of the existing work of input/output logic is that the derivation rules always work in bundles. When several derivation rules work together, the corresponding semantics will be rather complex, and insights of the machinery is therefore concealed. To achieve a deeper understanding of input/output logic, it is helpful to isolate every single rule and study them separately.

Theory of joining-systems introduced by Lindahl and Odelstad (2000, 2008, 2013); Odelstad and Lindahl (2000) is an algebraic framework for analyzing normative systems. In this thesis we develop two variants of theory of joining-systems: Boolean joining-systems and Heyting joining-systems. Within those algebraic frameworks, we define isomorphism and embedding between normative systems. Then we use them to study the similarity of normative systems as well as some other global properties of normative systems.

- **Objective 3**: Investigate the axiomatic representation of norms provided by input/output logic and the algebraic representation of norms provided by theory of joining-systems.

1.2.4 On reasoning about norms

The complexity of default logic has been extensively studied in Stillman (1992); Gottlob (1992); Rintanen (1998a). Rintanen investigates the complexity of three proposals of prioritized default logic: Brewka (1994), Baader and Hollunder (1995) and Rintanen (1998b). The complexity of deontic

defeasible logic is studied in Governatori et al. (2013). It is well-known in theoretical computer science that complexity is an indispensable component of every logic. So far, previous literature in norm-based deontic logics (except deontic defeasible logic) focuses on proof theory and semantics, and neglects complexity. In this thesis, we fill this gap by the extensive study of the complexity of norm-based deontic logic.

- **Objective 4**: Investigate the complexity of norm-based deontic logic.

1.2.5 On ethical agents

Ethical agents have been extensively studied in moral philosophy and in economics, and their studies have been identified as one of the thorniest challenges in artificial intelligence [†] (Deng, 2015). The intersection of these areas is the new field of machine ethics (Anderson and Anderson, 2011). "Machine ethics is concerned with giving machines ethical principles or a procedure for discovering a way to resolve the ethical dilemmas they might encounter, enabling them to function in an ethically responsible manner through their own ethical decision making." (Anderson and Anderson, 2011). The best known ethical principles designed for intelligent machines are the Three Laws of Robotics formulated by the science fiction author Isaac Asimov:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.

2. A robot must obey orders given it by human beings except where such orders would conflict with the First Law.

3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

Much research has emphasized using machine-learning techniques such as neural networks (Guarini, 2006), case-based reasoning (McLaren, 2006), and inductive logic programming (Anderson et al., 2006) to build ethical agents. Pereira and Saptawijaya (2009) illustrate how moral decisions can be drawn computationally by using prospective logic programs. Prospective logic programs are used to model moral dilemmas, as they are able to prospectively look ahead at the consequences of hypothetical moral judgments. With this knowledge of consequences, moral rules are then used to decide the appropriate moral judgments. The whole moral reasoning is achieved through a priori constraints and a posteriori preferences on abductive stable models, two features that are found in prospective logic programming.

[†] http://www.nature.com/news/machine-ethics-the-robot-s-dilemma-1.17881

Bringsjord et al. (2006) propose the application of deontic logic in machine ethics. The objective of their research is to arrive at a methodology that allows an agent to behave ethically as much as possible in an environment which requires such behavior. Lorini (2015) develops a dynamic logic of mental attitudes and joint actions and uses it to provide a logical analysis of moral agency. Instead of norms, Lorini uses an ideality function which maps every possible world to a real number, representing the degree of the ideality to characterized the moral aspect of an agent. Lorini left it as future work to investigate the relationships between an agent's moral values and norms.

Nagenborg (2007) identifies an artificial moral agent as an artificial agent guided by norms, which we as human beings consider to have moral content. BOID (belief, obligation, intention, desire) agent architecture (Broersen et al., 2005; Governatori and Rotolo, 2007) is an extension of the BDI model with obligations as a moral component.

Objective 5: Apply norm-based deontic logic to the construction of artificial ethical agents.

1.3 Research Methodology

Methodology for Objective 1

Our methodology of norm creation is to use convex optimization to efficiently compute an optimal correlated equilibrium, then transform correlated equilibrium to norms. We study different types of optimal correlated equilibrium:

1. Utilitarian correlated equilibria are those correlated equilibria which maximize the sum of the expected utility of all agents.
2. Egalitarian correlated equilibria are those correlated equilibria which maximize the expected utility of the poorest agent.
3. Elitist correlated equilibria are those correlated equilibria which maximize the expected utility of the happiest agent.
4. Nash-product correlated equilibria are those correlated equilibria which maximize the product of the expected utility of all agents.
5. Opportunity-balanced correlated equilibria are those correlated equilibria which are computed by taking the average of those correlated equilibria which maximize the expect utility of every single agent.

All these optimal correlated equilibria can be efficiently computed using techniques from convex optimization. We then propose two algorithms to transform correlated equilibria to norms.

The idea behind both algorithms is to translate the probability distribution characterized by the correlated equilibrium to randomized signals. In correspondence with those five types of optimal correlated equilibria, five types of norms are studied: utilitarian norms, egalitarian norms, Nash-product norms, elitist norms and opportunity-balanced norms.

Methodology for Objective 2

The general methodology for studying the emergence of norms in Alexander (2007) is the following:

1. Identify norms with a particular strategy in a two-player game.
2. Use replicator dynamics and multiagent learning to test whether norms emerge as result of the repeated play of the two-player game.
3. Test norm emerge with different social networks.

In this thesis, we follow Alexander's general methodology but we study norm emergence in a new game called Ali Baba and the Thief. We identify norms prescribing no harmful behavior with the strategy Ali Baba in this game. In our model this game is repeatedly played by a given amount of agents. Each agent adapts its strategy by using a learning rule among different playing rounds. We consider a norm as emerged in the population if:

(1) All agents are choosing and will continue to choose the action prescribed by the norm.

(2) Every agent believes that all agents, who are relevant in its social network, will choose the action prescribed by the norm in the next round.

(3) Every agent believes that all other agents, who are relevant in its social network, believe that it is good if the agent chooses the action prescribed by the norm.

In our model, individual agents repeatedly play Ali Baba and the Thief with their neighbors. An agent learns its strategy to play the game using replicator dynamics or the learning rule imitate-the-best. We use the Netlogo platform to simulate agents' behavior in the repeated play of this game.

Methodology for Objective 3

Our methodology of the study of the axiomatic representation of norms is to first analyze various derivation rules of input/output logic in isolation and study the corresponding semantics, then combine those results together to obtain alternative semantics for several input/output logics. Our

methodology of the study of the algebraic representation of norms is to introduce variants of the theory of joining-systems and to use those variants as the algebraic semantics of unconstrained input/output logic.

Methodology for Objective 4

We use standard methodology in complexity theory to study the complexity of norm-based deontic logic: we prove the hardness of a logic via reduction and the membership via providing an algorithm on a computational model such as a Turing machine. We show unconstrained input/output logic is in the 1st level of the polynomial hierarchy. Constrained input/output logic are complete for the 2ed level of the polynomial hierarchy. The hardness for the 2ed level of the polynomial hierarchy is proved by a polynomial reduction from the satisfiability/validity problem of quantified Boolean formulas (QBF)

- 2-QBF-SAT: given an arbitrary 2-QBF$^{\exists}$ $\exists p_1 \dots p_m \forall q_1 \dots q_n \Phi$, decide if it is satisfiable.

Negative and static permission checking in input/output logic are in the 1st level of the polynomial hierarchy while dynamic permission checking is complete for the 2ed level of the polynomial hierarchy. We show prioritized input/output logic out_1^p is complete for the 2ed level of the polynomial hierarchy while out_3^p is in the 3ed level of the polynomial hierarchy. We show that Hansen's imperative logic is complete for the 2ed level of the polynomial hierarchy and deontic default logic is in the 3ed level of the polynomial hierarchy. We prove that deontic default logic is Δ_3^p-hard by a polynomial time reduction from the following problem:

- Maximum 2-QBF: given an arbitrary 2-QBF$^{\exists}$ $\exists p_1 \dots p_m \forall q_1 \dots q_n \Phi$, decide if $V_1(p_m) = 1$ where V_1 is the **lexicographically maximal** valuations of $\{p_1, \dots, p_m\}$ such that for all valuation V_2 of $\{q_1, \dots, q_n\}$, $V_1 \cup V_2 \vDash \Phi$,

Here for two valuation of $\{p_1, \dots, p_m\}$, V_1 is lexicographically larger than V_2 iff there exists i such that $V_1(p_i) = 1$, $V_2(p_i) = 0$ and for all $j \in \{1, \dots, i-1\}$, $V_1(p_j) = V_2(p_j)$.

Methodology for Objective 5

Our methodology for objective 5 is to adopt a deontic logic+Boolean game approach to the construction of ethical agents. Boolean game is a class of games based on propositional logic. It was first introduced by Harrenstein et al. (2001) and further developed by several researchers (Harrenstein, 2004; Dunne et al., 2008; Bonzon et al., 2009). We use norm-based deontic logic to reason about norms and use Boolean game to represent the interaction of agents. We use norms

to assess the normative status of strategies. Then agents' preference is changed by the normative status of strategies. Agents of different types use different procedures to change their preferences. We characterize 6 types of ethical agents: moral, amoral, social, selfish, negatively impartial and positively impartial.

1. An *amoral* agent prefers strategy profiles with higher utility.
2. A *moral* agent prefers strategy profiles with higher normative status.
3. - A *selfish* agent first prefers strategy profiles with higher utility.
   - For two strategy profiles of the same utility, the agent prefers the one which contains his strategy of higher normative status.
4. - A *social* agent first prefers strategy profiles which contains his strategy of higher normative status.
   - For two strategy profiles of the same normative status, it prefers strategy profiles with higher utility.
5. - A *negatively impartial* agent first classifies strategies into negatively permitted category and prohibited category.
   - Then it ranks its strategies using utility within these two categories.
6. - A *positively impartial* agent first classifies strategies into positively permitted category and not positively permitted category.
   - Then it ranks its strategies using utility within these two categories.

We study some complexity issues related to normative reasoning/status and agents' preference change. When no restriction is imposed, those decision problems of interest to us are decidable but the complexity is high. Under certain restrictions we obtained intermediate and low complexity.

1.4 A brief introduction to norm-based deontic logic

Compared to standard deontic logic, norm-based deontic logic has the following advantages.

Advantage 1: Norm-based deontic logic solves Jorgensen's dilemma.

Although deontic logic studies normative concepts in general, it is rather difficult how there can be a logic of such concepts at all. Norms like imperatives, promises, legal codes and moral standards are usually not viewed as being true or false. Philosophically, it is widely acknowledged that

there is a distinction between norms on the one hand, and declarative statements on the other. Declarative statements are capable of being true or false; but norms are not. Norms may be complied or violated. But it makes no sense to describe norms as true or as false. For example, a norm "Mary, you may enter now!" does **not describe**, but **demand** a behavior of Mary. Being non-descriptive, norms cannot meaningfully be termed true or false. Lacking truth values, these expressions cannot be premise or conclusion in an inference, be termed consistent or contradictory, or be compounded by truth-functional operators. Hence, while there certainly exists a logical study of normative expressions and concepts, it seems there cannot be a logic of norms: this is Jorgensen's dilemma (Jörgensen, 1938). In norm-based deontic logic, norms do not have truth-value. Therefore norm-based deontic logic solves Jorgensen's dilemma at its starting line.

Advantage 2: Norm-based deontic logic solves the contrary-to-duty paradox.

The contrary-to-duty paradox is the most notorious paradox in deontic logic. The original phrasing of the paradox requires a formalization of the following scenario in which the sentences are mutually consistent and logically independent (Chisholm, 1963).

1. It ought to be that John goes to help his neighbours.
2. It ought to be that if John goes to help his neighbours, then he tells them he is coming.
3. If John doesn't go to help his neighbours, then he ought not to tell them he is coming.
4. John does not go to help.

But formalization using standard deontic logic is either inconsistent or not logically independent. Norm-based deontic logic gives consistent and logically independent formalization of the above scenario therefore solves the contrary-to-duty paradox. In general, norm-based deontic logic provides correct prescriptions in situations where some norms are already violated. Note that norm-based deontic logic is not the only approach to solve the contrary-to-duty paradox. In fact, solving this paradox is an advantage of norm-based logic shared with many other deontic modal logics such as preference-based deontic logic and deontic STIT logic.

Advantage 3: Norm-based deontic logic offers a formal mechanism to deal with moral conflicts.

Consider the following scenario taken from Hansen (2008), which is sometimes called the 'order puzzle': before you go to a party, you become the recipient of various imperative sentences:

1. Your mother says: if you drink anything, then don't drive.
2. Your best friend says: if you go to the party, then you drive.

3. Some acquaintance says: if you go to the party, then have a drink with me.

Assume mother is more important than best friend, who is more important than acquaintance. What will you do? Intuitively, you should obey your mother and your best friend, and hence do the driving and not accept your acquaintance's invitation. However, it is not so clear what formal mechanism could explain this reasoning. Norm-based deontic logic appears as suitable tools to formalize such reasoning.

Advantage 4: Norm-based deontic logic characterizes various notions of permission.

Philosophically, it is common to distinguish between two kinds of permissions: negative permission and positive permission. Negative permission is straightforward to describe: something is negatively permitted according to certain norms iff it is not prohibited by those norms. That is, iff there is no obligation to the contrary. Positive permission is more elusive. Intuitively, something is positively permitted according to certain norms iff it can be derived from those norms. But what exactly does "derive" mean? In mathematics we can derive theorems in a "straight" way or by contradiction. These two methods of derivation give two different notions of positive permission. Makinson and van der Torre (2003) introduces these two types of positive permission as static and dynamic permission. Here is an example from Makinson and van der Torre (2003) to distinguish these two kinds of positive permission.

Example 1.1. *Assume there are two norms:*

1. *A man is obliged to pay tax on condition of having salary.*
2. *A man is permitted to vote on condition of being older than 18.*

Now the question is, according to the given normative system:

Is a man permitted to vote on the condition of having salary?

In one sense the answer is no. If we stick to the straight derivation, a man has salary does not imply that he is older than 18. Therefore we cannot derive he is permitted to vote. This is one notion of positive permission, the static permission.

In another sense the answer is yes. The reason is: suppose we add "a man is not permitted to vote on the condition of having salary" to the normative system. We will make the normative system incoherent in the sense that when the normative system is applied to a man who has salary and is older than 18, then he is permitted to vote meanwhile not permitted to vote. This is another notion of positive permission, the dynamic permission.

Figure 1.1: Input/output logic

Other notions of permission, such as permission as *exception*, have been studied in Stolpe (2010c); Governatori et al. (2013). All these notions of permission can be captured by norm-based deontic logics. Having stated the advantages of norm-based deontic logic, in what follows we review input/output logic, imperative logic and deontic default logic. The complexity of those logics will be developed in later chapters. Here we do not review defeasible deontic logic because the complexity of this logic is already known in the literature (Governatori et al., 2013) and in this thesis we make no contribution to defeasible deontic logic.

1.4.1 Input/output logic

In the first volume of the deontic logic handbook (Gabbay et al., 2013), input/output logic, initiated by Makinson and van der Torre (2000, 2001, 2003) and further developed by Stolpe (2008a,b, 2010b,c); Parent (2011); Parent and van der Torre (2014a); Sun (2014, 2015c,d); Sun and van der Torre (2014), appears as one of the new achievements in deontic logic in recent years. Input/output logic takes its origin in the study of conditional norms. Unlike most deontic modal logic, which mainly adopts possible world semantics, input/output logic uses operational semantics. The basic idea is: norms are conceived as a deductive machine, like a black box which produces normative statements as output, when we feed it factual statements as input. Figure 1.1 is a brief visualization of input/output logic.

Input/output logic avoids assuming that conditional norms bear truth-values. Norms are not embedded in compound formulas using truth-functional connectives. To keep clear of all confusion, norms are not even treated as formulas, but simply as ordered pairs (a, x) of logical formulas. If (a, x) is a *mandatory* norm, then it is read as "given a, x is obligatory". If it is a *permissive* norm, then it is read as "given a, x is permitted".

Just like every logic, two indispensable components of input/output logic is a proof system and a semantics. The semantic construction of input/output logic is based on the intuition that norms are deductive machines which produces obligations as output, when we feed it facts as input. Formally, let $\mathbb{P} = \{p_0, p_1, \ldots\}$ be a countable set of propositional variables and $L_{\mathbb{P}}$ be the propositional language built upon $\mathbb{P}$. Let $O \subseteq L_{\mathbb{P}} \times L_{\mathbb{P}}$ be a set mandatory norms. O can be viewed as a function from $2^{L_{\mathbb{P}}}$ to $2^{L_{\mathbb{P}}}$ such that for a set of formulas A, $O(A) = \{x \in L_{\mathbb{P}} : (a, x) \in O$ for

some $a \in A\}$. Depending on the pairs included in O, a different output is produced against an input A. Makinson and van der Torre (2000) introduce four basic relevant outputs for an input A, called out_1, out_2, out_3, and out_4, which are functions taking O and A as arguments. The semantics of input/output logic from out_1 to out_4 are defined as follows:

- $out_1(O, A) = Cn(O(Cn(A)))$.
- $out_2(O, A) = \bigcap\{Cn(O(V)) : A \subseteq V, V \text{ is complete}\}$.
- $out_3(O, A) = \bigcap\{Cn(O(B)) : A \subseteq B = Cn(B) \supseteq O(B)\}$.
- $out_4(O, A) = \bigcap\{Cn(O(V) : A \subseteq V \supseteq O(V)), V \text{ is complete}\}$.

Here Cn is the classical consequence operator of propositional logic. That is $Cn(A) = \{a \in L_{\mathbb{P}} : A \vDash a\}$. A set of formulas is complete if it is either maximal consistent or equal to $L_{\mathbb{P}}$. These four operators are called 'simple-minded output', 'basic output', 'simple-minded reusable output' and 'basic reusable output' respectively. The typical understanding of $x \in out_i(O, A)$ is: give a set of mandatory norms O and fact A, x is obligatory. For each of these four operators, a throughput version that allows inputs to reappear as outputs is defined as $out_i^+(O, A) = out_i(O_{id}, A)$, where $O_{id} = O \cup \{(a, a) : a \in L_{\mathbb{P}}\}$. When A is a singleton, we write $out_i(O, a)$ for $out_i(O, \{a\})$.

The proof system of input/output logic is build on derivations of mandatory norms. We say that a mandatory norm (a, x) is derivable from a set O iff (a, x) is in the least set that extends $O \cup \{(\top, \top)\}$ and is closed under a number of derivation rules. The following are the derivation rules which are used by Makinson and van der Torre (2000) to construct the proof systems of input/output logic:

- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vdash a$.
- OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.
- WO (weakening the output): from (a, x) to (a, y) whenever $x \vdash y$.
- AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.
- CT (cumulative transitivity): from (a, x) and $(a \wedge x, y)$ to (a, y).
- ID (identity): from nothing to (a, a).

The proof system based on the rules SI, WO and AND is called $deriv_1$. Adding OR to $deriv_1$ gives $deriv_2$. Adding CT to $deriv_1$ gives $deriv_3$. The five rules together give $deriv_4$. Adding ID to $deriv_i$ gives $deriv_i^+$ for $i \in \{1, 2, 3, 4\}$. $deriv_i(O)$ is the smallest set that extends $O \cup \{(\top, \top)\}$ and is

closed under the rules of proof system $deriv_i$. In Makinson and van der Torre (2000), the following soundness and completeness theorems are given:

Theorem 1.1 (Makinson and van der Torre (2000)). *Given a set of mandatory norms O and formula a,*

- $x \in out_i(O, a)$ *iff* $(a, x) \in deriv_i(O)$, *for* $i \in \{1, 2, 3, 4\}$.
- $x \in out_i^+(O, a)$ *iff* $(a, x) \in deriv_i^+(O)$, *for* $i \in \{1, 2, 3, 4\}$.

Input/output logic containing the rule of WO is not free from Ross' paradox (Ross, 1944). In a nutshell, Ross' paradox says that on the one hand, it is unacceptable to claim "you ought to mail a letter" implies "you ought to mail the letter or burn it." On the other hand in standard deontic logic it is valid that $\bigcirc x \rightarrow \bigcirc(x \vee y)$. Therefore there is a confliction between natural language and standard deontic logic. Stolpe (2008a) develops the *mediated reusable input/output logic* such that Ross' paradox is avoided without damaging the power of WO. Stolpe achieves this by replacing WO and CT in $deriv_3$ by OEQ and MCT respectively.

- OEQ (output equivalence): from (a, x) and $x \dashv\vdash y$ to (a, y). Here $x \dashv\vdash y$ means $x \vdash y$ and $y \vdash x$.
- MCT (mediated cumulative transitivity): from (a, x'), $x' \vdash x$ and $(a \wedge x, y)$ to (a, y).

Before building mediated reusable input/output logic, Stolpe develops an input/output logic which is weaker than simple-minded input/output logic using rules SI, OEQ and AND. We call such logic *naive input/output logic*: $deriv_0(O)$ is the smallest set of norms such that $O \cup \{(\top, \top)\} \subseteq deriv_0(O)$ and $deriv_0(O)$ is closed under the rules SI, OEQ and AND. Stolpe introduces the semantics of naive input/output logic as follows:

$$x \in out_0(O, A) \text{ iff } x \in C_{ae}(O(Cn(A))).$$

Here $C_{ae}(A) = \{b \in L_{\mathbb{P}} : b \dashv\vdash \top \text{ or there exist } a_1, \ldots, a_n \in A, \text{ such that } a_1 \wedge \ldots \wedge a_n \dashv\vdash b\}$.

Theorem 1.2 (Stolpe (2008a)). $(a, x) \in deriv_0(O)$ *iff* $x \in out_0(O, a)$.

Stople's mediated reusable input/output logic is an extension of naive input/output logic by incorporating the derivation rule MCT: $deriv_5(O)$ is the smallest set of norms such that $O \cup \{(\top, \top)\} \subseteq deriv_5(O)$ and $deriv_5(O)$ is closed under the rules SI, OEQ, AND and MCT. The semantics of mediated reusable input/output logic is given by an inductive definition: $x \in out_5(O, A)$ iff x is equivalent to a subset of $\bigcup_{i=0}^{\infty} A_i$ where

- $A_0 = O(Cn(A))$, and

- $A_{n+1} = A_n \cup O(Cn(A_n \cup A))$

Theorem 1.3 (Stolpe (2008a)). $(a, x) \in deriv_5(O)$ *iff* $x \in out_5(O, a)$.

Aggregative input/output logic introduced by Parent and van der Torre (2014b) can be viewed as a variant of mediated reusable input/output logic. Parent and van der Torre (2014b) introduce aggregative input/output logic based on the following ideas: on one hand, deontic detachment or cumulative transitivity (CT) is fully in line with the tradition of deontic logic. On the other hand, they also observe that potential counterexamples to deontic detachment may be found in the literature. Parent and van der Torre illustrate this with the following example:

- You ought to exercise hard everyday.
- If you exercise hard everyday, you ought to eat heartily.
- You ought to eat heartily.

Intuitively, the obligation to eat heartily no longer holds, if you take no exercise. Like the others, Parent and van der Torre claim that this counterexample suggests an alternative form of detachment, which keeps track of what has been previously detached. They therefore reject the CT rule, and they accept a weaker rule ACT. As a consequence WO is no longer accepted.

- ACT (aggregative cumulative transitivity): from (a, x) and $(a \wedge x, y)$ to $(a, x \wedge y)$.

The proof system of aggregative input/output logic $deriv_6(O)$ is the smallest set that extends O and closed under the rules SI, OEQ and ACT [‡], while its semantics is defined as follows: $x \in out_6(O, A)$ iff there is finite $O' \subseteq O$ such that

- $O'(Cn(A)) \neq \emptyset$
- for all $B = Cn(B)$, if $A \cup O'(B) \subseteq B$ then $x \dashv\vdash \bigwedge O'(B)$.

Theorem 1.4 (Parent and van der Torre (2014b)). $(a, x) \in deriv_6(O)$ *iff* $x \in out_6(O, a)$.

Constrained and prioritized input/output logic

In the literature, out_0 to out_6 are called unconstrained input/output logic. To deal with the possible inconsistency between the input and the output and solve the most notorious paradox obstructing the development of deontic logic, the contrary-to-duty paradox, constrained input/output logic is introduced in Makinson and van der Torre (2003). Constrained input/output logic allows to determine which norms are triggered in a situation that already violates some of them.

[‡]Note that it is not required that $(\top, \top) \in deriv_6(O)$.

Definition 1.3 (Makinson and van der Torre (2001))**.** *Given a set of mandatory norms O, a set of facts $A \subseteq L_{\mathbb{P}}$ and a set of constrain $C \subseteq L_{\mathbb{P}}$, for $i \in \{1, \ldots, 4\}$,*

- $maxfamily_i(O, A, C) = \{O' \subseteq O : out_i(O', A) \cup C$ *is consistent,* $out_i(O'', A) \cup C$ *is not consistent, for every* $O' \subsetneq O'' \subseteq O\}$.
- $outfamily_i(O, A, C) = \{out_i(O', A) : O' \in maxfamily_i(O, A, C)\}$.
- $out_i^{\cup}(O, A, C) = \bigcup outfamily_i(O, A, C)$.
- $out_i^{\cap}(O, A, C) = \bigcap outfamily_i(O, A, C)$.

To understand these concepts, consider the following instance of the contrary-to-duty paradox from Prakken and Sergot (1996). Suppose we have the following two norms: "The cottage should not have a fence or a dog" and "if it has a dog it must have both a fence and a warning sign." We may formalize these as $O = \{(\top, \neg(f \vee d)), (d, f \wedge w)\}$. Suppose further that we are in the situation that the cottage has a dog, that is $A = \{d\}$. In the context of $A = C$, the first norm is violated. We have

- $maxfamily_1(O, A, C) = \{(d, f \wedge w)\}$
- $outfamily_1(O, A, C) = Cn(f \wedge w)$
- $out_1^{\cup}(O, A, C) = out_1^{\cap}(O, A, C) = Cn(f \wedge w)$

Prioritized input/output logic is an extension of constrained input/output logic by incorporating priorities of norms. Parent (2011), drawing from Boella and van der Torre (2003b), develops prioritized input/output which is capable of handling moral conflicts. Parent (2011) starts the construction by imposing a priority relation $\geq$, which is reflexive and transitive, over mandatory norms O. That is, $\geq$ is a binary relation over O such that for all $(a, x), (b, y), (c, z) \in O$,

- $(a, x) \geq (a, x)$,
- if $(a, x) \geq (b, y)$ and $(b, y) \geq (c, z)$ then $(a, x) \geq (c, z)$,

Here $(a, x) \geq (b, y)$ means (a, x) has higher priority than (b, y). The priority relation over norms is then lifted to priority over sets of norms. Parent uses the lifting originally introduced by Brass (1993): $O_1 \succeq O_2$ iff for all $(a_2, x_2) \in O_2 - O_1$ there is $(a_1, x_1) \in O_1 - O_2$ such that $(a_1, x_1) \geq (a_2, x_2)$. Let $O_1 \succ O_2$ denote that $O_1 \succeq O_2$ but it is not true that $O_2 \succeq O_1$. Let $O^{\geq} = (O, \geq)$ be a set mandatory norms with priority relation $\geq$ and A, C be two sets of formulas where A is the input

and C is the constrains. Parent and van der Torre (2014a) define prioritized input/output logic as follows: [§] for $i \in \{1,2,3,4\}$,

$$x \in out_i^p(O^{\geq}, A, C) \text{ iff } x \in \bigcap\{out_i(O', A) : O' \in preffamily_i(O^{\geq}, A, C)\}.$$

Here $preffamily_i(O^{\geq}, A, C)$ is the set of $\succeq$-maximal elements of $maxfamily_i(O, A, C)$. That is, $O' \in preffamily_i(O^{\geq}, A, C)\}$ if $O' \in maxfamily_i(O, A, C)$ and there is no $O'' \in maxfamily_i(O, A, C)$ such that $O'' \succ O'$.

Permissive input/output logic

Formal definitions of three types of permission are introduced in Makinson and van der Torre (2003).

Definition 1.4 (negative permission (Makinson and van der Torre, 2003))**.** *Given a normative system $N = (O, P)$, where O is a set of mandatory norms and P is a set of permissive norms, and a set of formulas A, $NegPerm_i(N, A) = \{x \in L_{\mathbb{P}} : \neg x \notin out_i(O, A)\}$, for $i \in \{1,2,3,4\}$.*

Intuitively, x is negatively permitted iff x is not forbidden. Since a formula is forbidden iff its negation is obligatory, x is not forbidden is equivalent to $\neg x$ is not obligatory. Permissive norms play no role in negative permission.

Definition 1.5 (static permission (Makinson and van der Torre, 2003))**.** *Given a set of formulas A and a normative system $N = (O, P)$, for $i \in \{1,2,3,4\}$,*

- *If $P = \emptyset$, then $StaPerm_i(N, A) = out_i(O, A)$.*
- *If $P \neq \emptyset$, then $StaPerm_i(N, A) = \{x \in L_{\mathbb{P}} : x \in out_i(O \cup \{(a', x')\}, A)$, for some $(a', x') \in P\}$.*

Intuitively, permissive norms are treated like weak mandatory norms, the only difference is that while the latter may be used jointly, the former may only be applied one by one. As an illustration of such difference, image a situation in which a man is permitted to date either one of two women, but this does not imply that he is permitted to date both of them.

Definition 1.6 (dynamic permission (Makinson and van der Torre, 2003))**.** *Given a finite set of formulas A, a normative system $N = (O, P)$, for $i \in \{1,2,3,4\}$,*

- *$x \in DyPerm_i(N, A)$ iff there is a consistent finite set of formulas C such that $StaPerm_i(N, C) \cup out_i(O \cup \{(\bigwedge A, \neg x)\}, C)$ is inconsistent.*

[§]The prioritized input/output logic introduced in Parent and van der Torre (2014a) is a simplification of the original prioritized input/output logic in Parent (2011).

In Makinson and van der Torre (2003), dynamic permission is defined using notions of derivation systems: $(a, x) \in DyPerm_i(O, P)$ iff $(c, \neg z) \in deriv_i(O \cup \{(a, \neg x)\})$ for some $(c, z) \in staPerm_i(O, P)$ with c being consistent. Here $(c, z) \in staPerm_i(O, P)$ iff $z \in StaPerm_i(N, c)$ where $N = (O, P)$. The readers can verify that our reformulation of dynamic permission is equivalent to the original definition.[¶]

Intuitively, x is dynamically permitted given facts A iff the prohibition of x under condition A, *i.e.* taking $(\bigwedge A, \neg x)$ as an mandatory norm, will create inconsistency of the normative system with respect to some consistent input C.

1.4.2 Imperative logic

In the imperative tradition of deontic logic (Stenius, 1971; Kanger, 1971; van Fraassen, 1973; Alchourrón and Bulygin, 1981; Niiniluoto, 1986; Hansen, 2004, 2005, 2006, 2008), a number of authors have deviated from deontic modal logic to the logical study starts with an explicitly given set of norms or imperatives. The general idea behind imperative logic is that to each imperative there is a descriptive sentence that describes what must be hold iff this imperative is satisfied. If a set of imperatives is under consideration, the set of corresponding descriptive sentences is then used to define deontic operators. The proper representation of this set of imperatives is controversial: directly representing imperatives by a set of descriptive sentences, as Kanger (1971) and Alchourrón and Bulygin (1981) have done, makes it appear as if norms can somehow be reduced to factual statements. Others like van Fraassen (1973), Niiniluoto (1986) and Hansen (2004, 2005, 2006, 2008), have more cautiously represented imperatives by a set of objects that refer to states of affairs or propositions, thereby following the doctrine that norms bear no truth value. In the latter a conditional imperative (*i.e.* mandatory norm) is represented as $a \Rightarrow !x$. An unconditional imperative $\top \Rightarrow !x$ is abbreviately denoted by $!x$.

Hansen's imperative logic

If some given conditional imperatives come into conflict, the best an agent can be expected to do is to follow a maximal subset of the imperatives. Intuitively, a priority ordering of the imperatives can

[¶]Here we sketch the proof: for the sake of simplicity, consider A and C are singletons $\{a\}$ and $\{c\}$ respectively. Suppose $(a, x) \in DyPerm_i(O, P)$, then $(c, \neg z) \in deriv_i(O \cup \{(a, \neg x)\})$ for some $(c, z) \in staPerm_i(O, P)$ with c being consistent. Therefore $\neg z \in out_i(O \cup \{(a, \neg x)\}, c)$ and $z \in StaPerm_i(N, c)$, which means $StaPerm_i(N, c) \cup out_i(O \cup \{(a, \neg x)\}, c)$ is inconsistent. Then we have $x \in DyPerm_i(N, a)$. Suppose $x \in DyPerm_i(N, a)$, then there is a consistent c such that $StaPerm_i(N, c) \cup out_i(O \cup \{(a, \neg x)\}, c)$ is inconsistent. If $StaPerm_i(N, c)$ or $out_i(O \cup \{(a, \neg x)\}, c)$ is inconsistent, say $StaPerm_i(N, c)$ is inconsistent, then $\bot \in StaPerm_i(N, c)$ and $\neg\bot \in out_i(O \cup \{(a, \neg x)\}, c)$. Then $(c, \bot) \in staPerm_i(O, P)$ and $(c, \neg\bot) \in deriv_i(O \cup \{(a, \neg x)\})$, which means $(a, x) \in DyPerm_i(O, P)$. The case for $out_i(O \cup \{(a, \neg x)\}, c)$ being inconsistent is similar. So we assume both $StaPerm_i(N, c)$ and $out_i(O \cup \{(a, \neg x)\}, c)$ are consistent but $StaPerm_i(N, c) \cup out_i(O \cup \{(a, \neg x)\}, c)$ is inconsistent. Then we know there is $z \in StaPerm_i(N, c)$ and $\neg z \in out_i(O \cup \{(a, \neg x)\}, c)$. Therefore $(c, z) \in staPerm_i(O, P)$ and $(c, \neg z) \in deriv_i(O \cup \{(a, \neg x)\})$, which means $(a, x) \in DyPerm_i(O, P)$.

be helpful in determining the relevant sets and resolve conflicts, but a formal resolution mechanism has been difficult to provide. In particular, reasoning about prioritized conditional imperatives is overshadowed by problems such as the *order puzzle* that are not satisfactorily resolved by many existing approaches. Based on unconstrained input/output logic, Hansen (2008) develops prioritized imperative logic to overcome those difficulties.

Hansen introduce *preferred maximally obeyable family* to characterize those norms which are still functioning in a given situation where not all norms can be obeyed. Given a set of prioritized norms or imperatives $O^{>}$, where $>$ is irreflexive and transitive. A full prioritization of $>$ is a strict linear order $\gtrsim$ such that if $\mathfrak{i} \gtrsim \mathfrak{j}$ then $\mathfrak{i} > \mathfrak{j}$ for all $\mathfrak{i}, \mathfrak{j} \in O$. The materialization of O is $m(O) = \{a \to x : (a, x) \in O\}$, which transforms a conditional norm to a material implication.

Definition 1.7 (preferred maximally obeyable family)**.** *(Hansen, 2008, p.29)*[‖] *Given a finite set of prioritized norms $O^{>}$ and a set of formulas A. $O' \in pomfamily(O^{>}, A)$ if there is $\gtrsim$ which is a full prioritization of $>$ such that $O' = \bigcup_{i=0}^{n} O_i$ where we list $\gtrsim$ by $(a_1, x_1), \ldots, (a_n, x_n)$ such that $(a_i, x_i) \gtrsim (a_{i+1}, x_{i+1})$ and*

1. $O_0 = \emptyset$,
2. $O_{i+1} = O_i \cup \{(a_i, x_i)\}$ *if* $A \cup m(O_i \cup \{(a_i, x_i)\})$ *is consistent. Otherwise* $O_{i+1} = O_i$.

Here note that every prioritization creates an element of $pomfamily$. The results after resolving moral conflicts is characterized by the following output operator: for $i \in \{1, 2, 3, 4\}$,

$$x \in out_i^h(O^{>}, A) \text{ iff } x \in \bigcap\{out_i(O', A) : O' \in pomfamily_i(O^{>}, A)\}.$$

1.4.3 Deontic default logic

Reiter's default logic (Reiter, 1980) is one of the most widely used non-monotonic logic in the artificial intelligence community. Extensions of Reiter's default logic by adding priorities over default rules can be found in Brewka (1994); Baader and Hollunder (1995); Rintanen (1998b). Horty's deontic default logic (Horty, 2003, 2007, 2012, 2014) can be viewed as an attempt to reconstruct Reiter's default logic to normative reasoning. Taken from Parent (2011), now we concisely introduce deontic default logic.

[‖]The notations we use are slightly different form Hansen's. In Hansen (2008), a set of norms O is obeyable with respect to facts A if $A \cup m(O)$ is consistent. O' is in $pomfamily(O^{>}, A)$ if it can be obtained from a full prioritization of $>$ by defining

$$O_{(a,x)} = \begin{cases} \bigcup_{(b,y) \gtrsim (a,x)} O_{(b,y)} \cup (a, x), & \text{if } \bigcup_{(b,y) \gtrsim (a,x)} \cup (a, x) \text{ is obeyable with respect to } A. \\ \bigcup_{(b,y) \gtrsim (a,x)} O_{(b,y)}, & \text{otherwise.} \end{cases} \tag{1.1}$$

for all $(a, x) \in O$ and letting $O' = \bigcup_{(a,x) \in O} O_{(a,x)}$ The interested readers can easily verify that despite the difference in notation, we define exactly the same notion as Hansen.

Using notation of input/output logic, a prioritized default theory is a triple $(O, >, A)$ where O is a set of defaults, which is the same as mandatory norms, and $>$ a priority relation over O which is irreflexive and transitive. Like in Reiter's default logic, the goal is to determine the extensions associated with a default theory $(O, >, A)$. Intuitively, an extension gathers all the agent's obligations that follow from what it knows about the world. However, the notion of extension in deontic default logic is not as central as it is in Reiter's theory. The key concept is that of *proper scenario* based on a default theory. A proper scenario is a subset of O satisfying certain conditions. The function of a proper scenario is similar to that of a preferred family in prioritized input/output logic and preferred maximally obeyable family in imperative logic. Intuitively, the defaults in a proper scenario tell us what counts as a binding (good, satisfactory, etc.) reason for what. Thus, if (a, x) is in the proper scenario O' based on a given default theory, then O' is said to provide a as a binding reason for x. The idea is to assume that the agent derives its obligations (part of the extension) from justifications or reasons for those obligations: in particular, that the agent is bounded by an obligation if it possesses a binding reason for that obligation.

Given a scenario $O' \subseteq O$, let $Conclusion(O') = \{x : (a, x) \in O'\}$. Formally, the notion of proper scenario is defined using three other notions. Each corresponds to a condition that a default must meet in order to be binding. The first notion is that of a default being triggered in scenario O', noted as $Triggered_{(O,>,A)}(O')$. The definition runs as follows:

$$Triggered_{(O,>,A)}(O') = \{(a, x) \in O : A \cup Conclusion(O') \vDash a\}.$$

The second notion is that of defaults being conflicted in O'. Let $Conflicted_{(O,\geq,A)}(O')$ denote the set of all such defaults. The definition reads:

$$Conflicted_{(O,>,A)}(O') = \{(a, x) \in O : A \cup Conclusion(O') \vDash \neg x\}.$$

The third notion is that of a default being defeated in O'. For $O_1, O_2 \subseteq O$, let $O_1 \succ O_2$ if for all $(a_1, x_1) \in O_1$, $(a_2, x_2) \in O_2$, $(a_1, x_1) > (a_2, x_2)$. Let $O^{O_1/O_2} = (O - O_1) \cup O_2$, $(a, x) \in Defeated_{(O,>,A)}(O_1)$ if $(a, x) \in O$ and there exists $O_2 \subseteq Triggered_{(O,>,A)}(O_1)$ such that

1. $O_2 \succ \{(a, x)\}$.
2. There is $O_3 \subseteq O_1$ with $O_2 \succ O_3$ such that
   (a) $A \cup Conclusion(O^{O_3/O_2})$ is consistent,
   (b) $A \cup Conclusion(O^{O_3/O_2}) \vDash \neg x$.

Here O_2 can be called a defeating set while O_3 can be called an accommodation set. The idea is that a default (a, x) is defeated by a set of defaults O_1 if we can find a set of defeating default O_2 which

is triggered by O_1 and we can find an accommodation set O_3 in O_1 such that if we replace O_3 by O_2, then the resulting set of defaults is consistent and implies $\neg x$. These three concepts are used to define the notion of a proper scenario.

Definition 1.8 (Proper scenario)**.** *(Horty, 2007, p.380) Let O' be a scenario based on the prioritized default theory $(O, >, A)$. Then O' is a proper scenario based on $(O, >, A)$, noted as $O' \in propScenario(O, >, A)$, just in case $O' = \bigcup_{i \geq 0} O'_i$ where*

- $O'_0 = \emptyset$,
- $O'_{i+1} = \{(a, x) \in O : (a, x) \in Triggered_{(O,>,A)}(O'_i),$
  $(a, x) \notin Conflicted_{(O,>,A)}(O'), (a, x) \notin Defeated_{(O,>,A)}(O')\}$.

The above definition exemplifies an approach to handling inconsistency that is familiar from the literature on non-monotonic reasoning. The key to the construction is to restrict the step-by-step application of defaults in order to guard against possible contradictions. The agent begins its reasoning process, at the initial stage O'_0, without believing in any defaults. Then, at each successive stage O'_i, it supplements its stock of defaults with those that have been triggered at the previous stage O'_{i-1} as long as they are neither conflicted nor defeated. Note that, at each stage O'_i, the constraining scenario against which the agent checks defaults for conflict or defeat is O' itself. With these concepts in place, it is a straightforward matter to define the notion of extension.

Definition 1.9 (Extension)**.** *(Horty, 2007, p.380) Let $(O, >, A)$ be a prioritized default theory and E a set of formulas. E is a an extension of $(O, >, A)$ if $E = Cn(A \cup Conclusion(O'))$ where O' is a proper scenario based on this default theory.*

1.5 A brief introduction to game theory

In deontic logic, norms are simply taken for granted. However, for a full account of norms, we must answer the question related to the generation of norms. For simple norms like imperatives, they can be created quite easily: an announcement of certain authority is sufficient to create a norm. The generation of these norms is not interesting. But there are other norms such as conventions and social norms of which their emergence is worthy of studying. Game theory plays an important role in the study of the generation of conventions and social norms.

Game theory was first formally introduced in the book *The Theory of Games and Economic Behavior* (von Neumann and Morgenstern, 1944). In this section we briefly review some basic notions of game theory. All of them are taken from the open source textbook (Vidal, 2006).

		Bob	
		γ	δ
Alice	α	1,2	4,3
	β	3,2	2,4

Figure 1.2: Sample game matrix in normal-form.

In the simplest game we have two agents each of which must take one of two possible actions. Agents take their actions at the same time. They will then each receive a utility value, or payoff, based on their joint actions. A game can be represented using a payoff matrix which shows the utility that the agents will receive given their actions. Figure 1.2 shows a sample payoff matrix. In this game Alice has two available actions α and β while Bob has action γ and δ. If Alice takes action α and Bob takes action γ then Alice will receive a utility of 1 and Bob a utility of 2. We can extend the payoff matrix to any number of players and actions. In these games we always assume that the players take their actions simultaneously.

A strategy for an agent is a probability distribution over all his actions. A strategy is pure if it assigns probability 1 to a specific action. It is common to express a pure strategy by the unique action which receives positive probability. A strategy profile s is a collection of strategies from agents, one for each agent. In our sample game, a strategy profile $s = (\alpha, \gamma)$ would give Alice a utility of 1 and Bob a utility of 2, in other words, $u_{\text{Alice}}(s) = 1$ and $u_{\text{Bob}}(s) = 2$. We also refer to Alice's strategy in s as s_{Alice}, which is α in this case. This strategy is also an example of a pure strategy: one where the agents take a specific action. Note that both α and γ are pure strategies. In contrast, a mixed strategy is one where the agents take different actions, each with some fixed probabilities. For example, a mixed strategy for Alice is to take action α with probability of 0.3 and action β with a probability of 0.7. Note that in a mixed strategy the probabilities for all actions of each agent have to add up to 1.

Formally, a normal-form game is a tuple $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ where $Agent = \{1, \ldots, n\}$ is the set of agent, Γ_i is the set of all actions of agent i, $u_i : \Gamma_1 \times \ldots \times \Gamma_n \mapsto \mathbb{R}$ is the utility function of agent i which maps every joint action a real number. Given a strategy profile $\mathbf{s} = (s_1, \ldots, s_n)$, the probability assigned to a joint action $\mathbf{a} = (\alpha_1, \ldots, \alpha_n)$ is $p_{\mathbf{s}}(\mathbf{a}) = \prod_i^n s_i(\alpha_i)$. The utility of $\mathbf{s}$ for i is the expected utility he can receive: $\mathbf{u}_i(\mathbf{s}) = \sum_{\mathbf{a} \in \Gamma_1 \times \ldots \times \Gamma_n} u_i(\mathbf{a}) \times p_{\mathbf{s}}(\mathbf{a})$.

Given a game we can't help but ask: what strategy should they use? Which is the best strategy in any given game? The problem, of course, is that there is no simple way to define what's best since what is best for one agent might not be good for another. As such, different solution concepts have been proposed, among which Nash equilibrium is the most famous.

		Bob	
		Stays Silent	Betrays
Alice	Stays Silent	Both serve six months.	B serves 10 years; A goes free.
	Betrays	A serves 10 years; B goes free.	Both serve two years.

Figure 1.3: Payoff matrix for original prisoner's dilemma problem.

We say that a strategy profile $\mathbf{s}$ is a Nash equilibrium if for all agents i, s_i is i's best strategy given that all the other players will play the strategies in $\mathbf{s}$. That is, if everyone else is playing the Nash equilibrium then the best response for everyone to do is to play the Nash equilibrium. Formally, given a strategy profile $\mathbf{s} = (s_1, \ldots, s_n)$, we use s_{-i} to denote $(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$. Let S_i be the set of all strategies of i, a strategy $s_i^* \in S_i$ is a best response for s_{-i} if for all $s_i' \in S_i$, $\mathbf{u}_i(s_i^*, s_{-i}) \geq \mathbf{u}_i(s_i', s_{-i})$. A strategy profile $\mathbf{s} = (s_1, \ldots, s_n)$ is a Nash equilibrium if for all i, s_i is a best response of s_{-i}.

Now we use some famous games to illustrate notions of game theory. The most famous game of all is the Prisoner's Dilemma. Its story typically goes something like the following.

> Two suspects Alice, Bob are arrested by the police. The police have insufficient evidence for a conviction, and having separated both prisoners, visit each of them and offer the same deal: if one testifies for the prosecution against the other and the other remains silent, the silent accomplice receives the full 10-year sentence and the betrayer goes free. If both stay silent, the police can only give both prisoners 6 months for a minor charge. If both betray each other, they receive a 2-year sentence each.

From this story we can generate a payoff matrix as shown in Figure 1.3. We can replace the prison sentences with utility values and arrive at the standard prisoner's dilemma payoff matrix shown in Figure 1.4, note that longer prison terms translate into lower utility. Thus, a 10 year sentence gets a utility of 0, a 2 year sentence has utility of 1, 6 months is 3, and no time served has utility of 5. The actions are labeled cooperate and defect because the suspects can either cooperate with each other and maintain their silence or defect from their coalition and tell on the other one.

Analysis of this matrix reveals that (Defeat, Defeat), or (D, D) for short, is the unique Nash equilibrium in this game. The Prisoner's dilemma is interesting because the best choice is not the rational choice. That is, even thought (C,C) is better than (D,D), a rational player will only play

		Bob	
		Cooperate	Defect
Alice	Cooperate	3,3	0,5
	Defect	5,0	1,1

Figure 1.4: Prisoner's dilemma with standard payoff values.

		Bob	
		Ice Hockey	Football
Alice	Ice Hockey	4,7	3,3
	Football	3,3	7,4

Figure 1.5: Battle of the sexes game.

D as that strategy is dominant. The game is widely studied because it applies to many real-world situations like nuclear disarmament and getting kids to clean up their room.

The battle of the sexes is another popular game. It is shown in Figure 1.5. In this game Alice and Bob like each other and would like to spend time together but must decide, without communicating with each other, where to go. Alice likes football while Bob likes ice hockey. As such, they have a coordination problem where each prefers to go to a different place but they would like to be together. This type of problem arises frequently in multiagent systems where we often have agents that want to cooperate with each other to achieve some larger goal but have conflicting priorities about how to achieve that goal.

After some analysis of this game we can determine that the Nash equilibrium solutions are (I,I) and (F,F). The problem here is that there are two strategies both of which are equally attractive. This is the type of problem that we could fix easily if we just have some norms to guide our behavior.

The chicken game, also known as the hawk-dove game, shown in Figure 1.6, is also common. In this story, two maladjusted teenagers drive their cars towards each other at high speed. The one who swerves first is a chicken and thus loses the game. But, if neither of them swerves then

		Bob	
		Continue	Swerve
Alice	Continue	0,0	5,1
	Swerve	1,5	4,4

Figure 1.6: The chicken game.

		Bob	
		Stag	Hare
Alice	Stag	10,10	0,8
	Hare	8,0	4,4

Figure 1.7: Stag Hunt

they both injured in a horrible crash. After some analysis we can see that this game is very similar to the battle of the sexes. (C,S) and (S,C) are Nash equilibria in this game. Once again there is a coordination problem, bad outcomes can be avoided by imposing norms to this game.

The last popular game we would like to present is the stag hunt. It is shown in Figure 1.7. In this game, two hunters can either jointly hunt a stag or individually hunt a hare. Hunting stags is quite challenging and requires mutual cooperation. If either hunts a stag alone, the chance of success is minimal. In this game the Nash equilibrium solutions are (S,S) and (H,H). Unlike the battle of sexes, in stag hunt one Nash equilibrium can be conceived as strictly better than the other. If there is a system designer in stag hunt, then to maximize the social welfare he will create a norm which prescribes each agent to choose Stag.

1.6 Interdisciplinary aspects and related topics

The study of norms is related to a bunch of scientific communities where normative concepts are interested.

Deontic Logic in Computer Science (DEON) (http://www.deon2014.ugent.be/) is a series of conferences which are designed to promote interdisciplinary cooperation among scholars interested in linking the formal-logical study of normative concepts and normative systems with computer science, artificial intelligence, philosophy, organization theory and law. Most deontic logics are invented by scholars from this community.

Workshops on Coordination, Organization, Institutions and Norms in Multiagent Systems (COIN) (http://coin2015.tbm.tudelft.nl/) is a series of workshops interested in analyzing the social, legal, economic and technological dimensions of agent organizations, and the co-evolution of agent interactions. Among many practical topics, norm generation is studied in this community (Mahmoud et al., 2011, 2012).

The *Formal Ethics* (http://www.formalethics.net/languages/en/index.html) conferences aims at providing an international platform for the discussion and promotion of formal approaches to ethics and to push the frontiers of the research being conduced in this field. The

workshop brings together researchers who are employing formal tools to address questions in ethics and/or political philosophy. The game theoretical perspective of norms is a hot topic within this community (Thoma, 2015).

The annual *International Web Rule Symposium (RuleML)* (http://2015.ruleml.org/) is the leading international event in the field of rules and their applications. Legal RuleML is a sub-track of RuleML which brings together practitioners interested in the theory and applications of legal rules or norms in academic research, industry, engineering, business and other diverse application areas. It provides a forum for stimulating co-operation and cross-fertilization between the many different communities focused on the research and development of legal rule-based systems. Deontic defeasible logic is extensively studied is this community (Governatori and Rotolo, 2010).

The *Workshops on Juris-informatics (JURISIN)* (http://research.nii.ac.jp/~ksatoh/jurisin2015/) focus on a research area which studies legal issues from the perspective of informatics. *International Conference on Legal Knowledge-based Systems (JURIX)* (http://jurix2015.di.uminho.pt/) provides an international forum for academics and practitioners for the advancement of cutting edge research in the interface between law and computer technology. The *International Conference on AI and Law (ICAIL)* (http://sites.sandiego.edu/icail/) provides a forum for the presentation and discussion of the latest research results and practical applications in AI and law and stimulates interdisciplinary and international collaboration. The purpose of those workshops is to discuss both the fundamental and practical issues among people from the various backgrounds such as law, social science, information and intelligent technology, logic and philosophy, including the conventional "AI and law" area.

Agreement technologies refer to computer systems in which autonomous software agents negotiate with one another in order to come to mutually acceptable agreements. An agent may choose whether to fulfill an agreement or not, and it should fulfill it when there is an obligation to do so derived from the standing agreements. The *International Conference Series on Agreement Technologies* (http://ai-group.ds.unipi.gr/eumas-at2015/at2015) is an interdisciplinary forum that brings together researchers and practitioners working on the various topics comprising this emergent and vibrant field. In the handbook of agreement technologies (Ossowski, 2013), 7 chapters are devoted to the study of norms.

The workshop on *Social Norms and Institutions* offers a platform for the exchange of ideas for experts developing and applying theories of social norms and institutions across a diverse range of different social sciences (http://www.socio.ethz.ch/en/news-and-events/events/sni2015.html). Modern research in the field of norms and institutions relies on new theories and

methods such as the concepts and theories of asymmetric information, signaling, social networks, classical and behavioral decision theory and game theory, psychological theories of motivation etc.

1.7 Outline of this thesis

The outline of this thesis is the following. In Chapter 2 and 3 we study the generation of norms in games. In Chapter 4 we study the axiomatic representation of norms. This chapter is an extension of Sun (2014). In Chapter 5 we study the algebraic representation of norms. This chapter is an extension of Sun (2013, 2015d). In Chapter 6 and Chapter 7 the complexity of norm-based deontic logic is studied. Some results of Chapter 6 are presented in Sun and Ambrossio (2015a,b). Only my contributions in Sun and Ambrossio (2015a,b) are included in this thesis. In Chapter 8 we use norm-based deontic logic and games to build ethical agents. This chapter is based on Sun (2015b,a); Sun and Robaldo (2015). Again, only my contributions in Sun and Robaldo (2015) are included in this thesis. Chapter 9 summarizes this thesis with proposed future work.

Chapter 2

Norm Creation in Games

Abstract

In this chapter we study how to create norms in games. We consider norms as normative rules which are used to guide agents' behavior. Such normative rules are created by using correlated equilibrium in games. Our proposal belongs to the offline norm creation approach. Agents' compliance and computational complexity have been identified as interesting problems to cope with in the offline norm creation approach. In this chapter, five types of norms are studied: utilitarian norms, egalitarian norms, Nash-product norms, elitist norms and opportunity-balanced norms. We show that in our framework all these five types of norms can be created in polynomial time and all rational agents will comply with those norms.

2.1 Introduction

Norms have been extensively studied as a mechanism for coordinating interactions within multiagent systems. Two general approaches to the design of norms have been investigated in the literature: online approaches (Shoham and Tennenholtz, 1997; Sen and Airiau, 2007; Morales et al., 2011, 2015) and offline approaches (Shoham and Tennenholtz, 1996; van der Hoek et al., 2007; Ågotnes and Wooldridge, 2010; Ågotnes et al., 2012). Online approaches aim to establish agents with the ability to dynamically coordinate their activities, for example by reasoning explicitly about coordination at run-time or learning from the interaction with other agents. Online approaches can also be termed as the *norm emergence* approaches because norms in these approaches are not designed by any legislator but come to exist by itself in the process of agents' repeated interactions. In contrast, offline approaches aim at developing a coordination device at design-time, and build this regulation into a system for use at run-time. Offline approaches can be termed as the *norm creation* approaches because norms in these approaches are created by system designers. There are arguments in favor of both approaches: online approaches are potentially more flexible, and may be more robust against unanticipated events, while offline approaches benefits from offline reasoning about coordination, thereby reducing the run-time decision-making burden on agents. This chapter belongs to the offline approach and the next chapter belongs to the online approach.

One of the most popular offline approaches is the social law paradigm, originally introduced by Shoham and Tennenholtz (1996). Social laws are understood as a set of rules imposed upon a multiagent system with the goal of ensuring some desirable states. Social laws work by constraining the behavior of the agents in the system, that is, by forbidding agents from performing certain actions in certain states.

There are two disadvantages of the social law paradigm. Firstly, the computational complexity of the creation of social laws that will effectively coordinate a multiagent system is intractable. In Shoham and Tennenholtz (1996), the design of social laws is to find suitable restrictions for the system, which is NP-hard. In van der Hoek et al. (2007), the design of social laws is framed as model checking problem of alternating-time temporal logic. It is again NP-hard. Ågotnes and Wooldridge (2010) extend the model by taking into account both the implementation costs of social laws and multiple (possibly conflicting) design objectives with different priorities. In this setting, the design of social laws becomes an optimization problem and is FP^{NP}-hard. Secondly, there is no guarantee whether agents will comply with the created social laws. There might be conflicts between the interest of agents and the designer. The complexity of the generation of social laws such that every agent will choose to comply is even higher.

In this chapter, following ideas from the game theoretical analysis of norms, we develop a different offline paradigm such that the complexity of the norm creation problem is tractable and every agent will comply with the created norms. We follow Gintis' proposal (Gintis, 2010) and present an alternative offline norm creation framework. Compared to the social law paradigm, the main features of our framework are the following:

1. Instead of constraints, norms in our framework works with randomized signals such as traffic lights, to guide agents' behavior. We generate randomized signals by computing correlated equilibrium of games. Such signals are involved in the description of the triggering condition of norms.

2. Five types of norms are created: utilitarian, egalitarian, elitist, Nash-product and opportunity-balanced norms.

   (a) Utilitarian norms are created from those correlated equilibria which maximize the sum of the expect utility of all agents.

   (b) Egalitarian norms are created from those correlated equilibria which maximize the expected utility of the poorest agent.

   (c) Elitist norms are created from those correlated equilibria which maximize the expected utility of the happiest agent.

   (d) Nash-product norms are created from those correlated equilibria which maximize the product of the expected utility of all agents.

   (e) Opportunity-balanced norms are created from those correlated equilibria which are computed by taking the average of those correlated equilibria which maximize the expect utility of every single agent.

The procedure of norm creation in this chapter is as follows: at first a normal-form game is given. Then we compute a correlated equilibrium of the given game. The resulting correlated equilibrium is a probability distribution over agents' action profiles. We then transform the probability distribution to randomized signals and norms to guide agents' behavior.

The contribution of this chapter is both conceptual and technical. Discussions on utilitarian and egalitarian norms are abound in philosophy and ethics (Sinnott-Armstrong, 2015; Arneson, 2013). While the idea of using correlated equilibrium to interpret norms goes back to Gintis (2010), he did not study specific types of norms. Utilitarian, egalitarian and elitist correlated equilibrium have been introduced in the multiagent learning literature (Greenwald and Hall, 2003). We have not find published articles investigating Nash-product and opportunity-balanced correlated equilibrium so far.

The structure of the rest of this chapter is as follows. In Section 2.2 we explain in detail our framework and show how norms can be created in polynomial time. In Section 2.3 we discuss related work. We summarize this chapter in Section 2.4.

2.2 Norms and correlated equilibrium

In game theory, correlated equilibrium is a solution concept that is more general than Nash equilibrium. But it is insufficiently studied within the normative multiagent system community, if not completely ignored. Correlated equilibrium was first discussed in Aumann (1974). A correlated equilibrium is a probability distribution over agents' action profiles, which can be understood as a public randomized signal, such that each agent can choose his action according to the recommendation of the signal. If no agent would want to deviate from the recommended strategy (assuming the others don't deviate), the probability distribution is called a correlated equilibrium. Formally, given a normal-form game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$, a correlated equilibrium is a probability distribution τ over $\Gamma = \Gamma_1 \times \ldots \times \Gamma_n$ such that for every $i \in Agent$, for every pair of actions $\alpha_i, \alpha_i' \in \Gamma_i$, we have:

$$\sum_{\alpha_{-i} \in \Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i, \alpha_{-i}) \geq \sum_{\alpha_{-i} \in \Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i', \alpha_{-i})$$

where $\Gamma_{-i} = \Gamma_1 \times \ldots \times \Gamma_{i-1} \times \Gamma_{i+1} \times \ldots \times \Gamma_n$.

Intuitively, in a correlated equilibrium the expected utility of an agent following the randomized signal produced by the correlated equilibrium ($\sum_{\alpha_{-i} \in \Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i, \alpha_{-i})$) is larger than his expected utility of deviating from the signal ($\sum_{\alpha_{-i} \in \Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i', \alpha_{-i})$). An immediate positive consequence of this properties is that all those norms created from correlated equilibrium will be obeyed by rational agents because they cannot be better-off by violating those norms. An equivalent characterization of correlated equilibrium is

$$\sum_{\alpha_{-i} \in \Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times (u_i(\alpha_i, \alpha_{-i}) - u_i(\alpha_i', \alpha_{-i})) \geq 0.$$

The difference between $u_i(\alpha_i, \alpha_{-i})$ and $u_i(\alpha_i', \alpha_{-i})$ can be understood as a measure of the *regret* of agent i choosing α_i' instead of α_i, when other agents choose α_{-i}. When $(u_i(\alpha_i, \alpha_{-i}) - u_i(\alpha_i', \alpha_{-i}))$ is positive, agent i has no regret for choosing α_i compared to α_i', when other agents choose α_{-i}. Therefore this characterization of correlated equilibrium shows that there is no expected regret for each agent in a correlated equilibrium, which is why all agents are willing to follow the signal produced by a correlated equilibrium.

Example 2.1 (chicken game, see Chapter 1)**.** *This game has two pure Nash equilibria: pure strategy profile* (C, S) *with expected utility vector* $(5, 1)$ *and* (S, C) *with expected utility vector* $(1, 5)$*. It has one mixed Nash equilibrium: the mixed strategy profile* $(1/2, 1/2)$*, which each agent assign probability* $1/2$ *to* S*, with expected utility vector* $(2.5, 2.5)$*.* τ *is a correlated equilibrium in this game if the following is satisfied:*

- $\tau(C,C) \times 0 + \tau(C,S) \times 5 \geq \tau(C,C) \times 1 + \tau(C,S) \times 4$
- $\tau(S,C) \times 1 + \tau(S,S) \times 4 \geq \tau(S,C) \times 0 + \tau(S,S) \times 5$
- $\tau(C,C) \times 0 + \tau(S,C) \times 5 \geq \tau(C,C) \times 1 + \tau(S,C) \times 4$
- $\tau(C,S) \times 1 + \tau(S,S) \times 4 \geq \tau(C,S) \times 0 + \tau(S,S) \times 5$

Therefore $\tau_1(C,C) = 0.1, \tau_1(C,S) = 0.4, \tau_1(S,C) = 0.3, \tau_1(S,S) = 0.2$ *is a correlated equilibrium with expected utility* $(3.1, 2.7)$*.* $\tau_2(C,S) = \tau_2(S,C) = \tau_2(S,S) = 1/3$ *is another correlated equilibrium with expected utility* $(10/3, 10/3)$*. Note that both* τ_1 *and* τ_2 *assign positive probability to* (S, S)*, in which each agent can be better off by deviate to* C*.*

Compared to Nash equilibrium, the most important advantage of correlated equilibrium in norm creation is that they are computationally less expensive than Nash equilibrium. This can be captured by the fact that computing a correlated equilibrium only requires solving a linear program whereas computing a Nash equilibrium requires solving a mixed integer programming problem. It is known from the algorithmic game theory literature that correlated equilibrium of normal-form games can be computed in polynomial time (Papadimitriou and Roughgarden, 2008), while computing Nash equilibrium is PPDA-complete (Daskalakis et al., 2009).

Although we accepted that norms are created from correlated equilibrium, this does not mean each correlated equilibrium should be implemented to norms. Instead, we are only interested in those correlated equilibria which satisfy certain desired properties, for example maximize social welfare, protect the weakest member of the society, or ensure fairness. In the theory of multiagent resource allocation (Chevaleyre et al., 2006), four types of social welfare are widely used:

- Utilitarian social welfare is the sum of the utilities of all agents.
- Egalitarian social welfare is the utility of the weakest agents.
- Nash-product social welfare is the product of the utilities of all agents.
- Elitist social welfare is the utility of the strongest agents.

In correspondence with those notions of social welfare, we are interested in studying utilitarian /egalitarian/Nash-product/elitist correlated equilibrium. In this chapter we moreover introduce opportunity-balanced correlated equilibrium, which in certain sense ensure fairness.

2.2.1 Utilitarian correlated equilibrium

For each agent i, his expected utility in a correlated equilibrium τ is calculated as follows:

$$EU_i(\tau) = \sum_{\mathbf{a}\in\Gamma} \tau(\mathbf{a}) \times u_i(\mathbf{a})$$

Given a normal-form game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ and a correlated equilibrium τ. The utilitarian social welfare induced by τ is

$$SW^u(\tau) = \sum_{i\in Agent} EU_i(\tau)$$

Utilitarian social welfare sums up the agents' expected utilities in a given correlated equilibrium, thus provides a useful measure of the overall benefit of the society.

Definition 2.1 (utilitarian correlated equilibrium (Greenwald and Hall, 2003))**.** *Given a normal-form game, and Θ the set of all correlated equilibrium of this game. $\tau \in \Theta$ is a utilitarian correlated equilibrium if τ maximizes $SW^u(\tau)$, i.e. for all $\tau' \in \Theta$, $SW^u(\tau) \geq SW^u(\tau')$.*

A utilitarian correlated equilibrium can be computed using linear programming with maximizing $SW^u(\tau)$ as the objective function and requirements of correlated equilibrium as constrains. Utilitarian norms are normative rules which describe those randomized signals implemented from utilitarian correlated equilibrium. For example, in the chicken game utilitarian correlated equilibrium are solutions of the following linear program:

$$\max \tau(C,C) \times (0+0) + \tau(C,S) \times (5+1) + \tau(S,C) \times (1+5) + \tau(S,S) \times (4+4)$$

subject to

- $\tau(C,C) \times 0 + \tau(C,S) \times 5 \geq \tau(C,C) \times 1 + \tau(C,S) \times 4$
- $\tau(S,C) \times 1 + \tau(S,S) \times 4 \geq \tau(S,C) \times 0 + \tau(S,C) \times 5$
- $\tau(C,C) \times 0 + \tau(S,C) \times 5 \geq \tau(C,C) \times 1 + \tau(S,C) \times 4$
- $\tau(C,S) \times 1 + \tau(S,S) \times 4 \geq \tau(C,S) \times 0 + \tau(S,S) \times 5$
- $\tau(C,C) + \tau(C,S) + \tau(S,C) + \tau(S,S) = 1$

- $\tau(C,C), \tau(C,S), \tau(S,C), \tau(S,S) \geq 0$

Let $\tau(S,S)$ be x_1, $\tau(C,S)$ be x_2, $\tau(S,C)$ be x_3 and $\tau(C,C)$ be x_4. This linear program is transformed to:

$$\max 8x_1 + 6x_2 + 6x_3$$

subject to

- $x_1 - x_3 \leq 0$
- $-x_2 + x_4 \leq 0$
- $x_1 - x_2 \leq 0$
- $-x_3 + x_4 \leq 0$
- $x_1 + x_2 + x_3 + x_4 = 1$
- $x_1, x_2, x_3, x_4 \geq 0$

Using techniques of linear programming, we find a solution for the linear program with $x_1 = x_2 = x_3 = \frac{1}{3}$. Such a utilitarian correlated equilibrium can be transformed to randomized signals. For example we can use a random number generator to generate a real number between 0 and 1. If the number is less than $\frac{1}{3}$, then we show a red signal to agent 1 and a green signal to agent 2. If the number is larger than $\frac{2}{3}$, then we show a green signal to agent 1 and a red signal to agent 2. Otherwise we show a red signal to both agents. Then we create the following two norms for both agent 1 and agent 2:

- if the signal you see is red, then you should *Swerve*.
- if the signal you see is green, then you should *Continue*.

Note that to ensure each agent to comply with those norms, we should make the probability distribution of signals public meanwhile keep signals private for each agent. When one agent sees both agents' signals, it is possible that he want to violate the norms. For example, when agent 1 see a red signal for himself as well as a red signal for agent 2, then he has an incentive to choose C, which is a violation of his norm. But if agent 1 only sees his own red signal, then he knows that the probability of the signal for agent 2 being red is $\frac{1}{2}$. Therefore if he chooses C, his expected utility will be $\frac{5}{2}$, which is no larger than his expected utility of choosing S. The incentive of violation disappears. Now consider a different example where one ambulance meets a bus in a crossroad.

	Stop	*Continue*
Stop	$(-10, 0)$	$(-10, 5)$
Continue	$(10, 0)$	$(-20, -10)$

Table 2.1: Ambulance game

Example 2.2. *(ambulance game) The game is depicted by the payoff matrix in Table 2.1, with ambulance being the row player. τ is a correlated equilibrium in this game if the following is satisfied:*

- $\tau(S,S) \times (-10) + \tau(S,C) \times (-10) \geq \tau(S,S) \times 10 + \tau(S,C) \times (-20)$
- $\tau(C,S) \times 10 + \tau(C,C) \times (-20) \geq \tau(C,S) \times (-10) + \tau(C,C) \times (-10)$
- $\tau(S,S) \times 0 + \tau(C,S) \times 0 \geq \tau(S,S) \times 5 + \tau(C,S) \times (-10)$
- $\tau(S,C) \times 5 + \tau(C,C) \times (-10) \geq \tau(S,C) \times 0 + \tau(C,C) \times 0$

Let $\tau(S,S)$ be x_1, $\tau(C,S)$ be x_2, $\tau(S,C)$ be x_3 and $\tau(C,C)$ be x_4. The linear program we need to calculate utilitarian correlated equilibria is:

$$max\ z = -10x_1 + 10x_2 - 5x_3 - 30x_4$$

subject to

- $20x_1 - 10x_3 \leq 0$
- $-20x_2 + 10x_4 \leq 0$
- $5x_1 - 10x_2 \leq 0$
- $-5x_3 + 10x_4 \leq 0$
- $x_1 + x_2 + x_3 + x_4 = 1$
- $x_1, x_2, x_3, x_4 \geq 0$

Using techniques of linear programming, we find a solution for the linear program with $x_2 = 1, x_1 = x_3 = x_4 = 0$. Such a utilitarian correlated equilibrium can be implemented to a norm:

- *The bus should stop and the Ambulance should continue.*

The norm created in the ambulance game explains real traffic rule quite well.

Theorem 2.1. *Utilitarian correlated equilibria can be computed in polynomial time.*

Proof. A utilitarian correlated equilibrium can be computed using linear programming with the objective function maximizing the sum of the expected utility of all agents. A linear program can be solved in polynomial time using standard techniques like ellipsoid algorithms (Alevras and Padberg, 2001). □

Not only correlated equilibria can be computed efficiently. Norms can also be created from correlated equilibria. The following algorithm shows how to create norms from an arbitrary correlated equilibria and use them to coordinate agents' behavior. Given a normal-form game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ and a correlated equilibrium τ,

1. For every agent i, for each of his action α_i, prepare a unique signal b_i.
2. For each action profile $(\alpha_1, \ldots, \alpha_n)$ such that $\tau(\alpha_1, \ldots, \alpha_n) \neq 0$. Create a norm of the form "if you see b_i, then you should do α_i".
3. Suppose $\Gamma_1 \times \ldots \times \Gamma_n = \{\mathbf{a}^1, \ldots, \mathbf{a}^k\}$ and $\tau(\mathbf{a}^j) = x_j$. We divide the interval $[0, 1)$ to sub-intervals $[0, x_1), [x_1, x_1 + x_2), \ldots, [x_1 + \ldots + x_{n-1}, 1)$.
4. Use a random number generator to generate a real number y between $[0, 1)$.
5. According to which sub-interval y belongs to, we choose a sub-interval. If y is in $[x_1 + \ldots + x_{k-1}, x_1 + \ldots + x_k)$, then we choose $\mathbf{a}^k = (\alpha_{k1}, \ldots, \alpha_{kn})$.
6. Send single b_{k1} to agent 1; send single b_{k2} to agent 2,$\cdots$, send single b_{kn} to agent n.

Although the above algorithm faithfully transforms correlated equilibria to norms, it has a drawback in the sense that many signals are involved in such transformation. It is possible to approximately transform correlated equilibria to norms without using artificial signals, as the following algorithm shows. Given a game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ and a correlated equilibrium τ,

1. Suppose $\Gamma_1 \times \ldots \times \Gamma_n = \{\mathbf{a}^1, \ldots, \mathbf{a}^k\}$ and $\tau(\mathbf{a}^j) = x_j$. We divide the interval $[0, 1)$ to sub-intervals $(0, x_1], (x_1, x_1 + x_2], \ldots, (x_1 + \ldots + x_{n-1}, 1]$.
2. Time every sub-interval with 365, so we have $(0, 365x_1], (365x_1, 365(x_1 + x_2)], \ldots, (365(x_1 + \ldots + x_{n-1}), 365]$.
3. For an integer $k \in \{1, \ldots, 365\}$, if $k \in (365(x_1 + \ldots + x_{k-1}), 365(x_1 + \ldots + x_k)]$, then we choose α^k.
4. Suppose $\mathbf{a}^k = (\alpha_{k1}, \ldots, \alpha_{kn})$. We create the following norms: "in the kth day of a year, agent 1 should do α_{k1}, agent 2 should do α_{k2},. . ., agent n should do α_{kn} "

This algorithm uses dates as a kind of natural signals in the creation of norms, therefore no artificial signal is needed.

Utilitarian norms are created from a utilitarian equilibrium in the form of normative rules via the above algorithms. Syntactically those norms are of the form "if x is true, y should be done" or "given the condition x, y is obligatory". Such representation of norms coincides with the norm representation in norm-based deontic logics. These two algorithms are also used to transform other correlated equilibria to norms.

2.2.2 Egalitarian correlated equilibrium

The concept of egalitarian social welfare is inspired by the work of Rawls (1971). Intuitively, egalitarian social welfare is measured by the situation of the poorest member of the society. It therefore provides a useful measure of fairness in cases where the minimum need of all agents are to be satisfied. For example, distributing humanitarian aid items among the needy population in a disaster area. Guaranteeing every survivor's continuing survival is the primary goal in such a situation, and it is best captured by the notion of egalitarian social welfare. Given a game $(Agent, \Gamma_1, \ldots \Gamma_n, u_1, \ldots, u_n)$ and a correlated equilibrium τ. The egalitarian social welfare induced by τ is

$$SW^e(\tau) = min\{EU_i(\tau) : i \in Agent\}$$

Definition 2.2 (egalitarian correlated equilibrium (Greenwald and Hall, 2003))**.** *Given a game and* Θ *the set of all correlated equilibrium of this game.* $\tau \in \Theta$ *is a egalitarian correlated equilibrium if* τ *maximizes* $SW^e(\tau)$*, i.e. for all* $\tau' \in \Theta$*,* $SW^e(\tau) \geq SW^e(\tau')$ *.*

Egalitarian norms are normative rules created from egalitarian correlated equilibria. Computing egalitarian correlated equilibrium seems to be more difficult than computing utilitarian correlated equilibrium at the first sight because maximizing egalitarian social welfare is not a linear function. However, the good news is, we can still use linear programming to compute egalitarian correlated equilibrium efficiently, as the following theorem shows.

Theorem 2.2. *Egalitarian correlated equilibrium can be computed in polynomial time.*

Proof. We use linear programming to compute egalitarian correlated equilibrium. The variables of the linear programming are random variables $\tau(a), \tau(a'), \ldots$, representing the probability assigned to each joint action of the game. The objective function of the linear programming is to maximize the egalitarian social welfare, which is represented by a variable z. There are four groups of constrains in this linear program.

1. $\tau(\mathbf{a}) \geq 0$, for all $\mathbf{a} \in \Gamma_1 \times \ldots \times \Gamma_n$.

2.

$$\sum_{\mathbf{a}\in\Gamma_1\times\ldots\times\Gamma_n} \tau(\mathbf{a}) = 1$$

These two groups of constrains together ensures that the function τ is a probability distribution over strategy profiles.

3. For all i, for all α_i and $\alpha_i' \in \Gamma_i$,

$$\sum_{\alpha_{-i}\in\Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i, \alpha_{-i}) \geq$$

$$\sum_{\alpha_{-i}\in\Gamma_{-i}} \tau(\alpha_i, \alpha_{-i}) \times u_i(\alpha_i', \alpha_{-i})$$

The third group of constrains say that the solution of this linear programming must be a correlated equilibrium.

4. For all i,

$$\sum_{\mathbf{a}\in\Gamma_1\times\ldots\times\Gamma_n} u_i(\mathbf{a})\tau(\mathbf{a}) \geq z$$

The fourth group of constrains ensures that z is no larger than the expected utility of any agent in this correlated equilibrium.

These four groups of constrains together ensure that the solution of this linear program indeed maximizes the egalitarian social welfare. It can be verified that the size of this linear program is polynomial to the size of the given game. Since a linear program can be solved in polynomial time, we know that egalitarian correlated equilibrium can be computed in polynomial time. □

2.2.3 Nash-product correlated equilibrium

Given a game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ in which utility function is non-negative. The Nash-product social welfare induced by a correlated equilibrium τ is

$$SW^n(\tau) = \prod_{i\in Agent} EU_i(\tau)$$

An outcome maximizes Nash-product social welfare if it maximizes the product of the individual agent utilities. This idea goes back to John Nash's famous solution to the bargaining problem (Nash, 1950). Nash-product social welfare can be regarded as a compromise between the

utilitarian and the egalitarian social welfare. On the one hand, just as utilitarian social welfare, the Nash-product increases with single increasing of individual utilities. On the other hand, just as egalitarian social welfare, the Nash-product reaches its maximum when the utilities distributed equally over all agents.

Definition 2.3 (Nash-product correlated equilibrium)**.** *Given a game and Θ the set of all correlated equilibrium of this game, $\tau \in \Theta$ is a Nash-product correlated equilibrium if τ maximizes $SW^n(\tau)$, i.e. for all $\tau' \in \Theta$, $SW^n(\tau) \geq SW^n(\tau')$.*

Nash-product norms are normative rules created from Nash-product correlated equilibria. The following theorem shows that the computation of Nash-product correlated equilibrium is tractable.

Theorem 2.3. *Nash-product correlated equilibrium can be computed in polynomial time.*

Proof. We use convex optimization to compute Nash-product correlated equilibrium.* Suppose $\mathbf{a}^1, \ldots, \mathbf{a}^k$ are all action profiles in the given game. We use $\tau(\mathbf{a}^1), \ldots, \tau(\mathbf{a}^k)$ to represent the probability assigned to each strategy profile of the game. To compute Nash-product correlated equilibrium we need to maximize the Nash-product social welfare:

$$\prod_{i \in Agent} \sum_{\mathbf{a} \in \Gamma} \tau(\mathbf{a}) \times u_i(\mathbf{a}) =$$

$$\prod_{i \in Agent} (\tau(\mathbf{a}^1) \times u_i(\mathbf{a}^1) + \ldots + \tau(\mathbf{a}^k) \times u_i(\mathbf{a}^k)) =$$

$$(\tau(\mathbf{a}^1) \times u_1(\mathbf{a}^1) + \ldots + \tau(\mathbf{a}^k) \times u_1(\mathbf{a}^k)) \times \ldots \times (\tau(\mathbf{a}^1) \times u_n(\mathbf{a}^1) + \ldots + \tau(\mathbf{a}^k) \times u_n(\mathbf{a}^k)) =$$

$$(\tau(\mathbf{a}^1))^n \times \prod_{i \in Agent} u_i(\mathbf{a}^1) + \ldots + (\tau(\mathbf{a}^k))^n \times \prod_{i \in Agent} u_i(\mathbf{a}^k)$$

Note that the above function is a convex function because every goal of every agent is non-negative, which ensures that $u_i(\alpha^j)$ is non-negative. The above function is maximized iff the following function is minimized

$$-\log((\tau(\mathbf{a}^1))^n \times \prod_{i \in Agent} u_i(\mathbf{a}^1) + \ldots + (\tau(\mathbf{a}^k))^n \times \prod_{i \in Agent} u_i(\mathbf{a}^k)) \qquad (\divideontimes)$$

Note that $(\divideontimes)$ is also a convex function because the function $f(x) = -\log x$ is a convex function and the composition of two convex functions is again a convex function. Now we take minimizing $(\divideontimes)$ as our objective function and built a convex optimization problem by imposing four groups of constrains the same as in the proof of Theorem 2. Such a convex optimization problem can be solved using standard techniques from convex optimization, say interior-point methods, in polynomial time. □

*A comprehensive introduction of convex optimization can be found in Boyd and Vandenberghe (2004).

2.2.4 Elitist correlated equilibrium

Given a game $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$ and a correlated equilibrium τ, the elitist social welfare induced by τ is

$$SW^{eli}(\tau) = max\{EU_i(\tau) : i \in Agent\}$$

Intuitively, elitist social welfare is measured by the situation of the happiest member of the society. Maximizing elitist social welfare reflects the famous *Matthew effect* in sociology which describes the phenomenon where "the rich get richer and the poor get poorer".[†] The elitist social welfare is clearly not a fair measure for social welfare, but it can be useful in cooperation based applications where we require only one agent to achieve its goals.

Definition 2.4 (elitist correlated equilibrium (Greenwald and Hall, 2003))**.** *Given a game and Θ the set of all correlated equilibrium of this game, $\tau \in \Theta$ is a elitist correlated equilibrium if τ maximizes $SW^{eli}(\tau)$, i.e. for all $\tau' \in \Theta$, $SW^{eli}(\tau) \geq SW^{eli}(\tau')$.*

Note that elitist correlated equilibrium is called republican correlated equilibrium in Greenwald and Hall (2003). We believe "elitist" is a better name therefore we will stick to this new name. Elitist norms are normative rules created from elitist correlated equilibria. Elitist correlated equilibrium can be computed in polynomial time using techniques from convex optimization.

Theorem 2.4. *Elitist correlated equilibrium can be computed in polynomial time.*

Proof. We use convex optimization to compute elitist correlated equilibrium. The variables of the linear programming are random variables $\tau(\mathbf{a}), \tau(\mathbf{a}'), \ldots$, representing the probability assigned to each strategy profile of the game. The objective function is

$$\min\{-EU_1(\tau), \ldots, -EU_n(\tau)\}.(*)$$

Note that $(*)$ is minimized iff $\max\{EU_1(\tau), \ldots, EU_n(\tau)\}$ is maximized, which means the elitist social welfare is maximized. Moreover, since $-EU_i(\tau) = \sum_{\mathbf{a}\in\Gamma} \tau(\mathbf{a}) \times (-u_i(\mathbf{a}))$ is convex we know $(*)$ is also convex. There are three groups of constrains in this convex optimization problem, which are exactly the same as the first three groups of constrains in computing egalitarian correlated equilibrium. Such a convex optimization problem can be solved using standard techniques from convex optimization in polynomial time. □

[†]https://en.wikipedia.org/wiki/Matthew_effect

2.2.5 Opportunity-balanced correlated equilibrium

A fifth kind of correlated equilibrium of interest to us is what we call opportunity-balanced correlated equilibrium. Intuitively, an opportunity-balanced correlated equilibrium is the average of those correlated equilibrium which maximizes each single agent's expected utility.

Definition 2.5 (opportunity-balanced correlated equilibrium)**.** *Given a game* $(Agent, \Gamma_1, \ldots, \Gamma_n, u_1, \ldots, u_n)$*, an opportunity-balanced correlated equilibrium is a correlated equilibrium* τ *such that* $\tau(\boldsymbol{a}) = (\tau_i(\boldsymbol{a}) + \ldots + \tau_n(\boldsymbol{a}))/n$*, where* τ_i *is a correlated equilibrium which maximizes the expected utility of agent* i*.*

The set of all correlated equilibria of a game is a convex set because correlated equilibrium is defined using linear constrains. Therefore the average of finite correlated equilibria is again a correlated equilibrium.

Example 2.3. *(opportunity-balanced correlated equilibrium for the chicken game) We first solve two linear programs with objective function max* $4x_1 + 5x_2 + x_3$ *and* $4x_1 + x_2 + 5x_3$ *respectively. Solving the first linear program gives us* τ_1 *with* $\tau_1(x_2) = 1, \tau_1(x_1) = \tau_1(x_3) = \tau_1(x_4) = 0$*. Solving the second linear program gives us* τ_2 *with* $\tau_2(x_3) = 1, \tau_2(x_1) = \tau_2(x_2) = \tau_2(x_4) = 0$*. Then we calculate the opportunity-balanced correlated equilibrium* τ *by taking the average of* τ_1 *and* τ_2*:* $\tau(x_2) = \tau(x_3) = \frac{1}{2}, \tau(x_1) = \tau(x_4) = 0$*.*

Example 2.4. *(opportunity-balanced correlated equilibrium for the ambulance game) We first solve two linear programs with objective function max* $-10x_1 + 10x_2 - 10x_3 - 20x_4$ *and* $5x_3 - 10x_4$ *respectively. Solving the first linear program gives us* τ_1 *with* $\tau_1(x_2) = 1, \tau_1(x_1) = \tau_1(x_3) = \tau_1(x_4) = 0$*. Solving the second linear program give us* τ_2 *with* $\tau_2(x_3) = 1, \tau_2(x_1) = \tau_2(x_2) = \tau_2(x_4) = 0$*. Then we calculate the opportunity-balanced correlated equilibrium* τ *by taking the average of* τ_1 *and* τ_2*:* $\tau(x_2) = \tau(x_3) = \frac{1}{2}, \tau(x_1) = \tau(x_4) = 0$*.*

Opportunity-balanced norms are normative rules created from opportunity-balanced correlated equilibria. The following theorem shows that the computation of opportunity-balanced correlated equilibrium is tractable.

Theorem 2.5. *Opportunity-balanced correlated equilibria can be computed in polynomial time.*

Proof. Given an arbitrary with n agents. An opportunity-balanced correlated equilibrium can be computed as follows:

1. For each agent, use linear programming to compute a correlated equilibrium which maximizes the agent's expected utility.

2. Take the average of the n correlated equilibria from the previous step.

Since each linear program can be solved in polynomial time, an opportunity-balanced correlated equilibrium can be computed in polynomial time. □

2.3 Related work

Except the literature mentioned in the introductory section, the generation of norms is also studied in sociology and philosophy. Bicchieri's analysis of norms (Bicchieri, 2006) originates from conditional behavioral rules. In her approach, a norm is a behavioral rule that agent i prefers to perform concerning her belief about other individuals' actions and what they expect i to conform. In her account, a norm is a function defines the agent's strategy in response to other agents' preferences. In Binmore (2005) norms are tools to solve the equilibrium selection problem in a coordination game with multiple Nash equilibria. His approach is based on evolutionary game theory as he believes that the interaction of agent's strategies are infinitely repeated games. While all equilibria in a game are not pareto optimal, norms arise to solve the equilibrium selection problem. An evolutionary game theoretical approach of norm emergence can be found in Alexander (2007) and Skyrms (2014). More discussions about Alexander (2007) and Skyrms (2014) can be found in the next chapter.

We understand creation and emergence as two different methods of norm generation corresponds to offline norm design and online norm design in computer science respectively. It seems concepts from evolutionary game theory, like evolutionary stable strategy and replicator dynamics, are more suitable for norm emergence, while concepts from traditional game theory, like Nash/correlated equilibrium, are more suitable for norm creation.

Boella and van der Torre (2003c) argue that to reason about the creation of norms, we need a model of norm-evading agents. A norm-evading agent is an agent who looks for ways to violate the norm while at the same time evading the sanction. Boella and van der Torre (2003a) present a model of norm-evading agents based on the attribution of mental attitudes to normative systems. Boella and van der Torre (2003c) address the following two questions: 1. How can the attribution of mental attitudes to normative systems be used to reason about norm creation? 2. How can we formalize norm creation using the attribution of mental attitudes to normative systems?

In the multiagent system community, non-game theoretical approaches to norm generation are introduced. Morales et al. (2014, 2015) introduce minimality and simplicity as two crucial factors in the evaluation of the process of norm generation. Simplicity and minimality are respectively referred to the computational complexity of the algorithm and the number of norms that are

generated in the process. These two metrics together imply the concept of norm compactness which provides more liberty for autonomous agents. The two most recent systems designed by Morales *et al* regarding minimality and simplicity are SIMON and LION (Morales et al., 2014, 2015).

2.4 Summary

In this chapter we have studied how to create norms in games. We have considered norms as normative rules which are used to guide agents' behavior. Such normative rules were created by using correlated equilibrium in games. Five types of norms have been studied: utilitarian norms, egalitarian norms, Nash-product norms, elitist norms and opportunity-balanced norms. All these norms can be created in polynomial time in our framework. Moreover, since all these norms are created from correlated equilibrium, it is to each agent's interest to comply with these norms.

Chapter 3

Norm Emergence in Games

Abstract

In this chapter we propose a model that supports the emergence of norms via multiagent learning in social networks. In our model, individual agents repeatedly interact with their neighbors in a game called Ali Baba and the Thief. An agent learns its strategy to play the game using the learning rule, imitate-the-best. Our results show that some norms prohibiting harmful behaviors, such as "you should not rob", can emerge after repeated interactions among agents inhabited in some social networks. Our experimental results suggest that there is a critical point of norm emergence which is decided by the quotient of the initial utility and the amount of robbery in Ali Baba and the Thief. When the quotient of the initial utility and the amount of robbery is smaller than the critical point, the probability of norm emergence is high. The probability drops dramatically as long as the quotient is larger than the critical point.

3.1 Introduction

In the literature of multiagent systems, the online approaches of norm generation (Shoham and Tennenholtz, 1997; Sen and Airiau, 2007; Morales et al., 2011, 2015) aim to establish agents with the ability to dynamically coordinate their activities, for example by reasoning explicitly about coordination at run-time or learning from the interaction with other agents. Online approaches can also be termed as the *norm emergence* approaches because norms in these approaches are not designed by any legislator but come to exist by themselves in the process of repeated interactions between agents.

Norm emergence is also studied by philosophers. An evolutionary game theoretical approach of norm emergence can be found in Alexander (2007) and Skyrms (2014). In this approach, two different features are emphasized: relatively simple learning processes and networked interactions. Both Alexander and Skyrms explored a variety of games such as the prisoner's dilemma and the stag hunt to illustrate the emergence of norms in different situations. Though Skyrms used the replicator dynamics occasionally, both of them tended to adopt simple learning rules like "imitate-the-best" because such rules are less cognitively demanding. Alexander justified the use of these simple rules on the grounds that they are extremely simple to follow for agents of bounded rationality. These simple learning rules provide the same function as the replicator dynamics: between different rounds of play, agents rely on their learning rules to decide which strategies to adopt.

The general methodology for studying the emergence of norms in Alexander (2007) is the following:

1. Identify norms with a particular strategy in a two-player game.
2. Use replicator dynamics and multiagent learning to test whether norms emerge as a result of the repeated play of the two-player game.
3. Test norm emerge with different social networks.

Two-player games studied in Alexander (2007) includes prisoner's dilemma, stag hunt, cake cutting and ultimatum game. Alexander used these games to analyze the emergence of norms of cooperation, trust, fair division and retaliation respectively. In this chapter, we follow Alexander's general methodology but we study norm emergence in a game called Ali Baba and the Thief, which is not explored in Alexander (2007). We propose Ali Baba and the Thief as a variant of the chicken game. In this 2-player game, each agent has two strategies: Ali Baba and Thief. Each agent has initial utility x. If both agents choose Ali Baba, then their utilities do not change. If they both

	Ali Baba	Thief
Ali Baba	x,x	$x-d$,$x+d$
Thief	$x+d$,$x-d$	0,0

Table 3.1: Ali Baba and the Thief

choose Thief, then there is a fight between them and they are both injured. The resulting utility is 0. If one chooses Ali Baba and the other chooses Thief, then Thief robs Ali Baba and the utility of the one who chooses Thief increases by d and the other one decreases by d, where $0 \leq d \leq x$. We call d the amount of robbery. The payoff matrix of this game is shown in Table 3.1.

We identify norms prescribing no harmful behavior with the strategy Ali Baba in this game. In our model this game is repeatedly played by a given amount of agents. Each agent adapts its strategy using a learning rule between different rounds of play. We say a norm has emerged in the population if:

(1) All agents are choosing and will continue to choose the action prescribed by the norm.

(2) Every agent believes that all agents, who are relevant in its social network, will choose the action prescribed by the norm in the next round.

(3) Every agent believes that all other agents, who are relevant in its social network, believe that it is good if the agent chooses the action prescribed by the norm.

The above three criteria of norm emergence is a reformulation of Lewis' famous analysis of conventions: "Everyone conforms, everyone expects others to conform, and everyone has good reasons to conform because conforming is in each person's best interest when everyone else plans to conform" (Lewis, 1969). In our game, we are interested in the situation where all agents choose Ali Baba. This can be understood as no agent is willing to be Thief, which shows norms like "you should not rob" or "don't harm others" have emerged.

The structure of this chapter is the following: in Section 3.2 we review some background knowledge on evolutionary game theory and learning in games. Then in Section 3.3 we study how norms emerge in Ali Baba and the Thief. Section 3.4 discusses some related work and Section 3.5 summarizes this chapter.

3.2 Background: evolutionary game theory and learning in games

Evolutionary game theory originates as an application of game theory to biology (Lewontin, 1961; Smith and Price, 1973). Recently it has become of increased interest to social scientists and

philosophers. There are two approaches to evolutionary game theory. The first approach employs the concept of an evolutionarily stable strategy as the principal tool of analysis. The second approach constructs an explicit model of the process of evolution by representing the frequency of strategies change in the population (Alexander, 2009). The second approach is closely related to learning in games (Fudenberg and Levine, 1998).

Typically, in the problem of learning in games we have two agents that face each other repeatedly in the same game, and each one tries to maximize the sum of its utility over time. The theory of learning in games studies the equilibrium concepts described by various simple learning mechanisms. Many learning mechanisms have been studied. In this chapter, we focus on imitate-the-best.

3.2.1 Replicator Dynamics

Replicator dynamics originates from evolutionary game theory and is a widely studied learning model in repeated games. Here our introduction of replicator dynamics is taken from the open source textbook (Vidal, 2006). This model assumes that the fraction of agents playing a particular strategy will grow in proportion to how well that strategy performs in the population. A homogeneous population of agents is assumed. The agents are randomly paired in order to play a symmetric game, that is, a game where both agents have the same set of possible strategies and receive the same payoffs for the same actions. The replicator dynamics model is meant to capture situations where agents reproduce in proportion to how well they are doing.

Formally, we let $\phi^t(s)$ be the number of agents using a pure strategy s at round t and S be the set of all strategies. We can then define

$$\theta^t(s) = \frac{\phi^t(s)}{\sum_{s' \in S} \phi^t(s')} \tag{3.1}$$

to be the proportion of agents playing s at round t. The expected utility for an agent playing strategy s at round t is defined as

$$u^t(s) = \sum_{s' \in S} \theta^t(s') u(s, s'), \tag{3.2}$$

where $u(s, s')$ is the utility that an agent playing s receives against an agent playing s'. Notice that this expected utility assumes that the agents face each other in pairs and choose their opponents randomly. The average utility at round t is defined as

$$u^t = \sum_{s \in S} \theta^t(s) u^t(s), \tag{3.3}$$

In the replicator dynamics the rate of strategy-change for each agent is proportional to how well it did on the previous round. Thus, the proportion of agents playing s at the next round is given by

$$\theta^{t+1}(s) = \theta^t(s)\frac{u^t(s)}{u^t}. \tag{3.4}$$

Notice that the number of agents playing a particular strategy will continue to increase as long as the expected utility for that strategy is greater than the average utility. Only strategies whose expected utility is less than average utility will decrease in population. As such, the size of a population will constantly fluctuate. However, when studying replicator dynamics we ignore the absolute size of the population and focus on the proportion of the population playing a particular strategy.

3.2.2 Imitate-the-best

Imitate-the-best is a very natural and common learning rule in the modeling literature (Nowak and May, 1992; Epstein, 1998). According to this rule, at each round of play, every agent surveys the utility of its neighbors and adopts the strategy of the one who did the best in the last round, where "best" means "received the highest utility". Here the neighborhood is defined in the setting of a social network. A social network is a graph $G = (Agent, Neighbor)$ where every node in this graph is an agent and every edge connecting two agents means that the two agents are neighbors. Formally,

$$s_i^{t+1} = s_k^t \text{ with } k \in \text{argmax}_{j \in N_i} u_j^t$$

where N_i is the set of all neighbors of agent i and u_j^t is the utility agent j received in round t and s_k^t is the pure strategy agent k adopts in round t.

It is also assumed that an agent will not switch strategies unless it has some incentives to do so. If the best neighbors of an agent still managed to get lower utility than it, the agent will not switch its strategy. Ties between best neighbors are broken randomly. So, for example, if the highest utility in an agent's neighborhood was obtained by 2 agents following strategy s_1, 1 agent following strategy s_2, and 2 agents following strategy s_3, that agent will adopt strategy s_1 with probability $\frac{2}{5}$, strategy s_2 with probability $\frac{1}{5}$, and strategy s_3 with probability $\frac{2}{5}$.

3.3 Ali Baba and Thief

In this section we study norm emergence in Ali Baba and the Thief. We use replicator dynamics and imitate-the-best as rules of learning. No social network is assumed when agents learn by using

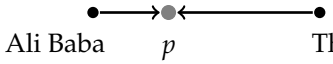

Figure 3.1: Ali Baba and Thief modeled using replicator dynamics

replicator dynamics while lattice model and small world model are used when agents use imitate-the-best.

3.3.1 Replicator dynamics

Let T denote the strategy Thief and A denote the strategy Ali Baba. Let p stands for the proportion of the population which chooses T, then the expected utility of an agent choosing T at round t is

$$u^t(\mathtt{T}) = p \cdot u(\mathtt{T},\mathtt{T}) + (1-p) \cdot u(\mathtt{T},\mathtt{A}) = (1-p) \cdot (x+d).$$

Similarly, the average utility of an agent choosing A is

$$u^t(\mathtt{A}) = p \cdot u(\mathtt{A},\mathtt{T}) + (1-p) \cdot u(\mathtt{A},\mathtt{A}) = p \cdot (x-d) + (1-p) \cdot x = x - pd.$$

The average utility of the population is

$$u^t = p \cdot u^t(\mathtt{T}) + (1-p) \cdot u^t(\mathtt{A}) = p \cdot (1-p) \cdot (x+d) + (1-p) \cdot (x-pd).$$

The number of agents choosing T increases exactly when $u^t(\mathtt{T}) > u^t$. This relation can be expressed in terms of the utilities as follows:

$$\begin{gathered}(1-p) \cdot (x+d) > p \cdot (1-p) \cdot (x+d) + (1-p) \cdot (x-pd), \\ (x+d) > p \cdot (x+d) + (x-pd), \\ x + d > xp + x, \\ p < \tfrac{d}{x}.\end{gathered}$$

Figure 3.1 illustrates what happens when we model Ali Baba and the Thief using replicator dynamics. The points on the right and left of the diagram correspond to the states where all agents choose Thief or Ali Baba, respectively. Points in the middle of the diagram represent mixed states of the population. A stable state exists at the point where the proportion of agents choosing Thief is $p = \frac{d}{x}$. The arrow from left to right in Figure 3.1 indicates that if at stage t the proportion of Thief is less than $\frac{d}{x}$, then more agents will choose Thief in stage $t+1$. Similarly, the arrow from right to left indicates that less agents will choose Thief if p is larger than $\frac{d}{x}$.

In the stable state, a pattern of behavior emerges. In this pattern, a proportion p of agents choose Thief and a proportion $1-p$ of agents choose Ali Baba. When d is very close to 0, norms saying "don't rob", "be peaceful" or "don't harm others" can be viewed as emerged because:

1. Almost all agents are choosing and will continue to choose Ali Baba.

2. All agents believe that most agents will choose Ali Baba in the next round. Here we conceive such belief is formed in the process of evolution: if an agent sees another agent keeping choosing a specific strategy for a long time, by default it believes that agent will keep choosing that strategy.

3. Every agent believes that most agents believe that it is good if it chooses Ali Baba. Indeed, every agent always believes (even knows) that it is good for itself that other agents choose Ali Baba.

On the other hand, when d is very close to x, although it is true that,

1. Almost all agents are choosing and will continue to choose Thief.

2. All agents believe most agents will choose Thief in the next round.

It is not plausible that every agent believes that most agents believe that it is good if it chooses Thief. Therefore norms prescribing "you should rob" do not emerge, even though most agents choose Thief in that state.

3.3.2 Imitate-the-best

We run simulations using the Netlogo platform.* We set the population of agents to be 100. Initially, 50% of agents choose Thief. Over time, however, through agent-agent interactions, a bias toward Ali Baba spreads through the entire network until 100% of the population choose Ali Baba. At this point, we say that a norm prescribing that there should be no harmful behavior has emerged. However, this process may sometime demonstrate complexity. For example, Figure 3.2 shows the percentage of Thieves fluctuate several times in the process of system evolution. Note that when all agents choose the same action at some stage, they will choose that action forever because the learning rule they use is imitate-the-best. Therefore such stages are stable (or absorbing) stages. It is plausible that several rounds after the stable stage, all agents starts to believe that their neighbors will keep the same behavior as in the stable stage based on their own experience of interaction.

Lattice model

Lattice models are a special kind of social network in which the connections between agents are defined spatially. Each agent is considered to be located at some cell on an N-dimensional grid, and every cell in the grid is occupied by exactly one agent. In the 1-dimensional regular lattice,

*http://ccl.northwestern.edu/netlogo/index.shtml

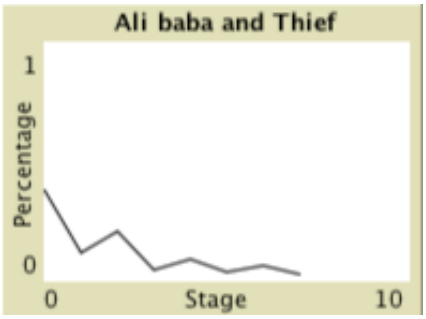

Figure 3.2: The process of norm emergence in Ali Baba and Thief

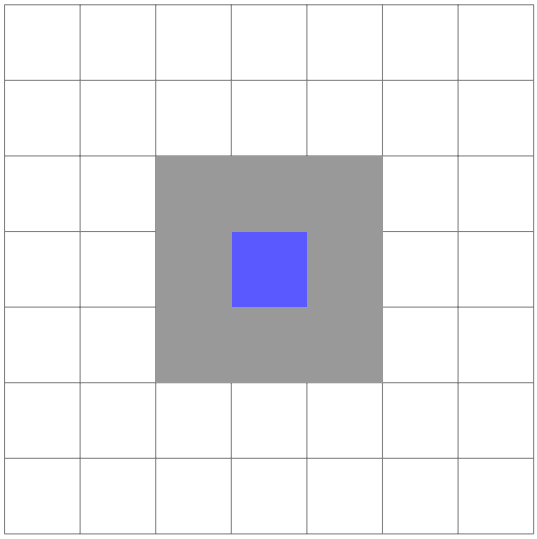

Figure 3.3: A 2-dimensional lattice

this means that agents live on a line. Every agent who does not live in the end point has exactly 2 neighbors. In the 2-dimensional regular lattice, the agents live on a grid. Every agent who does not live in the boundary has exactly 8 neighbors. As it is show in Figure 3.3, the agent in blue has 8 grey neighbors surrounding him. Circular lattice is a variant of regular lattice. In a 1-dimensional circular lattice, the agents lives in the left and right corner are neighbors. Therefore every agent has exactly 2 neighbors in a 1-dimensional circular lattice. Similarly, in a 2-dimensional circular lattice, we consider the leftmost and rightmost columns are neighboring columns and the top and bottom rows are neighboring rows. Therefore every agent in a 2-dimensional circular lattice has exactly 8 neighbors.

In our experiments on lattice models, we set the initial utility $x = 1000$ and study how the amount of robbery d changes the probability of norm emergence. The data of our experiment is recorded in Table 3.2. We let the amount of robbery d vary from 200 to 800. The initial percentage

Amount of robbery (d)	T_r	T_c
200	100	100
400	100	100
600	0	0
800	0	0

Table 3.2: Ali Baba and Thief, $x = 1000$, $d = 200, \ldots, 800$

Amount of robbery (d)	T_r	T_c
400	100	100
420	100	100
440	60	17
460	79	17
480	73	15
500	0	16
520	0	20
540	0	24
560	0	18
580	0	12
600	0	16

Table 3.3: Ali Baba and Thief, $x = 1000$, $d = 400, \ldots, 600$

of agents choosing Thief is set to be 50%. In each round, every agent play Ali Baba and Thief with his 8 neighbors one by one. Their utility in this round is the average utility of the 8 plays. At the end of a round, each agent compares its utility to all its neighbors. If the agent's utility is higher than all its neighbors, then the agent does not change its strategy in the next round. Otherwise the agent adopts the strategy chosen by its neighbor of highest utility. The simulation stops when no agent changes its strategy between two rounds, or the rounds reaches 100. We run the simulation for 100 times for each $d \in \{200, 400, 600, 800\}$. We say a norm emerges if the simulation stops with all agents choosing Ali Baba. We use T_r and T_c to denote the times of norm emergence in our simulations.

In general, our experiment shows that when the amount of robbery is high, the probability of norm emergence is low. When the amount of robbery decrease, the probability of norm emergence quickly increase. When d is less than 400, the norm "you should not rob" emerges for certain. This is in contrast with the analysis of replicator dynamics. If $d = 400$, then according to replicator dynamics a proportion of 40% of agents will choose Thief, therefore a norm saying there ought to be no robbery does not emerge. Note that there is a leap of the probability of norm emergence from $d = 600$ to $d = 400$. We are curious about such a leap. To have a better understanding we perform a second experiment in which the amount of robbery vary from 400 to 600.

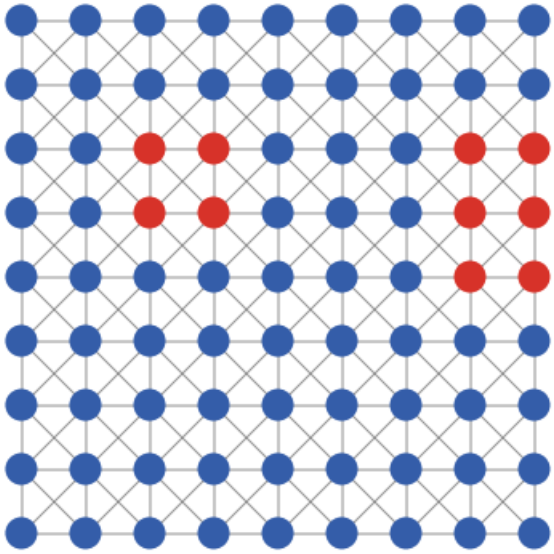

Figure 3.4: stable state without norm emergence (red agents are Thiefs)

The leap still exists, as Table 3.3 shows. Now the leap takes place when d decrease from 500 to 480 in the case of regular lattice. In circular lattice the leap appears when d decrease from 440 to 420. The existence of such a leap in the regular lattice can be explained as follows.

Assume $x = 1000, d = 500$. Suppose at round t an agent i chooses Thief and 3 of its neighbors also choose Thief while other 5 neighbors choose Ali Baba. The the utility of agent i in round t is $u_i^t = \frac{0+0+0+1500+1500+1500+1500+1500}{8} = 937.5$. For i's neighbor j who chooses Ali Baba, in the best case the uitility of j in round t is $u_j^t = \frac{500+1000+1000+1000+1000+1000+1000+1000}{8} = 937.5$. Therefore according to imitate-the-best, i will keep choosing Thief in round $t+1$. This explains why the state shown in Figure 3.4 is a stable state. The existence of such states explains why the probability of norm emergence drops dramatically in the regular lattice model when $d \geq 500$.

The sudden decrease of the probability of norm emergence in the circular lattice model is much more difficult to explain. By the above argument we believe that stable states without norm emergence also exist in circular lattices and it should be reached when $d = 500$. However, experimental results do not support our belief. Further experiments show that such a leap is a robust existence in circular lattices: a tiny change of d from 428.57142 to 428.57143 dramatically changes the probability of norm emergence (see Table 3.4). Those phenomena suggests there is a critical point of norm emergence in lattice models: as long as d is less than a certain value,

Amount of robbery (d)	Probability of norm emergence
428.57142	1
428.57143	0.16

Table 3.4: circular lattice, critical point, $x = 1000$

Initial utility (x)	value of d at critical points	$\frac{d}{x}$
100	42	0.42
200	85	0.425
300	128	0.426
400	171	0.4275
500	214	0.428
600	257	0.428
700	300	0.429
800	342	0.4275
900	385	0.428
1000	428	0.428

Table 3.5: circular lattice, critical point in general

evolution of the system will lead most agents choose Ali Baba with high probability. Otherwise the probability of norm emergence is low.

To further investigate the existence of critical points in circular lattices, we perform more experiments. The result of our experiments is present in Table 3.5. Those data confirms the existence of critical points. For a circular lattice, the critical point is reached when $\frac{d}{x} \approx 0.428$.

Small world

The principal merit of lattice models is that they work well for modeling social networks in which the social relations are associated with the spatial positions of the agents. For social systems in which the relevant relations are not associated with spatial position, other network models need to be developed.

A lot of social networks are well characterized by the *small-world property*, which says that any two agents in the network are connected by a short sequence of friends, family and acquaintances. A *small-world network* is a type of graph in which most nodes are not neighbors of one another, but most nodes can be reached from every other node by a small number of edges. The problem of norm emergence on small-world networks has been studied in Delgado (2002) and Alexander (2007). We choose the small-world network provided by the NetLogo models library.[†]

[†] http://ccl.northwestern.edu/netlogo/models/SmallWorlds

Amount of robbery (d)	Probability of norm emergence
100	1
200	0.68
300	0.06
400	0.02
500	0

Table 3.6: small world, $x = 1000$, $d = 100, \ldots, 500$

Amount of robbery (d)	Probability of norm emergence
220	0.73
240	0.75
260	0.72
280	0.05

Table 3.7: small world, $x = 1000$, $d = 220, \ldots, 280$

Just like in the lattice models, in our experiments in small world models, we set the population to be 100, the initial utility $x = 1000$ and study how the amount of robbery d changes the probability of norm emergence. The data of our simulations are recorded in Table 3.6. Just like in the lattice model, our experiment shows that the probability of norm emergence increases when the amount of robbery decreases. Note that there is also a leap of the probability of norm emergence from $d = 300$ to $d = 200$. To have a better understanding we perform more experiments to find the critical points. The results of those experiment are present in Table 3.7 and 3.8. Those data confirms the existence of critical points. For small world models, the critical point is reached when $\frac{d}{x} \approx 0.265$.

3.4 Related work

Axelrod (1986) presents a "norms game" in which agents choose to conform or deviate from norms probabilistically, and then other agents probabilistically choose to punish any deviations at some cost. Agents can choose over time to be more or less "bold", which determines the rate at which

Initial utility (x)	value of d at critical points	$\frac{d}{x}$
100	27	0.27
300	81	0.27
500	131	0.261
700	189	0.27
900	239	0.265

Table 3.8: Ali Baba and Thief, critical point in general

they deviate from norms, and they can likewise choose to be more or less "vengeful", which determines how often they punish. Axelrod noted that in such a game the stable state is constant defection and no punishment. However, if we introduce a meta-norm to this game, which requires one to punish people who fail to punish defectors, then we arrive at a stable norm in which there is no boldness, but very high levels of vengefulness. It is under these conditions that we find a norm emerge and remain stable. Axelrod's model aims to illustrate that norms require meta-norms. That is, failure to punish defection must be seen as equivalent to a defection itself.

Shoham and Tennenholtz (1997) proposes a reinforcement learning approach based on the rule *highest cumulative reward* to study the emergence of social norms. According to this rule, an agent chooses the strategy that has yielded the highest reward in the past iterations. The history of the strategies chosen and the rewards for each strategy are stored in a memory of a certain size. Their experiments show that the rate of updating strategy and interval between memory flushes had a significant impact on the efficiency of norm emergence.

Bicchieri et al. (2004) present a simulation of the dynamics of impersonal trust. Their model does not rely on a meta-norm of punishment - instead, it is purely driven by repeated interactions of conditional strategies. They show how a "trust and reciprocate" norm can emerge and stabilize in populations of conditional cooperators. The norm is not to be identified with a single strategy. It is instead supported by several conditional strategies that vary in the frequency and intensity of sanctions. In their model, agents play anywhere from 1 to 30 rounds of a trust game for 1,000 iterations, relying on the 4 unconditional strategies, and the 16 conditional strategies that are standard for the trust game. After each round, agents update their strategies based on replicator dynamics. As the number of rounds grows, a norm of impersonal trust/reciprocity emerges in the population.

Boella and van der Torre (2005a) study enforceable social laws in artificial social systems by using a control system. Boella and van der Torre (2005b) use enforceable social laws to address the question how artificial social systems can be extended to reason about the evolution of artificial social systems.

Sen and Airiau (2007) propose a framework for the emergence of norms through social learning in which agents learn norms based on private interactions. They experimented with different reinforcement learning algorithms and studied the influence of the population size, the set of possible actions and the heterogeneity of the population on norm emergence.

Sen and Sen (2009) evaluate how varying topologies of social networks affected the emergence of norms through social learning in these networks. Three different kinds of network topologies (i.e., scale-free, fully-connected and ring networks) were studied to show how quickly norms

converged in social networks depending on parameters such as the topology of the network, the population size and the number of actions available.

Savarimuthu and Cranefield (2011) make three contributions to the study of norms. Firstly, based on the simulation research on norms, they propose a life-cycle model for norms. Secondly, they discuss different mechanisms used by researchers to study norm creation, identification, spreading, enforcement and emergence. They also discuss the strengths and weaknesses of each of these mechanisms. Thirdly, in the context of identifying the desired characteristics of the simulation models of norms they discuss the research issues that need to be addressed.

Yu et al. (2013) propose a collective learning framework, which imitates the opinion aggregation process in human decision making, to study the impact of agent local collective behaviors on norm emergence in different situations. In their framework, each agent interacts repeatedly with all its neighbors. At each step, an agent first takes a best-response action towards each of its neighbors and then combines all these actions into a final action using ensemble learning methods. They conduct extensive experiments to evaluate the framework with respect to different network topologies, learning strategies, numbers of actions, and so on.

3.5 Summary

In this chapter we have proposed a model that supports the emergence of norms via multiagent learning in social networks. In our model, individual agents repeatedly interact with its neighbors over a given game called Ali Baba and the Thief. An agent learns its strategy to play the game by using the learning rule imitate-the-best. Our results have shown that some norms prohibiting harmful behaviors such as "you should not rob" can emerge after repeated interactions among agents inhabited in certain social networks. Our experimental results have suggested that there is a critical point of norm emergence which is decided by quotient of the initial utility and the amount of robbery in Ali Baba and the Thief. When the quotient of the initial utility and the amount of robbery is smaller than the critical point, the probability of norm emerge is high. The probability drops dramatically as long as the quotient of the initial utility and the amount of robbery is larger than the critical point.

Chapter 4

Axiomatics of Norms

Abstract

The derivation systems of unconstrained input/output logic are axiomatic representations of norms. In this chapter we analyze various derivation rules of input/output logic in isolation and define the corresponding semantics. Then we combine them together to achieve alternative semantics of several input/output logics. Our alternative semantics for out_3 and out_3^+ is useful in the study of the complexity of input/output logic. Our alternative semantics for constitutive input/output logic is adequate for the derivation system of constitutive input/output logic.

Input/output logic adopts mainly operational semantics: a normative system is conceived in input/output logic as a deductive machine, like a black box which produces normative statement as output, when we feed it descriptive statements as input. The procedure of the operational semantics is divided to three stages. In the first stage, we have in hand a set of propositions (call it the input) as a description of the current state. We then apply logical operators to this set, say, close the set by logical consequence. Then we give this set to the deductive machine and we reach the second stage. In the second stage, the machine accepts the input and produces a set of propositions as output. In the third stage, we accept the output and apply logical operators to it. On the axiomatic side, input/output logic is characterized by derivation rules about norms. A norm is represented by an ordered pair of formulas. Given a set of mandatory norms O, a derivation system is the smallest set which extends O and is closed under certain derivation rules.

In this chapter we study the axiomatics and operational semantics of input/output logic. Our methodology is to first analyze various derivation rules of input/output logic in isolation and study the corresponding semantics, then combine those results together to achieve adequate semantics for various input/output logics. One feature of the existing work of input/output logic is: the derivation rules always work in bundles. For example in simple-minded input/output logic, the derivation system is decided by three rules: strengthening the input (SI), weakening the output (WO) and conjunction in the output (AND). When several derivation rules work together, the corresponding operational semantics will be rather complex, and insights of the machinery is therefore concealed. To achieve a deeper understanding of input/output logic, it is helpful to isolate every single rule and study them separately.

4.1 Axiomatics of input/output logic

The proof system of input/output logic is build on derivations of norms. We say that a mandatory norm (a, x) is derivable from a set O iff (a, x) is in the least set that extends O and is closed under a number of derivation rules. The following are the derivation rules that have been used to build input/output logic:

- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vdash a$.
- IEQ (input equivalence): from (a, x) and $a \dashv\vdash b$ to (b, x). Here $a \dashv\vdash b$ means $a \vdash b$ and $b \vdash a$.
- OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.
- WO (weakening the output): from (a, x) to (a, y) whenever $x \vdash y$.
- OEQ (output equivalence): from (a, x) and $x \dashv\vdash y$ to (a, y).

- AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.
- Z (zero premise): from nothing to $(\top, \top)$.
- ID (identity): from nothing to (a, a), for every $a \in L_{\mathbb{P}}$.
- T (plain transitivity): from (a, x) and (x, y) to (a, y).
- CT (cumulative transitivity): from (a, x),$(a \wedge x, y)$ to (a, y) .
- MCT (mediated cumulative transitivity): from (a, x'), $x' \vdash x$ and $(a \wedge x, y)$ to (a, y).
- ACT (aggregative cumulative transitivity): from (a, x),$(a \wedge x, y)$ to $(a, x \wedge y)$.

The derivation system based on the rules SI, WO, AND and Z is called $deriv_1$.* Adding OR to $deriv_1$ gives $deriv_2$. Adding CT to $deriv_1$ gives $deriv_3$. These five rules together give $deriv_4$. Adding ID to $deriv_i$ gives $deriv_i^+$ for $i \in \{1, 2, 3, 4\}$. $(a, x) \in deriv(O)$ is used to denote that (a, x) is derivable from O using rules of derivation system $deriv$. The rules IEQ, OEQ, and T is used in the input/output logic of constitutive norms (Boella and van der Torre, 2006). MCT is introduced by Stolpe (2008a) in his mediated reusable input/output logic, while ACT is recently introduced by Parent and van der Torre in their aggregative input/output logic (Parent and van der Torre, 2014b).

4.2 Operational semantics for input/output logic

As mentioned in the introductory section, our methodology is to first analyze various derivation rules of input/output logic in isolation and study the corresponding semantics, then combine those results together to achieve adequate semantics for several input/output logics. Since the procedure of operational semantics is divided into three stages, we also classify derivation rules according to different stages: rules of input correspond to operations in the first stage; rules of output correspond to operations in the third stage; rules of normative system correspond to operations in the second stage.

4.2.1 Rules of input

In this subsection we investigate the following rules regulating the input:

- IEQ (input equivalence): from (a, x) and $a \dashv\vdash b$ to (b, x).
- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vdash a$.

*In Makinson and van der Torre (2000), $deriv_1(O)$ is characterized as the least set that extends $O \cup \{(\top, \top)\}$ and is closed under SI, WO and AND. It can be easily verified that our description of $deriv_1(O)$ is equivalent to Makinson and van der Torre's original characterization.

- OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.

IEQ is a basic rule in the logic of constitutive norms (Jones and Sergot, 1996). SI is involved in all input/output logic of Makinson and van der Torre. OR is valid in $deriv_2$ and $deriv_4$. The derivation systems decided by rules of input are defined as follows:

Definition 4.1. *$deriv_{ie}(O)$, $deriv_{si}(O)$, $deriv_{or}(O)$ are the derivation systems given by the rule IEQ, SI, OR respectively. That is, $deriv_{ie}(O)$ is the smallest set of norms such that $O \subseteq deriv_{ie}(O)$ and $deriv_{ie}(O)$ is closed under the IEQ rule, and similarly for $deriv_{si}(O)$ and $deriv_{or}(O)$.*

Now our task is to construct the semantics corresponding to those derivation systems. For the convenience of notation, we let $C_e(A) = \{b \in L_{\mathbb{P}} : \text{there is } a \in A, a \dashv\vdash b\}$, for a set $A \subseteq L_{\mathbb{P}}$. Moreover, we call a set A disjunctive if it satisfies the following: for all $x \vee y \in A$, either $x \in A$ or $y \in A$. The following is the semantics corresponding to the rules of input.

Definition 4.2. *For a set of norms O and a formula a, we define $out_{ie}(O, a) = O(C_e(\{a\}))$, $out_{si}(O, a) = O(Cn(a))$, $out_{or}(O, a) = \bigcap\{O(B) : a \in B, B \text{ is disjunctive}\}$.*

Theorem 4.1.

1. $(a, x) \in deriv_{ie}(O)$ *iff* $x \in out_{ie}(O, a)$.
2. $(a, x) \in deriv_{si}(O)$ *iff* $x \in out_{si}(O, a)$.
3. $(a, x) \in deriv_{or}(O)$ *iff* $x \in out_{or}(O, a)$.

Proof. 1. (left-to-right) Assume $(a, x) \in deriv_{ie}(O)$. We prove by induction on the length of the derivation. The base case is when $(a, x) \in O$. If $(a, x) \in O$, then $x \in O(\{a\}) \subseteq O(C_e(\{a\}))$ because $\{a\} \subseteq C_e(\{a\})$. Therefore $x \in out_{ie}(O, a)$.

For the inductive case, we assume (a, x) is derived by the IEQ rule. Then there is b such that $b \dashv\vdash a$ and $(b, x) \in deriv_{ie}(O)$. By induction hypothesis we know $x \in out_{ie}(O, b)$. Therefore $x \in O(C_e(\{b\})) = O(C_e(\{a\})) = out_{ie}(O, a)$.

(right-to-left) Assume $x \in out_{ie}(O, a)$. Then $x \in O(C_e(\{a\}))$. Therefore there is $b \in C_e(\{a\})$ such that $(b, x) \in O$. That is, there is $b \dashv\vdash a$ such that $(b, x) \in O$. Therefore using the IEQ rule we have $(a, x) \in deriv_{ie}(O)$.

2. (left-to-right) Assume $(a, x) \in deriv_{si}(O)$. Again we prove by induction on the length of the derivation. The base case is when $(a, x) \in O$. If $(a, x) \in O$, then $x \in O(a) \subseteq O(Cn(a)) = out_{si}(O, a)$.

For the inductive case, we assume (a, x) is derived by the SI rule. If (a, x) is derived by the SI rule, then there is b such that $(b, x) \in deriv_{si}(O)$ and $a \vdash b$. By induction hypothesis we know

$x \in out_{si}(O, b)$. Therefore $x \in O(Cn(b))$. Now by $a \vdash b$ we know $Cn(b) \subseteq Cn(a)$. Therefore by the monotony of O, which is easy to be checked, we know $O(Cn(b)) \subseteq O(Cn(a))$. Hence $x \in O(Cn(a))$ and $x \in out_{si}(O, a)$.

(right-to-left) Assume $x \in out_{si}(O, a)$. Then $x \in O(Cn(a))$. Therefore there exist $b \in Cn(a)$, $(b, x) \in O$. Therefore $a \vdash b$. Now using SI we have $(a, x) \in deriv_{si}(O)$.

3. (left-to-right) Assume $(a, x) \in deriv_{or}(O)$. Again we prove by induction on the length of the derivation. The base case is when $(a, x) \in O$. If $(a, x) \in O$, then for all disjunctive B which contains a, $x \in O(B)$. Therefore $x \in \bigcap\{O(B) : a \in B, B \text{ is disjunctive}\} = out_{or}(O, a)$.

   For the inductive case, we assume (a, x) is derived by the OR rule, then there are formulas b, c such that $(b, x) \in deriv_{or}(O)$, $(c, x) \in deriv_{or}(O)$ and a is $b \vee c$. By induction hypothesis we know $x \in out_{or}(O, b)$ and $x \in out_{or}(O, c)$. Now for every B^* such that $a \in B^*$ and B^* is disjunctive, we have $b \vee c \in B^*$ since a is $b \vee c$. Note that B^* is disjunctive, so we further have either $b \in B^*$ or $c \in B^*$. If $b \in B^*$, then B^* is a disjunctive set contains b. So we have $x \in out_{or}(O, b) = \bigcap\{O(B) : b \in B, B \text{ is a disjunctive set }\} \subseteq O(B^*)$. Hence $x \in O(B^*)$. If $c \in B^*$, we can similarly deduce $x \in O(B^*)$. Therefore no matter $b \in B^*$ or $c \in B^*$, we have $x \in O(B^*)$. Note that B^* is an arbitrary disjunctive set contains a, so we know $x \in \bigcap\{O(B) : a \in B, B \text{ is disjunctive}\} = out_{or}(O, a)$.

   (right-to-left)[†] Suppose $(a, x) \notin deriv_{or}(O)$. We construct a disjunctive set $B = \{a_0, \ldots, a_n\}$ containing a by means of the following algorithm (where $a_0 = a$).

   - $i = 0$
   - while a_i is of the form $a_i^1 \vee a_i^2$ do
     - if $(a_i^1, x) \notin deriv_{or}(O)$, let $a_{i+1} := a_i^1$, else let $a_{i+1} := a_i^2$
     - $i := i + 1$

   The procedure terminates in view of the fact that a is a finite string. It terminates at a_i iff a_i is not disjunctive. Note that for each $i \in \{0, \ldots, n\}$, $(a_i, x) \notin deriv_{or}(O)$. For a_0 this is so by our supposition. Suppose it holds for i. In case $a_{i+1} = a_i^1$, trivially $a_{i+1} \notin deriv_{or}(O)$. Suppose thus that $a_{i+1} = a_i^2$ and thus that $(a_i^1, x) \in deriv_{or}(O)$. If $(a_{i+1}, x) \in deriv_{or}(O)$ then by OR, $(a_i, x) \in deriv_{or}(O)$ which contradicts the induction hypothesis. Thus $(a_{i+1}, x) \notin deriv_{or}(O)$. Therefore it holds that for each $i \in \{0, \ldots, n\}$, $(a_i, x) \notin deriv_{or}(O)$. Then we know for each $i \in \{0, \ldots, n\}$, $(a_i, x) \notin O$. Hence $x \notin O(B)$. Note also that by the construction B is a disjunctive set that contains a. Thus $x \notin out_{or}(O, a)$.

[†]This proof is due to an anonymous reviewer of CLIMA2014. The original proof is much more complex than the current proof.

□

Remark 4.1. *The above result reveals that rules of input corresponding to operations in the first stage: SI means to close the input by logical consequence; IEQ means to close the input by logical equivalence; OR ensures the input has to be extended to satisfy disjunctive property.*

4.2.2 Rules of output

In this subsection we investigate the following rules regulating the output:

- OEQ (output equivalence): from (a, x) and $x \dashv\vdash y$ to (a, y).
- WO (weakening the output): from (a, x) to (a, y) whenever $x \vdash y$.
- AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.

OEQ is a basic rule in the logic of constitutive norms (Jones and Sergot, 1996). WO and AND are involved in all input/output logic of Makinson and van der Torre. The derivation systems decided by rules of output are defined as follows.

Definition 4.3. *$deriv_{oe}(O)$, $deriv_{wo}(O)$, $deriv_{and}(O)$ are the derivation systems given by the rule OEQ, WO, AND respectively.*

For a set of formulas $A \subseteq L_{\mathbb{P}}$, let $C_s(A) = \{b \in L_{\mathbb{P}} :$ there is $a \in A, a \vdash b\}$, $C_a(A) = \{x \in L_{\mathbb{P}} :$ there exist $x_1, \ldots, x_n \in A$, x is $x_1 \wedge \ldots \wedge x_n\}$. Here we understand $x_1 \wedge \ldots \wedge x_n$ as an abbreviation of $(\ldots((x_1 \wedge x_2) \wedge x_2) \wedge \ldots \wedge x_n)$. Intuitively, $C_s(A)$ is an operation that closes A by single consequence and $C_a(A)$ is an operation that closes A by aggregation or conjunction. The following is the semantics corresponding to the rules of output. For simplicity of notation, $O(a)$ is short for $O(\{a\})$.

Definition 4.4. *For every set of norms O and formula a, we define $out_{oe}(O, a) = C_e(O(a))$, $out_{wo}(O, a) = C_s(O(a))$, $out_{and}(O, a) = C_a(O(a))$.*

Theorem 4.2.

1. $(a, x) \in deriv_{oe}(O)$ *iff* $x \in out_{oe}(O, a)$.
2. $(a, x) \in deriv_{wo}(O)$ *iff* $x \in out_{wo}(O, a)$.
3. $(a, x) \in deriv_{and}(O)$ *iff* $x \in out_{and}(O, a)$.

Proof. 1. (left-to-right) Assume $(a, x) \in deriv_{oe}(O)$. We prove by induction on the length of derivation. The base case is easy to prove. Here we focus on the inductive case.

If (a, x) is derived by the OEQ rule, then there is y such that $y \dashv\vdash x$ and $(a, y) \in deriv_{oe}(O)$. By induction hypothesis we know $y \in out_{oe}(O, a)$. Therefore $y \in C_e(O(a))$. By $y \dashv\vdash x$ we have $x \in C_e(O(a))$. Therefore $x \in out_{oe}(O, a)$.

(right-to-left) Assume $x \in out_{oe}(O, a)$. Then $x \in C_e(O(a))$. Hence there is y such that $x \dashv\vdash y$ and $y \in O(a)$. Therefore $(a, y) \in O$. Now by applying the OEQ rule we know $(a, x) \in deriv_{oe}(O)$.

2. (left-to-right) Assume $(a, x) \in deriv_{wo}(O)$. We prove by induction on the length of derivation. The base case is easy to prove. Here we focus on the inductive case.

   If (a, x) is derived by the WO rule, then there is y such that $y \vdash x$, $(a, y) \in deriv_{wo}(O)$. By induction hypothesis we know $y \in out_{wo}(O, a)$, $y \in C_s(O(a))$. Therefore there is $z \in O(a)$ such that $z \vdash y$. Hence $z \vdash x$ and $x \in C_s(O(a))$. Therefore $x \in out_{wo}(O, a)$.

   (right-to-left) Assume $x \in out_{wo}(O, a)$. Then $x \in C_s(O(a))$. Therefore there is y such that $y \in O(a)$ and $y \vdash x$. Therefore $(a, y) \in O$. Now by applying the WO rule we know $(a, x) \in deriv_{wo}(O)$.

3. (left-to-right) Assume $(a, x) \in deriv_{and}(O)$. Again we prove by induction on the length of derivation and just focus on the inductive case.

   If (a, x) is derived by the AND rule, then there are y, z such that $(a, y) \in deriv_{and}(O)$, $(a, z) \in deriv_{and}(O)$ and x is $y \wedge z$. By induction hypothesis we know $y \in out_{and}(O, a)$ and $z \in out_{and}(O, a)$. Hence $y \in C_a(O(a))$ and $z \in C_a(O(a))$. Now by by the definition of C_a we have $y \wedge z \in C_a(O(a))$. That is, $x \in C_a(O(a))$. Hence $x \in out_{and}(O, a)$.

   (right-to-left) Assume $x \in out_{and}(O, a)$. Then $x \in C_a(O(a))$. Therefore there are $x_1, \ldots, x_n \in O(a)$ such that x is $x_1 \wedge \ldots \wedge x_n$. From $x_1, \ldots, x_n \in O(a)$ we can deduce $(a, x_1), \ldots, (a, x_n) \in O$. Now by applying the AND rule finite many times we know $(a, x_1 \wedge \ldots \wedge x_n) \in deriv_{and}(O)$. That is $(a, x) \in deriv_{and}(O)$.

□

Remark 4.2. *The above result reveals that rules of output corresponding to operations in the third stage: WO means close the output by logical consequence; OEQ means close the output by logical equivalence; AND ensures the output is closed under conjunction.*

4.2.3 Rules of normative system

While rules of input and output affect the first stage and the third stage respectively, rules of normative system affect the second stage. We investigate three rules of the normative system:

- Z (zero premise): from nothing to $(\top, \top)$.
- ID (identity): from nothing to (a, a), for every $a \in L_{\mathbb{P}}$.
- CD (conditioning) from nothing to (a, b), for every $a, b \in L_{\mathbb{P}}$ such that $a \vdash b$.

Z is a rule used in Stople's mediated reusable input/output logic (Stolpe, 2008a), and it is derivable in all Makinson and van der Torre's input/output logics. ID is used in Makinson and van der Torre's throughput input/output logic. CD is a rule used in many conditional logics.

Definition 4.5. *$deriv_z(O)$, $deriv_{id}(O)$, $deriv_{cd}(O)$ are the derivation systems given by the rule Z, ID and CD respectively.*

Definition 4.6. *For every set of norms O, let $O_z = O \cup \{(\top, \top)\}$, $O_{id} = O \cup \{(a, a) : a \in L_{\mathbb{P}}\}$, $O_{cd} = O \cup \{(a, b) : a, b \in L_{\mathbb{P}}, a \vdash b\}$. We define $out_z(O, a) = O_z(a)$, $out_{id}(O, a) = O_{id}(a)$, $out_{cd}(O, a) = O_{cd}(a)$.*

Theorem 4.3.

1. $(a, x) \in deriv_z(O)$ *iff* $x \in out_z(O, a)$.
2. $(a, x) \in deriv_{id}(O)$ *iff* $x \in out_{id}(O, a)$.
3. $(a, x) \in deriv_{cd}(O)$ *iff* $x \in out_{cd}(O, a)$.

Proof. 1. (left-to-right) Assume $(a, x) \in deriv_z(O)$. We prove by induction on the length of derivation. If $(a, x) \in O$ then $x \in O(a) \subseteq O_z(a) = out_z(O, a)$. If (a, x) is $(\top, \top)$ then $x \in \{(\top, \top)\}(a) \subseteq O_z(a) = out_z(O, a)$. In both cases we have $x \in out_z(O, a)$.

(right-to-left) Assume $x \in out_z(O, a)$. Then $x \in O_z(a) = O(a) \cup \{(\top, \top)\}(a)$. If $x \in O(a)$ then $(a, x) \in O \subseteq deriv_z(O)$. If $x \in \{(\top, \top)\}(a)$ then $(a, x) = (\top, \top) \in deriv_z(O)$.

2. (left-to-right) Assume $(a, x) \in deriv_{id}(O)$. We prove by induction on the length of derivation. If $(a, x) \in O$ then $x \in O(a) \subseteq O_{id}(a) = out_{id}(O, a)$. If (a, x) is (a, a) then $x \in \{(a, a)\}(a) \subseteq O_{id}(a) = out_{id}(O, a)$. In both cases we have $x \in out_{id}(O, a)$.

   (right-to-left) Assume $x \in out_{id}(O, a)$. Then $x \in O_{id}(a) = O(a) \cup \{(a, a) : a \in L_{\mathbb{P}}\}(a)$. If $x \in O(a)$ then $(a, x) \in O \subseteq deriv_{id}(O)$. If $x \in \{(a, a) : a \in L_{\mathbb{P}}\}(a)$ then (a, x) is of the form (a, a) which is contained in $deriv_{id}(O)$.

3. (left-to-right) Assume $(a, x) \in deriv_{cd}(O)$. We prove by induction on the length of derivation. If $(a, x) \in O$ then $x \in O(a) \subseteq O_{cd}(a) = out_{id}(O, a)$. If $a \vdash x$ then $x \in \{(a, b) : a, b \in L_{\mathbb{P}}, a \vdash b\}(a) \subseteq O_{cd}(a) = out_{cd}(O, a)$. In both cases we have $x \in out_{cd}(O, a)$.

   (right-to-left) Assume $x \in out_{cd}(O, a)$. Then $x \in O_{cd}(a) = O(a) \cup \{(a, b) : a, b \in L_{\mathbb{P}}, a \vdash b\}(a)$. If $x \in O(a)$ then $(a, x) \in O \subseteq deriv_{cd}(O)$. If $x \in \{(a, b) : a, b \in L_{\mathbb{P}}, a \vdash b\}(a)$ then $a \vdash x$. Therefore (a, x) is contained in $deriv_{cd}(O)$.

□

4.2.4 Cross-stage Rules

In this subsection we investigate cross-stage rules, which affect more than one stages. Such rules typically have the form of transitivity:

- T (plain transitivity): from (a, x) and (x, y) to (a, y).
- CT (cumulative transitivity): from (a, x), $(a \wedge x, y)$ to (a, y) .

T is used in the input/output logic for constitutive norms (Boella and van der Torre, 2006). CT is involved in $deriv_3$ and $deriv_4$.

Definition 4.7. *$deriv_t(O)$ is the smallest set of norms such that $O \subseteq deriv_t(O)$ and $deriv_t(O)$ is closed under the T rule.*

The corresponding semantics for $deriv_t(O)$ is defined in an inductive manner.

Definition 4.8. *For every set of norms O and formula a, we define $out_t(O, a) = \bigcup_{i=1}^{\infty} O_t^i(\{a\})$. Here for a set A, $O_t^i(A)$ is defined as follows:*

- $O_t^1(A) = O(A)$
- $O_t^{i+1}(A) = O(O_t^i(A))$

Theorem 4.4. *$(a, x) \in deriv_t(O)$ iff $x \in out_t(O, a)$.*

The following lemmas are needed to prove the above theorem.

Lemma 4.1. *For all $i \geq 1$, if $A \subseteq B$ the $O_t^i(A) \subseteq O_t^i(B)$.*

Proof. We prove by induction. If $i = 1$, then $O_t^1(A) = O(A) \subseteq O(B) \subseteq O_t^1(B)$. Assume the statement is true for k, consider $k + 1$. $O_t^{k+1}(A) = O(O_t^k(A))$, $O_t^{k+1}(B) = O(O_t^k(B))$. By induction hypothesis we have $O_t^k(A) \subseteq O_t^k(B)$. Therefore $O(O_t^k(A)) \subseteq O(O_t^k(B))$, $O_t^{k+1}(A) \subseteq O_t^{k+1}(B)$. □

Lemma 4.2. *For all $i, j \geq 1$, if $x \in O_t^i(a)$ and $y \in O_t^j(x)$, then $y \in O_t^{i+j}(a)$*

Proof. Suppose $x \in O_t^i(a)$ and $y \in O_t^j(x)$. Then by the above lemma we have $y \in O_t^j(x) \subseteq O_t^j(O_t^i(a))$. Now we show that $O_t^j(O_t^i(a)) = O_t^{i+j}(a)$. We prove by induction on j. If $j = 1$, then $O_t^j(O_t^i(a)) = O_t^1(O_t^i(a)) = O_t(O_t^i(a)) = O_t^{i+1}(a) = O_t^{i+j}(a)$. Suppose $O_t^k(O_t^i(a)) = O_t^{i+k}(a)$. Then $O_t^{k+1}(O_t^i(a)) = O^1(O_t^k(O_t^i(a))) = O^1(O_t^{i+k}(a)) = O(O_t^{i+k}(a)) = O_t^{i+k+1}(a)$. □

Lemma 4.3. *For all $i \geq 1$, if $x \in O_t^i(a)$ then $(a, x) \in deriv_t(O)$.*

Proof. We prove by induction. If $i = 1$, then from $x \in O_t^1(a) = O(a)$ we can deduce $(a, x) \in O \subseteq deriv_t(O)$. Now for $i = k + 1$, if $x \in O_t^{k+1}(a)$, then $x \in O(O_t^k(a))$. Therefore there is $y \in O_t^i(a)$, $(y, x) \in O$. By induction hypothesis we have $(a, y) \in deriv_t(O)$ and then use the rule of T we have $(a, x) \in deriv_t(O)$. □

Proof. (Theorem 4.4) (left to right) Assume $(a, x) \in deriv_t(O)$, then either $(a, x) \in O$ or (a, x) is derived by the T rule. The first case is easy to prove. Here we just focus on the second case.

Assume $(a, y) \in deriv_t(O)$ and it is deduced by the T rule. Then there exist $(a, x) \in deriv_t(O)$ and $(x, y) \in deriv_t(O)$. By induction hypothesis we have $x \in out_t(O, a)$ and $y \in out_t(O, x)$. That is, $x \in \bigcup_{i=1}^{\infty} O_t^i(a)$ and $y \in \bigcup_{i=1}^{\infty} O_t^i(x)$. Therefore there exist some i, j such that $x \in O_t^i(a)$ and $y \in O_t^j(x)$. Therefore we have $y \in O_t^{i+j}(a)$ by Lemma 4.2. Hence $y \in \bigcup_{i=1}^{\infty} O_t^i(a)$, $y \in out_t(O, a)$.

(right to left) Assume $x \in out_t(O, a)$, then $x \in \bigcup_{i=1}^{\infty} O_t^i(a)$. Then there exist some i, $x \in O_t^i(a)$. Now by Lemma 4.3 above we have $(a, x) \in deriv_t(O)$. □

Concerning the other cross-stage rule CT, on the one hand, it is difficult to define their corresponding semantics. On the other hand, in the next section we use an inductive semantics to define systems containing cross-stage rules together with other rules.

4.3 Alternative semantics for input/output logic

4.3.1 Alternative semantics for out_3 and out_3^+

Now we use results from previous sections to build adequate semantics for derivation systems decided by multiple rules. We start with an alternative semantics for out_3 and out_3^+. Such alternative semantics is useful in the study of the complexity of input/output logic in Chapter 6.

Theorem 4.5. *Let $B_A^O = \bigcup_{i=0}^{\infty} B_{A,i}^O$, where $B_{A,0}^O = Cn(A)$, $B_{A,i+1}^O = Cn(A \cup O(B_{A,i}^O))$. Let B_a^O be short for $B_{\{a\}}^O$. Then*

1. *$(a, x) \in deriv_3(O)$ iff $x \in Cn(O(B_a^O))$.*

2. $(a, x) \in deriv_3^+(O)$ *iff* $x \in Cn(O_{id}(B^{O_{id}}_{\{a\}}))$.

Here B^O_A is the least fixed point of function $f^O_A : 2^{L_{\mathbb{P}}} \to 2^{L_{\mathbb{P}}}$ such that $f^O_A(X) = Cn(A \cup O(X))$. It can be proved that f^O_A is monotonic with respect to the set theoretical inclusion $\subseteq$, and $(2^{L_{\mathbb{P}}}, \subseteq)$ is a complete lattice. Then by Tarski's fixed point theorem (Tarski, 1955) there exists a least fixed point of f^O_A. This inductive semantics is inspired by Stolpe (2008b). Stolpe defines an alternative semantics for out_3 which he calls it as bulk-increment semantics: $out^b_3(O, A) = \bigcup_{i=0}^{\infty} A_i$ where $A_0 = Cn(O(Cn(A)))$, $A_{i+1} = Cn(A_i \cup Cn(O(Cn(A_n \cup A))))$. To prove Theorem 4.5, we need the following lemmas.

Lemma 4.4. *For every* $A \subseteq L_{\mathbb{P}}, O \subseteq L_{\mathbb{P}} \times L_{\mathbb{P}}$, $A \subseteq B^O_A$

Proof. From the reflexivity of $\vdash$, we know $A \subseteq Cn(A) \subseteq B^O_A$. □

Lemma 4.5. *For every* $a \in L_{\mathbb{P}}, O \subseteq L_{\mathbb{P}} \times L_{\mathbb{P}}$, $B^O_a = Cn(B^O_a)$. *Here* B^O_a *is short for* $B^O_{\{a\}}$.

Proof. The left-to-right direction is trivial. Concerning the other direction, assume $x \in Cn(B^O_a)$. Then by the compactness of $\vdash$ we know there are $\{x_1, \ldots, x_n\} \subseteq B^O_a$ such that $\{x_1, \ldots, x_n\} \vdash x$. Without loss of generality, let $\{x_1, \ldots, x_n\} \subseteq B^O_{a,k}$, then $x \in Cn(B^O_{a,k}) = B^O_{a,k} \subseteq B^O_a$. □

Lemma 4.6. *For every* $a, b \in L_{\mathbb{P}}, O \subseteq L_{\mathbb{P}} \times L_{\mathbb{P}}$, *if* $a \vdash b$ *then* $B^O_b \subseteq B^O_a$.

Proof. We prove that for every natural number i, $B^O_{b,i} \subseteq B^O_{a,i}$. We prove by induction on i. If $i = 0$, then $B^O_{b,0} = Cn(b) \subseteq Cn(a) \subseteq B^O_{a,0}$. Assume $i = k + 1$ and $B^O_{b,k} \subseteq B^O_{a,k}$. Then $B^O_{b,k+1} = Cn(\{b\} \cup O(B^O_{b,k}))$. From $B^O_{b,k} \subseteq B^O_{a,k}$ we deduce $O(B^O_{b,k}) \subseteq O(B^O_{a,k})$. Now by the monotony of $\vdash$ we know $Cn(\{b\} \cup O(B^O_{b,k})) \subseteq Cn(\{b\} \cup O(B^O_{a,k}))$ and by transitivity we know $Cn(\{b\} \cup O(B^O_{a,k})) \subseteq Cn(\{a\} \cup O(B^O_{a,k}))$. Hence $B^O_{b,k+1} \subseteq B^O_{a,k+1}$. So we have proved for every i, $B^O_{b,i} \subseteq B^O_{a,i}$. With this result in hand, we can easily deduce that $B^O_b \subseteq B^O_a$. □

Lemma 4.7. *If* $x \in Cn(O(B^O_a))$, *then* $x \in B^O_a$.

Proof. By the definition of B^O_a, it is easy to verify that $O(B^O_a) \subseteq B^O_a$ and $Cn(B^O_a) \subseteq B^O_a$. The result then follows. □

Lemma 4.8. *If* $x \in Cn(O(B^O_a))$, *then* $B^O_a = B^O_{a \wedge x}$.

Proof. It's easy to prove that $B^O_a \subseteq B^O_{a \wedge x}$ since $a \wedge x \vdash x$. For the other direction, we need to prove that for every natural number i, $B^O_{a \wedge x, i} \subseteq B^O_a$. We prove this by induction on i.

- Base step: Let $i = 0$, we then have $B^O_{a \wedge x, i} = Cn(a \wedge x)$. By Lemma 4.4 we have $a \in B^O_a$. By Lemma 4.7 we have $x \in B^O_a$. Then by Lemma 4.5 we have $a \wedge x \in Cn(\{a, x\}) \subseteq Cn(B^O_a) = B^O_a$.

- Inductive step: Assume for $i = k$, $B^O_{a\wedge x,k} \subseteq B^O_a$. Then $B^O_{a\wedge x,k+1} = Cn(\{a \wedge x\} \cup O(B^O_{a\wedge x,k}))$. From $B^O_{a\wedge x,k} \subseteq B^O_a$ we know that $O(B^O_{a\wedge x,k})) \subseteq O(B^O_a)$. By the definition of B^O_a, it is easy to verify that $O(B^O_a) \subseteq B^O_a$. Therefore $O(B^O_{a\wedge x,k})) \subseteq B^O_a$. By the base step we have $a \wedge x \in B^O_a$. Then by Lemma 4.5 we know $Cn(\{a \wedge x\} \cup O(B^O_{a\wedge x,k})) \subseteq B^O_a$. That is, $B^O_{a\wedge x,k+1} \subseteq B^O_a$. This concludes the proof.

□

Lemma 4.9. *For all natural number i, if $b \in B^O_{a,i}$ and $(b, x) \in O$, then $(a, x) \in deriv_3(O)$*

Proof. We prove by induction on i.

- Base step: Let $i = 0$. Then $b \in B^O_{a,0} = Cn(a)$. Hence $a \vdash b$. Therefore we can apply SI to $a \vdash b$ and (b, x) to derive (a, x).

- Inductive step: Assume for $i = k$, if $b \in B^O_{a,k}$ and $(b, x) \in O$, then $(a, x) \in deriv_3(O)$. Now let $b \in B^O_{a,k+1}$. Then $b \in Cn(\{a\} \cup O(B^O_{a,k}))$, and there exist $b_1 \dots b_n \in O(B^O_{a,k})$ such that $\{a, b_1, \dots, b_n\} \vdash b$. Therefore $\{a \wedge b_1 \wedge \dots \wedge b_n\} \vdash b$

  Then apply SI to $(b, x) \in O$ and $a \wedge b_1 \wedge \dots \wedge b_n \vdash b$ we have $(a \wedge b_1 \wedge \dots \wedge b_n, x) \in deriv_3(O)$. Note that for each $i \in \{1, \dots, n\}$, from $b_i \in O(B^O_{a,k})$ we know there is $a_i \in B^O_{a,k}$ such that $(a_i, b_i) \in O$. Now by inductive hypothesis we have $(a, b_i) \in deriv_3(O)$. Then applying the AND rule we have $(a, b_1 \wedge \dots \wedge b_n) \in deriv_3(O)$. From $(a, b_1 \wedge \dots \wedge b_n) \in deriv_3(O)$ and $(a \wedge b_1 \wedge \dots \wedge b_n, x) \in deriv_3(O)$ we can adopt the CT rule to derive $(a, x) \in deriv_3(O)$.

□

Proof. (Theorem 4.5) We first prove the case for out_3.
(left to right) Assume $(a, x) \in deriv_3(O)$, we prove by induction on the length of derivation.

- (Base step) Assume $(a, x) \in O$, then by Lemma 4.4 we have $a \in B^O_a$. Hence $x \in O(B^O_a) \subseteq Cn(O(B^O_a))$.

- Assume (a, x) is $(\top, \top)$. We need to prove that $\top \in Cn(O(B^O_\top))$, which is trivial because $\top \in Cn(\emptyset)$.

- Assume $(b, x) \in deriv_3(O)$ and it is derived at the last step by using SI from $(a, x) \in deriv_3$ and $b \vdash a$. Then by inductive hypothesis we have $x \in Cn(O(B^O_a))$. By Lemma 4.6 we know $B^O_a \subseteq B^O_b$. Therefore we further have $O(B^O_a) \subseteq O(B^O_b)$, $Cn(O(B^O_a)) \subseteq Cn(O(B^O_b))$. Hence $x \in Cn(O(B^O_b))$.

- Assume $(a, x \wedge y) \in deriv_3(O)$ and it is derived at the last step by using AND from (a, x) and (a, y). Then by inductive hypothesis we have $x \in Cn(O(B_a^O))$ and $y \in Cn(O(B_a^O))$. Therefore $x \wedge y \in Cn(\{x, y\}) \subseteq Cn(O(B_a^O))$.

- Assume $(a, y) \in deriv_3(O)$ and it is derived by using WO form $(a, x) \in deriv_3(O)$ and $x \vdash y$. Then by inductive hypothesis we have $x \in Cn(O(B_a^O))$. Since $x \vdash y$, we can prove that $y \in Cn(O(B_a^O))$.

- Assume $(a, y) \in deriv_3(O)$ and it is derived by using CT form $(a, x) \in deriv_3(O)$ and $(a \wedge x, y) \in deriv_3(O)$. Then by inductive hypothesis we have $x \in Cn(O(B_a^O))$ and $y \in Cn(O(B_{a\wedge x}^O))$. Then by Lemma 4.8 we have $B_a^O = B_{a\wedge x}^O$. Therefore $y \in Cn(O(B_a^O))$.

(right to left) Assume $x \in Cn(O(B_a^O))$, then there exist $x_1, \ldots, x_n \in O(B_a^O)$ such that $\{x_1, \ldots, x_n\} \vdash x$. For each $i \in \{1, \ldots, n\}$, from $x_i \in O(B_a^O)$ we know there is $a_i \in B_a^O$ such that $(a_i, x_i) \in O$. From $a_i \in B_a^O$ we know there exist k such that $a_i \in B_{a,k}^O$. Now by Lemma 4.9 we know $(a, x_i) \in deriv_3(O)$. Then applying the AND rule we have $(a, x_1 \wedge \ldots \wedge x_n) \in deriv_3(O)$. Then by the WO rule we have $(a, x) \in deriv_3(O)$.

Now we prove the case for out_3^+.

(left to right) Assume $(a, x) \in deriv_3^+(O)$, we prove by induction on the length of derivation and we focus on the case when (a, x) is derived by the rule ID.

- Assume (a, x) is (a, a). We need to prove that $a \in Cn(O_{id}(B_a^{O_{id}}))$. Indeed, $a \in Cn(a) = B_{a,0}^{O_{id}} \subseteq B_a^{O_{id}}$. Therefore $a \in O_{id}(B_a^{O_{id}}) \subseteq Cn(O_{id}(B_a^{O_{id}}))$.

(right to left) Assume $x \in Cn(O_{id}(B_a^{O_{id}}))$, then there exist $x_1, \ldots, x_n \in O_{id}(B_a^{O_{id}})$ such that $\{x_1, \ldots, x_n\} \vdash x$. For each $i \in \{1, \ldots, n\}$, from $x_i \in O_{id}(B_a^{O_{id}})$ we know there is $a_i \in B_a^{O_{id}}$ such that $(a_i, x_i) \in O_{id}$. From $a_i \in B_a^{O_{id}}$ we know there exist k such that $a_i \in B_{a,k}^{O_{id}}$. Now by Lemma 4.9 we know $(a, x_i) \in deriv_3(O_{id})$. Then applying the AND rule we have $(a, x_1 \wedge \ldots \wedge x_n) \in deriv_3(O_{id})$. Then by the WO rule we have $(a, x) \in deriv_3(O_{id})$. Therefore $(a, x) \in deriv_3^+(O)$. □

4.3.2 Alternative semantics for constitutive input/output logic

Constitutive norms are a type of norms discussed in the handbook of deontic logic and normative systems (Gabbay et al., 2013). They are usually contrasted with regulative norms which regulate the behavior of human beings by indicating which behaviors are obligatory, permitted and forbidden. Constitutive norms do not regulate actions or states-of-affairs, but rather they define new possible actions or states of affairs. Formally, a constitutive norm can be represent by a conditional $a \Rightarrow_c x$ which is read as "a counts as x". An overview of the logical study of constitutive norms can be found in Grossi and Jones (2013).

Boella and van der Torre (2006) use a weak input/output logic, decided by rules of IEQ, OEQ, AND and T, to reason about constitutive norms. However, we discover the semantics defined by Boella and van der Torre (2006) is not adequate for the derivation system. In what follows, we first show the inadequacy of Boella and van der Torre's semantics, then we use the previous results in this chapter to build a adequate semantics for constitutive input/output logic.

Let $\mathfrak{N} \subseteq L_{\mathbb{P}} \times L_{\mathbb{P}}$ be a set of constitutive norms where each $(a, x) \in \mathfrak{N}$ is read as "a counts as x". Let $deriv_7(\mathfrak{N})$ be the smallest set such that $\mathfrak{N} \subseteq deriv_7(\mathfrak{N})$ and closed under the rules of IEQ, OEQ, AND and T. In Boella and van der Torre (2006), the semantics for $deriv_7(\mathfrak{N})$ is defined as follows: given be a set A of formulas, $Out(\mathfrak{N}, A) = \{\wedge Y : Y \subseteq \bigcup_{i=0}^{\infty} Out_i(\mathfrak{N}, A)\}$ is calculated as follows, assuming the replacements by logical equivalence:

- $Out_0(\mathfrak{N}, A) = \emptyset$
- $Out_{i+1}(\mathfrak{N}, A) = Out_i(\mathfrak{N}, A) \cup \{y : (\wedge X', y) \in \mathfrak{N}, X' \subseteq Out_i(\mathfrak{N}, A)\}$.

$deriv_7(\mathfrak{N})$ is not adequate with respect to this semantics. For an illustration, let $\mathfrak{N} = \{(p, q)\}$, where p and q are distinct propositional letters. Then $(p, q) \in deriv_7(\mathfrak{N})$. Following the definition of $Out(\mathfrak{N}, A)$, we have $Out_0(\mathfrak{N}, \{p\}) = \emptyset$. $Out_1(\mathfrak{N}, \{p\}) = \emptyset \cup \{y : (\wedge X', y) \in \mathfrak{N}, X' \subseteq \emptyset\} = \{y : (\wedge\emptyset, y) \in \mathfrak{N}\} = \{y : (\top, y) \in \mathfrak{N}\} = \emptyset$. And similarly, $Out_2(\mathfrak{N}, \{p\}) = Out_3(\mathfrak{N}, \{p\}) = \ldots = \emptyset$. Therefore $Out(\mathfrak{N}, \{p\}) = \emptyset$ and $q \notin Out(\mathfrak{N}, \{p\})$. This shows that the semantics $Out(\mathfrak{N}, A)$ is not adequate for $deriv_7(\mathfrak{N})$. Using the results of this chapter, an adequate semantics for $deriv_7(\mathfrak{N})$ is defined as follows.

Definition 4.9. *For every set of constitutive norms $\mathfrak{N}$ and formula a, let $out_7(\mathfrak{N}, a) = \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(\{a\})$. Here for a set of formulas A,*

- $\mathfrak{N}_7^1(A) = C_{ae}(\mathfrak{N}(C_e(A)))$
- $\mathfrak{N}_7^{i+1}(A) = C_{ae}(\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A)))$.

with $C_{ae}(A) = \{b \in L_{\mathbb{P}} : \exists a_1, \ldots, a_n \in A, a_1 \wedge \ldots \wedge a_n \dashv\vdash b\}$.

C_{ae}, read as"closed under aggregation and equivalence", is a combination of C_e defined in Section 4.2.1 and C_a defined in Section 4.2.2. For convenience we will use $\mathfrak{N}_7^i(a)$ to represent $\mathfrak{N}_7^i(\{a\})$.

Theorem 4.6. *$(a, x) \in deriv_7(\mathfrak{N})$ iff $x \in out_7(\mathfrak{N}, a)$.*

To prove this theorem we need the following lemmas.

Lemma 4.10. *For all A, if $i \leq j$ then $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^j(A)$*

Proof. Here we just prove $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^{i+1}(A)$. Since $\mathfrak{N}_7^{i+1}(A) = C_{ae}(\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A)))$, and it is easy to prove that $\mathfrak{N}_7^i(A) \subseteq C_{ae}(\mathfrak{N}_7^i(A))$. Moreover by the monotonicity of C_{ae} we can prove $C_{ae}(\mathfrak{N}_7^i(A)) \subseteq C_{ae}(\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A)))$. Therefore $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^{i+1}(A)$. □

Lemma 4.11. *For all $i \geq 1$, if $A \subseteq B$ the $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^i(B)$.*

Proof. We prove by induction on i. And we focus on the inductive step. Assume $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^i(B)$, consider $\mathfrak{N}_7^{i+1}(A)$ and $\mathfrak{N}_7^{i+1}(B)$. Note that $\mathfrak{N}_7^{i+1}(A) = C_{ae}(\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A)))$. By induction hypothesis we have $\mathfrak{N}_7^i(A) \subseteq \mathfrak{N}_7^i(B)$. By the monotonicity of $\mathfrak{N}(\bullet)$ we have $\mathfrak{N}(\mathfrak{N}_7^i(A)) \subseteq \mathfrak{N}(\mathfrak{N}_7^i(B))$. Therefore $\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A)) \subseteq \mathfrak{N}_7^i(B) \cup \mathfrak{N}(\mathfrak{N}_7^i(B))$. Therefore $C_{ae}(\mathfrak{N}_7^i(A) \cup \mathfrak{N}(\mathfrak{N}_7^i(A))) \subseteq C_{ae}(\mathfrak{N}_7^i(B) \cup \mathfrak{N}(\mathfrak{N}_7^i(B)))$ by the monotonicity of C_{ae}. That is, $\mathfrak{N}_7^{i+1}(A) \subseteq \mathfrak{N}_7^{i+1}(B)$. □

Lemma 4.12. *For all $i, j \geq 1$, for all set A, $\mathfrak{N}_7^i(\mathfrak{N}_7^j(A)) \subseteq \mathfrak{N}_7^{i+j}(A)$.*

Proof. We prove by induction on i.

If $i = 1$, then $\mathfrak{N}_7^1(\mathfrak{N}_7^j(A)) = C_{ae}(\mathfrak{N}(C_e(\mathfrak{N}_7^j(A)))) = C_{ae}(\mathfrak{N}(\mathfrak{N}_7^j(A)))$. $\mathfrak{N}_7^{1+j}(A) = C_{ae}(\mathfrak{N}_7^j(A) \cup \mathfrak{N}(\mathfrak{N}_7^j(A)))$. By monotonicity of C_{ae} we have that $C_{ae}(\mathfrak{N}(\mathfrak{N}_7^j(A))) \subseteq C_{ae}(\mathfrak{N}_7^j(A) \cup \mathfrak{N}(\mathfrak{N}_7^j(A)))$. Therefore $\mathfrak{N}_7^1(\mathfrak{N}_7^j(A)) \subseteq \mathfrak{N}_7^{1+j}(A)$.

Now for the inductive step. Consider $\mathfrak{N}_7^{i+1}(\mathfrak{N}_7^j(A))$ and $\mathfrak{N}_7^{i+1+j}(A)$. Note that we have $\mathfrak{N}_7^{i+1}(\mathfrak{N}_7^j(A)) = C_{ae}(\mathfrak{N}_7^i(\mathfrak{N}_7^j(A)) \cup \mathfrak{N}(\mathfrak{N}_7^i(\mathfrak{N}^j(A))))$. Moreover $\mathfrak{N}_7^{i+1+j}(A) = C_{ae}\ (\mathfrak{N}_7^{i+j}(A) \cup \mathfrak{N}(\mathfrak{N}_7^{i+j}(A)))$. By induction hypothesis we have $\mathfrak{N}_7^i(\mathfrak{N}_7^j(A)) \subseteq \mathfrak{N}_7^{i+j}(A)$, and by the monotonicity of $\mathfrak{N}(\bullet)$ we know that $\mathfrak{N}(\mathfrak{N}_7^i(\mathfrak{N}_7^j(A))) \subseteq N(N_7^{i+j}(A))$. Therefore $C_{ae}(\mathfrak{N}_7^i(\mathfrak{N}_7^j(A)) \cup \mathfrak{N}(\mathfrak{N}_7^i(\mathfrak{N}_7^j(A)))) \subseteq C_{ae}(\mathfrak{N}_7^{i+j}(A) \cup \mathfrak{N}(\mathfrak{N}_7^{i+j}(A)))$. therefore we have $\mathfrak{N}_7^{i+1}(\mathfrak{N}_7^j(A)) \subseteq \mathfrak{N}_7^{i+1+j}(A)$. □

Lemma 4.13. *For all $i, j \geq 1$, if $x \in \mathfrak{N}_7^i(a)$ and $y \in \mathfrak{N}_7^j(x)$, then there exist some k such that $y \in \mathfrak{N}_7^k(a)$*

Proof. Assume $x \in \mathfrak{N}_7^i(a)$ and $y \in \mathfrak{N}_7^j(x)$, then $\{x\} \subseteq \mathfrak{N}_7^i(a)$. Therefore by Lemma 4.11 we have $y \in \mathfrak{N}_7^j(\mathfrak{N}_7^i(a))$. Now by the lemma above we have $y \in \mathfrak{N}_7^{i+j}(a)$. □

Lemma 4.14. *For all $i \geq 1$, if $x \in \mathfrak{N}_7^i(a)$ and $y \in \mathfrak{N}_7^i(a)$, then $x \wedge y \in \mathfrak{N}_7^i(a)$*

Proof. Trivial. Simply because $\mathfrak{N}_7^i(a)$ is closed under C_{ae}. □

To prove the right to left direction of Theorem 4.6, we need the following lemma.

Lemma 4.15. *For all $i \geq 1$, if $x \in \mathfrak{N}_7^i(a)$ then $(a, x) \in deriv_7(\mathfrak{N})$.*

Proof. We prove by induction on i.

If $i = 1$, from $x \in \mathfrak{N}_7^1(a)$ we know $x \in C_{ae}(\mathfrak{N}(C_e(a)))$. Therefore there exist $x_1 \ldots x_m \in \mathfrak{N}(C_e(a))$ such that $x \dashv\vdash x_1 \wedge \ldots \wedge x_m$. From $x_1 \ldots x_m \in \mathfrak{N}(C_e(a))$ we can deduce that there exist $(a_1, x_1), \ldots, (a_n, x_m) \in \mathfrak{N}$ such that $a_1, \ldots, a_m \in C_e(a)$. Therefore $a \dashv\vdash a_1, \ldots, a \dashv\vdash a_m$. Now

we use IEQ we have $(a, x_1), \ldots, (a, x_m) \in deriv_7(\mathfrak{N})$. And use the AND rule finite times we have $(a, x_1 \wedge \ldots \wedge x_m) \in deriv_7(\mathfrak{N})$. Then by OEQ we know $(a, x) \in deriv_7(\mathfrak{N})$.

Now for the inductive step. Assume $x \in \mathfrak{N}_7^{i+1}(a)$,then $x \in C_{ae}(\mathfrak{N}_7^i(a) \cup \mathfrak{N}(\mathfrak{N}_7^i(a)))$. Therefore there exist $x_1, \ldots, x_m \in \mathfrak{N}_7^i(a)$ and $y_1, \ldots, y_n \in \mathfrak{N}(\mathfrak{N}_7^i(a))$ such that $x \dashv\vdash x_1 \wedge \ldots \wedge x_m \wedge y_1 \wedge \ldots \wedge y_n$. By induction hypothesis we can deduce $(a, x_1), \ldots, (a, x_m) \in deriv_7(\mathfrak{N})$ from $x_1, \ldots, x_m \in \mathfrak{N}_7^i(a)$. From $y_1, \ldots, y_n \in \mathfrak{N}(\mathfrak{N}_7^i(a))$ we know there exist $a_1, \ldots, a_n \in \mathfrak{N}_7^i(a)$ such that $(a_1, y_1), \ldots, (a_n, y_n) \in \mathfrak{N}$. By induction hypothesis we can deduce $(a, a_1), \ldots, (a, a_n) \in deriv_7(\mathfrak{N})$ from $a_1, \ldots, a_n \in \mathfrak{N}_7^i(a)$. Now by using the T rule n times we have $(a, y_1), \ldots, (a, y_n) \in deriv_7(\mathfrak{N})$. Then by using the AND rule we have $(a, x_1 \wedge \ldots \wedge x_m \wedge y_1 \wedge \ldots y_n) \in deriv_7(\mathfrak{N})$. Then use OEQ we have $(a, x) \in deriv_7(\mathfrak{N})$. □

Proof. (Theorem 4.6) (left to right) Assume $(a, x) \in deriv_7(\mathfrak{N})$, then either $(a, x) \in \mathfrak{N}$, or (a, x) is derived by using at the last step one of the rules IEQ, OEQ, T and AND. Here we only deal with the last two cases. Other cases are easy.

Assume $(a, x) \in deriv_7(\mathfrak{N})$ and it is deduced by the T rule at the last step. Then there exist $(a, y) \in deriv_7(\mathfrak{N})$ and $(y, x) \in deriv_7(\mathfrak{N})$. By induction hypothesis we have $y \in out_7(\mathfrak{N}, a)$ and $x \in out_7(\mathfrak{N}, y)$. That is, $y \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$ and $x \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(y)$. Therefore there exist some i, j such that $y \in \mathfrak{N}_7^i(a)$ and $x \in \mathfrak{N}_7^j(y)$. Therefore we have $x \in \mathfrak{N}_7^k(a)$ for some k by Lemma 4.13. Hence $x \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$, $x \in out_7(\mathfrak{N}, a)$.

Assume $(a, x) \in deriv_7(\mathfrak{N})$ and it is deduced by the AND rule at the last step. Then there exist x_1, x_2 such that x is $x_1 \wedge x_2$ and $(a, x_1), (a, x_2) \in deriv_7(\mathfrak{N})$. By induction hypothesis we have $x_1 \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$ and $x_2 \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$. Therefore for some m, n we have $x_1 \in \mathfrak{N}_7^m(a)$ and $x_2 \in \mathfrak{N}_7^n(a)$. Let $k = max\{m, n\}$, then by Lemma 4.10 we have $x_1, x_2 \in \mathfrak{N}_7^k(a)$. Then by Lemma 4.14 we have $x_1 \wedge x_2 \in \mathfrak{N}_7^k(a)$. That is, $x \in \mathfrak{N}_7^k(a)$, $x \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$ and $x \in out_7(\mathfrak{N}, a)$.

(right to left) Assume $x \in out_7(\mathfrak{N}, a)$, then $x \in \bigcup_{i=1}^{\infty} \mathfrak{N}_7^i(a)$. Then there exist some k such that $x \in \mathfrak{N}_7^k(a)$. Now by Lemma 4.15 we have $(a, x) \in deriv_7(\mathfrak{N})$. □

4.4 Summary

The derivation systems in unconstrained input/output logic are axiomatic representations of norms. In this chapter we have analyzed various derivation rules of input/output logic in isolation and defined the corresponding semantics. Then we combined them together to achieve alternative semantics of several input/output logics. Our alternative semantics for out_3 and out_3^+ will be used in the study of the complexity of input/output logic. Our alternative semantics for constitutive input/output logic is adequate for the derivation system of constitutive input/output logic.

Chapter 5

Algebra of Norms

Abstract

Lindahl and Odelstad's theory of joining-systems is an algebraic approach to normative systems. In this chapter we develop two variants of the theory of joining-systems: Boolean joining-systems and Heyting joining-systems. Those two variants algebraically characterize unconstrained input/output logic in the sense that a norm (a, x) is derivable from a set of norms O if and only if it is in the space of norms algebraically generated by O. Within those algebraic frameworks, we define isomorphism and embedding between normative systems. Then we use them to study the similarity of normative systems as well as some other global properties of normative systems.

5.1 Introduction

One feature of input/output logic is that it adopts operational rather than possible world semantics. There is no exterior structure in such operational semantics. Therefore tools to compare the similarity of structures, like bisimulation and isomorphism, play no role in input/output logic. This feature makes it difficult to analyze the *similarity* of normative systems using input/output logic, although the *equivalence* of normative systems can be represented within the input/output framework (Boella et al., 2008a).

An algebraic framework for analyzing normative systems is introduced by Lindahl and Odelstad (2000, 2008, 2013); Odelstad and Lindahl (2000). The most general form of the theory is called theory of joining-systems. A complete introduction of this theory is Lindahl and Odelstad (2013), which also contains references to earlier publications. A joining-system is a triple (B_1, B_2, S) where B_1, B_2 are two ordered algebraic structures and S a relation between B_1 and B_2 satisfying some conditions. Introducing variants of the theory of joining-systems, in this chapter we develop algebraic semantics for unconstrained input/output logic and use it to study the similarity of normative systems as well as other global properties of normative systems.

Some readers may wonder why do we need another algebraic semantics for input/output logic, given that input/output logic can be translated into modal logic and the algebraic semantics of modal logic is already well established.* The reason is, modal algebra in fact does not provide an algebraic semantics for input/output logic. The modal translation of input/output logic is only a fragment of modal logic, of which the modal depth is only 1. Note that modal logic of modal depth 1 is not closed under the modal operator, while modal algebra, as well as all algebras, requires the algebra to be closed under all its operators. Therefore the modal translation of input/output logic is in fact not a fragment of modal algebra (or any algebra).

The layout of this chapter is as follows. In Section 5.2 we give a brief introduction to the theory of joining-systems. Then, in Section 5.3 and 5.4 we develop algebraic semantics for input/output logic and intuitionistic input/output logic respectively. Section 5.5 presents some applications of the algebraic semantics. We discuss related work in Section 5.6. Finally in Section 5.7 we summarize this chapter.

5.2 Background: theory of joining-systems

The algebraic structure Lindahl and Odelstad (2000) use for their theory of joining-systems is Boolean quasi-ordering, which is an extension of Boolean algebra.

*More introduction about the modal translation of input/output logic will be reviewed in Chapter 6.

Definition 5.1 (Boolean algebra (Givant and Halmos, 2009)). *A structure $\mathfrak{A} = (A, +, \cdot, -, 0, 1)$ where A is a set, $+$ and $\cdot$ are binary operators on A, $-$ is a unitary operator on A and $0, 1 \in A$, is a Boolean algebra if it satisfies the following identities: for all $x, y, z \in A$,*

1. $x + y = y + x,\ x \cdot y = y \cdot x$
2. $x + (y + z) = (x + y) + z,\ x \cdot (y \cdot z) = (x \cdot y) \cdot z$
3. $x + 0 = x,\ x \cdot 1 = x$
4. $x + (-x) = 1,\ x \cdot (-x) = 0$
5. $x + (y \cdot z) = (x + y) \cdot (x + z),\ x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

The elements of a Boolean algebra are ordered as $x \leq y$ iff $x \cdot y = x$. It can be proved that $\leq$ is reflexive, transitive, and anti-symmetric, therefore $\leq$ is a partial order.

Example 5.1. *Given arbitrary set X, $(2^X, \cup, \cap, -, \emptyset, X)$ is a Boolean algebra where $\cup, \cap, -, \emptyset$ and X is respectively understood as $+, \cdot, -, 0$ and 1. And $\subseteq$ is understood as $\leq$.*

Definition 5.2 (Boolean quasi-ordering (Lindahl and Odelstad, 2013)). *A Boolean quasi-ordering is a structure $\mathfrak{B} = (B, +, \cdot, -, 0, 1, R)$ such that $(B, +, \cdot, -, 0, 1)$ is a Boolean algebra, and $R \subseteq B \times B$ is a reflexive and transitive relation on B which satisfies the following conditions for all a, b and c in B:*

- *aRb and aRc implies $aR(b \cdot c)$*
- *aRb implies $(-b)R(-a)$*
- *$(a \cdot b)Ra$*
- *not $1R0$*

A reflexive and transitive relation is called a quasi-ordering in Lindahl and Odelstad (2000).

Definition 5.3 (joining-systems (Lindahl and Odelstad, 2013)). *A joining-systems is a structure $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ such that $\mathfrak{A}$ and $\mathfrak{B}$ are Boolean quasi-orderings and $S \subseteq A \times B$ satisfies the following conditions:*

1. *If $(a, x) \in S$, bRa and xRy, then $(b, y) \in S$.*
2. *For all $X \subseteq B$, if for all $x \in X$,$(a, x) \in S$, then $(a, y) \in S$ for all $y \in glb(X)$.*

   Here glb is an abbreviation of greatest lower bound. Formally, $glb(X) = \{b \in B : \forall x \in X, bRx$ and $\forall a \in B$, if $\forall x \in X, aRx$, then $aRb\}$.

3. *For all* $X \subseteq A$, *if for all* $x \in X$, $(x, b) \in S$, *then* $(y, b) \in S$ *for all* $y \in lub(X)$.

   Here lub *is an abbreviation of least upper bound. Formally,* $lub(X) = \{a \in A : \forall x \in X, xRa$ *and* $\forall b \in A$, *if* $\forall x \in X, xRb$, *then* $aRb\}$.

Lindahl and Odelstad (2013) show a variant of joining-systems that is more close to input/output logic.

Definition 5.4 (prejoining-systems (Lindahl and Odelstad, 2013))**.** *A prejoining-systems is a structure* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ *such that* $\mathfrak{A}$ *and* $\mathfrak{B}$ *are Boolean quasi-orderings and* $S \subseteq A \times B$ *satisfies the following conditions:*

1. *If* $(a, x) \in S$, bRa *and* xRy, *then* $(b, y) \in S$.
2. *For all* ***finite*** $X \subseteq B$, *if for all* $x \in X$,$(a, x) \in S$, *then* $(a, y) \in S$ *for all* $y \in glb(X)$.
3. *For all* ***finite*** $X \subseteq A$, *if for all* $x \in X$, $(x, b) \in S$, *then* $(y, b) \in S$ *for all* $y \in lub(X)$.

5.3 Input/output logic and Boolean joining-systems

Given two Boolean algebras $\mathfrak{A} = (A, +_A, \cdot_A, -_A, 0_A, 1_A)$ and $\mathfrak{B} = (B, +_B, \cdot_B, -_B, 0_B, 1_B)$ with ordering $\leq_A$ and $\leq_B$ respectively. For two ordered pairs (a, x), $(b, y) \in A \times B$, following Lindahl and Odelstad, we define $(a, x) \preceq (b, y)$ iff $b \leq_A a$ and $x \leq_B y$. (a, x) is said to be narrower than (b, y) if $(a, x) \preceq (b, y)$.[†] Lindahl and Odelstad use their narrowness relation to define joining-systems to algebraically represent normative systems. Lindahl and Odelstad's joining-systems are based on Boolean quasi-ordering. To build an algebraic semantics for input/output logic, here we introduce joining-systems which are variants of Lindahl and Odelstad's.

We define those variants based on Boolean algebra. Resembling the names of input/output logic, we call those variants of joining-systems *simple-minded*, *basic*, *simple-minded reusable* and *basic reusable* respectively. The basic Boolean joining-systems is almost the same as Lindahl and Odelstad's prejoining-systems, with the only difference being the underlying algebraic structure. The other three variants will be introduced later.

Definition 5.5 (basic Boolean joining-systems)**.** *A basic Boolean joining-systems is a prejoining-systems based on Boolean algebra. That is, basic Boolean joining-systems is a structure* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ *such that* $\mathfrak{A}$ *and* $\mathfrak{B}$ *are Boolean algebras and* $S \subseteq A \times B$ *satisfies the following conditions:*

1. *If* $(a, x) \in S$ *and* $(a, x) \preceq (b, y)$, *then* $(b, y) \in S$.

[†]Such a narrowness relation is called subinterval relation in Odelstad and Boman (2004).

2. *For all finite* $X \subseteq B$, *if for all* $x \in X$,$(a, x) \in S$, *then* $(a, y) \in S$ *for all* $y \in glb(X)$.

3. *For all finite* $X \subseteq A$, *if for all* $x \in X$, $(x, b) \in S$, *then* $(y, b) \in S$ *for all* $y \in lub(X)$.

If $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ is a basic Boolean joining-systems, then we call S a basic Boolean joining space. We equivalently replace condition 2 and 3 by the following conditions and use them in later proofs:

$2'$ If $(a, x) \in S$ and $(a, y) \in S$, then $(a, x \cdot_B y) \in S$

$3'$ If $(a, x) \in S$ and $(b, x) \in S$, then $(a +_A b, x) \in S$

Lemma 5.1. *Given a basic Boolean joining-systems* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$, *the following are equivalent:*

(1) For all finite $X \subseteq B$, *if for all* $x \in X$,$(a, x) \in S$, *then* $(a, y) \in S$ *for all* $y \in glb(X)$.

(2) If $(a, x) \in S$ *and* $(a, y) \in S$, *then* $(a, x \cdot_B y) \in S$

Proof. $(1) \Rightarrow (2)$: If $(a, x) \in S$ and $(a, y) \in S$, then $\{x, y\} \subseteq B$ and $glb(\{x, y\}) = \{x \cdot_B y\}$. Therefore by (1) we know $(a, x \cdot_B y) \in S$.
$(2) \Rightarrow (1)$: Let $X = \{x_1, \ldots, x_n\}$ be a finite subset of B. Then $glb(X) = \{x_1 \cdot_B \ldots \cdot_B x_n\}$. Assume $(a, x_i) \in S$ for all $i \in \{1, \ldots, n\}$. Then by (2) we know $(a, x_1 \cdot_B x_2) \in S$. Moreover we can deduce that $(a, x_1 \cdot_B x_2 \cdot_B x_3) \in S, \ldots, (a, x_1 \cdot_B \ldots \cdot_B x_n) \in S$. □

Lemma 5.2. *Given a basic Boolean joining-systems* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$, *the following are equivalent:*

(1) For all finite $X \subseteq A$, *if for all* $x \in X$, $(x, b) \in S$, *then* $(y, b) \in S$ *for all* $y \in lub(X)$.

(2) If $(a, x) \in S$ *and* $(b, x) \in S$, *then* $(a +_A b, x) \in S$

Proof. $(1) \Rightarrow (2)$: If $(a, x) \in S$ and $(b, x) \in S$, then $\{a, b\} \subseteq A$ and $lub(\{a, b\}) = \{a +_A b\}$. Therefore by (1) we know $(a +_A b, x) \in S$.
$(2) \Rightarrow (1)$: Let $X = \{x_1, \ldots, x_n\}$ be a finite subset of A. Then $lub(X) = \{x_1 +_A \ldots +_A x_n\}$. Assume $(x_i, b) \in S$ for all $i \in \{1, \ldots, n\}$. Then by (2) we know $(x_1 +_A x_2, b) \in S$. Moreover we can deduce that $(x_1 +_A x_2 +_A x_3, b) \in S, \ldots, (x_1 +_A \ldots +_A x_n, b) \in S$. □

Moreover, we can equivalently define basic Boolean joining space using *ideal* and *filter*, which are standard algebraic notions.

Definition 5.6 (ideal (Givant and Halmos, 2009))**.** *Let* $\mathfrak{A}$ *be a Boolean algebra and* $I \subseteq A$. *For* I *to be an ideal of* $\mathfrak{A}$, *it is necessary and sufficient that the following three conditions be satisfied:*

1. $0_A \in I$

2. *for all* $x, y \in I$, $x +_A y \in I$

3. *for all* $x \in I$ *and* $y \in A$, *if* $y \leq_A x$ *then* $y \in I$

Definition 5.7 (filter (Givant and Halmos, 2009))**.** *Let* $\mathfrak{A}$ *be a Boolean algebra and* $F \subseteq A$. *For* F *to be a filter of* $\mathfrak{A}$, *it is necessary and sufficient that the following three conditions are satisfied:*

1. $1_A \in F$
2. *for all* $x, y \in F$, $x \cdot_A y \in F$
3. *for all* $x \in F$ *and* $y \in A$, *if* $x \leq_A y$ *then* $y \in F$

Example 5.2. *Given arbitrary infinite set* X, *then* $(2^X, \cup, \cap, -, \emptyset, X)$ *is a Boolean algebra. Let* $I = \{A' \subseteq X : A' \text{ is finite}\}$ *and* $F = \{A' \subseteq X : A' \text{ is cofinite}\}$, *where* A' *is cofinite means* $A - A'$ *is finite. Then* I *is an ideal and* F *is a filter.*

Let $F_\uparrow(X)$ be the filter generated by X, which means $F_\uparrow(X)$ is the smallest filter contains X, and $I_\downarrow(X)$ be the ideal generated by X, which means $I_\uparrow(X)$ is the smallest ideal contains X. Then we have the following proposition defining joining space by ideal and filter:

Proposition 5.1. *Given a structure* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$, *where* $\mathfrak{A}, \mathfrak{B}$ *are Boolean algebras and* $S \subseteq A \times B$, S *is a basic Boolean joining space in* $\mathbb{S}$ *if and only if it satisfies the following conditions:*

1. *For every finite set* $X \subseteq B$ *and* $a \in A$, *if for every* $x \in X$, $(a, x) \in S$, *then* $(a, y) \in S$, *for every* $y \in F_\uparrow(X)$.
2. *For every finite set* $X \subseteq A$ *and* $b \in B$, *if for every* $x \in X$, $(x, b) \in S$, *then* $(y, b) \in S$, *for every* $y \in I_\downarrow(X)$.

Proof. Assume S is a basic Boolean joining space. For the first condition, let X be an arbitrary finite subset of B. Without loss of generality, we let $X = \{x_1, \ldots, x_n\}$. Suppose $\forall x \in X$, $(a, x) \in S$. Then by applying clause $2'$ of Definition 5.5 finitely many times we have $(a, x_1 \cdot_B \ldots \cdot_B x_n) \in S$. Since for all $y \in F_\uparrow(X)$, $x_1 \cdot_B \ldots \cdot_B x_n \leq_B y$, we then know $(a, x_1 \cdot_B \ldots \cdot_B x_n) \preceq (a, y)$. Therefore $(a, y) \in S$ by Definition 5.5. Similarly we can prove that the second condition is satisfied.

Now assume S satisfies the two conditions in this proposition. Assume $(a, x) \in S$ and $(a, x) \preceq (b, y)$, then $x \leq_B y$ and $y \in F_\uparrow(x)$, hence $(a, y) \in S$. Moreover we have $b \leq_A a$ and $b \in I_\downarrow(a)$, so we have $(b, y) \in S$. Assume $(a, x) \in S$ and $(a, y) \in S$. Since $x \cdot_B y \in F_\uparrow(\{x, y\})$, we know $(a, x \cdot_B y) \in S$. Similarly we can prove if $(a, x) \in S$ and $(b, x) \in S$, then $(a +_A b, x) \in S$. Therefore S is a basic joining space. □

Up to now, we have defined basic Boolean joining-systems and joining space. But does a basic Boolean joining space always exist? The answer is positive. As the following proposition shows, the largest and the smallest basic Boolean joining space always exists.

Proposition 5.2. *Given two Boolean algebras* $\mathfrak{A}, \mathfrak{B}$,

1. $(\mathfrak{A}, \mathfrak{B}, A \times B)$ *is a basic Boolean joining-systems.*
2. *Let* $\mathfrak{I}$ *be a set of indexes, if for all* $i \in \mathfrak{I}$, $(\mathfrak{A}, \mathfrak{B}, S_i)$ *is a basic Boolean joining-systems, then* $(\mathfrak{A}, \mathfrak{B}, \cap_{i\in\mathfrak{I}} S_i)$ *is a basic Boolean joining-systems.*

Proof. The first item is trivial:

(1) If $(a, x) \in A \times B$ and $(a, x) \preceq (b, y)$, then $(b, y) \in A \times B$.

(2) If $(a, x) \in A \times B$ and $(a, y) \in A \times B$, then $(a, x \cdot_B y) \in A \times B$.

(3) If $(a, x) \in A \times B$ and $(b, x) \in A \times B$, then $(a +_A b, x) \in A \times B$.

Now we prove the second item. For every finite set $X \subseteq A$, if for every $x \in X$, $(x, b) \in \cap_{i\in\mathfrak{I}} S_i$, then $(x, b) \in S_i$ for every $i \in \mathfrak{I}$. Therefore by Proposition 5.1 we have $\forall y \in I_{\downarrow}(X)$, $(y, b) \in S_i$. So we must have $(y, b) \in \cap_{i\in\mathfrak{I}} S_i$. Similarly we can make use of the first item of Proposition 1. Therefore $\cap_{i\in\mathfrak{I}} S_i$ is a joining space of $\mathfrak{A} \times \mathfrak{B}$. □

5.3.1 Basic input/output logic and basic Boolean joining-systems

In this subsection, we use basic Boolean joining-systems to develop an algebraic semantics for basic input/output logic. We prove that for a set of mandatory norms O, (a, x) is derivable from O in basic input/output logic, if and only if it is in the basic Boolean joining space generated by O. To show this, we use a special Boolean algebra named Lindenbaum-Tarski algebra.

Let $\equiv$ be the provable equivalence relation on $L_{\mathbb{P}}$, i.e. for every formula $x, y \in L_{\mathbb{P}}$, $x \equiv y$ iff $\vdash x \leftrightarrow y$. Let $L_{\mathbb{P}}^{\equiv}$ be the set of equivalence classes that $\equiv$ induces on $L_{\mathbb{P}}$. For any formula $x \in L_{\mathbb{P}}$, let $[x]$ denote the equivalence class containing x.

Definition 5.8 (Lindenbaum-Tarski algebra (Blackburn et al., 2001))**.** *The Lindenbaum-Tarski algebra for propositional logic* $L_{\mathbb{P}}$ *is a structure* $\mathfrak{L} = (L_{\mathbb{P}}^{\equiv}, +, \cdot, -, 0, 1)$ *where* $[x] + [y] = [x \vee y]$, $[x] \cdot [y] = [x \wedge y]$, $-[x] = [\neg x]$, $0 = [\bot]$ *and* $1 = [\top]$.

For more details of Lindenbaum-Tarski algebra, readers are suggested to consult Chapter 5 of Blackburn et al. (2001). Every Lindenbaum-Tarski algebra is a Boolean algebra.

Let $O^{\equiv} = \{([a], [x]) : (a, x) \in O\}$. Let $\mathbb{S} = (\mathfrak{L}, \mathfrak{L}, S)$ be a basic Boolean joining-systems such that $O^{\equiv} \subseteq S$. By Proposition 5.2 we know such basic Boolean joining-systems always exist. Moreover, there is a smallest basic Boolean joining space O_2 such that $O^{\equiv} \subseteq O_2$ and for every basic Boolean joining space S that extends $O^{\equiv}$, $O_2 \subseteq S$. Here we use the notation O_2 for the resemblance of basic input/output logic. Such O_2 is the basic Boolean joining space generated by $O^{\equiv}$. The following proposition shows how we construct O_2.

Proposition 5.3. *Let* $O_2' = \bigcup_{i=0}^{\infty} O_2^i$ *be constructed as follows:* [‡]

- $O_2^0 = O^{\equiv}$
- O_2^{i+1} *contains all* $([a],[x])$ *for which:*

  (1) there is $([b],[y]) \in O_2^i$, $([b],[y]) \preceq ([a],[x])$*;*

  (2) there are $([a],[y]),([a],[z]) \in O_2^i$ *such that* $[x] = [y] \cdot [z]$*;*

  (3) there are $([b],[x]),([c],[x]) \in O_2^i$ *such that* $[a] = [b] + [c]$*.*

  Then $O_2' = O_2$*.*

Proof. We have to show three things: (a) $O^{\equiv} \subseteq O_2'$. (b) O_2' is a basic Boolean joining space. (c) every basic Boolean joining space S that extends $O^{\equiv}$, $O_2' \subseteq S$

(a) This is obvious in view of the construction.

(b) We show that 1, 2′ and 3′ from Definition 5.5 hold for O'.

- Let $([a],[x]) \in O_2'$ and $([a],[x]) \preceq ([b],[y])$. Hence there is $i \geq 0$ such that $([a],[x]) \in O_2^i$. By (1), $([b],[y]) \in O_2^{i+1} \subseteq O_2'$.
- Let $([a],[x]),([a],[y]) \in O_2'$. Hence there is $i,j \geq 0$ such that $([a],[x]) \in O_2^i$ and $([a],[y]) \in O_2^j$. Let $k = max(\{i,j\})$. Note that $O_2^i \subseteq O_2^{i+1}$ by the construction of O_2^i. Therefore we know $([a],[x]),([a],[y]) \in O_2^k$. Then by (2) we have $([a],[x] \cdot [y]) \in O_2^{k+1} \subseteq O_2'$.
- 3′ is shown analogously.

(c) Let S be an arbitrary basic Boolean joining space S that extends $O^{\equiv}$. We now prove that $O_2^i \subseteq S$, for all i. This can be proved by introduction on i. Indeed, $O_2^0 = O^{\equiv} \subseteq S$. Assume $O_2^k \subseteq S$, then for all $([a],[x]) \in O_2^{k+1}$,

1. if there is $([b],[y]) \in O_2^i$ and $([b],[y]) \preceq ([a],[x])$. Then $([b],[y]) \in S$. Since S is basic Boolean joining space, we know that $([a],[x]) \in S$.
2. If there are $([a],[y]),([a],[z]) \in O_2^i$ such that $[x] = [y] \cdot [z]$. Then $([a],[y]),([a],[z]) \in S$. Since S is basic Boolean joining space, we know that $([a],[x]) \in S$.
3. If there are $([b],[x]),([c],[x]) \in O_2^i$ such that $[a] = [b] + [c]$. Then $([b],[x]),([c],[x]) \in S$. Since S is basic Boolean joining space, we know that $([a],[x]) \in S$.

□

[‡]I thank a reviewer of Journal of Logic and Computation for his/her contribution to this proposition.

With proposition 5.3 at hand, we now prove a correspondence result. Intuitively, this result states that every (a, x) is logically derivable from O if and only if it is in the space of norms algebraically generated by O.

Theorem 5.1. *The following three propositions are equivalent:*

1. $(a, x) \in deriv_2(O)$.
2. $([a], [x]) \in O_2$.
3. $x \in out_2(O, a)$.

Proof. $1 \Rightarrow 2$: This can be proved simply by induction one the length of derivation.
$2 \Rightarrow 3$: Assume $([a], [x]) \in O_2$. Hence there is an $i \geq 0$ such that $([a], [x]) \in O_2^i$ (see Proposition 5.3). We show that for each $([a], [x]) \in O_2$, $x \in out_2(O, a)$ by induction over i. For the induction base let $([a], [x]) \in O_2^0 = O^{\equiv}$. Trivially $x \in out_2(O, a)$. For the inductive cases, assume the conclusion is true for O_2^i, consider $([a], [x]) \in O_2^{i+1}$. By proposition 5.3 we need to deal with three cases.

- If for some $([b], [y]) \in O_2^i$, $([b], [y]) \preceq ([a], [x])$, then by induction hypotheses we know $y \in out_2(O, b) = \bigcap\{Cn(O(V)) : b \in V, V \text{ is complete}\}$. Since $[a] \leq [b]$ and $[y] \leq [x]$ we know $a \vdash b, y \vdash x$ and $x \in Cn(y)$. Hence $x \in \bigcap\{Cn(O(V)) : b \in V, V \text{ is complete}\}$. Moreover, every complete set V contains a must contain b, hence $\bigcap\{Cn(O(V)) : b \in V, V \text{ is complete}\} \subseteq \bigcap\{Cn(O(V)) : a \in V, V \text{ is complete}\}$. Therefore $x \in \bigcap\{Cn(O(V)) : a \in V, V \text{ is complete}\}$, $x \in out_2(O, a)$.

- If there exist $([a], [y]), ([a], [z]) \in O_2^i$ such that $[x] = [y] \cdot [z]$. Then by induction hypotheses we know $y, z \in out_2(O, a) = \bigcap\{Cn(O(V)) : a \in V, V \text{ is complete}\}$. Therefore $y \wedge z \in \bigcap\{Cn(O(V)) : a \in V, V \text{ is complete}\}$ and $x \in \bigcap\{Cn(O(V)) : a \in V, V \text{ is complete}\}$. That is, $x \in out_2(O, a)$.

- If there exist $([b], [x]), ([c], [x]) \in O_2^i$ such that $[a] = [b] + [c]$. Then by induction hypotheses we know $x \in out_2(O, b)$ and $x \in out_2(O, c)$. Therefore $x \in \bigcap\{Cn(O(V)) : b \in V, V \text{ is complete}\}$ and $x \in \bigcap\{Cn(O(V)) : c \in V, V \text{ is complete}\}$. For every complete set V such that $b \vee c \in V$, it must be that either $b \in V$ or $c \in V$. Therefore, for every complete set V that contains $b \vee c$, $x \in Cn(V)$, which means $x \in \bigcap\{Cn(O(V)) : b \vee c \in V, V \text{ is complete}\}$, i.e. $x \in out_2(O, b \vee c)$, $x \in out_2(O, a)$.

$3 \Rightarrow 1$: This is a special case of the completeness of input/output logic. □

5.3.2 Input/output logics and joining-systems

The previous subsection proves a correspondence result between basic input/output logic and basic Boolean joining-systems. We now prove correspondence results between other input/output logics and other Boolean joining-systems.

Definition 5.9 (Boolean joining-systems)**.** *Given a structure* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ *where* $\mathfrak{A}$,$\mathfrak{B}$ *are Boolean algebras. Given the following conditions:*

1. *If* $(a, x) \in S$ *and* $(a, x) \preceq (b, y)$*, then* $(b, y) \in S$*.*
2. *If* $(a, x) \in S$ *and* $(a, y) \in S$*, then* $(a, x \cdot_B y) \in S$*.*
3. *If* $(a, x) \in S$ *and* $(b, x) \in S$*, then* $(a +_A b, x) \in S$*.*
4. *if* $(a, x) \in S$ *and* $(a \cdot_A x, y) \in S$*, then* $(a, y) \in S$*.*

- *If* $\mathbb{S}$ *satisfies (1) and (2), then* $\mathbb{S}$ *is a simple-minded Boolean joining-systems. S is a simple-minded Boolean joining space of* $\mathbb{S}$*.*
- *If* $\mathbb{S}$ *satisfies (1), (2) and (4), then* $\mathbb{S}$ *is a simple-minded reusable Boolean joining-systems. S is a simple-minded reusable Boolean joining space of* $\mathbb{S}$*.*
- *If* $\mathbb{S}$ *satisfies (1), (2), (3) and (4), then* $\mathbb{S}$ *is a basic reusable Boolean joining-systems. S is a basic reusable Boolean joining space of* $\mathbb{S}$*.*

Similar to Proposition 5.2, we prove the existence of the largest and the smallest simple-minded/basic/simple-minded reusable/basic reusable joining space.

Proposition 5.4. *Given two Boolean algebras* $\mathfrak{A}, \mathfrak{B}$*,*

1. $(\mathfrak{A}, \mathfrak{B}, A \times B)$ *is a simple-minded/simple-minded reusable/basic reusable Boolean joining-systems.*
2. *If for all* $i \in \mathfrak{I}$*,* $(\mathfrak{A}, \mathfrak{B}, S_i)$ *is a simple-minded Boolean joining-systems, then* $(\mathfrak{A}, \mathfrak{B}, \cap_{i\in\mathfrak{I}} S_i)$ *is a simple-minded Boolean joining-systems. And similarly for simple-minded reusable/basic reusable joining space.*

Proof. Similar to the proof of Proposition 5.2. Here we only prove that for reusable joining-systems, if $(a, x) \in \cap_{i\in\mathfrak{I}} S_i$ and $(a \cdot_A x, y) \in \cap_{i\in\mathfrak{I}} S_i$, then $(a, y) \in \cap_{i\in\mathfrak{I}} S_i$.

Assume $(a, x) \in \cap_{i\in\mathfrak{I}} S_i$ and $(a \cdot_A x, y) \in \cap_{i\in\mathfrak{I}} S_i$. Therefore for an arbitrary $i \in \mathfrak{I}$,$(a, x) \in S_i$ and $(a \cdot_A x, y) \in S_i$. Then we know $(a, y) \in S_i$. Therefore $(a, y) \in \cap_{i\in\mathfrak{I}} S_i$. □

Let O_1 to O_4 be respectively the smallest simple-minded/simple-minded reusable/basic reusable joining space based on Lindenbaum-Tarski algebra $\mathfrak{L}$ that extends $O^{\equiv}$. We have the the following correspondence result:

Theorem 5.2. *For $i \in \{1, 3, 4\}$, the following three statements are equivalent:*

1. $(a, x) \in deriv_i(O)$.
2. $([a], [x]) \in O_i$.
3. $x \in out_i(O, a)$.

Proof. Similar to the proof of Theorem 5.1. The key step is to establish an inductive construction of O_i. Here we only sketch the case for $i = 3$.

We first give an inductive construction of O_3 as follows: Let $O_3 = \bigcup_{i=0}^{\infty} O_3^i$, where

- $O_3^0 = O^{\equiv}$
- O_3^{i+1} contains all $([a], [x])$ for which:
  - (1) there is $([b], [y]) \in O_3^i$, $([b], [y]) \preceq ([a], [x])$;
  - (2) there is $([a], [y]), ([a], [z]) \in O_3^i$ such that $[x] = [y] \cdot [z]$;
  - (3) there is $([a], [y]), ([a] \cdot [y], [x]) \in O_3^i$.

Then we prove no matter how $([a], [x])$ is generated in O_3, we have $x \in out_3(O, a)$. For example if $([a], [x]) \in O_3$ and it is because of the existence of $([a], [y]), ([a] \cdot [y], [x]) \in O_3$. Then use inductive hypothesis we know $y \in out_3(O, a)$ and $x \in out_3(O, a \wedge y)$. Then by applying the definition of out_3 we prove $x \in out_3(O, a)$. □

5.4 Intuitionistic input/output logic and Heyting joining systems

A frequent belief about input/output logic is that it presupposes classical propositional logic. Parent et al. (2014) show that this is a misunderstanding by building input/output logic on top of intuitionistic logic. In this section, we show that there is an algebraic companion for intuitionistic input/output logic, the Heyting joining-systems. To do this we first introduce intuitionistic input/output logic and Heyting joining-systems, then we construct the correspondence.

5.4.1 Intuitionistic input/output logic

Intuitionistic logic (Van Dalen, 1986) is different from classical logic by omitting the principle of excluded middle and the *reductio ad absurdum* rule. Intuitionistic input/output logic is based on

intuitionistic propositional logic (IPL), the propositional fragment of intuitionistic logic. Given a set of propositional letters $\mathbb{P}$, the language of intuitionistic propositional logic L_I is defined as follows:

$$a, b ::= \bot \mid p \mid (a \wedge b) \mid (a \vee b) \mid (a \to b)$$

Here $p \in \mathbb{P}$ and we use $\neg a$ as an abbreviation of $a \to \bot$. A proof system of intuitionistic propositional logic used in Parent et al. (2014) is defined via the following sequent calculus:

- Group 1: Let A, B be finite set of formulas
  - (Ref) If $a \in A$, then $A \vdash_I a$
  - (Mon) $\frac{A \vdash_I a}{A \cup B \vdash_I a}$
  - (Cut) $\frac{A \vdash_I a \quad A \cup \{a\} \vdash_I b}{A \vdash_I b}$

  The labels (Ref) and (Mon) are mnemonic for "reflexivity" and "monotony" respectively.

- Group 2:
  - $\frac{A \vdash_I a \quad A \vdash_I b}{A \vdash_I a \wedge b}(\wedge\text{:I})$
  - $\frac{A \vdash_I a \wedge b}{A \vdash_I a}(\wedge\text{:E})$
  - $\frac{A \vdash_I a}{A \vdash_I a \vee b}(\vee\text{:I})$
  - $\frac{A \cup \{a\} \vdash_I c \quad A \cup \{b\} \vdash_I c \quad A \vdash_I a \vee b}{A \vdash_I c}(\vee\text{:E})$
  - $\frac{A \cup \{a\} \vdash_I b}{A \vdash_I a \to b}(\to\text{:I})$
  - $\frac{A \vdash_I a \quad A \vdash_I a \to b}{A \vdash_I b}(\to\text{:E})$
  - $\frac{A \vdash_I \bot}{A \vdash_I a}(\bot\text{:E})$

If $\emptyset \vdash_I a$ then we say a is provable in IPL. If A is infinite, then we let $A \vdash_I a$ iff $A' \vdash_I a$ for some finite $A' \subseteq A$.

Let O be a set of ordered pairs of formulas of L_I. For a set of formulas $A \subseteq L_I$, let $Cn^I(A) = \{a \in L_I : A \vdash_I a\}$. To define intuitionistic input/output logic the concept of saturated set is needed.

Definition 5.10 (saturated set (Thomason, 1968))**.** *A set $A \subseteq L_I$ is said to be saturated if the following three conditions hold:*

1. $A \nvdash_I \bot$
2. *if $a \vee b \in A$ then $a \in A$ or $b \in A$*
3. *if $A \vdash_I a$ then $a \in A$*

Parent et al. (2014) develop intuitionistic input/output logic as follows:

- $out_1^I(O, A) = Cn^I(O(Cn^I(A)))$.
- $out_2^I(O, A) = \bigcap\{Cn^I(O(B)) : A \subseteq B, B \text{ is saturated or } B = L_I\}$.
- $out_3^I(O, A) = \bigcap\{Cn^I(O(B)) : A \cup O(B) \subseteq B = Cn^I(B)\}$.
- $out_4^I(O, A) = \bigcap\{Cn^I(O(B)) : A \cup O(B) \subseteq B, B \text{ is saturated or } B = L_I\}$.

The proof system of intuitionistic input/output logics are similar to its propositional counterpart. Parent et al. (2014) use AND, OR, CT and the intuitionistic version of SI and WO:

- SI^I (intuitionistic strengthening the input): from (a, x) to (b, x) whenever $b \vdash_I a$
- WO^I (intuitionistic weakening the output): from (a, x) to (a, y) whenever $x \vdash_I y$

The derivation system based on the rules SI_I, WO_I and AND is called $deriv_1^I$. Adding OR and CT to $deriv_1^I$ gives $deriv_2^I$ and $deriv_3^I$ respectively. The five rules together define $deriv_4^I$. In Parent et al. (2014), the following theorems are given:

Theorem 5.3. *(Parent et al., 2014)*

- *For* $i \in \{1, 2, 3\}$, $x \in out_i^I(O, a)$ *iff* $(a, x) \in deriv_i^I(O)$.
- *If* $(a, x) \in deriv_4^I(O)$, *then* $x \in out_4^I(O, a)$.[§]

5.4.2 Heyting joining-systems

Heyting algebra was introduced by Arend Heyting in 1930s to formalize intuitionistic logic. Heyting algebra generalize Boolean algebra in the sense that a Heyting algebra satisfying $x + (-x) = 1$ is a Boolean algebra.

Definition 5.11 (Heyting algebra (Heyting, 1930))**.** *A Heyting algebra is a partially ordered set* $(H, 0, 1 \leq, \cdot, +, \rightarrow)$ *with a smallest elements* 0, *a largest element* 1 *and three operators* $\cdot$, $+$ *and* $\rightarrow$ *satisfying the following conditions, for all* $x, y, z \in H$

1. $x \leq 1$.
2. $x \cdot y \leq x$.
3. $x \cdot y \leq y$.

[§]It is an open problem whether the other direction of the implication holds.

	$x \cdot y$			$x + y$			$x \to y$		
$x \diagdown y$	0	$\frac{1}{2}$	1	0	$\frac{1}{2}$	1	0	$\frac{1}{2}$	1
0	0	0	0	0	$\frac{1}{2}$	1	1	1	1
$\frac{1}{2}$	0	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	1	0	1	1
1	0	$\frac{1}{2}$	1	1	1	1	0	$\frac{1}{2}$	1

Figure 5.1: Operations on $\{0, \frac{1}{2}, 1\}$

4. $z \leq x$ *and* $z \leq y$ *implies* $z \leq x \cdot y$.
5. $0 \leq x$.
6. $x \leq x + y$.
7. $y \leq x + y$.
8. $x \leq z$ *and* $y \leq z$ *implies* $x + y \leq z$.
9. $z \leq (x \to y)$ *iff* $z \cdot x \leq y$.

Example 5.3. *Let* $\mathfrak{H} = (H, 0, 1 \leq, \cdot, +, \to)$ *where* $H = \{0, \frac{1}{2}, 1\}$, $0 \leq \frac{1}{2} \leq 1$, $\cdot, +$ *and* $\to$ *are represented by Figure 5.1. Then* $\mathfrak{H}$ *is a Heyting algebra, but not a Boolean algebra.*

A valuation from IPL to a Heyting algebra is a function $V : \mathbb{P} \to H$. V is extended to arbitrary formulas by putting:

$$V(\bot) = 0.$$
$$V(\top) = 1.$$
$$V(a \wedge b) = V(a) \cdot V(b).$$
$$V(a \vee b) = V(a) + V(b).$$
$$V(a \to b) = V(a) \to V(b).$$

A formula a is said to be H-valid if $V(a) = 1$ for all valuations V on a Heyting algebra H.

Theorem 5.4. *(Troelstra and van Dalen, 1988) A formula a is provable in IPL iff a is H-valid for all Heyting algebra H.*

Heyting joining-systems are defined in a similar way to Boolean joining-systems.

Definition 5.12 (Heyting joining-systems)**.** *A Heyting joining-systems is a structure* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ *such that* $\mathfrak{A} = (A, 0_A, 1_A \leq_A, \cdot_A, +_A, \to_A)$, $\mathfrak{B} = (B, 0_B, 1_B, \leq_B, \cdot_B, +_B, \to_B)$ *are Heyting algebras and* $S \subseteq A \times B$ *satisfies certain conditions:*

1. *If* $(a,x) \in S$ *and* $(a,x) \preceq (b,y)$, *then* $(b,y) \in S$.

2. *If* $(a,x) \in S$ *and* $(a,y) \in S$, *then* $(a, x \cdot_B y) \in S$.

3. *If* $(a,x) \in S$ *and* $(b,x) \in S$, *then* $(a +_A b, x) \in S$.

4. *if* $(a,x) \in S$ *and* $(a \cdot_A x, y) \in S$, *then* $(a,y) \in S$.

The Heyting joining-systems satisfies 1 and 2 is called simple-minded. Adding 3 and 4 produces basic and simple-minded reusable Heyting joining-systems respectively. The four condition together give rise to basic reusable Heyting joining-systems.

Let $\equiv_I$ be the provable equivalence relation on L_I, i.e. for every formula $x, y \in L_I$, $x \equiv_I y$ iff $\vdash_I x \leftrightarrow y$. Let $L_I^{\equiv_I}$ be the equivalence classes that $\equiv_I$ induces on L_I. For any formula $x \in L_I$, let $[x]^I$ denote the equivalence class containing x. The intuitionistic Lindenbaum-Tarski algebra for L_I is a structure $(L_I^{\equiv_I}, 0, 1, \leq, +, \cdot, \rightarrow)$ where $0 = [\bot]^I$, $1 = [\top]^I$, $[x]^I \leq [y]^I$ iff $[x]^I \cdot [y]^I = [x]^I$, $[x]^I + [y]^I = [x \vee y]^I$, $[x]^I \cdot [y]^I = [x \wedge y]^I$, $[x]^I \rightarrow [y]^I = [x \rightarrow y]^I$.

It can be verified that an intuitionistic Lindenbaum-Tarski algebra is a Heyting algebra. Let O be a set of ordered pairs of formulas of L_I. Let $O^{\equiv_I} = \{([a]^I, [x]^I) : (a,x) \in O\}$, and O_1^I to O_4^I be the simple-minded/basic/simple-minded reusable/basic reusable Heyting joining-systems generated by $O^{\equiv_I}$ respectively. Then we have the following correspondence results between intuitionistic input/output logics and Heyting joining-systems:

Theorem 5.5. *For* $i \in \{1,2,3\}$, *the following three statements are equivalent:*

1. $(a,x) \in deriv_i^I(O)$.

2. $([a],[x]) \in O_i^I$.

3. $x \in out_i^I(O,a)$.

Proof. Similar to the proof of Theorem 5.1. The key step is to establish an inductive construction of O_i. Here we only sketch the case for $i = 3$. Other cases are similar.

We first give an inductive construction of O_3^I as follows: Let $O_3^I = \bigcup_{i=0}^{\infty} O_3^{I,i}$, where

- $O_3^{I,0} = O^{\equiv_I}$.
- $O_3^{I,i+1}$ contains all $([a],[x])$ for which:
  - (1) there is $([b],[y]) \in O_3^{I,i}$, $([b],[y]) \preceq ([a],[x])$;
  - (2) there is $([a],[y]),([a],[z]) \in O_3^{I,i}$ such that $[x] = [y] \cdot [z]$;
  - (3) there is $([a],[y]),([a] \cdot [y],[x]) \in O_3^{I,i}$.

Then we prove no matter how $([a],[x])$ is generated in O_3^I, we have $x \in out_3^I(O,a)$. For example if $([a],[x]) \in O_3^I$ and it is because of the existence of $([a],[y]),([a]\cdot[y],[x]) \in O_3^I$. Then use inductive hypothesis we know $y \in out_3^I(O,a)$ and $x \in out_3^I(O,a \wedge y)$. Then by applying the definition of out_3^I we prove $x \in out_3^I(O,a)$. □

5.5 Application of the algebraic representation

Lindahl and Odelstad's joining-systems, as well as our Boolean and Heyting joining-systems, provide algebraic representation of norms and normative system. One advantage of such an algebraic representation is that we can use them to study the *similarity* of normative systems. Algebraic notions such as homomorphism and isomorphism are natural tools to explore the similarity of structures. We approach the similarity of normative systems by introducing isomorphism and embedding between normative systems.

5.5.1 Similarity of normative systems

For two algebraic structures $\mathfrak{A}$ and $\mathfrak{B}$, if they are isomorphic then they are essentially the same. We can extend the isomorphism of Boolean algebra to Boolean joining-systems.

Definition 5.13 (isomorphism of Boolean algebra (Givant and Halmos, 2009))**.** *For two Boolean algebras $\mathfrak{A} = (A, +_A, \cdot_A, -_A, 0_A, 1_A)$ and $\mathfrak{A}' = (A', +_{A'}, \cdot_{A'}, -_{A'}, 0_{A'}, 1_{A'})$ and h a map from A to A'. We say that h is an isomorphism from $\mathfrak{A}$ to $\mathfrak{A}'$ iff for any $x, y \in A$, h satisfies the following conditions:*

1. *h is bijective.*
2. $h(x +_A y) = h(x) +_{A'} h(y)$.
3. $h(x \cdot_A y) = h(x) \cdot_{A'} h(y)$.
4. $h(1_A) = 1_{A'}$.

Given an isoporphism h from $\mathfrak{A}$ to $\mathfrak{A}'$, it is easy to check that for all $x, y \in A$ and $x', y' \in A'$, if $h(x) = x'$ and $h(y) = y'$, then $x \leq_A y$ iff $x' \leq_{A'} y'$. Now we extend isomorphism to Boolean joining-systems.

Definition 5.14 (isomorphism of joining-systems)**.** *For two Boolean joining-systems $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ and $\mathbb{S}' = (\mathfrak{A}', \mathfrak{B}', S')$ and h a map from $A \cup B$ to $A' \cup B'$. We say that h is an isomorphism from $\mathbb{S}$ to $\mathbb{S}'$ iff h satisfies the following conditions:*

1. *h is bijective.*

2. *the restriction of h on A is an isomorphism from A to A'.*

3. *the restriction of h on B is an isomorphism from B to B'.*

4. *$(a, x) \in S$ iff $(h(a), h(x)) \in S'$.*

If there is an isomorphism form $\mathbb{S}$ to $\mathbb{S}'$, then we say $\mathbb{S}$ and $\mathbb{S}'$ are isomorphic. Two isomorphic joining-systems can naturally be understood as structurally similar.

In political philosophy, research on totalitarianism (Arendt, 1958; Armstrong, 1961) views the ideology of Nazi Germany and Soviet Union to be similar. Stalinism and Nazism are described as "totalitarian twins". Using the algebraic representation of normative systems we can describe such similarity in a mathematical flavor.

Example 5.4. *Let H be "you worship Hitler", S be "you respect Stalin", N be "you are a member of the Nazi Party", C be "you are a member of the communist party", R be "you are against to the rich people", J be "you hate Jews". Let $\mathfrak{B}_1$ be the Boolean algebra generated by $\{H, N, J\}$, $\mathfrak{B}_2$ be the Boolean algebra generated by $\{S, C, R\}$. Let $S_1 = \{(\top, H), (N, J)\}$ be the normative system saying "you are obligated to worship Hitler" and "you are obligated to hate Jews, given the condition that you are a member of the Nazi Party". Let $S_2 = \{(\top, S), (C, R)\}$ be the normative system saying "you are obligated to respect Stalin" and "you are obligated to be against to the rich people, given the condition that you are a member of the communist party". For joining-systems $(\mathfrak{B}_1, \mathfrak{B}_1, S_1)$ and $(\mathfrak{B}_2, \mathfrak{B}_2, S_2)$, we can build an isomorphism h such that $h(H) = S, h(N) = C, h(J) = R$. We therefore conclude that S_1 and S_2 are similar.*

Not only isomorphism can be used as an algebraic tool to analyze the similarity of normative systems, embedding is also a useful tool.

Definition 5.15 (Embedding of joining-systems)**.** *For two joining-systems $\mathbb{S} = (A, B, S)$ and $\mathbb{S}' = (A', B', S')$ and h a map from $A \cup B$ to $A' \cup B'$. We say that h is an embedding from $\mathbb{S}$ to $\mathbb{S}'$ iff h satisfies the following conditions:*

1. *h is injective.*

2. *the restriction of h on A is an isomorphism from A to $h(A)$.*

3. *the restriction of h on B is an isomorphism from B to $h(B)$.*

4. *if $(a, x) \in S$ then $(h(a), h(x)) \in S'$.*

In political philosophy, totalitarianism is generally viewed as an extreme version of authoritarianism (Sondrol, 2009). This suggests that a normative system of totalitarianism can be considered as an extension of a normative system of authoritarianism. Therefore mathematically there should

be an embedding from the normative system of authoritarianism to the normative system of totalitarianism. The following is an illustration.

Example 5.5. *Let $\mathfrak{B}_1, S_1$ be same as in the previous example. Let G be "you like Gaddafi", F be "you are a follower of Gaddafi", A be "you dislike America". Let $\mathfrak{B}_3$ be the Boolean algebra generated by $\{G, F, A\}$. Let $S_3 = \{(\top, G)\}$ be the normative system saying "you are obligatory to like Gaddafi". For joining-systems $(\mathfrak{B}_1, \mathfrak{B}_1, S_1)$ and $(\mathfrak{B}_3, \mathfrak{B}_3, S_3)$, we can build an embedding h such that $h(G) = H, h(F) = N, h(A) = J$. We therefore say that S_1 is an extension of S_3.*

It must be confessed that our examples greatly simplify the complexity of political philosophy to a degree that lots of valuable information is lost. However, we still believe that the algebraic approach offers an interesting mathematical tool for political philosophy in the sense that more complex normative systems can also be characterized by joining-systems based on more expressive algebras. For example the expressive power of polyadic algebra is the same as first-order logic (Ságí, 2013). We may develop polyadic joining-systems to represent normative systems. Using polyadic joining-systems we can discuss the similarity of different ideology without losing too much valuable information.

5.5.2 The core of a normative system

In Section 5.3 the narrowness relation $\preceq$ is defined as $(a, x) \preceq (b, y)$ iff $b \leq_A a$ and $x \leq_B y$. We further define the strict narrowness relation $\prec$ as $(a, x) \prec (b, y)$ iff $(a, x) \preceq (b, y)$ and not $(b, y) \preceq (a, x)$. Intuitively, if $(a, x) \prec (b, y)$ then (a, x) is stronger than (b, y) in the sense that if (a, x) is in a joining-systems, then (b, y) must also be in the joining-systems, but not vice verse.

We now use the strict narrowness relation to define the core of a normative system. A norm (a, x) is minimal in a joining-systems $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ iff there is no $(b, y) \in S$ such that $(b, y) \prec (a, x)$. In Odelstad and Lindahl (2002), such a minimal norm is called a connection from $\mathfrak{A}$ to $\mathfrak{B}$. As noticed by Odelstad and Lindahl (2002), the set of all minimal elements of a joining-systems can be viewed as the core of the system in the sense that the whole system is uniquely determined by its minimal norms. Therefore we can logically deduce the whole system, if we know the core of the system. For a joining-systems $\mathbb{S}$, let $core(\mathbb{S}) = \{(a, x) \in S : (a, x) \text{ is minimal in } \mathbb{S}\}$ denote the set of all its minimal norms. The following are some formal statements about the properties of the core of finite joining-systems.

Observation 5.1. *For every joining-systems $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$, if S is finite then $core(\mathbb{S}) \neq \emptyset$.*

Proof. The proof is trivial. Due to the fact that S is finite, there is no infinite descending chain on $\prec$. □

Observation 5.2. *For all joining-systems* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$*, if* S *is finite, then for any* $(a, x) \in S$*, there exists* $(b, y) \in core(\mathbb{S})$ *such that* $(b, y) \preceq (a, x)$*.*

Proof. Let (a, x) be an arbitrary norm in S. If $(a, x) \in core(\mathbb{S})$, then $(a, x) \preceq (a, x)$ and we are done. If $(a, x) \notin core(\mathbb{S})$, then (a, x) is not a minimal norm. Hence there exist some (b, y) such that $(b, y) \prec (a, x)$. If $(b, y) \in core(\mathbb{S})$ then we are done. If not, then there exist some (c, z) such that $(c, z) \prec (b, y)$. Since S is finite, this procedure will stop at some point. Then by transitivity of $\preceq$, there must exist some $(a', x') \in core(\mathbb{S})$ such that $(a', x') \preceq (a, x)$. □

Observation 5.3. *For two joining-systems* $\mathbb{S} = (\mathfrak{A}, \mathfrak{B}, S)$ *and* $\mathbb{S}' = (\mathfrak{A}, \mathfrak{B}, S')$*,if both* S *and* S' *are finite, then* $core(\mathbb{S}) = core(\mathbb{S}')$ *iff* $S = S'$*.*

Proof. The right to left direction is trivial. For the left to right direction. Assume $core(\mathbb{S}) = core(\mathbb{S}')$. For any $(a, x) \in S$, by Observation 2 there exist $(b, y) \in core(\mathbb{S})$ such that $(b, y) \preceq (a, x)$. By assumption we know $(b, y) \in core(\mathbb{S}')$. Then by the definition of joining space we know $(a, x) \in S'$. Therefore $S \subseteq S'$. Similarly we can prove $S \supseteq S'$. □

All the three observations are valid only for finite joining-systems. Observation 5.1 states that the core always exist. Observation 5.3 states that the whole system is determined by the core. Regarding observation 5.2, it states that for any norms which is not in the core, there is a norm in the core which is stronger and therefore able to generate it. Observation 5.2 partly answers the problem of norms redundancy, which is raised by Boella et al. (2008b) and addressed by van der Torre (2010a). According to observation 5.2, all norms which are not in the core are redundant.

5.6 Related work

In Makinson and van der Torre (2000) , norms are simply ordered pairs of formula. Operators on norms are not defined. Such limitation is overcame in Stolpe (2010a) by introducing conjunction ($\overline{\wedge}$) and disjunction ($\underline{\vee}$) of norms. For two norms (a, x) and (b, y), Stolpe defines:

- $(a, x)\overline{\wedge}(b, y) := (a \vee b, x \wedge y)$
- $(a, x)\underline{\vee}(b, y) := (a \wedge b, x \vee y)$

Using such definition, Stolpe shows that the structure $(L_{\mathbb{P}} \times L_{\mathbb{P}}, \overline{\wedge}, \underline{\vee}, (\top, \bot), (\bot, \top))$ is indeed a bounded lattice with $(\top, \bot)$ the bottom element and $(\bot, \top)$ the top element. The negation of a norm (a, x) is then naturally defined as $-(a, x) := (\neg a, \neg x)$ because $(a, x)\overline{\wedge}(\neg a, \neg x) = (a \vee \neg a, x \wedge \neg x) = (\top, \bot)$.

It is well known in lattice theory (Garg, 2015) that every lattice can alternatively be defined as an ordered structure. Take the Lindenbaum-Tarski algebra $\mathfrak{L} = (L_{\equiv_{\mathbb{P}}}, +, \cdot, -, 0, 1)$ and the narrowness relation $\preceq$ over $\mathfrak{L} \times \mathfrak{L}$. Then it can be verified that $(\mathfrak{L} \times \mathfrak{L}, \preceq, ([1], [0]), ([0], [1]))$ is also a bounded lattice. These two approaches to define the lattice of norms reveals further connections between input/output logic and theory of joining-systems.

Stolpe (2015) provides a semantics for input/input output logic based on formal concept analysis. Stolpe offers powerful analytical techniques for classifying, visualising and analysing input/output relations, revealing implicit hierarchical structure and/or natural clusterings and dependencies.

5.7 Summary

The main contribution of this chapter is the presentation of some algebraic frameworks to normative systems. We have introduced Boolean joining-systems as the algebraic semantics for input/output logic and Heyting joining-systems as the algebraic semantics for intuitionistic input/output logic. Those algebraic semantics provides algebraic representation of norms and normative systems. We have introduced isomorphism and embedding of joining-systems and used them to study the similarity of normative systems.

Chapter 6

On the Complexity of Norm-based Deontic Logic I

Abstract

It is well-known in theoretical computer science that complexity is an indispensable component of every logic. So far, previous literature in input/output logic focuses on proof theory and semantics, and neglects complexity. This chapter adds the missing important component by giving the complexity results of several decision problems of input/output logic. Our results show that input/output logic is coNP-hard and in the 2nd level of the polynomial hierarchy.

6.1 Introduction

Recently, deontic logic has found its application in legal informatics (Governatori et al., 2013). Legal informatics is experiencing growth in activity in recent years, also at the industrial level. Several research projects aimed at designing web services for helping legal professionals to retrieve the information they are interested in have been approved recently by the EU commission and other institutions, e.g. Legivoc*. These projects discover inter-links between legal documents or classify and connect them to legal ontologies, by exploiting natural language processing (NLP) tools such as parsers and statistical algorithms (Boella et al., 2014, 2013). Although these techniques provide valid solutions to help navigate legislation, the overall usefulness of the systems are limited due to their focus on terminological issues and information retrieval while disregarding the specific semantic aspects, which allow legal reasoning. It becomes necessary to study the *logical architecture* of legal text, in order to enable deeper understanding of legislative text by human users and intelligent systems.

Nevertheless, such applications to legal informatics can only be developed if the complexity of norm-based deontic logics is well studied. Moreover, it is well-known in theoretical computer science that complexity is an indispensable component of every logic. So far, previous literature on norm-based deontic logics (except deontic defeasible logic) focuses on proof theory and semantics, and neglects complexity. This and the next chapter add the missing important component of norm-based deontic logics. Our results in this chapter show that most decision problems of input/output logic are NP-hard and in the 2nd level of the polynomial hierarchy, which means that although the complexity of input/output logic is not low, it is not astonishingly high. For example modal logic is at least as complex as input/output logic because modal logic is PSPACE-complete. We study the complexity of other norm-based deontic logic in the next chapter.

The rest of this chapter is organized as follows. We recap some background knowledge of complexity theory in Section 6.2. Then from Section 6.3 to 6.5 we study the complexity of unconstrained/constrained/permissive input/output logic respectively. In Section 6.6 we show how to reduce the complexity of input/output logic by imposing syntactical constrains. We summarize this chapter in Section 6.7.

6.2 Background: computational complexity theory

Computational complexity theory is the theory to investigate the time, memory, or other resources required for solving computational problems. In this section we briefly review some concepts and

*http://www.legivoc.eu

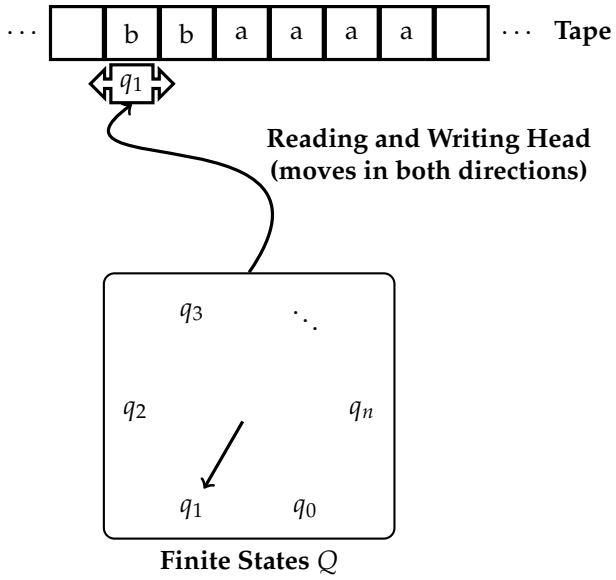

Figure 6.1: Turing machine visualized

results from computational complexity theory which will be used in this chapter. A comprehensive introduction of complexity theory can be found in Arora and Barak (2009) and Sipser (2012).

Definition 6.1 (Turing machine (Sipser, 2012))**.** *A Turing machine is a 7-tuple,* $(Q, \Sigma, \Gamma, \delta, q_0, q_{accept}, q_{reject})$*, where* Q, Σ, Γ *are all finite sets and*

1. Q *is the set of states,*
2. Σ *is the input alphabet not containing the blank symbol* $\llcorner\lrcorner$*,*
3. Γ *is the tape alphabet, where* $\llcorner\lrcorner \in \Gamma$ *and* $\Sigma \subseteq \Gamma$*,*
4. $\delta : Q \times \Gamma \mapsto Q \times \Gamma \times \{L, R\}$ *is the transition function,*
5. $q_0 \in Q$ *is the start state,*
6. $q_{accept} \in Q$ *is the accept state, and*
7. $q_{reject} \in Q$ *is the reject state, where* $q_{accept} \neq q_{reject}$

Figure 6.1 is a visualization of a Turing machine. Given a string $w = w_1 w_2 \dots w_n \in \Sigma^*$ and a Turing machine $M = (Q, \Sigma, \Gamma, \delta, q_{accept}, q_{reject})$, M computes w as follows.

- Initially w is put in the tape and the machine is in the start state q_0 and reading the first symbol of w using its tape head.

- At each step, M is in some state q and reading a symbol from the tape, say s. M then looks up the transition function δ and moves accordingly. For example, if $\delta(q, s) = (q', s', L)$ then M move to state q', overwrites the symbol s with s' on the tape and moves the tape head to the left.

- M stops if it is in the state q_{accept} or q_{reject}. These two states are called halting states. M accepts w if it stops with state q_{accept}. M rejects w if it stops with state q_{reject}.

When we start a Turing machine on an input, three outcomes are possible: accept, reject, or loop. By loop we mean the machine never enters a halting state. A Turing machine is called a decider if it halts on every input. Given a decider M, the language decided by M is the set of strings accepted by M, denoted $L(M)$. A function $f : \Sigma^* \mapsto \Sigma^*$ is computable if there is a decider M, on every input w, halts with just $f(w)$ on its tape.

A decision problem consists of a set of inputs and a question with a "yes" or "no" answer for each input. Every language $L \subseteq \Sigma^*$ gives rise to the following decision problem: given $x \in \Sigma^*$, is $x \in L$? Conversely, every decision problem can be thought of as arising from a language, namely, the language consisting of all inputs with a "yes" answer.

A non-deterministic Turing machine is defined similarly to a deterministic Turing machine. The only difference is that the transition function of a non-deterministic Turing machine has the form

$$\delta : Q \times \Gamma \mapsto 2^{Q \times \Gamma \times \{L,R\}}.$$

The computation of a non-deterministic Turing machine is a tree whose branches correspond to different possibilities of computation. If some branches of the computation leads to the accept state, the machine accepts the input. A non-deterministic decider is a non-deterministic Turing machine such that for all input, the computation tree has no infinite branch. Every decider, deterministic or non-deterministic, has its time complexity.

Definition 6.2 (Time complexity (Sipser, 2012))**.** *Given a deterministic decider M, the time complexity of M is the function $f : \mathbb{N} \mapsto \mathbb{N}$, where $f(n)$ is the maximum number of steps that M uses on any input of length n. Given a non-deterministic decider M, the time complexity of M is the function $f : \mathbb{N} \mapsto \mathbb{N}$, where $f(n)$ is the maximum number of steps that M uses on any branch of its computation on any input of length n.*

Let f and g be functions $f, g : \mathbb{N} \mapsto \mathbb{N}$. Say $f \in \mathbf{O}(g)$ if positive integers c and n_0 exist such that for every integer $n \geq n_0$

$$f(n) \leq cg(n).$$

A decider M is polynomial in time if its time complexity is in $\mathbf{O}(n^c)$ for some natural number c. Here $n^c : \mathbb{N} \mapsto \mathbb{N}$ is understood as a function such that for all $k \in \mathbb{N}$, $n^c(k) = k^c$.

Definition 6.3 (the P class (Sipser, 2012)). *P is the class of languages that have a polynomial time deterministic decider.*

Definition 6.4 (the NP class (Sipser, 2012)). *NP is the class of languages that have a polynomial time non-deterministic decider.*

The complement of a language $L \subseteq \Sigma^*$ is the language $\overline{L} = \{w \in \Sigma^* : w \notin L\}$.

Definition 6.5 (the coNP class (Sipser, 2012)). *coNP is the class of languages of which the complement is in NP .*

Definition 6.6 (Reduction (Sipser, 2012)). *Given two language $L \subseteq \Sigma^*$ and $L' \subseteq \Sigma'^*$, a reduction from L to L' is a function $f : \Sigma \mapsto \Sigma'$ such that for all $w \in \Sigma^*$, $w \in L$ iff $f(w) \in L'$. A reduction f is polynomial if f is computable by a polynomial time Turing machine.*

Definition 6.7 (NP-complete and coNP-complete (Sipser, 2012)). *A language L is NP (resp. coNP)-complete if it is in NP (resp. coNP) and for every L' in NP (resp. coNP) there is a polynomial reduction from L' to L.*

Definition 6.8 (NP-hard and coNP-hard (Sipser, 2012)). *A language L is NP (resp. coNP)-hard if for every L' in NP (resp. coNP) there is a polynomial reduction from L' to L.*

A well known NP-complete problem (proven by Cook (1971) and Levin (1975)) is the satisfiability problem of propositional logic (SAT): given a propositional formula x, is x satisfiable? As a consequence, the validity problem of propositional logic (given a propositional formula x, is x valid?) is coNP-complete.

The boolean hierarchy is the hierarchy of boolean combinations (intersection, union and complementation) of NP classes. BH_1 is the same as NP. BH_2 is the class of languages which are the intersection of a language in NP and a language in coNP. Wagner (1986) shows that the following 2-parity SAT problem is complete for BH_2:

Given two propositional formulas x_1 and x_2 such that if x_2 is satisfiable then x_1 is satisfiable, is it true that x_1 is satisfiable while x_2 is not?

Oracle Turing machine and two complexity classes related to oracle Turing machine will be used in this thesis.

Definition 6.9. ***3.1 (oracle Turing machine Arora and Barak (2009))*** *An oracle for a language L is a device that is capable of reporting whether any string w is a member of L. An oracle Turing machine M^L is a modified Turing machine that has the additional capability of querying an oracle. Whenever M^L writes a string on a special oracle tape it is informed whether that string is a member of L, in a single computation step.*

$\mathtt{P}^{\mathtt{NP}}$ is the class of problems solvable by a deterministic polynomial time Turing machine with an $\mathtt{NP}$ oracle. $\mathtt{NP}^{\mathtt{NP}}$ is the class of problems solvable by a non-deterministic polynomial time Turing machine with an $\mathtt{NP}$ oracle. $\mathtt{P}^{\mathtt{NP}}$, $\mathtt{NP}^{\mathtt{NP}}$ and $\mathtt{coNP}^{\mathtt{NP}}$ are also termed as Δ_2^p, Σ_2^p and Π_2^p respectively, to mark the fact that $\mathtt{P}^{\mathtt{NP}}$, $\mathtt{NP}^{\mathtt{NP}}$ and $\mathtt{coNP}^{\mathtt{NP}}$ belong to the *2nd* level of the polynomial hierarchy. Δ_{i+1}^p is the class of problems solvable by a deterministic polynomial time Turing machine with a Δ_i^p oracle. Σ_{i+1}^p is the class of problems solvable by a non-deterministic polynomial time Turing machine with a Σ_i^p oracle. Π_{i+1}^p is the complement of Σ_{i+1}^p.

6.2.1 Modal logic

Some complexity results about modal logic will be used in this chapter. For the sake of self-containment, we here give a very short review of modal logic.

Definition 6.10 (modal language (Blackburn et al., 2001))**.** *Given a set $\mathbb{P}$ of propositional letters, the language of modal logic L_M is the smallest set such that:*

1. $\mathbb{P} \subseteq L_M$.
2. *if $x \in L_M$, then $\neg x \in L_M$.*
3. *if $x \in L_M$ and $y \in L_M$, then $x \wedge y \in L_M$.*
4. *if $x \in L_M$, then $\Box x \in L_M$.*

$\Diamond x$ is used as an abbreviation of $\neg\Box\neg x$. The modal depth of a modal formula is define as

- $md(p) = 0$, for all $p \in \mathbb{P}$,
- $md(\neg x) = md(x)$,
- $md(x \wedge y) = max\{md(x), md(y)\}$
- $md(\Box x) = md(x) + 1$

The semantics of modal logic is constructed using relational models.

Definition 6.11 (Relational model (Blackburn et al., 2001))**.** *A relational model $M = (W, R, V)$ is a tuple where:*

- *W is a (non-empty) set of possible worlds: $w, w', \ldots$*
- *$R \subseteq W \times W$ is a binary relation over W.*
- *$V : \mathbb{P} \mapsto 2^W$ is a valuation function for propositional letters such that $V(p) \subseteq W$.*

Definition 6.12 (Satisfaction (Blackburn et al., 2001))**.** *Given a relational model $M = (W, R, V)$ and a world $w \in W$, the satisfaction relation $M, w \vDash x$ (read as "world w satisfies x in model M") is defined by induction on the structure of x using the following clauses*

- *$M, w \vDash p$ iff $w \in V(p)$.*
- *$M, w \vDash \neg x$ iff $M, w \nvDash x$.*
- *$M, w \vDash x \wedge y$ iff $M, w \vDash x$ and $M, w \vDash y$.*
- *$M, w \vDash \Box x$ iff for all $w' \in W$, if $(w, w') \in R$ then $M, w' \vDash x$.*

A modal logic formula x is K-valid, denoted as $\vDash_K x$, if for all relational modal $M = (W, R, V)$ and all world $w \in W$, $M, w \vDash x$ holds. A modal logic formula x is T-valid, denoted as $\vDash_T x$, if for all relational modal $M = (W, R, V)$ where R is reflexive and all world $w \in W$, $M, w \vDash x$ holds. The K-validity problem of modal logic is: give a modal formula x, does $\vDash_K x$ hold? Similarly for the T-validity problem. The satisfiability problem is the complement of the validity problem. The following complexity results of modal logic will be used in this chapter:

Theorem 6.1. *(Halpern, 1995)*

1. *The K-satisfiability problem for formulas of model depth 1 is* NP*-complete.*
2. *The T-satisfiability problem for formulas of model depth 1 is* NP*-complete.*
3. *The K-validity problem for formulas of model depth 1 is* coNP*-complete.*
4. *The T-validity problem for formulas ofmodel depth 1 is* coNP*-complete.*

Given a set of modal formulas A, $A \vDash_K x$ holds if for all relational model $M = (W, R, V)$ and all worlds $w \in W$, if $M, w \vDash a$, for all $a \in A$, then $M, w \vDash x$. $A \vDash_T x$ holds if for all relational model $M = (W, R, V)$ where R is reflexive and all worlds $w \in W$, if $M, w \vDash a$, for all $a \in A$, then $M, w \vDash x$. When A is finite, it is proven that $A \vDash_K x$ iff $\vDash_K \bigwedge A \rightarrow x$ and $A \vDash_T x$ iff $\vDash_T \bigwedge A \rightarrow x$ (Blackburn et al., 2001).

6.3 Complexity of unconstrained input/output logic

The complexity of input/output logic was sparsely studied in the past. Although the reversibility of derivations rules as a proof re-writing mechanism was studied for input/output logic (Makinson and van der Torre, 2000), the length or complexity of such proofs was not developed. We approach the complexity of unconstrained input/output logic from a semantic point of view. We focus on the following *fulfillment problem*:

- Given a finite set of mandatory norms O, a finite set of formulas A and a formula x, is $x \in out(O, A)$?

6.3.1 Complexity of out_1 and out_1^+

We start with the complexity of out_1. The following theorem shows that out_1 has the same complexity as propositional logic in the sense that both the fulfillment problem of out_1 and the validity problem of propositional logic are coNP-complete.

Theorem 6.2. *The fulfillment problem of simple-minded input/output logic out_1 is coNP-complete.*

Proof. Concerning the coNP hardness, we prove by reducing the validity problem of propositional logic to the fulfillment problem of simple-minded input/output logic: given an arbitrary $x \in L_{\mathbb{P}}$, $\vdash x$ iff $x \in Cn(\top)$ iff $x \in Cn(O(Cn(A)))$ where $O = \emptyset$ iff $x \in out_1(O, A)$ where $O = \emptyset$.

Now we prove the coNP membership. We provide the following non-deterministic Turing machine to solve the complement of our problem. Let $O = \{(a_1, x_1), \ldots, (a_n, x_n)\}$, A be a finite set of formulas and x be a formula.

1. Guess a sequence of valuations $V_1, \ldots, V_n$ and V' on the propositional letters appearing in $A \cup \{a_1, \ldots, a_n\} \cup \{x_1, \ldots, x_n\} \cup \{x\}$. Every guess creates a branch in the computation tree of the non-deterministic Turing machine.
2. Let $O' \subseteq O$ be the set of norms which contain all (a_i, x_i) such that $V_i(A) = 1$ and $V_i(a_i) = 0$.
3. Let $X = \{x : (a, x) \in O - O'\}$.
4. If $V'(X) = 1$ and $V'(x) = 0$. Then return "accept" on this branch. Otherwise return "reject" on this branch.

It can be verified that $x \notin Cn(O(Cn(A)))$ iff the algorithm returns "accept" on some branches and the time complexity of the Turing machine is polynomial. The main intuition of the Turing machine is: O' collects all norms which *cannot* be triggered[†] by A. On some branches we must have

[†]We say a norm (a, x) is triggered by A if $a \in Cn(A)$).

that O' contains exactly all those norms which are not triggered by A. In those lucky branches X is the same as $O(Cn(A))$. If there is a valuation V' such that $V'(X) = 1$ and $V'(x) = 0$, then we know $x \notin Cn(X) = Cn(O(Cn(A)))$. □

To solve the complexity of out_1^+ we make use of the following lemma, which gives suggestions to a procedure to solve the fulfillment problem of out_1^+.

Lemma 6.1. $out_1^+(O, A) = Cn(A \cup O(Cn(A)))$.

Proof. Assume $x \in out_1^+(O, A) = out_1(O_{id}, A)$. Then $x \in Cn(O_{id}(Cn(A))) = Cn((O \cup \{(a, a) : a \in L_{\mathbb{P}}\})(Cn(A))) = Cn(O(Cn(A)) \cup (\{(a, a) : a \in L_{\mathbb{P}}\})(Cn(A))) = Cn(O(Cn(A)) \cup Cn(A))$. Therefore there are some $x_1, \ldots, x_n \in O(Cn(A)) \cup Cn(A)$ such that $x_1 \wedge \ldots \wedge x_n \vdash x$. Without loss of generality, assume $x_1, \ldots, x_{n-1} \in O(Cn(A))$ and $x_n \in Cn(A)$. Then there are some $y_1, \ldots, y_m$ such that $y_1 \wedge \ldots \wedge y_m \vdash x_n$. Then we know $x_1, \ldots, x_{n-1}, y_1, \ldots, y_m \in O(Cn(A)) \cup A$ and $x_1 \wedge \ldots, x_{n-1} \wedge y_1 \wedge \ldots \wedge y_m \vdash x$. Therefore $x \in Cn(A \cup O(Cn(A)))$.

Assume $x \in Cn(A \cup O(Cn(A)))$, then $x \in Cn(Cn(A) \cup O(Cn(A))) = Cn(O(Cn(A)) \cup (\{(a, a) : a \in L_{\mathbb{P}}\})(Cn(A))) = Cn(O_{id}(Cn(A))) = out_1^+(O, A)$. □

Theorem 6.3. *The fulfillment problem of simple-minded output throughput input/output logic out_1^+ is* coNP*-complete.*

Proof. Concerning the lower bound, we prove by a reduction from the validity problem of propositional logic: given arbitrary $x \in L_{\mathbb{P}}$, $\vdash x$ iff $x \in Cn(\top)$ iff $x \in Cn(A \cup O(Cn(A)))$ where $O = \emptyset = A$ iff $x \in out_1^+(O, A)$ where $O = \emptyset = A$.

Concerning the upper bound, we prove by giving a non-deterministic Turing machine similar to the one in the proof of Theorem 6.2. The only change is now in step 4 we test if $V'(A \cup X) = 1$ and $V'(x) = 0$. It can be verified that $x \notin Cn(A \cup O(Cn(A)))$ iff the non-deterministic Turing machine returns "accept" on some branch. By Lemma 6.1 we know this Turing machine solves our problem. □

6.3.2 Complexity of out_2 and out_2^+

Makinson and van der Torre (2000) introduce $s(O) = \{x : (a, x) \in O\}$ as the projection of O to the second component of its consisting norms and $O^{\Box} = \{a \rightarrow \Box x : (a, x) \in O\}$. Here $\Box$ is the necessity modality of modal logic. The following theorem reveals the relation between basic input/output logic and modal logic, and is useful in the study of complexity.

Theorem 6.4. *(Makinson and van der Torre, 2000)*

$$x \in out_2(O, A) \text{ iff } x \in Cn(s(O)) \text{ and } O^{\Box} \cup A \vDash_K \Box x.$$

Theorem 6.5. *The fulfillment problem of basic input/output logic out_2 is* coNP*-complete.*

Proof. Concerning the lower bound, we prove by a reduction from the validity problem of propositional logic: given arbitrary $x \in L_{\mathbb{P}}$, $\vdash x$ iff $x \in Cn(\emptyset)$ iff $x \in out_2(O, A)$ where $O = \emptyset$.

Concerning the upper bound, we prove by polynomially reducing the fulfillment problem of basic input/output logic to the validity problem of modal logic K with modal depth 1. Theorem 6.4 gives us the key idea of the reduction. By Theorem 6.4, $x \in out_2(O, A)$ iff $\vDash_K (\bigwedge s(O) \rightarrow x) \wedge ((\bigwedge O^{\Box} \wedge A) \rightarrow \Box x)$. Then by Corollary 6.1 we know the upper bound is coNP. □

Now we study the complexity of out_2^+ and out_4^+. We make use of the materialisation of norms introduced by Makinson and van der Torre (2000). Let $m(O) = \{a \rightarrow x : (a, x) \in O\}$ be materialisation of O. That is, the operator $m(\bullet)$ transforms a norms (a, x) into an classical implication $a \rightarrow x$. The following theorem shows that out_2^+ and out_4^+ can be reduced to propositional logic via the materialisation of norms.

Theorem 6.6. *(Makinson and van der Torre, 2000)* $out_2^+(O, A) = out_4^+(O, A) = Cn(A \cup m(O))$.

Theorem 6.7.

1. *The fulfillment problem of basic throughput input/output logic out_2^+ is* coNP*-complete.*
2. *The fulfillment problem of basic reusable throughput input/output logic out_4^+ is* coNP*-complete.*

Proof. Given Theorem 6.6, the proof is routine: $x \in out_2^+(O, A)$ iff $x \in Cn(A \cup m(O))$ iff $A \cup m(O) \vdash x$ iff $\vdash \bigwedge(A \cup m(O)) \rightarrow x$. □

6.3.3 Complexity of out_3 and out_3^+

Theorem 6.8. *The fulfillment problem of simple-minded reusable input/output logic out_3 is between* coNP *and* P^{NP}.

Proof. The lower bound can be proved using the same reduction as in the the proof of the lower bound of out_1. Concerning the upper bound, we provide the following algorithm on a oracle Turing machine with an NP oracle.

Let $O = \{(a_1, x_1), \ldots, (a_n, x_n)\}$, A be a finite set of formulas and x be a formula.

1. Let $X := A, Y := Z := O, U := \emptyset$.
2. for each $(a_i, x_i) \in Y$, ask the oracle if $X \vdash a_i$ holds.

(a) If "yes", then let $X := X \cup \{x_i\}$, $Z := Z - \{(a_i, x_i)\}$.

(b) Otherwise do nothing.

3. If $Y = Z$, then goto 4. Otherwise let $Y := Z$, goto step 2.

4. for each $(a_i, x_i) \in O$, ask the oracle if $X \vdash a_i$ holds.

   (a) If "yes", then let $U := U \cup \{x_i\}$.

   (b) Otherwise do nothing

5. Ask the oracle if $U \vdash x$ holds.

   (a) If "yes", then return "accept".

   (b) Otherwise return "reject".

The intuition of this Turing machine is: we start with input X equals to A, and then we add to X gradually all the consequences of the norms that are triggered. Y and Z collects the norms that at step i have not been processed yet and we use whether $Y = Z$ to test if we have triggered all those norms which can possibly be triggered. At step 4, we have expanded the input X such that it contains both facts A and all consequences of those norms which can be triggered. Then we use U to collect all the consequences of those norms which is triggered by X. Finally we test if U implies x.

The inductive characterization presented in Section 4.3.1 is useful to prove the correctness of the above algorithm. Here we just state some crucial points: from step 1 to step 3, the algorithm generates X as B^O_A. At step 4, X contains both A and all consequences of those norms which can be triggered by B^O_A. Then we generate U, which is understood as $O(X)$.

Concerning the time complexity, the times of the for-loop in step 2 is at most n. Each loop can be finished in polynomial time. Therefore all the loops in step 2 can be done in polynomial time. Step 3 calls for step 2 for at most n times. Therefore it can still be done in polynomial time. The times of loop in step 4 is exactly n. Each loop can be finished in polynomial time. Therefore all the loops in step 4 can be done in polynomial time. Step 5 can be done in polynomial time. Therefore the algorithm is polynomial. □

We use the following examples to illustrate how the above algorithm works.

Example 6.1. *Let p, q, r, s and t be propositional letters. Let $O = \{(p, q), (q, r), (p \wedge r, s), (t, t)\}$, $A = \{p\}$, $t \to s \in out_3(O, A)$ is computed as follows:*

1. *Let $X = A = \{p\}$, $Y = Z = O = \{(p, q), (q, r), (p \wedge r, s), (t, t)\}$, $U = \emptyset$.*

2. *Compute whether* $\{p\} \vdash p$, $\{p\} \vdash q$, $\{p\} \vdash p \wedge r$, $\{p\} \vdash t$.

3. *Let* $X = \{p, q\}$, $Z = \{(q, r), (p \wedge r, s), (t, t)\}$.

4. *Compare if* $Y = Z$. *The result is negative. So let* $Y = \{(q, r), (p \wedge r, s), (t, t)\}$

5. *Compute whether* $\{p, q\} \vdash q$, $\{p, q\} \vdash p \wedge r$, $\{p\} \vdash t$.

6. *Let* $X = \{p, q, r\}$. $Z = \{(p \wedge r, s), (t, t)\}$.

7. *Compare if* $Y = Z$. *The result is negative. So let* $Y = \{(p \wedge r, s), (t, t)\}$.

8. *Compute whether* $\{p, q, r\} \vdash p \wedge r$, $\{p\} \vdash t$.

9. *Let* $X = \{p, q, r, s\}$. $Z = \{(t, t)\}$.

10. *Compare if* $Y = Z$. *The result is negative. So let* $Y = \{(t, t)\}$.

11. *Compute whether* $\{p, q, r, x\} \vdash t$.

12. *Let* $X = \{p, q, r, s\}$. $Z = \{(t, t)\}$.

13. *Compare if* $Y = Z$. *The result is positive.*

14. *Compute whether* $\{p, q, r, s\} \vdash p$, $\{p, q, r, s\} \vdash q$, $\{p, q, r, s\} \vdash p \wedge r$, $\{p, q, r, s\} \vdash t$.

15. *Let* $U = \{q, r, s\}$.

16. *Compute whether* $\{q, r, s\} \vdash t \rightarrow s$. *The answer is positive, so we conclude* $t \rightarrow s \in out_3(O, A)$.

Theorem 6.9. *The fulfillment problem of simple-minded reusable throughput input/output logic* out_3^+ *is between* coNP *and* $\mathtt{P}^{\mathtt{NP}}$.

Proof. The lower bound can be proved using the same reduction as in the the proof of the lower bound of out_1. Concerning the upper bound, we prove by giving an algorithm similar to the one in the proof of Theorem 6.8. We make the following change:

- In step 2 and 4 we ask the oracle if $A \cup X \vdash a_i$ holds.
- In step 5 we ask the oracle if $A \cup U \vdash x$ is holds.

□

Example 6.2. *Let* p, q, r *and* s *be propositional letters. Let* $O = \{(p, q \wedge r), (p \wedge r, s)\}$, $A = \{p\}$, $\neg p \wedge s \in out_3^+(O, A)$ *is computed as follows:*

1. *Let* $X = A = \{p\}$, $Y = Z = O = \{(p, q \wedge r), (p \wedge r, s)\}$, $U = \emptyset$.

2. *Compute whether* $\{p\} \vdash p$, $\{p\} \vdash p \wedge r$.

3. *Let* $X = \{p, q \wedge r\}$, $Z = \{(p \wedge r, s)\}$.

4. *Compare if* $Y = Z$. *The result is negative. So let* $Y = \{(p \wedge r, s)\}$.

5. *Compute whether* $\{p, q \wedge r\} \vdash p \wedge r$.

6. *Let* $X = \{p, q \wedge r, s\}$. $Z = \emptyset$.

7. *Compare if* $Y = Z$. *The result is negative. So let* $Y = \emptyset$.

8. *Compute whether* $\{p, q \wedge r, s\} \vdash p$, $\{p, q \wedge r, s\} \vdash p \wedge r$

9. *Let* $U = \{r, s\}$.

10. *Compute whether* $A \cup U = \{p, r, s\} \vdash \neg p \wedge s$. *The answer is negative, so we conclude* $\neg p \wedge s \notin out_3^+(O, A)$.

6.3.4 Complexity of out_4

Similar to out_2, out_4 can also be translated to modal logic.

Theorem 6.10. *(Makinson and van der Torre, 2000)* $x \in out_4(O, A)$ *iff* $x \in Cn(s(O))$ *and* $O^{\Box} \cup A \vDash_T \Box x$.

Theorem 6.11. *The fulfillment problem of basic reusable input/output logic* out_4 *is* coNP*-complete.*

Proof. The lower bound can be proved using the same reduction as in the the proof of the lower bound of out_1. Concerning the upper bound, we prove by polynomially reducing the fulfillment problem of basic reusable input/output logic to the validity problem of modal logic T with modal depth 1. Theorem 6.10 gives us the key idea of the reduction. By Theorem 6.10, $x \in out_4(O, A)$ iff $\vDash_T (\bigwedge s(O) \rightarrow x) \wedge ((\bigwedge O^{\Box} \wedge A) \rightarrow \Box x)$. Then by Theorem 6.1 we know the upper bound is coNP. □

6.4 Complexity of constrained input/output logic

In the constrained setting, a finite set of mandatory norms O and a subset $O' \subseteq O$, a finite set of input A and a finite set of constrains C are given. We study the complexity of the following problems:

- consistency checking: is $out_i(O, A)$ consistent with C?

- maxfamily membership: is $O' \in maxfamily_i(O, A, C)$?
- full-join fulfillment: is $x \in out_i^{\cup}(O, A, C)$?
- full-meet fulfillment: is $x \in out_i^{\cap}(O, A, C)$?

Theorem 6.12.

- *For $i \in \{1, 2, 4, 1^+, 2^+, 4^+\}$, the consistency checking problem is* NP*-complete.*
- *For $i \in \{3, 3^+\}$, the consistency checking problem is* NP*-hard and in* P^{NP}.

Proof. The NP-hard problem SAT can be reduced to the consistency checking problem, which gives us the result of the lower bound. The consistency checking problem can be solved by a reduction to the complement of the fulfillment problem, which gives us the result of the upper bound: $out_i(O, A)$ is consistent with C iff $C \cup out_i(O, A)$ is satisfiable iff $out_i(O, A) \not\models \neg \bigwedge C$ iff $\neg \bigwedge C \notin out_i(O, A)$. □

We show now that the maxfamily membership problem is BH_2-complete, where BH_2 is the class of languages which are the intersection of a language in NP and a language in coNP (cf. section 6.2 above).

Theorem 6.13. *For $i \in \{1, 2, 4, 1^+, 2^+, 4^+\}$, the maxfamily membership problem is BH_2-complete.*

Proof. The BH_2 hardness can be proved by a reduction from the 2-Parity SAT problem. Given two propositional formulas x_1 and x_2 such that if x_2 is satisfiable then x_1 is satisfiable. Our aim is to decide if x_1 is satisfiable meanwhile x_2 is not satisfiable.

Let $O = \{(\top, x_1 \vee x_2), (\top, x_2)\}$, $A = C = \emptyset$, $O' = \{(\top, x_1 \vee x_2)\}$.

- If $O' \in maxfamily_i(O, A, C)$ then $out_i(O', A) = Cn(x_1 \vee x_2)$ is consistent and $out_i(O, A) = Cn((x_1 \vee x_2) \wedge x_2)$ is inconsistent. Therefore $x_1 \vee x_2$ is satisfiable and $(x_1 \vee x_2) \wedge x_2$ is not satisfiable. Then we know x_2 is not satisfiable. It then follows that x_1 is satisfiable because otherwise $x_1 \vee x_2$ is not satisfiable.
- If x_1 is satisfiable and x_2 is not satisfiable, then $x_1 \vee x_2$ is satisfiable. Therefore $out_i(O', A)$ is consistent and $out_i(O, A)$ is inconsistent. It then follows that $O' \in maxfamily_i(O, A, C)$.

So we have proved x_1 is satisfiable and x_2 is not satisfiable iff $O' \in maxfamily_i(O, A, C)$, which proves the BH_2 hardness.

Concerning the BH_2 membership, let $O = O' \cup \{(a_1, x_1), \ldots, (a_n, x_n)\}$. Then $O' \in maxfamily_i(O, A, C)$ iff C is consistent with $out_i(O', A)$ but not consistent with $out_i(O' \cup \{(a_j, x_j)\}, A)$ for every

$j \in \{1, \ldots, n\}$. Since deciding if C is consistent with $out_i(O', A)$ is in NP and deciding if C is not consistent with $out_i(O' \cup \{(a_j, x_j)\}, A)$ is in coNP, we know that deciding if $O' \in maxfamily_i(O, A, C)$ is in BH_2. □

Theorem 6.14. *For $i \in \{3, 3^+\}$, the maxfamily membership problem is BH_2-hard and in $\mathtt{P}^{\mathtt{NP}}$.*

Proof. The BH_2 hardness can be proved just like other input/output logics.

Concerning the upper bound, let $O = O' \cup \{(a_1, x_1), \ldots, (a_n, x_n)\}$. Then $O' \in maxfamily(O, A, C)$ iff C is consistent with $out_i(O', A)$ but not consistent with $out_i(O' \cup \{(a_j, x_j)\}, A)$ for every $j \in \{1, \ldots, n\}$. Since deciding if C is consistent with $out_i(O', A)$ is in $\mathtt{P}^{\mathtt{NP}}$ and deciding if C is not consistent with $out_i(O' \cup \{(a_j, x_j)\}, A)$ is also in $\mathtt{P}^{\mathtt{NP}}$, we know that deciding if $O' \in maxfamily_i(O, A, C)$ is in $\mathtt{P}^{\mathtt{NP}}$. □

Theorem 6.15. *For $i \in \{1, 2, 3, 4, 1^+, 2^+, 3^+, 4^+\}$, the full-join fulfillment problem is $\mathtt{NP}^{\mathtt{NP}}$-complete.*

Proof. Concerning the $\mathtt{NP}^{\mathtt{NP}}$ membership, we prove by giving the following algorithm on a non-deterministic Turing machine with an $\mathtt{NP}$ oracle to solve our problem.

1. Guess a subset $O' \subseteq O$.
2. Use the $\mathtt{NP}$ oracle to test if $O' \in maxfamily_i(O, A, C)$. If no, return "reject" on this branch. Otherwise continue.
3. Use the $\mathtt{NP}$ oracle to test if $x \in out_i(O', A)$. If $x \in out_i(O', A)$, then return "accept" on this branch. Otherwise return "reject" on this branch.

It can be verified that $x \in out_i^{\cup}(O, A, C)$ iff the non-deterministic Turing machine return "accept" on some branches. Step 2 can be done in polynomial time steps because the maxfamily membership problem is in $\mathtt{P}^{\mathtt{NP}}$. Step 3 can also be done in polynomial time steps because the fulfillment problem is also in $\mathtt{P}^{\mathtt{NP}}$. Therefore the time complexity of this non- deterministic Turing machine is polynomial.

Concerning the $\mathtt{NP}^{\mathtt{NP}}$ hardness, we show that the validity problem of 2-QBF$^{\exists}$ can be reduced to the full-join fulfillment problem.

Let $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ be a 2-QBF$^{\exists}$ where Φ is a propositional formula with variables in $\{p_1, \ldots, p_m, q_1, \ldots, q_n\}$. Let $A = C = \emptyset$, $O = \{(\top, p_1), \ldots, (\top, p_m), (\top, \neg p_1), \ldots, (\top, \neg p_m), (\top, \neg\Phi)\}$. Our aim is to show that this 2-QBF$^{\exists}$ is valid iff $\Phi \in out_i^{\cup}(O, A, C)$.

- If $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is valid, then there is a valuation V for $\{p_1, \ldots, p_m\}$ such that for all valuations V' for $\{q_1, \ldots, q_n\}$, $V \cup V'$ gives truth value 1 to Φ and 0 to $\neg\Phi$. Let $O' = \{(\top, p_1'), \ldots, (\top, p_m')\}$, where each p_i' is p_i if $p_i \in V$ and it is $\neg p_i$ if $p_i \notin V$. Then $O' \in$

$maxfamily_i(O, A, C)$ because $out_i(O', A) = Cn(\{p'_1, \ldots, p'_m\})$ is consistent with C and adding anything from $\{(\top, \neg p_1), \ldots, (\top, \neg p_m), (\top, \neg\Phi)\}$ to O' will destroy the consistency. Note that $\Phi \in Cn(\{p'_1, \ldots, p'_m\})$ by the construction of $\{p'_1, \ldots, p'_m\}$. Therefore $\Phi \in out_i(O', A)$, which further implies that $\Phi \in out_i^{\cup}(O, A, C)$.

- If $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is not valid, then for all valuation V for $\{p_1, \ldots, p_m\}$ there is a valuations V' for $\{q_1, \ldots, q_n\}$ such that $V \cup V'$ gives truth value 0 to Φ and 1 to $\neg\Phi$. Let $O' = \{(\top, p'_1), \ldots, (\top, p'_m), (\top, \neg\Phi)\}$ be an arbitrary set such that each p'_i is either p_i or $\neg p_i$. Then $out_i(O', A) = Cn(\{p'_1 \ldots, p'_m, \neg\Phi\})$, which is consistent. Moreover it can be verified that $O' \in maxfamily_i(O, A, C)$. Therefore $\neg\Phi \in out_i(O', A)$ and $\Phi \notin out_i(O', A)$. By the construction we can further verified that O' ranges over all elements of $maxfamily_i(O, A, C)$. Then we conclude $\Phi \notin out_i^{\cup}(O, A, C)$.

So we have reduced the validity problem of 2-QBF$^{\exists}$ to the full-join fulfillment problem, which shows the latter is $\mathtt{NP}^{\mathtt{NP}}$-hard. □

In the setting of normative/legal reasoning, the full-meet and full-join fulfillment problems are the two most important decision problems of input/output logic.

Theorem 6.16. *For $i \in \{1, 2, 3, 4, 1^+, 2^+, 3^+, 4^+\}$, the full-meet fulfillment problem is $\mathtt{coNP}^{\mathtt{NP}}$-complete.*

Proof. Concerning the $\mathtt{coNP}^{\mathtt{NP}}$ membership, we prove by giving the following algorithm on a non-deterministic Turing machine with an $\mathtt{NP}$ oracle to solve the complement of our problem.

1. Guess a subset $O' \subseteq O$.
2. Use the $\mathtt{NP}$ oracle to test if $O' \in maxfamily_i(O, A, C)$. If no, return "reject" on this branch. Otherwise continue.
3. Use the $\mathtt{NP}$ oracle to test if $x \notin out_i(O', A)$. If $x \notin out_i(O', A)$, then return "accept" on this branch. Otherwise return "reject" on this branch.

It can be verified that $x \notin out^{\cap}(O, A, C)$ iff the non-deterministic Turing machine returns "accept" on some branches. Step 2 can be done in polynomial time steps because the maxfamily membership problem is in $\mathrm{P}^{\mathtt{NP}}$. Step 3 can also be done in polynomial time steps because the fulfillment problem is also in $\mathrm{P}^{\mathtt{NP}}$. Therefore the time complexity of this non-deterministic Turing machine is polynomial.

Concerning the $\mathtt{coNP}^{\mathtt{NP}}$ hardness, we show that the validity problem of 2-QBF$^{\forall}$ can be reduced to the full-meet fulfillment problem.

Let $\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ be a 2-QBF$^{\forall}$ where Φ is a propositional formula with variables in $\{p_1, \dots, p_m, q_1, \dots, q_n\}$. Let $A = C = \emptyset$, $O = \{(\top, p_1), \dots, (\top, p_m), (\top, \neg p_1), \dots, (\top, \neg p_m), (\top, \Phi)\}$. Our aim is to show that this 2-QBF$^{\forall}$ is valid iff $\Phi \in out_i^{\cap}(O, A, C)$.

- If $\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ is valid, then for all valuation V for $\{p_1, \dots, p_m\}$ there is a valuation V' for $\{q_1, \dots, q_n\}$ such that $V \cup V'$ gives truth value 1 to Φ and 0 to $\neg\Phi$.

  Let $O' = \{(\top, p_1'), \dots, (\top, p_m'), (\top, \Phi)\}$ be an arbitrary set such that each p_i' is either p_i or $\neg p_i$. Then $out_i(O', A) = Cn(\{p_1' \dots, p_m', \Phi\})$, which is consistent. Moreover it can be verified that $O' \in maxfamily_i(O, A, C)$. Therefore $\Phi \in out_i(O', A)$. By the construction we can further verify that O' range over all elements of $maxfamily_i(O, A, C)$. Then we conclude $\Phi \in out_i^{\cap}(O, A, C)$.

- If $\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ is not valid, then there is a valuation V for $\{p_1, \dots, p_m\}$ such that for all valuations V' for $\{q_1, \dots, q_n\}$, $V \cup V'$ gives truth value 0 to Φ and 1 to $\neg\Phi$.

  Let $O' = \{(\top, p_1'), \dots, (\top, p_m')\}$, where each p_i' is p_i if $p_i \in V$ and it is $\neg p_i$ if $p_i \notin V$. Then $O' \in maxfamily_i(O, A, C)$ because $out_i(O', A) = Cn(\{p_1', \dots, p_m'\})$ is consistent with C and adding anything from $\{(\top, \neg p_1), \dots, (\top, \neg p_m), (\top, \Phi)\}$ to O' will destroy the consistency. Note that $\neg\Phi \in Cn(\{p_1', \dots, p_m'\})$ by the construction of $\{p_1', \dots, p_m'\}$. Therefore $\Phi \notin out_i(O', A)$, which further implies that $\Phi \notin out_i^{\cap}(O, A, C)$.

So, we have reduced the validity problem of 2-QBF$^{\forall}$ to the full-meet fulfillment problem, which shows the latter is $\mathtt{coNP}^{\mathtt{NP}}$-hard. □

6.5 Complexity of permissive input/output logic

In this section we study the complexity of the following decision problems about permissive input/output logic: given a finite normative system $N = (O, P)$, a finite set of input A and a formula x:

- negative permission checking: is $x \in NegPerm_i(N, A)$?
- positive-static permission checking: is $x \in StaPerm_i(N, A)$?
- positive-dynamic permission checking: is $x \in DyPerm_i(N, A)$?

Negative permission checking is relatively easy because it is simply the complement of the fulfillment problem.

Theorem 6.17.

1. *For $i \in \{1,2,4\}$, the negative permission checking is* NP*-complete.*

2. *For $i = 3$, negative permission checking is* NP*-hard and in* $\mathtt{P}^{\mathtt{NP}}$.

Proof. The negative permission checking is complement to the fulfillment problem. That is, $x \in NegPerm_i(N, A)$ iff $\neg x \notin out_i(O, A)$. Therefore the complexity of the negative permission checking problem belongs to the complement complexity class of the fulfillment problem. □

Positive-static permission checking is no harder than the fulfillment problem because both the class coNP and $\mathtt{P}^{\mathtt{NP}}$ are closed under finite union.

Theorem 6.18.

1. *For $i \in \{1,2,4\}$, the positive-static permission checking is* coNP*-complete.*

2. *For $i = 3$, the positive-static permission checking is* coNP*-hard and in* $\mathtt{P}^{\mathtt{NP}}$.

Proof. 1. Let $P = \{(a_1, x_1), \ldots, (a_n, x_n)\}$. Then $x \in StaPerm_i(N, A)$ iff $x \in out_i(O \cup \{(a_1, x_1)\}, A) \cup \ldots \cup out_i(O \cup \{(a_n, x_n)\}, A)$ iff $x \in out_i(O \cup \{(a_1, x_1)\}, A)$ or $x \in out_i(O \cup \{(a_2, x_2)\}, A)$ or ... or $x \in out_i(O \cup \{(a_n, x_n)\}, A)$. Since the coNP class is closed under finite union, we know that the positive-static permission checking problem is in coNP. The coNP hardness can be proved by setting $P = \emptyset$ and reduce the fulfillment problem to the static permission checking problem.

2. Similar to the above item. The $\mathtt{P}^{\mathtt{NP}}$ membership follows from the fact that the $\mathtt{P}^{\mathtt{NP}}$ class is closed under finite union.

□

Positive-dynamic permission checking is harder than other permission checking as the following theorem shows. The main source of complexity is that in positive-dynamic permission checking we have to first guess a consistent input and then check if it produced some inconsistency.

Theorem 6.19. *For $i \in \{1,2,3,4\}$, the positive-dynamic permission checking is* $\mathtt{NP}^{\mathtt{NP}}$*-complete.*

Proof. The $\mathtt{NP}^{\mathtt{NP}}$ membership follows by guessing a consistent set of formulas $C \subseteq f(O) \cup f(P) \cup A$, where $f(O) = \{a : (a, x) \in O\}$ and $f(P) = \{b : (b, y) \in P\}$, then using an NP oracle to check if $\bot \in StaPerm_i(N, C) \cup out_i(O \cup \{(\bigwedge A, \neg x)\}, C)$.

Concerning the $\mathtt{NP}^{\mathtt{NP}}$ hardness, we show that the validity problem of 2-$\mathtt{QBF}^{\exists}$ can be reduced to the positive-dynamic permission checking problem.

Let $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ be a 2-QBF$^{\exists}$ where Φ is a propositional formula contains variables only in $\{p_1, \ldots p_m, q_1, \ldots, q_n\}$.

Let $N = (O, P)$ where $O = \{(p_1, p_1), \ldots, (p_m, p_m), (\neg p_1, \neg p_1), \ldots, (\neg p_m, \neg p_m), (\Phi, \bot), (\neg\Phi, \top)\}$, $P = \emptyset$, $A = \emptyset$, $x = \bot$. Our aim is to prove that $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is valid iff $x \in DyPerm_i(N, A)$.

Note that $x \in DyPerm_i(N, A)$ iff there is a consistent set C such that $StaPerm_i(N, C) \cup out_i(O \cup \{(\bigwedge A, \neg x)\}, C)$ is inconsistent, which means there is a consistent set C such that $out_i(O, C) \cup out_i(O \cup \{(\top, \neg\bot)\}, C)$ is inconsistent. This is equivalent to say that there is a consistent set C such that $out_i(O, C)$ is inconsistent. Moreover, the following are equivalent:

- There is a consistent set C such that $out_i(O, C)$ is inconsistent.
- There is a consistent C which is a subset of $f(O)$ such that $out_i(O, C)$ is inconsistent.
- There is a set C which is a maximal consistent subset of $f(O)$ such that $out_i(O, C)$ is inconsistent.

Therefore $x \in DyPerm_i(N, A)$ iff there is a set C which is a maximal consistent subset of $f(O)$ such that $out_i(O, C)$ is inconsistent. We now show that $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is valid iff there is a set C which is a maximal consistent subset of $f(O)$ such that $out_i(O, C)$ is inconsistent.

- If $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is valid, then there is a valuation V for $\{p_1, \ldots, p_m\}$ such that for all valuations V' for $\{q_1, \ldots, q_n\}$, $V \cup V'$ gives truth value 1 to Φ.

  Let $C = \{p'_1, \ldots, p'_m, \Phi\}$, where each p'_i is p_i if $p_i \in V$ and it is $\neg p_i$ if $p_i \notin V$. Then C is a maximal consistent subset of $f(O)$. Moreover $\bot \in out_i(O, C)$ because $\Phi \in C$ and $(\Phi, \bot) \in O$.

- If $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$ is not valid, then for all valuation V for $\{p_1, \ldots, p_m\}$ there is a valuations V' for $\{q_1, \ldots, q_n\}$ such that $V \cup V'$ gives truth value 0 to Φ and 1 to $\neg\phi$. Let $C = \{p'_1, \ldots, p'_m, \neg\Phi\}$, where each p'_i is p_i if $p_i \in V$ and it is $\neg p_i$ if $p_i \notin V$. Then C is a maximal consistent subset of $f(O)$. Therefore $out_i(O, C) = Cn(\{p'_1, \ldots, p'_m, \top\})$ which is consistent.

So we have reduced the validity problem of 2-QBF$^{\exists}$ to the dynamic permission checking problem, which shows the latter is $\mathtt{NP}^{\mathtt{NP}}$-hard. □

6.6 Tractable fragments of input/output logic

Results from Section 6.3 show that all unconstrained input/output logic are intractable. In this section our task is to impose syntactic restrictions such that the decision problem of unconstrained input/output logic is tractable (in $\mathtt{P}$). For out_1 and out_3, we achieve tractability via Post Lattice. For out_2 and out_4, Post Lattice seems not helpful in identifying tractable fragments. Alternatively, we

conjecture that the tractable fragments of out_2 and out_4 can be achieved via modal Horn formula, but we left it for future work. Tractable fragments of constrained/prioritized input/output logic will be studied in Chapter 8.

SAT is the most fundamental and historically the first NP-complete problem. A natural question, posed by Lewis (1979) is what is the sources of hardness of SAT. More precisely, Lewis systematically restricted the language of propositional formula and shows that the complexity of the SAT depending on the set of allowed boolean connectives. Lewis (1979) proved that SAT is NP-complete iff the negation of implication, $x \wedge \neg y$ can be simulated by the allowed connectives. To simulate a logical connective f by a set of logical connectives $\mathcal{B}$ means that f can be obtained from functions from $\mathcal{B}$ by composition. We can express this fact by saying that f is in the clone generated by $\mathcal{B}$, in symbols $f \in [\mathcal{B}]$. For a clone $\mathcal{B}$, every set $\mathcal{B}' \subseteq \mathcal{B}$ with $[\mathcal{B}'] = \mathcal{B}$ is called a basis (or base) of $\mathcal{B}$.

Formally, an *n-ary* Boolean function is a function from $\{0,1\}^n$ to $\{0,1\}$. *id* is a special Boolean function denotes the unary identity, $id(x) = x$. Let $\mathcal{B}$ be a set of Boolean functions. $[\mathcal{B}]$ is the smallest set which extends $\mathcal{B} \cup \{id\}$ and is closed under the following rule: If $f(x_1, \ldots, x_n) \in [\mathcal{B}]$ and $X_1, \ldots, X_n$ are either Boolean variables or elements from $[\mathcal{B}]$, then $f(X_1, \ldots, X_n) \in [\mathcal{B}]$. We say that a class of Boolean functions $[\mathcal{B}]$ is *closed* if $\mathcal{B} = [\mathcal{B}]$. Closed classes are also referred to as *clones*.

This brings us into the realm of Post ' s lattice, the lattice of all Boolean clones (Post, 1941). Post identified all clones of Boolean functions and found a finite basis for each of them. He also discovered the inclusion structure of the classes. We refer to some basic Boolean functions with notations listed below:

- 0-*ary* Boolean functions: $c_0 =_{def} 0$ and $c_1 =_{def} 1$.
- 1-*ary* Boolean functions: $not(x) = 1$ iff $x = 0$. (In formulas we use $\neg x$ or $\overline{x}$ for $not(x)$.)
- some 2-*ary* Boolean functions: $and(x,y) = 1$ iff $x = y = 1$, $or(x,y) = 0$ iff $x = y = 0$, $xor(x,y) = 1$ iff $x \neq y$, $eq(x,y) = 1$ iff $x = y$, $imp(x,y) = 0$ iff $x = 1$ and $y = 0$. (In formulas we use $\oplus$ for xor.)

We briefly introduce some Boolean clones, a more detailed introduction to Boolean clones can be found in Böhler et al. (2003):

- BF is the class of all Boolean functions.
- For $a \in \{0,1\}$, a Boolean function f is called a-reproducing if $f(a, \ldots, a) = a$. The clones R_a contain all a-reproducing Boolean functions.

- For $(a_1,\ldots,a_n),(b_1,\ldots,b_n) \in \{0,1\}^n$, we say $(a_1,\ldots,a_n) \leq (b_1,\ldots,b_n)$ if $a_i \leq b_i$ for $1 \leq i \leq n$. An n-ary Boolean function f is called monotonic if for all $(a_1,\ldots,a_n),(b_1,\ldots,b_n) \in \{0,1\}^n$ it holds that : if $(a_1,\ldots,a_n) \leq (b_1,\ldots,b_n)$ then $f(a_1,\ldots,a_n) \leq f(b_1,\ldots,b_n)$. The class of all monotonic Boolean functions is denoted by M.

- A Boolean function f is called self-dual if for all $a_1,\ldots,a_n \in \{0,1\}$ we have $f(a_1,\ldots,a_n) = \neg f(\overline{a_1},\ldots,\overline{a_n})$. The class of all self-dual Boolean functions is called D.

- An n-ary Boolean function f is linear if there exist constants $e_0,\ldots,e_n \in \{0,1\}$ such that $f(x_1,\ldots,x_n) = e_0 \oplus e_1x_1 \oplus \ldots \oplus e_nx_n$. The class of all linear Boolean functions is called L.

- Let $T \subseteq \{0,1\}^n$ and $a \in \{0,1\}$. We call T a-separating if there exists an $i \in \{1,\ldots,n\}$ such that for all $(b_1,\ldots,b_n) \in T$ it holds that $b_i = a$. A Boolean function f is called a-separating if $f^{-1}(a)$ is a-separating. The function f is called a-separating of level k if every $T \subseteq f^{-1}(a)$ with $\mid T \mid = k$ is a-separating. The classes of all a-separating functions are called S_a and the classes of a-separating functions of level k are called S_a^k.

- The class E is the class of all Boolean functions that can be described by formulas build over $\wedge$, 0 and 1: $E = \{f \in \text{BF} : F(x_1,\ldots,x_n) = c_0 \wedge (c_1 \vee x_1) \wedge \ldots \wedge (c_n \vee x_n)$ for some constants $c_i, 0 \leq i \leq n\}$. Analogously, V is the class of Boolean functions that can be described by formulas build over $\vee$, 0 and 1.

- The class I_2 contains all projections (i.e., all Boolean functions I_k^n with $I_k^n(a_1,\ldots,a_n) = a_k$ for all $a_1,\ldots,a_n \in \{0,1\}$ and I contains all projections and additionally all constants (i.e., all Boolean functions c_a^n, $a \in \{0,1\}$, with $c_a^n(a_1,\ldots,a_n) = a$ for all $a_1,\ldots,a_n \in \{0,1\}$. N_2 contains all projections and all negations of projections. The class N contains N-2 and all constants.

For a finite set $\mathcal{B}$ of Boolean functions, Beyersdorff et al. (2009) define the Implication Problem for $\mathcal{B}$-formula IMP($\mathcal{B}$) as the following computational task: Given a finite set A of $\mathcal{B}$-formulas and a $\mathcal{B}$-formula x, decide whether $A \vdash x$ holds. The complexity of the implication problem is classified in Beyersdorff et al. (2009). The results relevant to this chapter are summarized in the following theorem.

Theorem 6.20 (Beyersdorff et al. (2009))**.** *Let $\mathcal{B}$ be a finite set of Boolean functions. Then IMP($\mathcal{B}$) is*

- coNP*-complete if $S_{00} \subseteq [\mathcal{B}]$, $S_{10} \subseteq [\mathcal{B}]$ or $D_2 \subseteq [\mathcal{B}]$ and*

- *in* P *for all other cases.*

Since the source of complexity for $out_1, out_3, out_1^+, out_3^+$ is the consequence relation $\vdash$ of propositional logic, we immediately have the following tractability result:

Corollary 6.1. *Let $\mathcal{B}$ be a finite set of Boolean functions such that $S_{00} \not\subseteq [\mathcal{B}]$, $S_{10} \not\subseteq \mathcal{B}$ and $D_2 \not\subseteq \mathcal{B}$. Let A be a set of $\mathcal{B}$-formulas, all formulas appear in O are $\mathcal{B}$-formulas and x a $\mathcal{B}$-formula. Deciding if $x \in out_i(O, A)$ is in* P*, for $i \in \{1, 3, 1^+, 3^+\}$.*

Proof. We only proof for $i = 1, 3$.

1. For the case of out_1. Given finite O, A and x, let $B = O(Cn(A))$. The set B can be constructed in polynomial time because now IMP($\mathcal{B}$) is in P. We then test if $x \in Cn(B)$. Since IMP($\mathcal{B}$) is in P, this step can be finished in polynomial time.

2. For the case of out_3. Now since IMP($\mathcal{B}$) is in P, $O(B_a^O)$ can be constructed in polynomial time. Whether $x \in Cn(O(B_a^O))$ can also be solved in polynomial time.

□

6.7 Summary

The present chapter is the first relevant work that gives a full characterization of basic computational tasks in input/output logic, namely: (1) fulfillment problem of unconstrained input/output logic (2) consistency checking, (3) *maxfamily* membership, (4) full join/meet fulfillment, and (5) negative, positive-static, and positive-dynamic permission checking. Our main finding is that the computational tasks from (1) to (5) are coNP-hard and in the 2nd level of the polynomial hierarchy. More complexity results on norm-based deontic logics will be presented in the next chapter.

Chapter 7

On the Complexity of Norm-based Deontic Logic II

Abstract

This chapter is a continuation of the Chapter 6. In this chapter we study the complexity of normative reasoning by investigating the complexity of prioritized input/output logic, prioritized imperative logic and deontic default logic. We show prioritized input/output logic out_1^p, as well as prioritized imperative logic, is complete for the 2ed level of the polynomial hierarchy while deontic default logic is located in the 3ed level of the polynomial hierarchy.

This chapter is a continuation of the Chapter 6. In Chapter 6, we do not consider priority between norms. In this chapter we bring priority back to our logic. We study the complexity of normative reasoning by investigating the complexity of prioritized input/output logic, prioritized imperative logic and deontic default logic. We say a logic is located in the n-th level of the polynomial hierarchy if it is Δ_n^p-hard and in either Σ_n^p or Π_n^p. Our results in this chapter show that prioritized input/output logic out_1^p, as well as prioritized imperative logic, is complete for the 2ed level of the polynomial hierarchy while deontic default logic is located in the 3ed level of the polynomial hierarchy.

Similar to Chapter 6, we mainly study the complexity of the *fulfillment problem* of norm-based deontic logic. For prioritized input/output logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{\geq}$, a set of input A, a set of constrains C and a formula x, decide if $x \in out_i^p(O^{\geq}, A, C)$.

For prioritized imperative logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{>}$, a set of input A and a formula x, decide if $x \in out_i^h(O^{\geq}, A)$.

For deontic default logic, we introduce $out^d(O^{>}, A)$ through a combination of Horty's framework and input/output logic. The idea is to view proper scenario as something similar to preferred family.

Definition 7.1 (proper output)**.** $x \in out^d(O^{>}, A)$ *iff* $x \in \bigcap\{out_1(O', A) : O' \in propScenario(O, >, A)\}$.

For deontic default logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{>}$, a set of input A and a formula x, decide if $x \in out^d(O^{>}, A)$.

The complexity of prioritized default logic has been studied in Rintanen (1998a). Rintanen investigates the complexity of three proposals of prioritized default logic: Brewka (1994), Baader and Hollunder (1995) and Rintanen (1998b). Rintanen shows that Brewka's logic has the same complexity as the logic of Baader and Hollunder. Both of them are Π_2^p-complete. The logic of Rintanen (1998b), however, is Δ_3^p-hard and in Π_3^p. Our results in this chapter show that out_1^p and prioritized imperative logic have the same complexity as the prioritized default logic of Brewka, Baader and Hollunder, while deontic default logic has similar complexity to Rintanen's prioritized default logic.

7.1 Prioritized input/output logic

Recall that prioritized input/output logic is defined in Parent and van der Torre (2014a) as follows:

$$x \in out_i^p(O^{\geq}, A, C) \text{ iff } x \in \bigcap\{out_i(O', A) : O' \in preffamily_i(O^{\geq}, A, C)\}.$$

Here $preffamily_i(O^{\geq}, A, C)$ is the set of $\succeq$-maximal elements of $maxfamily_i(O, A, C)$. And $\succeq$ is defined via $O_1 \succeq O_2$ iff for all $(a_2, x_2) \in O_2 - O_1$, there is $(a_1, x_1) \in O_1 - O_2$ such that $(a_1, x_1) \geq (a_2, x_2)$.

Lemma 7.1. *Given $O^{\geq} = (O, \geq)$, $O' \subseteq O$ and A, C two sets of formulas. Assume $O' \in maxfamily_1(O, A, C)$. Then $O' \in preffamily_1(O^{\geq}, A, C)$ iff for all $(a, x) \in O$, if $\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\}) \succ O'$ then $out_1(\{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\}, A) \cup C$ is inconsistent.*

Proof. ($\Rightarrow$) Assume $O' \in preffamily_1(O^{\geq}, A, C)$. Suppose there is $(a, x) \in O$ such that $\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\} \succ O'$ and $out_1(\{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\}, A) \cup C$ is consistent. Then $\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\}$ can be extended to a maximal subset of O, say O_1, such that $out_1(O_1, A) \cup C$ is consistent. Then $O_1 \in maxfamily_1(O, A, C)$. Note that $O_1 \succ O'$. This is because $O_1 \supseteq \{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\}$, $\{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\} \succ O'$ and the ordering relation $\succ$ has the following property (which can be routinely verified): if $X \supseteq Y$ and $Y \succ Z$, then $X \succ Z$. Note that $O_1 \succ O'$ contradicts our assumption that $O' \in preffamily_1(O^{\geq}, A, C)$.

($\Leftarrow$) Assume $O' \notin preffamily_1(O^{\geq}, A, C)$. Our aim is to find an $(a, x) \in O$ such that $\{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\} \succ O'$ and $out_1(\{(a, x)\} \cup \{(b, y) \in O' : (b, y) \geq (a, x)\}, A) \cup C$ is consistent.

From $O' \notin preffamily_1(O^{\geq}, A, C)$ we know there is $O_1 \in maxfamily_1(O, A, C)$ such that $O_1 \succ O'$, because it is assumed that $O' \in maxfamily_1(O, A, C)$. That is, $O_1 \succeq O'$ but $O' \not\succeq O_1$. Since $\geq$ is a weak linear order, we know there is some $(a, x) \in O_1 - O'$ such that for all $(b, y) \in O' - O_1$, $(a, x) > (b, y)$.$(*)$

Let $O_2 = \{(a, x)\} \cup \{(c, z) \in O' : (c, z) \geq (a, x)\}$.

We claim that $O_2 \succ O'$. That is, $O_2 \succeq O'$ but $O' \not\succeq O_2$. We first show that $O_2 \succeq O'$. Indeed, for all $(a', x') \in O' - O_2$, by the construction of O_2 we know that $(a', x') \not\geq (a, x)$. Since $\geq$ is linear, we infer that $(a, x) > (a', x')$. Note that $(a, x) \notin O'$. Therefore we conclude that $O_2 \succeq O'$. From $(a, x) > (a', x')$ for all $(a', x') \in O' - O_2$, we also deduce that $O' \not\succeq O_2$.

We claim $O_2 \subseteq O_1$. Suppose there is some $(c', z') \in O_2$ but not in O_1. Then $(c', z') \in O'$ and $(c', z') \geq (a, x)$. Then $(c', z') \in O' - O_1$, which contradict to $(*)$.

Then we know $out_1(O_2, A) \subseteq out_1(O_1, A)$. From $O_1 \in maxfamily_1(O, A, C)$, we know that $out_1(O_1, A) \cup C$ is consistent. Hence $out_1(O_2, A) \cup C$ is consistent. □

Lemma 7.2. *Given $O^{\geq} = (O, \geq)$, $O' \subseteq O$ and A, C two sets of formulas. Deciding if $O' \in$ $preffamily_1(O^{\geq}, A, C)$ is in $\mathtt{P}^{\mathtt{NP}}$.*

Proof. To decide if $O' \in preffamily_1(O^{\geq}, A, C)$, we first decide if $O' \in maxfamily_1(O, A, C)$. Using results from Chapter 6 we know that this can be done in polynomial time in a deterministic Turing machine with an NP oracle. If $O' \notin maxfamily_1(O, A, C)$, then we conclude $O' \notin preffamily_1(O^{\geq}, A, C)$. Otherwise we continue.

Now using Lemma 7.1, to decide whether $O' \in preffamily_1(O^{\geq}, A, C)$, we simply enumerate all $(a, x) \in O$, construct $\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\})$, decide if $\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\}) \succ O'$, and finally decide if $out_1(\{(a, x)\} \cup (\{(b, y) \in O' : (b, y) \geq (a, x)\}), A) \cup C$ is inconsistent. All those steps can be done in polynomial time in a deterministic Turing machine with an NP oracle. □

Theorem 7.1. *Given $O^{\geq} = (O, \geq)$ where $\geq$ is a weak linear order. Let A and C be two sets of formulas and x be a formula. Deciding if $x \in out_1^p(O^{\geq}, A, C)$ is Π_2^p-complete.*

Proof. Concerning the Π_2^p membership, we prove by giving the following algorithm on a non-deterministic Turing machine with an NP oracle to solve the complement of our problem.

1. Guess a subset $O' \subseteq O$.
2. Use the NP oracle to test if $O' \in preffamily_1(O^{\geq}, A, C)$. If no, return "reject" on this branch. Otherwise continue.
3. Use the NP oracle to test if $x \notin out_1(O', A)$. If $x \notin out_1(O', A)$, then return "accept" on this branch. Otherwise return"reject" on this branch.

It can be verified that $x \notin out_1^p(O^{\geq}, A, C)$ iff the non-deterministic Turing machine returns "accept" on some branches. Lemma 7.2 shows that step 2 can be finished in polynomial time. Step 3 can also be done in polynomial time steps because the fulfillment problem is also in $\mathtt{P}^{\mathtt{NP}}$. Therefore the time complexity of this non-deterministic Turing machine is polynomial.

For the Π_2^p hardness, notice that if $\geq = O \times O$, then $preffamily_1(O^{\geq}, A, C) = maxfamily_1(O, A, C)$ because for all $O' \in maxfamily_1(O, A, C)$ there is no $O'' \in maxfamily_1(O, A, C)$ such that $O'' \succ O'$. Our problem is equivalent to the constrained full-meet compliance problem, which is in Π_2^p. □

The above result shows that adding priority (weak linear order) does not increase the complexity for out_1.

7.2 Hansen's prioritized imperative logic

Just like prioritized input/output logic, Hansen introduces preferred maximally obeyable family to characterize those norms which are still functioning in a given situation with possible conflicts. Given a set of prioritized norms $O^{>}$ where $>$ is irreflexive and transitive. A prioritization of $>$ is a strict linear order $\gtrsim$ such that if $\mathfrak{i} \gtrsim \mathfrak{j}$ then $\mathfrak{i} > \mathfrak{j}$ for all $\mathfrak{i}, \mathfrak{j} \in O$. The materialization of O is $m(O) = \{a \to x : (a, x) \in O\}$, which transforms a conditional norm or conditional imperative to a material implication.

Definition 7.2 (preferred obeyable maximal family (Hansen, 2008))**.** *Given a finite set of prioritized commands $O^{>}$ and a set of formulas A. $O' \in pomfamily(O^{>}, A)$ if there is $\gtrsim$ which is a prioritization of $>$ such that $O' = \bigcup_{i=0}^{n} O_i$ where we list $\gtrsim$ by $(a_1, x_n), \ldots, (a_n, x_n)$ such that $(a_i, x_i) \gtrsim (a_{i+1}, x_{i+1})$ and*

1. *$O_0 = \emptyset$,*
2. *$O_{i+1} = O_i \cup \{(a_i, x_i)\}$ if $A \cup m(O_i \cup \{(a_i, x_i)\})$ is consistent. Otherwise $O_{i+1} = O_i$,*

Note that every prioritization induces an element of the *pomfamily*. The results after resolving moral conflicts is characterized by the following output operator.

$$x \in out_i^h(O^{>}, A) \text{ iff } x \in \bigcap\{out_i(O', A) : O' \in pomfamily(O^{\geq}, A)\}.$$

Theorem 7.2. *Given $O^{>} = (O, >)$ where $>$ is irreflexive and transitive. Let A be a set of formulas and x be a formula. Deciding if $x \in out_i^h(O^{\geq}, A)$ is Π_2^p-complete, for $i \in \{1, 2, 3, 4\}$.*

Proof. Concerning the Π_2^p hardness, we show that the validity problem of 2-QBF$^{\forall}$ can be reduced to our problem.

Let $\forall p_1 \ldots p_m \exists q_1 \ldots q_n \Phi$ be a 2-QBF$^{\forall}$ where Φ is a propositional formula with variables in $\{p_1, \ldots, p_m, q_1, \ldots, q_n\}$. Let $A = \emptyset$, $O = \{(\top, p_1), \ldots, (\top, p_m), (\top, \neg p_1), \ldots, (\top, \neg p_m), (\top, \Phi)\}$, $>= \emptyset$. Our aim is to show that this 2-QBF$^{\forall}$ is valid iff $\Phi \in out_i^h(O^{>}, A)$.

- If $\forall p_1 \ldots p_m \exists q_1 \ldots q_n \Phi$ is valid, then for all valuation V for $\{p_1, \ldots, p_m\}$ there is a valuation V' for $\{q_1, \ldots, q_n\}$ such that $V \cup V'$ gives truth value 1 to Φ and 0 to $\neg\Phi$.

  Let $O' = \{(\top, p_1'), \ldots, (\top, p_m'), (\top, \Phi)\}$ be an arbitrary set such that each p_i' is either p_i or $\neg p_i$. Then it can be verified that $O' \in pomfamily(O^{>}, A)$. Indeed, let $>$ be a strict linear order over O such that $(\top, p_1') > \ldots > (\top, p_m') > (\top, \Phi) > (\top, \sim p_1') > \ldots > (\top, \sim p_m')$. Then O' is a preferred obeyable maximal family generated by $>$. By the construction we can further verify that O' ranges over all elements of $pomfamily(O^{>}, A)$. Note that $out_i(O', A) = Cn(\{p_1' \ldots, p_m', \Phi\})$. Therefore $\Phi \in out_i(O', A)$. Then we conclude $\Phi \in out_i^h(O^{>}, A)$.

- If $\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ is not valid, then there is a valuation V for $\{p_1, \dots, p_m\}$ such that for all valuations V' for $\{q_1, \dots, q_n\}$, $V \cup V'$ gives truth value 0 to Φ and 1 to $\neg\Phi$.

  Let $O' = \{(\top, p'_1), \dots, (\top, p'_m)\}$, where each p'_i is p_i if $p_i \in V$ and it is $\neg p_i$ if $p_i \notin V$. Then $O' \in pomfamily(O^{>}, A)$ because $A \cup m(O') = \{p'_1, \dots, p'_m\}$ is consistent and adding anything from $m(\{(\top, \neg p_1), \dots, (\top, \neg p_m), (\top, \Phi)\})$ will destroy the consistency. Note that $\neg\Phi \in Cn(\{p'_1, \dots, p'_m\})$ by the construction of $\{p'_1, \dots, p'_m\}$. Therefore $\Phi \notin out_i(O', A)$, which further implies that $\Phi \notin out_i^h(O^{>}, A)$.

So, we have reduced the validity problem of 2-QBF$^{\forall}$ to our fulfillment problem, which shows the latter is Π_2^p-hard.

Concerning the Π_2^p-membership, we prove by giving the following algorithm on a non-deterministic Turing machine with an NP oracle to solve the complement of our problem.

1. Guess a subset $O' \subseteq O$.
2. Guess a prioritization of $>$.
3. Use the NP oracle to test if $O' \in pomfamily(O^{>}, A)$. If no, return "reject" on this branch. Otherwise continue.
4. Use the NP oracle to test if $x \notin out_i(O', A)$. If $x \notin out_i(O', A)$, then return "accept" on this branch. Otherwise return "reject" on this branch.

It can be verified that $x \notin out_i^h(O^{>}, A)$ iff the non-deterministic Turing machine returns "accept" on some branches. Step 3 can be done in polynomial time steps because the *pomfamily* membership can be decided in P^{NP}. Step 4 can also be done in polynomial time steps because the fulfillment problem of input/output logic is also in P^{NP}. Therefore the time complexity of this non-deterministic Turing machine is polynomial. □

7.3 Horty's deontic default logic

To refresh the readers, we remind the readers that the key notion of Horty's framework in the proper scenario is defined as follows:

Definition 7.3 (Proper scenario (Horty, 2007))**.** *Let O' be a scenario based on the prioritized default theory $(O, >, A)$. Then O' is a proper scenario based on $(O, >, A)$, noted as $O' \in propScenario(O, >, A)$, just in case $O' = \bigcup_{i \geq 0} O'_i$ where*

- $O'_0 = \emptyset$

- $O'_{i+1} = \{(a,x) \in O : (a,x) \in Triggered_{(O,>,A)}(O'_i),$
  $(a,x) \notin Conflicted_{(O,>,A)}(O'), (a,x) \notin Defeated_{(O,>,A)}(O')\}$

We will prove thatdeontic default logic is Δ_3^p-hard and in Π_3^p. For the Δ_3^p-hardness, we make use of the following result from Krentel (1992). Krentel (1992) shows that the following problem is Δ_3^p-complete:

- Maximum 2-QBF: given an arbitrary 2-QBF$^\exists$ $\exists p_1 \dots p_m \forall q_1 \dots q_n \Phi$, decide if $V_1(p_m) = 1$ where V_1 is the **lexicographically maximal** valuation of $\{p_1, \dots, p_m\}$ such that for all valuation V_2 of $\{q_1, \dots, q_n\}$, $V_1 \cup V_2 \vDash \Phi$,

Here for two valuation of $\{p_1, \dots, p_m\}$, V_1 is lexicographically larger than V_2 iff there exists i such that $V_1(p_i) = 1$, $V_2(p_i) = 0$ and for all $j \in \{1, \dots, i-1\}$, $V_1(p_j) = V_2(p_j)$.

Theorem 7.3. *Given $O^> = (O, >)$ where $>$ is irreflexive and transitive. Let A be a set of formulas and x be a formula. Deciding if $x \in out_i^d(O^>, A)$ is Δ_3^p-hard.*

Proof. We prove the Δ_3^p hardness by reducing Maximum 2-QBF to our problem. Given an arbitrary 2-QBF$^\exists$ $\exists p_1 \dots p_m \forall q_1 \dots q_n \Phi$, we construct $O = \{(\top, p_1), (\top, \neg p_1), \dots, (\top, p_m), (\top, \neg p_m), (\Phi, x)\}$, where x is a formula contains no propositional variable from $\{p_1, \dots, p_m, q_1, \dots, q_n\}$. We further let the priority relation be the universal relation, *i.e.* $>= \emptyset$. Our aim is to show that to decide if $V_1(p_m) = 1$ where V_1 is the lexicographically maximal valuations of $\{p_1, \dots, p_m\}$ such that for all valuation V_2 of $\{q_1, \dots, q_n\}$ it holds that $V_1 \cup V_2 \vDash \Phi$, we only need to decide if $p_m \in out^d(O^>, \emptyset)$.

We first show that the following are equivalent: for arbitrary $O' \subseteq O - \{(\Phi, x)\}$ and $P' \subseteq \{p_1, \dots, p_m\}$ satisfying that $(\top, p_i) \in O'$ iff $p_i \in P'$ and $(\top, \neg p_i) \in O'$ iff $p_i \notin P'$,

1. $O' \cup \{(\Phi, x)\} \in propScenario(O, >, \emptyset)$ and $x \in Cn(Conclusion(O' \cup \{(\Phi, x)\}))$.
2. P' is a lexicographically maximal valuation for $\{p_1, \dots, p_m\}$ such that for all $Q' \subseteq \{q_1, \dots, q_n\}$, $P' \cup Q' \models \Phi$.

Assume P' is the lexicographically maximal valuation for $\{p_1, \dots, p_m\}$ such that for all $Q' \subseteq \{q_1, \dots, q_n\}$, $P' \cup Q' \models \Phi$. We show that $O' \cup \{(\Phi, x)\} \in propScenario(O, >, \emptyset)$. Indeed, we can construct $[O' \cup \{(\Phi, x)\}]_0 = \emptyset$, $[O' \cup \{(\Phi, x)\}]_1 = \{(a,x) \in O : (a,x) \in Triggered_{(O,>,A)}([\{O' \cup (\Phi, x)\}]_0), (a,x) \notin Conflicted_{(O,>,A)}(O' \cup \{(\Phi, x)\}), (a,x) \notin Defeated_{(O,>,A)}(O' \cup \{(\Phi, x)\})\}$. Here we have $O' \subseteq [O' \cup \{(\Phi, x)\}]_1$ because for all $(\top, l_i) \in O'$,

1. $(\top, l_i) \in Triggered_{(O,>,A)}(\emptyset)$
2. $(\top, l_i) \notin Conflicted_{(O,>,A)}(O' \cup \{(\Phi, x)\})$.

3. $(\top, l_i) \notin Defeated_{(O,>,A)}(O' \cup \{(\Phi, x)\})$ because $(\top, \sim l_i) \not> (\top, l_i)$ since $> = \emptyset$.

We further have $[O' \cup \{(\Phi, x)\}]_2 = \{(a, x) \in O : (a, x) \in Triggered_{(O,>,A)}([O' \cup \{(\Phi, x)\}]_1), (a, x) \notin Conflicted_{(O,>,A)}(O' \cup \{(\Phi, x)\}), (a, x) \notin Defeated_{(O,>,A)}(O' \cup \{(\Phi, x)\})\}$. Now we prove $[O' \cup \{(\Phi, x)\}]_2 = O' \cup \{((\Phi, x))\}$. This is because

1. for all $(\top, l_i) \notin O'$, $(\top, l_i) \in Conflicted_{(O,>,A)}(O' \cup \{(\Phi, x)\})$.
2. $O' \subseteq [O' \cup \{(\Phi, x)\}]_1 \subseteq [O' \cup \{(\Phi, x)\}]_2$
3. $(\Phi, x) \in [O' \cup \{(\Phi, x)\}]_2$. The reason is: from $Cn(P') \models \Phi$ we derive $Consequence(O') \models \Phi$. Then we know $(\Phi, x) \in Triggered_{(O,>,A)}([O' \cup \{(\Phi, x)\}]_1)$. Meanwhile, $(\Phi, x) \notin Conflicted_{(O,>,A)}(O' \cup \{(\Phi, x)\})$ and $(\Phi, x) \notin Defeated_{(O,>,A)}(O' \cup \{(\Phi, x)\})\}$.

We further have $[O' \cup \{(\Phi, x)\}]_i = [O' \cup \{(\Phi, x)\}]_2$, for all $i \geq 3$. Therefore $O' \cup \{(\Phi, x)\} = \bigcup_{i \geq 0}[O' \cup \{(\Phi, x)\}]_i$, which proves $O' \cup \{(\Phi, x)\} \in propScenario(O, >, \emptyset)$. Then trivially we have $x \in Cn(Conclusion(O' \cup \{(\Phi, x)\}))$.

Assume $O' \cup \{(\Phi, x)\} \in propScenario(O, >, \emptyset)$ and $x \in Cn(Conclusion(O' \cup \{(\Phi, x)\}))$. Then we know $(\Phi, x) \in [O' \cup \{(\Phi, x)\}]_i$ for some i. It cannot be that $(\Phi, x) \in [O' \cup \{(\Phi, x)\}]_0$ because $[O' \cup \{(\Phi, x)\}]_0 = \emptyset$.

- If $(\Phi, x) \in [O' \cup \{(\Phi, x)\}]_1$, then $(\Phi, x) \in Triggered_{(O,>,A)}(\emptyset)$, which means $\emptyset \vDash \Phi$. Then we know $P' \cup Q' \models \Phi$, where P' is the lexicographically maximal valuation for $\{p_1, \ldots, p_m\}$ such that for all $Q' \subseteq \{q_1, \ldots, q_n\}$.
- If $(\Phi, x) \notin [O' \cup \{(\Phi, x)\}]_1$ but $(\Phi, x) \in [O' \cup \{(\Phi, x)\}]_2$, then $[O' \cup \{(\Phi, x)\}]_1 = O'$ and $(\Phi, x) \in Triggered_{(O,>,A)}(O')$. Therefore $Conclusion(O') \vDash \Phi$. Then by the relationship between O' and P', we know that for all $Q' \subseteq \{q_1, \ldots, q_n\}$, $P' \cup Q' \models \Phi$.
- If $(\Phi, x) \notin [O' \cup \{(\Phi, x)\}]_2$, then $[O' \cup \{(\Phi, x)\}]_2 = [O' \cup \{(\Phi, x)\}]_1 = O'$. Moreover $\bigcup_{i \geq 0}[O' \cup \{(\Phi, x)\}]_i = [O' \cup \{(\Phi, x)\}]_1 = O'$, which contradicts to $\bigcup_{i \geq 0}[O' \cup \{(\Phi, x)\}]_i = O' \cup \{(\Phi, x)\}$.

Then we can conclude that $P' \cup Q' \models \Phi$, where P' is the lexicographically maximal valuation for $\{p_1, \ldots, p_m\}$ such that for all $Q' \subseteq \{q_1, \ldots, q_n\}$.

Now we finish our reduction: given an arbitrary 2-QBF$^{\exists}$ $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$, if V_1 is the lexicographically maximal valuations of $\{p_1, \ldots, p_m\}$ such that for all valuation V_2 of $\{q_1, \ldots, q_n\}$, $V_1 \cup V_2 \vDash \Phi$, to decide if $V_1(p_m) = 1$, we only need to decide if $p_m \in out^d(O^>, \emptyset)$. Such reduction is polynomial in the size of $\exists p_1 \ldots p_m \forall q_1 \ldots q_n \Phi$, which proves the Δ_3^p hardness. $\square$

Lemma 7.3. *Given a prioritized default theory $(O, >, A)$, a scenario O' and a default (a, x)*

1. *deciding if $(a, x) \in Triggered_{(O,>,A)}(O')$ is* coNP*-complete.*
2. *deciding if $(a, x) \in Conflicted_{(O,>,A)}(O')$ is* coNP*-complete.*
3. *deciding if $(a, x) \in Defeated_{(O,>,A)}(O')$ is in Σ_2^p.*

Proof. Item 1 and 2 are trivial. Item 3 can be proved by a simple guess and check procedure on a non-deterministic Turing machine with an NP oracle. Here we omit the details. □

Theorem 7.4. *Given $O^> = (O, >)$ where $>$ is a irreflexive and transitive. Let A be a set of formulas and x be a formula. Deciding if $x \in out^d(O^>, A)$ is in Π_3^p.*

Proof. With Lemma 7.3 at hand. This theorem can be proved by a simple guess and check procedure on a non-deterministic Turing machine with an Σ_2^p oracle. Here we omit the details. □

7.4 Summary

In this chapter we have studied the complexity of normative reasoning by investigating the complexity of prioritized input/output logic, prioritized imperative logic and deontic default logic. We have shown that prioritized input/output logic out_1^p, as well as prioritized imperative logic, is complete for the 2ed level of the polynomial hierarchy while deontic default logic is located in the 3ed level of the polynomial hierarchy. Our results have shown that out_1^p and prioritized imperative logic have the same complexity as the prioritized default logic of Brewka, Baader and Hollunder, while deontic default logic has similar complexity to Rintanen's prioritized default logic.

Chapter 8

Application: Logic and Games for Ethical Agents

Abstract

The aim of this chapter is to provide a formal analysis of ethical agents. We adopt a deontic logic+Boolean game approach to the construction of ethical agents. We use deontic logic to reason about norms and use Boolean games to represent the interaction of agents. We use norms to assess the normative status of strategies. Then agents' preferences are changed by the normative status of strategies. Agents of different types use different procedures to change their preference. We characterize 6 types of ethical agents: moral, amoral, social, selfish, negatively impartial and positively impartial. We study some complexity issues related to normative reasoning/status and agents' preference change. When no restriction is imposed, those decision problems of interest to us are decidable but the complexity are high. Under certain restrictions we obtain intermediate and low complexity.

8.1 Introduction

The aim of this chapter is to provide a formal model of ethical agents. In social science, it is acknowledged that decisions of human agents are often affected by both agent's desire and by motivations based on moral issues (Fehr and Schmidt, 2003; Gintis et al., 2005). Intuitively, it seems acceptable that different agents have different reactions when there are conflicts between their obligations (moral value) and desires (personal utility). At least the following types of agents exist or could be constructed.

1. An *amoral* agent prefers actions with higher utility, and ignores the moral aspect of his actions.
2. A *moral* agent prefers actions with higher moral value and ignores the utility of his actions.
3. A *selfish* agent first prefers actions of higher utility. For two action of the same utility, the agent prefers the one with higher moral value.
4. A *social* agent first prefers actions of higher moral value. For two actions of the same moral value, it prefers the action with higher utility.
5. A *negatively impartial* agent first classifies actions into prohibited category and non-prohibited category. Then it ranks its actions using utility within these two categories.
6. A *positively impartial* agent first classifies strategies into permitted category and non-permitted category. Then it ranks its actions using utility within these two categories.

Based on such intuition, our main research concern in this chapter is to answer the following question:

How to formally characterize different types of ethical agents?

Our success criteria is to build formal models of ethical agents such that norms play an important role in these agents' decision-making procedure and such procedures are decidable in general and computationally tractable under certain restrictions. This research question is understood in the setting of normative multiagent systems. Normative multiagent system (Boella et al., 2008b) is a new interdisciplinary academic area developed in recent years bringing together researchers from multiagent system (Wooldridge, 2009), deontic logic and normative system (Ågotnes et al., 2007; Herzig et al., 2011; Alechina et al., 2013). Our methodology to solve the research problem is to adopt a Boolean game+deontic logic approach to the construction of ethical agents and normative multiagent system.

Boolean games (Harrenstein et al., 2001; Bonzon et al., 2009) are a class of games based on propositional logic. In the Boolean game theoretical setting, each agent controls a set of

propositional variables. A strategy of an agent is a truth assignment to the variables he controls. Norms are used to classify strategies as moral, permitted or prohibited. Such classification is used to transform the game by changing the preference relation in the Boolean game. To represent norms in Boolean games, we make use of deontic logic. Using deontic logic, the normative status of strategies is introduced. The preference relations in Boolean games are changed by the normative status of strategies. Agents of different types use different deontic logics for normative reasoning and have different procedures of preference change. The deontic logic and the procedure of preference change characterize different types of ethical agents.

Shoham and Tennenholtz's early work on behavior change under norms has considered only a relatively simple view of norms (Shoham and Tennenholtz, 1992, 1996), where some actions or states are designated as violations. Alechina et al. (2013) studies how conditional norms regulate agents' behaviors, but permissive norms plays no role in their framework. In this chapter, agents' behavior are regulated by conditional norms including permissive norms.

The structure of this chapter is the following: we present some background knowledge on Boolean game and deontic logic in Section 8.2. Ethical agents are introduced in Section 8.3. We study the complexity issues related to the construction of ethical agents in Section 8.4. We discuss some related work in Section 8.5. Then we summarize and conclude this chapter in Section 8.6.

8.2 Boolean games and deontic logic

8.2.1 Boolean games

Boolean games is a class of games based on propositional logic. It was firstly introduced by Harrenstein et al. (2001) and further developed by several researchers (Harrenstein, 2004; Dunne et al., 2008; Bonzon et al., 2009). In a Boolean game, each agent i is assumed to have a goal, represented by a propositional formula x_i over some set of propositional variables $\mathbb{P}$. Each agent i is associated with some subset $\mathbb{P}_i \subseteq \mathbb{P}$ of the variables, which are under the unique control of agent i. The actions, or strategies, available to i correspond to all the possible assignments of truth or falsity to the variables in $\mathbb{P}_i$. An agent will try to choose an assignment so as to satisfy his goal x_i. Strategic concerns arise because whether i's goal is in fact satisfied will depend on the actions made by other agents.

Formally, let $\mathbb{P} = \{p_0, p_1, \ldots\}$ be a finite set of propositional variables and $L_{\mathbb{P}}$ be the propositional language built from $\mathbb{P}$ and constants $\top$ (true) and $\bot$ (false) with the usual connectives $\neg, \vee, \wedge, \rightarrow$ and $\leftrightarrow$. $2^{\mathbb{P}}$ is the set of the valuations for $\mathbb{P}$, with the usual convention that for $V \in 2^{\mathbb{P}}$ and $p \in V$, V gives the value true to p if $p \in V$ and false otherwise. $\models$ denotes the classical logical

consequence relation. Let $X \subseteq \mathbb{P}$, 2^X is the set of X-valuations. A partial valuation (for $\mathbb{P}$) is an X-valuation for some $X \subseteq \mathbb{P}$. Partial valuations are denoted by listing all variables of X, with a " $+$ " symbol when the variable is set to be true and a " $-$ " symbol when the variable is set to be false: for instance, let $X = \{p, q, r\}$, then the X-valuation $V = \{p, r\}$ is denoted $\{+p, -q, +r\}$. If $\{\mathbb{P}_1, \ldots, \mathbb{P}_n\}$ is a partition of $\mathbb{P}$ and $V_1, \ldots, V_n$ are partial valuations, where $V_i \in 2^{\mathbb{P}_i}$, $(V_1, \ldots, V_n)$ denotes the valuation $V_1 \cup \ldots \cup V_n$.

Definition 8.1 (Boolean game (Bonzon et al., 2006))**.** *A Boolean game is a 4-tuple* $(Agent, \mathbb{P}, \pi, Goal)$*, where*

1. $Agent = \{1, \ldots, n\}$ *is a set of agents.*
2. $\mathbb{P}$ *is a finite set of propositional variables.*
3. $\pi : Agent \mapsto 2^{\mathbb{P}}$ *is a control assignment function such that* $\{\pi(1), \ldots, \pi(n)\}$ *forms a partition of* $\mathbb{P}$*. For each agent* i*,* $2^{\pi(i)}$ *is the strategy space of* i*.*
4. $Goal = \langle x_1, \ldots, x_n \rangle$ *is a sequence of formulas of* $L_{\mathbb{P}}$*.* x_i *is the goal which agent* i *want to achieve.*

A strategy for agent i is a partial valuation for all the variables i controls. Note that since $\{\pi(1), \ldots, \pi(n)\}$ forms a partition of $\mathbb{P}$, a strategy profile S is a valuation for $\mathbb{P}$. In the rest of the chapter we make use of the following notation, which is standard in game theory. Let $G = (Agent, \mathbb{P}, \pi, Goal)$ be a Boolean game with $Agent = \{1, \ldots, n\}$, $S = (s_1, \ldots, s_n)$ be a strategy profile, we use S_{-i} to denote the projection of S on $Agent - \{i\}$: $S_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$ and S_i to denote the projection of S on i's strategy. Agents' utilities in Boolean games are induced by their goals. For every agent i and every strategy profiles S, $u_i(S) = 1$ if $S \vDash x_i$, otherwise $u_i(S) = 0$. Agent's preference over strategy profile is induced by his utility function naturally: $S \leq_i S'$ iff $u_i(S) \leq u_i(S')$.

8.2.2 Deontic logic

Norm-based deontic logic are convenient tools for the purpose of this chapter. For the ease of exposition, in this chapter we choose prioritized simple-minded input/output logic out_1^p to illustrate how we build ethical agents. Ethical agents can be built similarly using other norm-based deontic logics.

We make use of both mandatory norms O and permissive norms P. We assume norms are equipped with a priority relation $\geq$. For the sake of simplicity, we assume that $\geq$ is reflexive, transitive and total. That is, $\geq$ is a binary relation over $O \cup P$ such that for all $(a, x), (b, y), (c, z) \in O \cup P$,

- $(a, x) \geq (a, x)$,
- if $(a, x) \geq (b, y)$ and $(b, y) \geq (c, z)$ then $(a, x) \geq (c, z)$,
- either $(a, x) \geq (b, y)$ or $(b, y) \geq (a, x)$.

Here $(a, x) \geq (a', x')$ is understood as (a, x) has higher priority than (a', x'). We call $N = (O, P, \geq)$ a prioritized normative system. Recall that $x \in out_1^p(O^{\geq}, A, C)$ iff $x \in \bigcap\{out_1(O', A) : O' \in \mathit{preffamily}_1(O^{\geq}, A, C)\}$.

Several notions of permission are introduced in input/output logic (Makinson and van der Torre, 2003; Stolpe, 2010c). For the ease of explanation, we choose negative and static positive permission from Makinson and van der Torre (2003) and reformulate them in the setting of prioritized normative system as follows:

Definition 8.2. *Given a normative system $N = (O, P, \geq)$ and an input set A,*

1. $NegPerm_1(N, A) = \{x \in L_{\mathbb{P}} : \neg x \notin out_1^p(O^{\geq}, A, \emptyset)\}$.
2. - *If $P \neq \emptyset$, then $StaPerm_1(N, A) = \{x \in L_{\mathbb{P}} : x \in out_1^p((O \cup \{(a', x')\})^{\geq}, A, \emptyset)$, for some $(a', x') \in P\}$.*
   - *If $P = \emptyset$, then $StaPerm_1(N, A) = out^p(O^{\geq}, A, \emptyset)$.*

If a permissive norm (a, x) has higher priority than a mandatory norm $(\top, \neg x)$, static permission can be understood as *exception* which says although x is forbidden in general, there is an exception which allows x, when a is the case. Detailed discussions of exception as a notion of permission can be found in Stolpe (2010c) and Governatori et al. (2013).

8.3 Ethical agents

A normative multiagent system contains a multiagent system, a normative system and a collection of facts which we call it environment. We use a Boolean game to represent a multiagent system and norm-based deontic logic to reason about and represent norms.

Definition 8.3 (normative multiagent system)**.** *A normative multiagent system is a tuple (G, N, E) where*

- *$G = (Agent, \mathbb{P}, \pi, Goal)$ is a Boolean game.*
- *$N = (O, P, \geq)$ is a finite prioritized normative system.*
- *$E \subseteq L_{\mathbb{P}}$ is the environment, which is a finite set of formulas representing facts.*

In a normative multiagent system, strategies are classified as moral, positively permitted, negatively permitted or prohibited. These concepts are defined using input/output logic.

Definition 8.4 (mandatory, permitted and prohibited strategies)**.** *Given a normative multiagent system (G, N, E), for each agent i, a strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is mandatory if*

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_1^p(O^{\geq}, E, \emptyset).$$

The strategy is positively permitted if

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in StaPerm_1(N, E).$$

The strategy is negatively permitted if

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in NegPerm_1(N, E).$$

The strategy is prohibited if

$$\neg(p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n) \in out_1^p(O^{\geq}, E, \emptyset).$$

Mandatory, positively permitted, negatively permitted and prohibited are four normative positions of strategies. We stipulate that the normative position degrades from mandatory to positively permitted, then to negatively permitted, and finally to prohibited. The **normative status** of a strategy is the highest normative position it has. There can be more than one normative positions for a strategy. But every strategy has a unique normative status. We define normative status using the highest normative position because typically a strategy of higher normative position also has a lower normative position, for example if the normative position of a strategy is mandatory, then it must be negatively permitted. Normative status offers a measure of the moral value of strategies. Normative status is a norm-based classification of an agent's strategies, which are truth assignments to the variables he controls. Such a definition is consistent with the ethical principle *ought implies can* proposed by Immanuel Kant, one of the most important philosopher in human history.

Example 8.1. *Let (G, N, E) be a normative multiagent system as follows:*

- *$G = (Agent, \mathbb{P}, \pi, Goal)$ is a Boolean game with*
  - *$Agent = \{1, 2\}$,*
  - *$\mathbb{P} = \{p, q\}$,*
  - *$\pi(1) = \{p\}$, $\pi(2) = \{q\}$,*
  - *$Goal = \langle p \wedge q, p \vee q \rangle$.*

- $N = (O, P, \geq)$ *where* $O = \{(\top, p)\}, P = \{(\top, q)\}, \geq = (O \cup P) \times (O \cup P)$.
- $E = \emptyset$.

Then $out_1(O, E) = Cn(\{p\}) = out_1^p(O^{\geq}, E, \emptyset)$, $StaPerm_1(N, E) = Cn(\{p, q\})$. *Therefore the normative status of* $+p, +q, -q, -p$ *is respectively mandatory, positively permitted, negatively permitted and prohibited.*

In a normative multiagent system, an agent's preference over strategy profiles is changed by the normative status of strategies. Different types of ethical agents change their preference in different ways. Informally, we let agents change their preference as follows:

1. An *amoral* agent prefers strategy profiles with higher utility.
2. A *moral* agent prefers strategy profiles with higher normative status.
3. - A *selfish* agent first prefers strategy profiles with higher utility.
   - For two strategy profiles of the same utility, the agent prefers the one which contains its strategy of higher normative status.
4. - A *social* agent first prefers strategy profiles which contains its strategy of higher normative status.
   - For two strategy profiles of the same normative status, it prefers strategy profiles with higher utility.
5. - A *negatively impartial* agent first classifies strategies into the negatively permitted category and the prohibited category.
   - Then it ranks its strategies using utility within these two categories.
6. - A *positively impartial* agent first classifies strategies into the positively permitted category and the not positively permitted category.
   - Then it ranks his strategies using utility within these two categories.

We call amoral, selfish, negatively impartial, positively impartial, social and moral agents type-0, type-1, ..., type-5 agents respectively. In Lorini (2015), the *degree of moral sensitivity* is used to measure the strength of an agent's moral value on its preference. That is, an agent is more moral if the degree of moral sensitivity is higher. Combining our terminology with Lorini's, the degree of moral sensitivity of type-i agents is higher than that of type-j agents iff $i > j$.

Given a normative multiagent system, it induces a normative Boolean game, which models the interaction of multiple ethical agents, by changing the preference of agents.

Definition 8.5 (normative Boolean game)**.** *Given a normative multiagent system* (G, N, E) *where* $G = (Agent, \mathbb{P}, \pi, Goal)$*, it induces a normative Boolean game* $G^N = (Agent, \mathbb{P}, \pi, \preceq_1, \ldots \preceq_n)$ *where* $\preceq_i$ *is the preference of i over strategy profiles such that*

1. *if i is type-*0 *(amoral), then* $s \preceq_i s'$ *if*
   - $u_i(s) < u_i(s')$.
2. *if i is type-*1 *(selfish), then* $s \preceq_i s'$ *if*
   - $u_i(s) < u_i(s')$*, or*
   - $u_i(s) = u_i(s')$ *and the normative status of* s'_i *is higher than that of* s_i.
3. *if i is type-*2 *(negatively impartial), then* $s \preceq_i s'$ *if*
   - s_i *is prohibited (not negatively permitted) and* s'_i *is negatively permitted, or*
   - *both* s_i *and* s'_i *are prohibited and* $u_i(s) < u_i(s')$*, or*
   - *both* s_i *and* s'_i *are negatively permitted and* $u_i(s) < u_i(s')$.
4. *if i is type-*3 *(positively impartial), then* $s \preceq_i s'$ *if*
   - s_i *is not positively permitted and* s'_i *is positively permitted, or*
   - *both* s_i *and* s'_i *are not positively permitted and* $u_i(s) < u_i(s')$*, or*
   - *both* s_i *and* s'_i *are positively permitted and* $u_i(s) < u_i(s')$.
5. *if i is type-*4 *(social), then* $s \preceq_i s'$ *if*
   - *the normative status of* s'_i *is higher than that of* s_i*, or*
   - *the normative status of* s'_i *is equal to* s_i *and* $u_i(s) < u_i(s')$.
6. *if i is type-*5 *(moral), then* $s \preceq_i s'$ *if*
   - *the normative status of* s'_i *is higher than that of* s_i.

8.4 Complexity issues

In order to practically build ethical agents, we have to study the complexity of normative reasoning and decision making in our framework. This section shows that the decision problem of interest to us are all decidable. When no restriction is imposed, the complexity is high. Under reasonable restrictions, the complexity turns out to be tractable. Under strong restrictions, the complexity is low.

8.4.1 High complexity

According to the complexity results of prioritized input/output logic provided in Chapter 7 (Theorem 7.1), we immediately have the following theorem showing that deciding whether a strategy is mandatory is in Π_2^p.

Theorem 8.1. *Given a normative multiagent system (G, N, E), a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding whether this strategy is mandatory is in Π_2^p, for $k \in \{0, 1, 2, 3, 4, 5\}$.*

The hardness of deciding whether a strategy is mandatory can be proved using a reduction from the validity problem of 2-QBF$^\forall$.

Theorem 8.2. *Given a normative multiagent system (G, N, E), a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding whether this strategy is mandatory is Π_2^p-hard for $k \in \{1, 2, 3, 4, 5\}$.*

Proof. We show that the validity problem of 2-QBF$^\forall$ can be reduced to our problem.

Let $\forall r_1 \ldots r_m \exists t_1 \ldots t_n \Phi$ be a 2-QBF$^\forall$ where Φ is a propositional formula with variables in $\{r_1, \ldots, r_m, t_1, \ldots, t_n\}$, which is disjoint from $\{p_1, \ldots, p_m, q_1, \ldots, q_n\}$. Let $E = \emptyset$, $O = \{(\top, r_1), \ldots, (\top, r_m), (\top, \neg r_1), \ldots, (\top, \neg r_m), (\top, \Phi), (\top, \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n))\}$, $P = \emptyset$, $\geq = O \times O$. Our aim is to show that this 2-QBF$^\forall$ is valid iff $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_1^p(O, E, \emptyset)$.

- If $\forall r_1 \ldots r_m \exists t_1 \ldots t_n \Phi$ is valid, then for all valuation V for $\{r_1, \ldots, r_m\}$ there is a valuation V' for $\{t_1, \ldots, t_n\}$ such that $V \cup V'$ gives truth value 1 to Φ and 0 to $\neg\Phi$.

  Let $O' = \{(\top, r_1'), \ldots, (\top, r_m'), (\top, \Phi), (\top, \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n))\}$ be a set such that each r_i' is either r_i or $\neg r_i$. Then $out_1(O', E) = Cn(\{r_1' \ldots, r_m', \Phi, \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n)\})$, which is consistent. Moreover it can be verified that $O' \in maxfamily_1(O, E, \emptyset)$. Therefore $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_1(O', E)$. By the construction we can further verify that O' range over all elements of $maxfamily_1(O, E, \emptyset)$. Since $\geq = \emptyset$, we know $maxfamily_1(O, E, \emptyset) = preffamily_1(O^\geq, E, \emptyset)$. Then we conclude $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_1^p(O, E, \emptyset)$.

- If $\forall r_1 \ldots r_m \exists t_1 \ldots t_n \Phi$ is not valid, then there is a valuation V for $\{r_1, \ldots, r_m\}$ such that for all valuations V' for $\{t_1, \ldots, t_n\}$, $V \cup V'$ gives truth value 0 to Φ and 1 to $\neg\Phi$.

  Let $O' = \{(\top, r_1'), \ldots, (\top, r_m'), (\top, \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n))\}$, where each r_i' is r_i if $r_i \in V$ and it is $\neg r_i$ if $r_i \notin V$. Then $O' \in maxfamily_1(O, E, \emptyset)$ because $out_1(O', E) = Cn(\{r_1', \ldots, r_m', \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n)\})$ is consistent and adding anything from $\{(\top, \neg r_1), \ldots, (\top, \neg r_m), (\top, \Phi)\}$ to O' will destroy the consistency. Note that $\neg\Phi \in$

$Cn(\{r'_1, \ldots, r'_m, \Phi \rightarrow (p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n)\})$ by the construction of $\{r'_1, \ldots, r'_m\}$. Therefore $\Phi \notin out_1(O', E)$ and $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \notin out_1(O', E)$, which further implies that $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \notin out_1^p(O, E, \emptyset)$.

So, we have reduced the validity problem of 2-QBF$^{\forall}$ to our target problem, which shows the latter is Π_2^p-hard. □

Corollary 8.1. *Given a normative multiagent system (G, N, E), a type-k agent, where $k \in \{1, 2, 3, 4, 5\}$, and its strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$,*

1. *deciding whether this strategy is prohibited is Π_2^p-complete.*
2. *deciding whether this strategy is negatively permitted is Σ_2^p-complete.*

Theorem 8.3. *Given a normative multiagent system (G, N, E), a type-k agent, where $k \in \{1, 2, 3, 4, 5\}$, and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding whether this strategy is positively permitted is Π_2^p-complete.*

Proof. The Π_2^p hardness is trivial. Here we omit the details. Concerning the Π_2^p membership, let $N = (O, P, \geq)$, $P = \{(a_1, x_1), \ldots, (a_m, x_m)\}$. Note that $StaPerm_1(N, E) = out_1^p((O \cup \{(a_1, x_1)\})^{\geq}, E) \cup \ldots \cup out_1^p((O \cup \{(a_m, x_m)\})^{\geq}, E)$. The Π_2^p membership follows from the fact that the Π_2^p class is closed under union. □

With the above theorems at hand, we can easily prove that deciding the normative status of a strategy is Π_2^p-hard and in $\Delta_3^p = \mathrm{P}^{\Sigma_2^p}$. Moreover, we can now obtain the complexity result of deciding which strategy is better in a normative multiagent system.

Theorem 8.4. *Given a normative multiagent system (G, N, E), an agent i and two strategy profiles s and s', deciding whether $s \prec_i s'$ is in Δ_3^p.*

Proof. This problem can be solved by a polynomial time deterministic Turing machine with an Σ_2^p oracle. We only need to call the oracle to test if s is mandatory, positively permitted, negative permitted or prohibited. And the same test for s'. The utility of s and s' can be calculated in polynomial time. □

The complexity results shown above are not so comforting with respect to the goal of building ethical agents. But we are still optimistic about the future of deontic logic+Boolean game approach to ethical agents for the following reasons:

1. Defeasible deontic logic is computationally efficient. If we take defeasible deontic logic out of our arsenal and use it to replace input/output logic. All those decision problems in this subsection can be solved efficiently.

2. Reasonable restrictions may be imposed to input/output logic such that all decision problems studied in this chapter become tractable. In the following two subsections we study how to lighten the complexity by adding restrictions to input/output logic.

8.4.2 Low complexity

Inspired by defeasible deontic logic, which uses not all propositional formulas but only (modal) propositional literals, we propose the following tractable fragment of input/output logic. Let $Lit_{\mathbb{P}} = \mathbb{P} \cup \{\neg p : p \in \mathbb{P}\}$ be the set of literals build on $\mathbb{P}$. Let $L^{cnl}_{\mathbb{P}}$ be the conjunctions of literals (CNL) of $\mathbb{P}$. That is, $L^{cnl}_{\mathbb{P}}$ is the smallest set such that:

- $Lit_{\mathbb{P}} \subseteq L^{cnl}_{\mathbb{P}}$
- if $a \in L^{cnl}_{\mathbb{P}}$ and $b \in L^{cnl}_{\mathbb{P}}$ then $a \wedge b \in L^{cnl}_{\mathbb{P}}$

For a literal l, we use $\sim l$ to denote its complement. That is, if l is a propositional atom p, then $\sim l$ is $\neg p$. If l is $\neg p$, then $\sim l$ is p. For ease of exposition, we slightly abuse the notation. For a literal l and a CNL x, we say $l \in x$ if l appears as a conjunct in x. If X is a set of CNLs, then we say $l \in X$ if $l \in x$ for some $x \in L$.

Lemma 8.1. *Let A be a finite set of formulas from $L^{cnl}_{\mathbb{P}}$, and $x \in L^{cnl}_{\mathbb{P}}$, then whether $A \vdash x$ can be decided in polynomial time.*

Proof. We can use the following simple algorithm to decide if $A \vdash x$.

1. First check if A contains two complementary literals p and $\neg p$. If yes, then A is inconsistent and $A \vdash x$ holds. Otherwise continue.
2. Check if every literal l which appears in x also appears in A. If yes, then $A \vdash x$ holds. Otherwise $A \nvdash x$.

It can be easily verified that this algorithm costs only polynomial time. □

Corollary 8.2. *Let A be a finite set of formulas from $L^{cnl}_{\mathbb{P}}$, $O \subseteq L^{cnl}_{\mathbb{P}} \times L^{cnl}_{\mathbb{P}}$ be a finite set of norms, and $x \in L^{cnl}_{\mathbb{P}}$. Then $x \in out_1(O, A)$ can be decided in polynomial time.*

Theorem 8.5. *Let A be a finite set of formulas from $L^{cnl}_{\mathbb{P}}$, $O \subseteq L^{cnl}_{\mathbb{P}} \times Lit_{\mathbb{P}}$ be a finite set of norms, and $l \in Lit_{\mathbb{P}}$. Then $l \in out^{p}_{1}(O^{\geq}, A, \emptyset)$ can be decided in polynomial time.*

Proof. To decide whether a literal $l \in out^{p}_{1}(O^{\geq}, A, \emptyset)$, we use the following procedure:

- If there is $(a, l) \in O$ such that $a \in Cn(A)$ and for all $(b, \sim l) \in O$ such that $b \in Cn(A)$, we have $(a, l) > (b, \sim l)$, then we conclude that $l \in out_1^p(O^{\geq}, A, \emptyset)$. Otherwise we conclude that $l \notin out_1^p(O^{\geq}, A, \emptyset)$.

This procedure can be executed in polynomial time because in the current setting whether $a, b \in Cn(A)$ can be decided in polynomial time. Now we prove that this procedure gives the correct answer.

- Assume there is $(a, l) \in O$ such that $a \in Cn(A)$ and for all $(b, \sim l) \in O$ such that $b \in Cn(A)$, we have $(a, l) > (b, \sim l)$.

  1. If there is no $(b, \sim l) \in O$ such that $b \in Cn(A)$. Then $(a, l) \in O'$ for every $O' \in maxfamily_1(O, A, \emptyset)$. Therefore $(a, l) \in O''$ for every $O'' \in preffamily_1(O^{\geq}, A, \emptyset)$. Then we know $l \in out_1^p(O^{\geq}, A, \emptyset)$.
  2. If there is some $(b, \sim l) \in O$ such that $b \in Cn(A)$. Suppose there is an $O' \in preffamily_1(O^{\geq}, A, \emptyset)$ such that $(a, l) \notin O'$. Let $O'' = (O' - \{(b, \sim l) : b \in Cn(A)\}) \cup \{(a, l)\}$. Then $O'' \succ O'$ because $(a, l) > (b, \sim l)$. Note that $out_1(O'', A)$ is consistent. Therefore O'' can be extended to O''' such that $O''' \in maxfamily_1(O, A, \emptyset)$. Then we know $O''' \succ O'$, which contradicts to $O' \in preffamily_1(O^{\geq}, A, \emptyset)$.

     Hence there is no $O' \in preffamily_1(O^{\geq}, A, \emptyset)$ such that $(a, l) \notin O'$. Therefore $(a, l) \in O'$ for every $O' \in preffamily_1(O^{\geq}, A, \emptyset)$, $l \in out_1^p(O^{\geq}, A, \emptyset)$.

- Assume it is not the case that there is $(a, l) \in O$ such that $a \in Cn(A)$ and for all $(b, \sim l) \in O$ such that $b \in Cn(A)$, we have $(a, l) > (b, \sim l)$. Then trivially we have $l \notin out_1^p(O^{\geq}, A, \emptyset)$.

□

Theorem 8.6. *Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{cnl}$, $O \subseteq L_{\mathbb{P}}^{cnl} \times Lit_{\mathbb{P}}$, a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding whether this strategy is mandatory/positively permitted/negative permitted/prohibited is in polynomial time, for $k \in \{0, 1, 2, 3, 4, 5\}$.*

Proof. To decide if $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is a mandatory strategy, we simply check if $p_1 \in out_1^p(O^{\geq}, E, \emptyset)$, …, $p_m \in out_1^p(O^{\geq}, E, \emptyset)$, $\neg q_1 \in out_1^p(O^{\geq}, E, \emptyset)$, …, $\neg q_n \in out_1^p(O^{\geq}, E, \emptyset)$. Other cased are similar. □

In light of the above theorem, we know that under those restrictions introduced in this subsection, the normative status of any strategy can be computed in polynomial time.

Corollary 8.3. *Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{cnl}$, $O \subseteq L_{\mathbb{P}}^{cnl} \times Lit_{\mathbb{P}}$, an agent i and two strategy profiles s and s', deciding whether $s \prec_i s'$ is in polynomial time.*

8.4.3 Intermediate complexity

Restricting formulas to literals or CNLs seems too stringent and limits the expressive power of the logical language. To find a balance between expressive power and complexity, we make use of Horn formulas, which are well studied in propositional logic programming (Dantsin et al., 2001).

A strict Horn clause is a non-empty disjunction of exactly one propositional atom and zero or more negated atoms. A strict Horn formula is a conjunction of strict Horn clauses. Let $L_{\mathbb{P}}^{Horn}$ be the set of strict Horn formulas build from $\mathbb{P}$. For a set of strict Horn formulas $A \subseteq L_{\mathbb{P}}^{Horn}$, a least model $LM(A)$ of A is a smallest set $P' \subseteq \mathbb{P}$ such that $P' \vDash A$. It is known in propositional logic programming that every set of strict Horn formulas has a unique least model.

Lemma 8.2. *(Dantsin et al., 2001) Let A be a finite set of strict Horn formulas from $L_{\mathbb{P}}^{Horn}$, and $p \in \mathbb{P}$, then deciding whether $p \in LM(A)$ is* P*-complete.*

Corollary 8.4. *Let A be a finite set of formulas from $L_{\mathbb{P}}^{Horn}$, and $x \in L_{\mathbb{P}}^{Horn}$, then deciding whether $A \vdash x$ is* P*-complete.*

Proof. It is P-hard because decided whether $p \in LM(A)$ can be reduced to this problem. This problem is in P because to we can decide if $A \vdash x$ by testing if every atom in x is also in $LM(A)$. □

Corollary 8.5. *Let A be a finite set of formulas from $L_{\mathbb{P}}^{Horn}$, $O \subseteq L_{\mathbb{P}}^{Horn} \times L_{\mathbb{P}}^{Horn}$ be a finite set of norms, and $x \in L_{\mathbb{P}}^{Horn}$. Then deciding whether $x \in out_1(O, A)$ is* P*-complete.*

Theorem 8.7. *Let A be a finite set of formulas from $L_{\mathbb{P}}^{Horn}$, $O \subseteq L_{\mathbb{P}}^{Horn} \times Lit_{\mathbb{P}}$ be a finite set of norms, and $l \in Lit_{\mathbb{P}}$. Then deciding whether $l \in out_1^p(O^{\geq}, A, \emptyset)$ is* P*-complete.*

Proof. To decide whether $l \in out_1^p(O^{\geq}, A, \emptyset)$, we use the same procedure as the proof of Theorem 8.5:

- If there is $(a, l) \in O$ such that $a \in Cn(A)$ and for all $(b, \sim l) \in O$ such that $b \in Cn(A)$, we have $(a, l) \succ (b, \sim l)$, then we conclude that $l \in out_1^p(O^{\geq}, A, \emptyset)$. Otherwise we conclude that $l \notin out_1^p(O^{\geq}, A, \emptyset)$.

The time complexity of this procedure is P-hard because in the current setting deciding whether $a, b \in Cn(A)$ is P-hard. □

Theorem 8.8. *Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{Horn}$, $O \subseteq L_{\mathbb{P}}^{Horn} \times Lit_{\mathbb{P}}$, a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding whether this strategy is mandatory/positively permitted/negative permitted/prohibited is* P*-complete, for $k \in \{1, 2, 3, 4, 5\}$.*

Proof. To decide if $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is a mandatory strategy, we simply check if $p_1 \in out_1^p(O^{\geq}, E, \emptyset)$, $\ldots$, $p_m \in out_1^p(O^{\geq}, E, \emptyset)$, $\neg q_1 \in out_1^p(O^{\geq}, E, \emptyset)$, $\ldots$, $\neg q_n \in out_1^p(O^{\geq}, E, \emptyset)$. Other cased are similar. □

In light of the above theorem, we know that under those restrictions introduced in this subsection, the computation of the normative status of any strategy is P-complete.

Corollary 8.6. *Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{Horn}$, $O \subseteq L_{\mathbb{P}}^{Horn} \times Lit_{\mathbb{P}}$, an agent i which is of type k, $k \in \{1, 2, 3, 4, 5\}$, and two strategy profiles s and s', deciding whether $s \prec_i s'$ is P-complete.*

8.5 Related work

There are many types of ethical theories in the literature. At least two of them have found applications in machine ethics (Gips, 1994): consequentialist and deontological. In consequentialist theories, actions are evaluated by their consequences. The best action to take now is the action that results in the best situation in the future. Discussions on whether we should install utilitarian reasoning to robots can be found in Grau (2011).

In a deontological ethical theory, actions however, may be thought to be moral or immoral independent of the specific consequences they may cause. Typically, in a deontological ethical theory actions are judged by deontological moral systems. One of the oldest examples of a deontological moral system is the Ten Commandments in the Old Testament. Another well-known deontological moral system is Kant's categorical imperative, which states "Act only on that maxim which you can at the same time will to be a universal law." An example of a modern deontological moral system is the 10 moral rules proposed by Gert (2005):

1. Don't kill.
2. Don't cause pain.
3. Don't disable.
4. Don't deprive of freedom.
5. Don't deprive of pleasure.
6. Don't deceive.
7. Keep your promise.
8. Don't cheat.

9. Obey the law.

10. Do your duty.

Powers (2006) considers the first formulation of Kant's categorical imperative to determine "what computational structures such a view would require and to see what challenges remain for its successful implementation." Powers proposes to use nonmonotonic logic, especially default logic, to model Kant's categorical imperatives.

Our approach can be viewed as another attempt to impose deontological ethical theories (not Kant's theory) to machines/robots/artificial agents. Gips (1994) points out that "Whenever a multi-rule system is proposed, there is the possibility of conflict between the rules." This creates a problem that needs to be resolved so that robots will know what to do when conflicts arise. Our approach (partially) solves this problem, thanks to the power of resolving moral conflicts from norm-based deontic logic.

8.6 Summary

The aim of this chapter is to provide a formal analysis of ethical agents. Ethical agents have been extensively studied in moral philosophy and in economics, and their study is identified as one of the thorniest challenges in artificial intelligence. We have adopted a deontic logic+Boolean game approach to the construction of ethical agents. We have used deontic logic to reason about norms and used Boolean games to represent the interaction of agents. We have used norms to assess the normative status of strategies. Then agents' preference are changed by the normative status of strategies. Agents of different types use different procedures to change their preference. We characterize 6 types of ethical agents: moral, amoral, social, selfish, negatively impartial and positively impartial. We have studied some complexity issues related to normative reasoning/status and agents' preference change. When no restriction is imposed, those decision problems of interest to us are decidable but the complexity is high. Under certain restrictions we obtained intermediate and low complexity.

Chapter 9

Summary and Future Work

9.1 Summary

9.1.1 Norm creation in games

In Chapter 2, we followed Gintis' proposal (Gintis, 2010) and presented an alternative offline norm creation framework such that the complexity of norm creation is tractable and every agent will comply with the created norms. Compared to the social law paradigm, the main features of our framework are the following:

1. Instead of constraints, norms in our framework work with randomized signals, like traffic lights, to guide agents' behavior. We generate randomized signals by computing correlated equilibrium of games. Such signals are involved in the description of the triggering condition of norms.

2. Five types of norms were created in Chapter 2: utilitarian, egalitarian, elitist, Nash-product and opportunity-balanced norms.

   (a) Utilitarian norms are created from those correlated equilibria that maximize the sum of the expect utility of all agents. A utilitarian correlated equilibrium is a correlated equilibrium that maximizes utilitarian social welfare. In a given game, utilitarian social welfare sums up the agents' expected utilities, thus providing a useful measure of the overall benefit of the society. A utilitarian correlated equilibrium can be computed by using linear programming with maximizing utilitarian social welfare as the objective function and requirements of the correlated equilibrium as constrains.

(b) Egalitarian norms are created from those correlated equilibria which maximize the expected utility of the poorest agent. Egalitarian social welfare is measured by the situation of the poorest member of the society. It therefore provides a useful measure of fairness in cases where the minimum need of all agents are to be satisfied. An egalitarian correlated equilibrium is a correlated equilibrium that maximizes egalitarian social welfare. We use linear programming to compute egalitarian correlated equilibria. The variables of the linear program are random variables representing the probability assigned to each joint action of the game. The objective function of the linear program is to maximize the egalitarian social welfare.

(c) Elitist norms are created from those correlated equilibria which maximize the expected utility of the happiest agent. Elitist social welfare is measured by the situation of the happiest member of the society. Maximizing elitist social welfare reflects the famous *Matthew effect* in sociology which describes the phenomenon where "the rich get richer and the poor get poorer". The elitist social welfare is clearly not a fair measure for social welfare, but it can be useful in cooperation based applications where we require only one agent to achieve its goals. An elitist correlated equilibrium is a correlated equilibrium that maximizes elitist social welfare. We use convex optimization to compute elitist correlated equilibrium. The variables of the linear program are random variables representing the probability assigned to each strategy profile of the game. The objective function is to minimize elitist social welfare.

(d) Nash-product norms are created from those correlated equilibria which maximize the product of the expected utility of all agents. Given a game in which utility function is non-negative, the Nash-product social welfare is the product of the expected utility of all agents. Nash-product social welfare can be understood as a compromise between utilitarian and egalitarian social welfare. On the one hand, just as utilitarian social welfare, Nash-product social welfare increases with single increasing of individual utilities. On the other hand, just as egalitarian social welfare, Nash-product social welfare reaches its maximum when the utilities distributed equally over all agents. A Nash-product correlated equilibrium is a correlated equilibrium that maximizes Nash-product social welfare. We use convex optimization to compute Nash-product correlated equilibrium.

(e) Opportunity-balanced norms are created from those correlated equilibria which are computed by taking the average of those correlated equilibria which maximize the expect utility of every single agent. The set of all correlated equilibria of a game is a

convex set because correlated equilibrium is defined using linear constrains. Therefore the average of finite many correlated equilibria is again a correlated equilibrium. Given an arbitrary game with n agents, an opportunity-balanced correlated equilibrium can be computed as follows:

i. For each agent, use linear programming to compute a correlated equilibrium which maximizes the agent's expected utility.

ii. Take the average of the n correlated equilibria from the previous step.

The procedure of norm creation in Chapter 2 is as follows: at first a normal-form game is given. Then we computed a correlated equilibrium of the given game. The resulting correlated equilibrium is a probability distribution over agents' action profiles. We then transform the probability distribution to randomized signals and create norms and signals to guide agents' behavior.

9.1.2 Norm emergence in games

In Chapter 3 we proposed a model that supports the emergence of norms via multiagent learning in social networks. In our model, individual agents repeatedly interact with their neighbors in a game called Ali Baba and the Thief. In this 2-player game, each agent has two strategies: Ali Baba and Thief. Each agent has initial utility x. If both agents choose Ali Baba, then their utilities do not change. If they both choose Thief, then there will be a fight between them and they are both injured. The resulting utility is 0. If one chooses Ali Baba and the other chooses Thief, then Thief robs Ali Baba and the utility of the one who chooses Thief increases by d and the other one decreases by d, where $0 \leq d \leq x$. We call d the amount of robbery.

We identified norms prescribing no harmful behavior with the strategy of Ali Baba in this game. In our model this game is repeatedly played by a given amount of agents. Each agent adapts its strategy by using a learning rule between different rounds of play. We say a norm has emerged in the population if:

(1) All agents are choosing and will continue to choose the action prescribed by the norm.

(2) Every agent believes that all agents who are relevant in its social network, will choose the action prescribed by the norm in the next round.

(3) Every agent believes that all other agents who are relevant in its social network, believe that it is good if the agent chooses the action prescribed by the norm.

We have used replicator dynamics and imitate-the-best as rules of learning. No social network is assumed when agents learn by using replicator dynamics while lattice model and small world model are used when agents use imitate-the-best.

In the stable state of replicator dynamics, a pattern of behavior emerges. In this pattern, a proportion p of agents choose Thief and a proportion $1 - p$ of agents choose Ali Baba. When d is very close to 0, norms saying "don't rob", "be peaceful" or "don't harm others" can be viewed as emerged. On the other hand, when d is very close to x, although it is true that,

- Almost all agents are choosing and will continue to choose Thief.
- All agents believe most agents will choose Thief in the next round.

It is not the case that every agent believe that most agents believe that it is good if the agent chooses Thief. Therefore norms prescribing "you should rob" do not emerge, even though most agents choose Thief in the stable state.

We then run simulations for imitate-the-best using the Netlogo platform. We set the population of agents to a fixed number. Initially, 50% of agents choose Thief. Over time, however, through agent-agent interactions, a bias toward Ali Baba spreads through the entire network until 100% of the population choose Ali Baba. At this point, we say that a norm prescribing that there should be no harmful behavior has emerged.

In the lattice model, our experiments showed that when the amount of robbery is high, the probability of norm emergence is low. When the amount of robbery decrease, the probability of norm emergence quickly increase. Given the initial utility $x = 1000$, when d is less than 400, the norm "you should not rob" emerges for certain. This is in contrast with the analysis of replicator dynamics. If $d = 400$, then according to replicator dynamics a proportion of 40% of agents will choose Thief, therefore a norm saying there ought to be no robbery does not emerge. Our experiments also show that in the lattice model there is a leap of the probability of norm emergence from $d = 600$ to $d = 400$.

Just like in the lattice model, our experiments in small world model showed that the probability of norm emergence increases when the amount of robbery decreases. There is a leap of the probability of norm emergence from $d = 300$ to $d = 200$.

Such leaps suggest that there are critical points of norm emergence which are decided by the quotient of the initial utility and the amount of robbery in Ali Baba and the Thief. When the quotient of the initial utility and the amount of robbery is smaller than the critical point, the probability of norm emergence is high. The probability drops dramatically as long as the quotient is larger than the critical point.

9.1.3 Axiomatics of norms

Input/output logic adopts mainly operational semantics: a normative system is conceived in input/output logic as a deductive machine, like a black box that produces normative statement as output, when we feed it descriptive statements as input. The procedure of the operational semantics is divided to three stages. In the first stage, we have in hand a set of propositions (call it the input) as a description of the current state. We then apply logical operators to this set, say, close the set by logical consequence. Then we pass this set to the deductive machine and we reach the second stage. In the second stage, the machine takes the input and produces a set of propositions as output. In the third stage, we apply logical operators to the output. On the axiomatic side, input/output logic is characterized by derivation rules about norms. The derivation systems of input/output logic are axiomatic representations of norms. A norm is represented by an ordered pair of formulas. Given a set of mandatory norms O, a derivation system is the smallest set which extends O and is closed under certain derivation rules. The following are the derivation rules that have been used to build input/output logic:

- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vdash a$.
- IEQ (input equivalence): from (a, x) and $a \dashv\vdash b$ to (b, x). Here $a \dashv\vdash b$ means $a \vdash b$ and $b \vdash a$.
- OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.
- WO (weakening the output): from (a, x) to (a, y) whenever $x \vdash y$.
- OEQ (output equivalence): from (a, x) and $x \dashv\vdash y$ to (a, y).
- AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.
- Z (zero premise): from nothing to $(\top, \top)$.
- ID (identity): from nothing to (a, a), for every $a \in L_{\mathbb{P}}$.
- T (plain transitivity): from (a, x) and (x, y) to (a, y).
- CT (cumulative transitivity): from (a, x),$(a \wedge x, y)$ to (a, y) .
- MCT (mediated cumulative transitivity): from (a, x'), $x' \vdash x$ and $(a \wedge x, y)$ to (a, y).
- ACT (aggregative cumulative transitivity): from (a, x),$(a \wedge x, y)$ to $(a, x \wedge y)$.

The derivation system based on the rules SI, WO, AND and Z is called $deriv_1$. Adding OR to $deriv_1$ gives $deriv_2$. Adding CT to $deriv_1$ gives $deriv_3$. These five rules together give $deriv_4$. Adding ID to $deriv_i$ gives $deriv_i^+$ for $i \in \{1, 2, 3, 4\}$. $(a, x) \in deriv(O)$ is used to denote that (a, x) is

derivable from O using rules of derivation system *deriv*. The rules IEQ, OEQ, and T is used in the input/output logic of constitutive norms. MCT is introduced by Stolpe (2008a) in his mediated reusable input/output logic, while ACT is recently introduced by Parent and van der Torre in their aggregative input/output logic.

One feature of the existing work of input/output logic is: the derivation rules always work in bundles. When several derivation rules work together, the corresponding operational semantics is rather complex, and insights of the machinery is therefore concealed. To achieve a deeper understanding of input/output logic, it is helpful to isolate every single rule and study them separately.

In Chapter 4 we analyzed various derivation rules of input/output logic in isolation and defined the corresponding semantics. Then we combine them together to achieve alternative semantics for several input/output logics. Since the procedure of operational semantics is divided into three stages, we also classify derivation rules according to different stages:

- Rules of input correspond to operations in the first stage. SI means to close the input by logical consequence; IEQ means to close the input by logical equivalence; OR ensures that the input has to be extended to satisfy disjunctive property.
- Rules of output correspond to operations in the third stage. WO means close the output by logical consequence; OEQ means close the output by logical equivalence; AND ensures that the output is closed under conjunction.
- Rules of normative system correspond to operations in the second stage. Z means the normative system O is extended to $O \cup \{(\top, \top)\}$. ID means to add all norms of the form (a, a) to the normative system.
- Cross-stage rules affect more than one stages. Such rules typically have the form of transitivity and some of them can be characterized by fixed-point formalism.

In Chapter 4 we developed alternative semantics for out_3 and out_3^+. Such alternative semantics is useful in the study of the complexity of input/output logic. Our alternative semantics for constitutive input/output logic is adequate for the derivation system of constitutive input/output logic.

9.1.4 Algebra of norms

Since input/output logic adopts operational rather than possible world semantics, there is no exterior structure in such operational semantics. Therefore tools to compare the similarity of

structures, like bisimulation and isomorphism, play no role in input/output logic. This feature makes it difficult to analyze the *similarity* of normative systems using input/output logic, although the *equivalence* of normative systems can be represented within the input/output framework.

An algebraic framework for analyzing normative systems is introduced by Lindahl and Odelstad (2000, 2008, 2013); Odelstad and Lindahl (2000). The most general form of the theory is called theory of joining-systems. A joining-system is a triple (B_1, B_2, S) where B_1, B_2 are two ordered algebraic structures and S a relation between B_1 and B_2 satisfying some conditions.

In Chapter 5 we developed two variants of theory of joining-systems: Boolean joining-systems and Heyting joining-systems. A Boolean algebra is a structure $\mathfrak{A} = (A, +, \cdot, -, 0, 1)$ where A is a set, $+$ and $\cdot$ are binary operators on A, $-$ is a unitary operator on A and $0, 1 \in A$, which satisfies the following identities: for all $x, y, z \in A$,

1. $x + y = y + x,\ x \cdot y = y \cdot x$
2. $x + (y + z) = (x + y) + z,\ x \cdot (y \cdot z) = (x \cdot y) \cdot z$
3. $x + 0 = x,\ x \cdot 1 = x$
4. $x + (-x) = 1,\ x \cdot (-x) = 0$
5. $x + (y \cdot z) = (x + y) \cdot (x + z),\ x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

A Boolean joining-system is a triple (B_1, B_2, S) where B_1, B_2 are two Boolean algebras and S is a relation between B_1 and B_2 satisfying some conditions. Boolean joining-system algebraically characterizes unconstrained input/output logic in the sense that a norm (a, x) is derivable from a set of norms O if and only if it is in the space of norms algebraically generated by O.

A frequent belief about input/output logic is that it presupposes classical propositional logic. Parent et al. (2014) show that this is a misunderstanding by building input/output logic on top of intuitionistic logic. In Chapter 5 we showed that Heyting joining-systems is an algebraic companion for intuitionistic input/output logic. Heyting algebra was introduced by Arend Heyting in 1930s to formalize intuitionistic logic. Heyting algebra generalizes Boolean algebra in the sense that a Heyting algebra satisfying $x + (-x) = 1$ is a Boolean algebra. A Heyting algebra is a partially ordered set $(H, 0, 1 \leq, \cdot, +, \rightarrow)$ with a smallest elements 0, a largest element 1 and three operators $\cdot$, $+$ and $\rightarrow$ satisfying the following conditions, for all $x, y, z \in H$

1. $x \leq 1$
2. $x \cdot y \leq x$
3. $x \cdot y \leq y$

4. $z \leq x$ and $z \leq y$ implies $z \leq x \cdot y$

5. $0 \leq x$

6. $x \leq x + y$

7. $y \leq x + y$

8. $x \leq z$ and $y \leq z$ implies $x + y \leq z$

9. $z \leq (x \to y)$ iff $z \cdot x \leq y$

A Heyting joining-system is a triple (B_1, B_2, S) where B_1, B_2 are two Heyting algebras and S a relation between B_1 and B_2 satisfying some conditions. Heyting joining-system algebraically characterizes unconstrained intuitionistic input/output logic in the same sense as Boolean joining-system algebraically characterizes unconstrained input/output logic.

Lindahl and Odelstad's joining-systems, as well as our Boolean and Heyting joining-systems, provide algebraic representations of norms and normative system. One advantage of such an algebraic representation is that we can use them to study the *similarity* of normative systems. In Chapter 5 we approached the similarity of normative systems by introducing isomorphism and embedding between normative systems. Using theory of joining-systems, in Chapter 5 we defined the core of a normative system. We showed that for two finite joining-systems, they are the same iff their cores are the same.

9.1.5 On the complexity of norm-based deontic logic

We have provided complexity results of many norm-based deontic logics. In Chapter 6 we showed unconstrained input/output logic is in the 1st level of the polynomial hierarchy. We focused on the fulfillment problem of unconstrained input/output logic:

- Given a finite set of mandatory norms O, a finite set of formulas A and a formula x, is $x \in out_i(O, A)$?

We showed that the fulfillment problem of input/output logic $out_1, out_1^+, out_2, out_2^+, out_4$ and out_4^+ is $\mathtt{coNP}$-complete. For out_3 and out_3^+, we showed that the complexity of the fulfillment problem is $\mathtt{coNP}$-hard and in $\mathtt{P}^{\mathtt{NP}}$.

We showed that constrained input/output logic are complete for the 2ed level of the polynomial hierarchy. In the constrained setting, a finite set of mandatory norms O, a subset $O' \subseteq O$, a finite set of input A and a finite set of constrains C are given. We study the complexity of the following problems:

- consistency checking: is $out_i(O, A)$ consistent with C?
- maxfamily membership: is $O' \in maxfamily_i(O, A, C)$?
- full-join fulfillment: is $x \in out_i^{\cup}(O, A, C)$?
- full-meet fulfillment: is $x \in out_i^{\cap}(O, A, C)$?

We showed that for $i \in \{1, 2, 4, 1^+, 2^+, 4^+\}$, the consistency checking problem is NP-complete, while for $i \in \{3, 3^+\}$, the consistency checking problem is NP-hard and in P^{NP}. Then we proved that for $i \in \{1, 2, 4, 1^+, 2^+, 4^+\}$, the maxfamily membership problem is BH_2-complete and for $i \in \{3, 3^+\}$, the maxfamily membership problem is BH_2-hard and in P^{NP}. We further showed that for $i \in \{1, 2, 3, 4, 1^+, 2^+, 3^+, 4^+\}$, full-join fulfillment problem is $\mathrm{NP}^{\mathrm{NP}}$-complete ($\Sigma_2^p$-complete) and the full-meet fulfillment problem is $\mathrm{coNP}^{\mathrm{NP}}$-complete ($\Pi_2^p$-complete).

In Chapter 6 we also studied the complexity of the following decision problems about permissive input/output logic: given a finite normative system $N = (O, P)$, a finite set of input A and a formula x:

- negative permission checking: is $x \in NegPerm_i(N, A)$?
- positive-static permission checking: is $x \in StaPerm_i(N, A)$?
- positive-dynamic permission checking: is $x \in DyPerm_i(N, A)$?

Negative and static permission checking in input/output logic are in the 1st level of the polynomial hierarchy while dynamic permission checking is complete for the 2ed level of the polynomial hierarchy. We showed that for $i \in \{1, 2, 4\}$, negative permission checking is NP-complete and for $i = 3$, negative permission checking is NP-hard and in P^{NP}. For $i \in \{1, 2, 4\}$, the positive-static permission checking is coNP-complete and for $i = 3$, the positive-static permission checking is coNP-hard and in P^{NP}. Positive-dynamic permission checking is harder than other permission checking: For $i \in \{1, 2, 3, 4\}$, positive-dynamic permission checking is $\mathrm{NP}^{\mathrm{NP}}$-complete. The main source of complexity is that in positive-dynamic permission checking we have to first guess a consistent input and then check if it produces some inconsistency.

In Chapter 6 we also studied how to impose syntactic restrictions such that decision problems of unconstrained input/output logic are tractable (in P). For out_1 and out_3, we achieve tractability via Post Lattice. Our results showed that if $\mathcal{B}$ is a finite set of Boolean functions such that $S_{00} \not\subseteq [\mathcal{B}]$, $S_{10} \not\subseteq \mathcal{B}$ and $D_2 \not\subseteq \mathcal{B}$, and A is a set of $\mathcal{B}$-formulas, all formulas appear in O are $\mathcal{B}$-formulas and x a $\mathcal{B}$-formula, then deciding if $x \in out_i(O, A)$ is in P, for $i \in \{1, 3, 1^+, 3^+\}$.

Chapter 7 is a continuation of Chapter 6. In Chapter 6, we did not consider priority between norms. In Chapter 7 we brought priority back to our logic. We studied the complexity of normative

reasoning by investigating the complexity of prioritized input/output logic, prioritized imperative logic and deontic default logic. Similar to Chapter 6, we mainly studied the complexity of the *fulfillment problem* of norm-based deontic logic. For prioritized input/output logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{\geq}$, a set of input A, a set of constrains C and a formula x, decide if $x \in out_i^p(O^{\geq}, A, C)$.

For prioritized imperative logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{>}$, a set of input A and a formula x, decide if $x \in out_i^h(O^{>}, A)$.

For deontic default logic, the fulfillment problem is:

- Given a set of prioritized norms $O^{>}$, a set of input A and a formula x, decide if $x \in out^d(O^{>}, A)$.

Our results in Chapter 7 showed that prioritized input/output logic out_1^p, as well as prioritized imperative logic, is complete for the 2ed level of the polynomial hierarchy while deontic default logic is located in the 3ed level of the polynomial hierarchy.

9.1.6 Logic and games for ethical agents

In Chapter 8 we adopted a deontic logic+Boolean game approach to the construction of ethical agents. We used norm-based deontic logic to reason about norms and use Boolean games to represent the interaction of agents. We used norms to assess the normative status of strategies. Given a normative multiagent system (G, N, E), for each agent i, a strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is mandatory if

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_1^p(O^{\geq}, E, \emptyset).$$

The strategy is positively permitted if

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in StaPerm_1(N, E).$$

The strategy is negatively permitted if

$$p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in NegPerm_1(N, E).$$

The strategy is prohibited if

$$\neg(p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n) \in out_1^p(O^{\geq}, E, \emptyset).$$

Then agents' preferences are changed by the normative status of strategies. Agents of different types use different procedures to change their preferences. We characterize 6 types of ethical agents: moral, amoral, social, selfish, negatively impartial and positively impartial.

1. An *amoral* agent prefers strategy profiles with higher utility.
2. A *moral* agent prefers strategy profiles with higher normative status.
3. - A *selfish* agent first prefers strategy profiles with higher utility.
   - For two strategy profiles of the same utility, the agent prefers the one which contains his strategy of higher normative status.
4. - A *social* agent first prefers strategy profiles which contains his strategy of higher normative status.
   - For two strategy profiles of the same normative status, it prefers strategy profiles with higher utility.
5. - A *negatively impartial* agent first classifies strategies into negatively permitted category and prohibited category.
   - Then it ranks its strategies using utility within these two categories.
6. - A *positively impartial* agent first classifies strategies into positively permitted category and not positively permitted category.
   - Then it ranks its strategies using utility within these two categories.

We studied some complexity issues related to normative reasoning/status and agents' preference change. If no restriction is imposed, then for two strategy profiles s and s', deciding whether s is better than s' is in Δ_3^p. By restricting to strict Horn formulas, the complexity becomes P-complete. By restricting to literals, the complexity is even lower.

9.2 Future work

There are two modules in this thesis: norm generation in games and norm-based deontic logic. Both modules leave many open problems worthy of studying. In this final section of this thesis, we first propose future work from a broad perspective, then we give some more concrete problems.

In the first module, we created norms by computing solutions of games and studied the emergence of norms by running simulations of repeated games. In the sub-module of norm

creation, we studied the creation of 5 types of norms. Each of these types corresponds to a meaningful measurement of social welfare. This result suggests some connections between norm creation and the theory of social choice. In computational social choice, especially in multiagent resource allocation (Chevaleyre et al., 2006), the egalitarian and the elitist social welfare are both representatives of the family of k-rank dictator social welfare. In the future we will explore the creation of other types of norms that maximize some interesting social welfare studied in social choice theory such as k-rank dictator social welfare.

We are also interested in investigating the limitations and alternatives of our norm creation framework. In our framework, signals play an important role in the process of creating norms from correlated equilibrium. Those signals are used to indicate calssical probability distribution over strategy profiles. Therefore they are classical signals in contrast to quantum signals. In the literature of quantum game theory, Huberman and Hogg (2003) and La Mura (2005) show that in some games, if the signals the agents receive are entangled in an intrinsically quantum-mechanical fashion, then by following the quantum signals, the sum of the expected utility of all agents is higher than the sum of expected utility in every utilitarian correlated equilibrium. Those results reveal one limitation of our framework. We leave the exploration of the power of quantum signals in norm creation as future work.

Dodis et al. (2000) raise the question whether there is a mechanism that eliminates the need for the mediator (norm creator) to implement correlated equilibrium yet allows the agents to maintain the high payoffs offered by mediator-assisted strategies. They partially solved this problem by providing a cryptographic protocol to the two agent correlated element selection problem. Their results showed that cryptographic protocols can be viewed as alternatives to norms in coordinating agent's behavior in games. A comparison between social norms and cryptographic protocols are worthy of deeper investigation.

In the sub-module of norm emergence, we study norm emergence in the lattice and small world network using the learning rule imitate-the-best. In the future we are interested in studying how different network topologies affect the emergence of norms. If we pay attention to the topology of real networks, we will find out that most of them have a very particular topology: they are complex networks (Adamic, 1999) with non-trivial wiring schemes. The Internet, is among the most prominent complex networks found in the real world. Complex networks are well characterized by some special properties, such as the connectivity distribution (either exponential or power-law) or the small-world property (Amaral et al., 2000). Our long-term objective is to use multi-agent learning to study the emergence of norms in various interesting games in complex networks.

Now we turn to the second module. How is deontic logic possible on a positivistic philosophy of norms? Makinson (1999) considers this question the 'fundamental problem of deontic logic', and

called to reconstruct deontic logic as a logic of reasoning about norms. Norm-based deontic logic offers a solution to this problem. In the past two decades, logicians, philosophers and computer scientists have made significant progresses on norm-based deontic logic. They have established semantics and proof theory, proved completeness theorems, solved deontic paradoxes and find applications in different fields such as artificial intelligence, information security and legal theory.

This thesis contributes some computational results to norm-based deontic logic and extends its application to machine ethics. We believe norm-based deontic logic is an interesting topic which is worthy of further development in both theory and application. On the theoretical side, one limitation of current norm-based deontic logic is that they are all based on propositional logic. Therefore the expressive power of norm-based deontic logic needs to be increased. The need of increasing expressive power is also raised by the application of deontic logic to information security.

Several authors have used deontic logic to specify security policies (Glasgow et al., 1992; Jones and Sergot, 1992; Demolombe and Jones, 1996; Cuppens-Boulahia and Cuppens, 2008; Cuppens et al., 2013). These authors outlined the main features of this formalism to provide a flexible and expressive language for specifying security policies. Deontic logic has been a useful tool in the specification and reasoning of security policies because key notions in security such as permission, authorization, prohibition and obligation are exactly the subjects of deontic logic. To apply norm-based deontic logic to information security, we have to find a balance between expressive power and computational complexity. On the one hand, propositional logic is too weak to express crucial notions in security policy such as authority, action and agent. On the other hand, we have proved that the complexity of norm-based deontic logic is already intractable even if the base logic is propositional logic. To solve this problem we need to build norm-based deontic logic on top of an expressive logic, meanwhile use only a fragment of such logic to keep the complexity tractable. After this is done, we can further proceed to implement norm-based deontic logic using some programming languages.

Having outlined the future work for the long term, now we discuss more concrete problems which can possibly be solved in near future.

9.2.1 On norm creation

k-rank dictator

In the literature of multiagent resource allocation (Chevaleyre et al., 2006), the egalitarian and the elitist social welfare are both representatives of the family of k-rank dictator social welfare. Given a game and a correlated equilibrium, let $EU(\tau) = (EU_1(\tau), \ldots, EU_n(\tau))$ be the expected utility

vector under τ. The ordered expected utility vector $EU(\tau)^*$ is defined as the vector we obtain when we rearrange the elements of $EU(\tau)$ in increasing order. The k-rank dictator social welfare for a natural number k is measured by the k-poorest agent:

$$SW_k(\tau) = EU(\tau)^*_k$$

Here $EU(\tau)^*_k$ means the kth element of the vector $EU(\tau)$. For $k = 1$, we obtain the egalitarian social welfare. For $k = n$ we obtain the elitist social welfare. A special case of particular interest is the median rank dictator social welfare which is defined as $SW_k(\tau)$ with $k = \frac{n}{2}$ in case n is even and $k = \frac{n+1}{2}$ in case n is odd. Indeed, for certain applications the individual level of welfare on an agent that does at least as well as half of the agents in the system but not better than the other half may be considered as suitable indicator for overall system performance. Whether k-rank dictator norms can be created efficiently is left as future work.

VCG mechanism and norm creation

The fact that norms can be created efficiently in our framework is a positive result. But in situations where the system designer are required to collect the utilities function of each agent, this positive result has a negative side in the sense that agents can manipulate the procedure of norm creation by misrepresenting their true utility function. To minimize the possibility of manipulation, we use a variant of the Vickrey-Clarke-Groves mechanism from the theory of mechanism design (Nisan and Ronen, 2001) to elicit the true utility function of agents.

Our basic idea is summarized as follows. Given a game $G = (Agent, \Gamma_1, \ldots \Gamma_n, u_1, \ldots, u_n)$, let the contraction of this game excluding i be $G^{-i} = (Agent - \{i\}, \Gamma_1, \ldots, \Gamma_{i-1}, \Gamma_{i+1}, \ldots, \Gamma_n, u'_1, \ldots, u'_{i-1}, u'_{i+1}, \ldots, u'_n)$, Where $u'_j(\alpha_1, \ldots, \alpha_{j-1}, \alpha_{j+1}, \ldots, \alpha_n) = (u_j(\alpha_1, \ldots, \alpha^1_i, \ldots, \alpha_n) + \ldots + u_j(\alpha_1, \ldots, \alpha^k_i, \ldots, \alpha_n))/k$, if $|\Gamma_i| = k$. Given a game $(Agent, \Gamma_1, \ldots \Gamma_n, u_1, \ldots, u_n)$, an agent i and its reported utility function $\hat{u}_i$,

1. Compute a utilitarian correlated equilibrium $\hat{\tau}$ for the game $(Agent, \Gamma_1, \ldots \Gamma_n, u_1, \ldots, \hat{u}_i, \ldots, u_n)$.
2. Compute a utilitarian correlated equilibrium τ^{-i} for the game G^{-i}.
3. Extend τ^{-i} to $\widehat{\tau^{-i}}$, which is a probability distribution over $\Gamma_1 \times \ldots \times \Gamma_n$ such that for all $\alpha_i \in \Gamma_i$, $\widehat{\tau^{-i}}(\alpha_1, \ldots, \alpha_i, \ldots \alpha_n) = \frac{1}{|\Gamma_i|}\tau^{-i}(\alpha_1, \ldots, \alpha_{j-1}, \alpha_{j+1}, \ldots, \alpha_n)$.
4. Let the tax imposed to i for reporting $\hat{u}_i$ be $tax_i(\hat{u}_i) = EU_i(\hat{\tau}) - EU_i(\widehat{\tau^{-i}})$

Suppose u_i is the true utility function of agent i, then the incentive for i to represent a fake utility function $\hat{u}_i$ is $(EU_i(\hat{\tau}) - tax_i(\hat{u}_i)) - (EU_i(\tau) - tax_i(u_i)) = EU_i(\widehat{\tau^{-i}}) - EU_i(\widehat{\tau^{-i}}) = 0$. This result

shows that given the taxation rule explained above, there is no incentive for an agent to report a fake utility function. A detailed formation of such mechanisms for eliciting the true utility function is left as future work.

9.2.2 On norm emergence

In the future it is worthy of studying the problem of norm emergence in Ali Baba and the Thief where agents are situated in a social network different from lattice or the small world model and use some other learning rules. In particular, we are interested in scale-free networks and fictitious play.

Scale-free network

A social network's diameter is defined as the largest distance between any two nodes in the network. The diameter represents the largest path within the network and characterizes the compactness and connectivity of the network. A network with a small diameter is very well-connected, and thus the average path length of the network will be small. On the other hand, a network with a large diameter will be very sparsely-connected, and the average path length can be large.

Scale-free networks have the structural property that the connectivity of the network follows a power law distribution. This means that the network has a small number of nodes which have a very high connectivity. However, most of the nodes in the network are sparsely-connected. Two examples that have been studied extensively are the collaboration of movie actors in films and the co-authorship by mathematicians of papers. The diameter of a scale-free network can be approximated as the largest distance among hubs plus 2 since this is the distance between a neighbor of one hub of the longest path and a neighbor of the other hub of the longest path.

Fictitious play

Fictitious play is one of the most important model of learning in games (Fudenberg and Levine, 1998). In this model agents assume that their opponents are playing a fixed strategy. The agents use their past experiences to build a model of the opponent's strategy and use this model to choose their own action.

Fictitious play uses a simple form of learning where an agent remembers everything the other agent has done and uses this information to build a probability distribution for the other agent's expected strategy. Formally, for the two agent case we say that agent i maintains a weight function $k_i : S_j \rightarrow \mathcal{R}^+$. The weight function changes over time as the agent learns. The weight function at

time t is represented by k_i^t. It maintains a count of how many times each strategy has been played by player j. When at time $t-1$ opponent j plays strategy s_j^{t-1} then i updates its weight function with

$$k_i^t(s_j) = k_i^{t-1}(s_j) + \begin{cases} 1 & \text{if } s_j^{t-1} = s_j, \\ 0 & \text{if } s_j^{t-1} \neq s_j. \end{cases} \tag{9.1}$$

Using this weight function, agent i can assign a probability to j playing any of its $s_j \in S_j$ strategies with

$$\mathbf{Pr}_i^t[s_j] = \frac{k_i^t(s_j)}{\sum_{\tilde{s}_j \in S_j} k_i^t(\tilde{s}_j)}. \tag{9.2}$$

That is, i assumes j will pick its action proportional to the values in $k_i(s_j)$. Player i then determines the strategy that will give it the highest expected utility given that j will play each of its $s_j \in S_j$ with probability $\mathbf{Pr}_i^t[s_j]$. In other words, i determines its best response to a probability distribution over j's possible strategies.

9.2.3 Axiomatics of norms

Abstract input/output logic

A frequent belief about input/output logic is that it presupposes classical logic (Parent et al., 2014). In other words, since all works in input/output logic before Parent et al. (2014) used propositional logic as base logic, it was believed that input/output logic cannot be wrapped around a logic having a higher expressivity.

Parent et al. (2014) and Sun (2015c) provide evidence that this conjecture is not true. They take intuitionistic logic and STIT logic (Belnap et al., 2001) respectively as the base logic and built input/output logic on top of it. Although Parent et al. (2014) and Sun (2015c) prove that propositional logic is not the only available choice for the object logic of input/output systems, their findings still need to be generalized before being considered acceptable. Each of them considers a *single* base logic.

Carnielli et al. (2013) moves forward on the same path of Parent et al. (2014) and Sun (2015c) by building input/output logic on top of an arbitrary Tarskian logic. Carnielli *et al* take an arbitrary Tarskian logic as base logic and build abstract input/output logic on top of it. Tarskian logic is the most general logic we can have: intuitionistic logic, STIT logic, first order logic, deontic logic, and so forth are only special instantiations of Tarskian logic. A Tarskian logic is a pair $\mathfrak{L} = (L, \vdash)$, where $\vdash$ is termed as "Tarskian consequence relation" and it is defined as follows.

Definition 9.1. *[Tarskian consequence Wójcicki (1988)] A Tarskian consequence relation over a language L is a relation $\vdash$, included in or equal to $2^L \times L$, that satisfies the following properties:*

1. ***reflexivity**: For all $A \subseteq L$, $x \in L$, if $x \in A$ then $A \vdash x$.*
2. ***cumulative transitivity**: If $A \vdash x$ and for all $a \in A$, $B \vdash a$ then $B \vdash x$.*
3. ***monotony**: If $A \vdash x$ then $A \cup B \vdash x$.*

A Tarskian consequence relation is *compact* if it satisfies the following:

(1) if $A \vdash x$ then $A_0 \vdash x$ for some finite $A_0 \subseteq A$.

We say that a logic $\mathfrak{L}$ has true constant and false constant if L contains formulas $\top$ and $\bot$ satisfying $x \vdash \top$ and $\bot \vdash x$ for all $x \in L$. Carnielli et al. (2013) say that a logic $\mathfrak{L}$ has conjunction and disjunction if $\mathfrak{L}$ has binary connectives $\wedge$ and $\vee$ satisfying usual classical properties. It is not explicitly stated what exactly those properties are in Carnielli et al. (2013). Here we explicitly require $\wedge$ and $\vee$ to satisfy the following properties:

(2)
a. $\{x \wedge y\} \vdash x$
b. $\{x \wedge y\} \vdash y$
c. $\{x, y\} \vdash x \wedge y$
d. $\{x, y\} \vdash y \wedge x$
e. $\{x\} \vdash x \vee y$
f. $\{y\} \vdash x \vee y$

Note that we do not require $\wedge$ and $\vee$ to satisfy distributivity, which means here we do not exclude quantum logic which usually falsifies distributivity. Let $Cn(\bullet)$ be the consequence operator of an arbitrary compact Tarskian logic with true and false constant as well as conjunction and disjunction. Carnielli et al. (2013) introduce the proof theory and semantics of *abstract simple-mined reusable input/output logic* (out_3^T) but do not prove the completeness. Here we give an alternative semantics for out_3^T:

Definition 9.2. *Let $B_A^O = \bigcup_{i=0}^{\infty} B_{A,i}^O$, where $B_{A,0}^O = Cn(A)$, $B_{A,i+1}^O = Cn(A \cup O(B_{A,i}^O))$,*

$$out_3^T(O, A) = Cn(O(B_A^O)).$$

Let $deriv_3^T(O)$ be the proof system of abstract simple-mined reusable input/output logic proposed in Carnielli et al. (2013). We make the following conjecture and leave the proof of this conjecture as future work.

Conjecture 9.1. *$(a, x) \in deriv_3^T(O)$ iff $x \in out_3^T(O, a)$.*

Parameterized logic programming

Gonçalves and Alferes (2012) show that propositional input/output logic can be embedded into parametrized logic programming. Parametrized logic programming (Gonçalves and Alferes, 2010) was introduced as an extension of answer set programming (Gelfond and Lifschitz, 1988) such that complex formulas are allowed to appear in the head and body of a rule. The main idea is to fix a Tarskian logic $\mathfrak{L}$ as parameter logic, and build up logic programs using formulas of L.

The formulas of L are called (parametrized) atoms and a (parametrized) literal is either a parametrized atom x or its negation *not* x, where as usual *not* denotes negation as failure.

Definition 9.3. *(Gonçalves and Alferes, 2010) A normal $\mathfrak{L}$-parametrized logic program is a set of rules*

$$x \leftarrow a_1, \ldots, a_m, not\ b_1, \ldots, not\ b_n$$

where $a_1, \ldots, a_m, b_1, \ldots, b_n, x \in L$. A definite $\mathfrak{L}$-parametrized logic program is a set of rules without negation as failure, i.e. of the form $x \leftarrow a_1, \ldots, a_m$ where $a_1, \ldots, a_m, x \in L$.

A theory of $\mathfrak{L}$ is a set of formulas that is closed under the inference relation $\vdash$. That is, $A \subseteq L$ is a theory of $\mathfrak{L}$ if $A = Cn(A)$. An *interpretation* of an $\mathfrak{L}$-parametrized logic program is a theory of $\mathfrak{L}$. If I and J are two interpretations then we say that $I \preceq J$ if $I \subseteq J$.

Definition 9.4 (Gonçalves and Alferes (2010))**.** *A interpretation I satisfies a rule $x \leftarrow a_1, \ldots, a_m, not\ b_1, \ldots, not\ b_n$ if $x \in I$ whenever $a_i \in I$ for all $i \in \{1, \ldots, m\}$ and $b_j \notin I$ for every $j \in \{1, \ldots, n\}$.*

Definition 9.5 (Gonçalves and Alferes (2010))**.** *An interpretation is a model of an $\mathfrak{L}$-parametrized logic program $\mathcal{P}$ if it satisfies every rule of $\mathcal{P}$. We denote by $Mod^{\mathfrak{L}}(\mathcal{P})$ the set of models of $\mathcal{P}$.*

Definition 9.6 (Gonçalves and Alferes (2010))**.** *The stable model semantics of a definite $\mathfrak{L}$-parametrized logic program $\mathcal{P}$, denoted as $S^{\mathfrak{L}}_{\mathcal{P}}$, is its least model with respect to the $\preceq$ relation. That is , I is the stable model semantics of $\mathcal{P}$ if I is a model of $\mathcal{P}$ and for all J which is a model of $\mathcal{P}$, $I \preceq J$.*

The above notion is well-defined because it is proved in Gonçalves and Alferes (2010) that every definite $\mathfrak{L}$-parametrized logic program has a unique least model.

We define a modal expansion of a Tarskian logic $\mathfrak{L} = (L, \vdash)$ to be another logic $\mathfrak{L}^{\Box} = (L^{\Box}, \vdash^{+})$ where $\Box$ is a distinct modal operator that does not appear in L such that:

- $L^{\Box} = L \cup \{\Box x : x \in L\}$.
- $\vdash^{+} \subseteq 2^{L^{\Box}} \times L^{\Box}$ is a Tarksian consequence relation which satisfies the following:
  1. $\vdash \subseteq \vdash^{+}$.
  2. If $x \vdash y$ then $\Box x \vdash^{+} \Box y$.

3. $\Box x \wedge \Box y \vdash^{+} \Box(x \wedge y)$.

We conjecture that abstract input/output logic based on $\mathfrak{L}$ can be embedded into $\mathfrak{L}^{\Box}$-parametrized logic program.

Conjecture 9.2. *Given a Tarskian logic* $\mathfrak{L} = (L, \vdash)$, $O \subseteq L \times L$, $A \subseteq L$, $\mathfrak{L}^{\Box} = (L^{\Box}, \vdash^{+})$ *a modal expansion of* $\mathfrak{L}$, *let* $\mathcal{P}_3 = \{x \leftarrow a : (a, x) \in O\} \cup \{\Box x \leftarrow a : (a, x) \in O\} \cup \{a \leftarrow : a \in A\}$. *Then* $out_3^T(O, A) \subseteq \{x : \Box x \in S_{\mathcal{P}_3}^{\mathfrak{L}^{\Box}}\}$.

9.2.4 Algebra of norms

We use more advanced logic and algebra to relate input/output logic and joining-systems and use them to give a more accurate formal characterization of normative systems. One recent result from algebraic logic shed lights on this problem: algebraic hybrid logic (Litak, 2006). It is worth investigating whether similar results can be obtained if we use hybrid logic and its algebraic companion to replace the logic and algebra we used in this thesis.

9.2.5 On the complexity of normative reasoning

The following directions of future work are worthy of studying:

1. What is the exact complexity of out_0, out_3, out_5, out_6, out_2^p, out_3^p and out_4^p?
2. What is the exact complexity of deontic default logic?
3. Other notions of permission have been studied in Stolpe (2010c). What is the complexity of Stolpe's permissive input/output logic?

9.2.6 On ethical agents

One of the most interesting avenues for future research is to integrate other mental attitudes such as belief and intention into the architecture of agents. More normative positions and types should be studied in the future. For example we can identify x as *weak mandatory* if $x \in \bigcup\{out_i(O', A) : O' \in preffamily_i(O^{\geq}, A, C)\}$ and let a *weak moral* agents first classify his strategies into two categories: weak mandatory and not weak mandatory, then rank his strategies using utility within these two categories. Another topic worthy of investigating is to extend the expressivity of Boolean games. The Boolean games we used only allow to express binary preference. Extension of Boolean games to express more complicated preference has been developed by Bonzon et al. (2009). In the future we will study solution concepts of normative Boolean games with complicated preference. In

Boolean games, an action is simply a truth assignment. Extensions are made in dynamic logic of propositional assignments (Herzig et al., 2011; Balbiani et al., 2013) by allowing action to be combined using action operators such as sequence, choice and iteration. In the future we will investigate how to incorporate complex actions into our framework.

9.2.7 Other related directions

On norm change

Theories of normative system change are studied in Boella et al. (2009b) and Stolpe (2010a). They generalise classical revision theory of the AGM brand to sets of norms. This is achieved in Stolpe (2010a) by substituting input/output logic for classical logic and tracking the changes. Operations of derogation and amendment – analogues of contraction and revision – are defined and characterized, and the precise relationship between contraction and derogation, on the one hand, and derogation and amendment on the other, is established. The complexity of norm change in their framework is worthy of studying in the future.

Graded causal theory and prioritized causal calculas

There is a close relation between deontic logic and causal theory. Makinson (1993) shows that conditional obligation in deontic logic and counterfactual conditionals, an important source of causal theory, are actually two sides of the same coin. Bochman (2004) uses a variant of input/output logic to develop causal calculus.

One important problem in causal theory is the problem of isomorphism, the problem that one collection of data supports many different causal patterns. By incorporating normality into an account of actual causation, Halpern and Hitchcock (2015) develop graded causal theory to solve the problem of isomorphism. It seems that we can use a different approach to develop something similar to graded causal theory. Bochman's causal calculus is a variant of non-prioritized input/output logic. By using prioritized input/output logic we can develop a prioritized causal calculas. Bochman and Lifschitz (2015) prove that causal calculus is to a large extend the same as Pearl's causal theory (Pearl, 2000). To what extend will the prioritized causal calculus be the same as the graded causal theory? How can these two formal frameworks benefit from each other?

Procedural norms and social software

Procedural norms guide agents to achieve justice and fairness. Up to now, deontic logicians pay little attention to procedural norms (Boella and van der Torre, 2008). Some formal study of

procedural norms can be found in the literature of social software (van Eijck and Verbrugge, 2015). A systematical study of the syntax, semantics and complexity of procedural norms is worthy of further development.

Bibliography

Adamic, L. (1999). The small world web. In Abiteboul, S. and Vercoustre, A., editors, *Research and Advanced Technology for Digital Libraries, Third European Conference, ECDL'99,* Paris, France, September 22-24, 1999, pages 443–452, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 157

Ågotnes, T., van der Hoek, W., Rodríguez-Aguilar, J. A., Sierra, C., and Wooldridge, M. (2007). On the logic of normative systems. In Sangal et al. (2007), pages 1175–1180. 132

Ågotnes, T., van der Hoek, W., and Wooldridge, M. (2012). Conservative social laws. In Raedt, L. D., Bessière, C., Dubois, D., Doherty, P., Frasconi, P., Heintz, F., and Lucas, P. J. F., editors, *ECAI 2012 - 20th European Conference on Artificial Intelligence,* Montpellier, France, August 27-31 , 2012, pages 49–54, Amsterdam, The Netherlands. IOS Press. 4, 35

Ågotnes, T. and Wooldridge, M. (2010). Optimal social laws. In van der Hoek, W., Kaminka, G. A., Lespérance, Y., Luck, M., and Sen, S., editors, *9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2010),* Toronto, Canada, May 10-14, 2010, Volume 1-3, pages 667–674, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 4, 35

Airiau, S., Sen, S., and Villatoro, D. (2014). Emergence of conventions through social learning - heterogeneous learners in complex networks. *Autonomous Agents and Multi-Agent Systems*, 28(5):779–804. 4

Alchourrón, C. and Bulygin, E. (1981). The expressive conception of norms. In Hilpinen, R., editor, *New Studies in Deontic Logic: Norms, Actions, and the Foundations of Ethics*, pages 95–124, Dordrecht, the Netherlands. Springer Netherlands. 6, 24

Aldewereld, H. (2009). Autonomy vs. conformity: An institutional perspective on norms and protocols. *Knowledge Engineering Review*, 24(4):410–411. 7

Aldewereld, H., Dignum, F., García-Camino, A., Noriega, P., Rodríguez-Aguilar, J. A., and Sierra, C. (2006). Operationalisation of norms for usage in electronic institutions. In Nakashima, H., Wellman, M. P., Weiss, G., and Stone, P., editors, *5th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2006)*, Hakodate, Japan, May 8-12, 2006, pages 223–225, New York, USA. ACM Press. 4

Alechina, N., Dastani, M., and Logan, B. (2013). Reasoning about normative update. In Rossi, F., editor, *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, Beijing, China, August 3-9, 2013, San Francisco, USA. AAAI Press. 132, 133

Alevras, D. and Padberg, M. W. (2001). *Linear Optimization and Extensions: Problems and Solutions*. Springer-Verlag Berlin Heidelberg, Berlin and Heidelberg, Germany. 42

Alexander, J. M. (2007). *The Structural Evolution of Morality*. Cambridge University Press, New York, USA. 7, 10, 13, 48, 51, 60

Alexander, J. M. (2009). Evolutionary game theory. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Fall 2009 edition. 53

Amaral, L. A. N., Scala, A., Barthélémy, M., and Stanley, H. E. (2000). Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):11149–11152. 157

Anderson, M. and Anderson, S. L. (2011). *Machine Ethics*. Cambridge University Press, New York, USA. 11

Anderson, M., Anderson, S. L., and Armen, C. (2006). Medethex: A prototype medical ethics advisor. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, July 16-20, 2006, Boston, Massachusetts, pages 1759–1765, San Francisco, USA. AAAI Press. 11

Andrighetto, G., Governatori, G., Noriega, P., and van der Torre, L., editors (2013). *Normative Multi-Agent Systems*, volume 4 of *Dagstuhl Follow-Ups*. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany. 3

Antoniou, G., Dimaresis, N., and Governatori, G. (2007). A system for modal and deontic defeasible reasoning. In Orgun, M. A. and Thornton, J., editors, *AI 2007: Advances in Artificial Intelligence, 20th Australian Joint Conference on Artificial Intelligence*, Gold Coast, Australia, December 2-6, 2007, pages 609–613, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 6

Antoniou, G., Dimaresis, N., and Governatori, G. (2009). A modal and deontic defeasible reasoning system for modelling policies and multi-agent systems. *Expert Systems With Applications*, 36(2):4125–4134. 6

Åqvist, L. (1986). Some results on dyadic deontic logic and the logic of preference. *Synthese*, 66(1):95–110. 6

Arendt, H. (1958). *The Origins of Totalitarianism*. The World Publishing Company, Cleveland and New York, USA. 96

Armstrong, J. A. (1961). *The Politics of Totalitarianism*. Random House, New York, USA. 96

Arneson, R. (2013). Egalitarianism. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Summer 2013 edition. 36

Arora, S. and Barak, B. (2009). *Computational Complexity: A Modern Approach*. Cambridge University Press, New York, USA. 102, 105

Aumann, R. (1974). Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1):67–96. 8, 37

Aumann, R. and Brandenburger, A. (1995). Epistemic conditions for Nash equilibrium. *Econometrica*, 63(5):1161–1180. 9

Aumann, R. J. (1987). Correlated equilibrium as an expression of bayesian rationality. *Econometrica: Journal of the Econometric Society*, 55(1):1–18. 9

Axelrod, R. (1986). An evolutionary approach to norms. *American Political Science Review*, 80(4):1095–1111. 4, 7, 61

Axelrod, R. (1997). *The complexity of cooperation: Agemt-based models of competition and collaboration*. Princeton University Press, New Jersey, USA. 3

Baader, F. and Hollunder, B. (1995). Priorities on defaults with prerequisites, and their application in treating specificity in terminological default logic. *Journal of Automated Reasoning*, 15(1):41–68. 10, 25, 123

Balbiani, P., Herzig, A., and Troquard, N. (2013). Dynamic logic of propositional assignments: A well-behaved variant of PDL. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013*, New Orleans, LA, USA, June 25-28, 2013, pages 143–152, Washington, DC, USA. IEEE Computer Society. 165

Balke, T., Dignum, F., van Riemsdijk, M. B., and Chopra, A. K., editors (2014). *Coordination, Organizations, Institutions, and Norms in Agent Systems IX - COIN 2013 International Workshops, COIN@AAMAS*, St. Paul, MN, USA, May 6, 2013, COIN@PRIMA, Dunedin, New Zealand, December 3, 2013, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 175, 178, 179, 186

Beirlaen, M., Straßer, C., and Meheus, J. (2013). An inconsistency-adaptive deontic logic for normative conflicts. *Journal of Philosophical Logic*, 42(2):285–315. 6

Belnap, N., Perloff, M., and Xu, M. (2001). *Facing the Future: Agents and Choice in Our Indeterminist World*. Oxford, New York, USA. 161

Bendor, J. and Swistak, P. (2001). The evolution of norms. *The American Journal of Sociology*, 106(6):1493–1545. 3

Beyersdorff, O., Meier, A., Thomas, M., and Vollmer, H. (2009). The complexity of propositional implication. *Information Processing Letters*, 109(18):1071–1077. 120

Bicchieri, C. (2006). *The Grammar of Society: The Nature and Dynamics of Social Norms*. Cambridge University Press, New York, USA. 7, 48

Bicchieri, C., Duffy, J., and Tolle, G. (2004). Trust among strangers. *Philosophy of Science*, 71(3):286–319. 62

Bikakis, A. and Zheng, X., editors (2015). *Multi-disciplinary Trends in Artificial Intelligence - 9th International Workshop*, Switzerland. Springer International Publishing. 185, 186

Binmore, K. (2005). *Natural Justice*. Oxford University Press, New York, USA. 7, 48

Blackburn, P., De Rijke, M., and Venema, Y. (2001). *Modal logic*. Cambridge University Press, Cambridge, UK. 5, 86, 105, 106

Bochman, A. (2004). A causal approach to nonmonotonic reasoning. *Artificial intelligence*, 160(1-2):105–143. 165

Bochman, A. and Lifschitz, V. (2015). Pearl's causality in a logical setting. In Bonet, B. and Koenig, S., editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, January 25-30, 2015, Austin, Texas, USA, pages 1446–1452, San Francisco, USA. AAAI Press. 165

Boella, G., Broersen, J., and van der Torre, L. (2008a). Reasoning about constitutive norms, counts-as conditionals, institutions, deadlines and violations. In Bui, T. D., Ho, T. V., and Ha, Q.-T., editors, *Intelligent Agents and Multi-Agent Systems, 11th Pacific Rim International Conference on*

Multi-Agents, PRIMA 2008, Hanoi, Vietnam, December 15-16, 2008, pages 86–97, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 81

Boella, G., Caire, P., and van der Torre, L. (2009a). Norm negotiation in online multi-player games. *Knowledge and Information Systems*, 18(2):137–156. 8

Boella, G., Caro, L. D., and Robaldo, L. (2013). Semantic relation extraction from legislative text using generalized syntactic dependencies and support vector machines. In Morgenstern, L., Stefaneas, P. S., Lévy, F., Wyner, A. Z., and Paschke, A., editors, *Theory, Practice, and Applications of Rules on the Web - 7th International Symposium, RuleML 2013,* Seattle, WA, USA, July 11-13, 2013, pages 218–225, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 101

Boella, G., Di Caro, L., Ruggeri, A., and Robaldo, L. (2014). Learning from syntax generalizations for automatic semantic annotation. *Journal of Intelligent Information Systems*, 43(2):231–246. 101

Boella, G., Pigozzi, G., and van der Torre, L. (2009b). Normative framework for normative system change. In Sierra, C., Castelfranchi, C., Decker, K. S., and Sichman, J. S., editors, *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009),* Budapest, Hungary, May 10-15, 2009, Volume 1, pages 169–176, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 165

Boella, G. and van der Torre, L. (2003a). Attributing mental attitudes to normative systems. In *The Second International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2003,* July 14-18, 2003, Melbourne, Victoria, Australia, pages 942–943, New York, USA. ACM Press. 48

Boella, G. and van der Torre, L. (2003b). Permissions and obligations in hierarchical normative systems. In Zeleznikow and Sartor (2003), pages 109–118. 22

Boella, G. and van der Torre, L. (2003c). Rational norm creation: Attributing mental attitudes to normative systems, part 2. In Zeleznikow and Sartor (2003), pages 81–82. 48

Boella, G. and van der Torre, L. (2005a). Enforceable social laws. In Dignum, F., Dignum, V., Koenig, S., Kraus, S., Singh, M. P., and Wooldridge, M., editors, *4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2005),* July 25-29, 2005, Utrecht, The Netherlands, pages 682–689, New York, USA. ACM Press. 62

Boella, G. and van der Torre, L. (2005b). The evolution of artificial social systems. In Kaelbling, L. P. and Saffiotti, A., editors, *IJCAI-05, Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence,* Edinburgh, Scotland, UK, July 30-August 5, 2005, pages 1655–1556, San Francisco, USA. Morgan Kaufmann Publishers Inc. 62

Boella, G. and van der Torre, L. (2006). A logical architecture of a normative system. In Goble, L. and Meyer, J.-J. C., editors, *Deontic Logic and Artificial Normative Systems, 8th International Workshop on Deontic Logic in Computer Science, DEON 2006,* Utrecht, The Netherlands, July 12-14, 2006, pages 24–35, Utrecht, The Netherlands. Springer Netherlands. 4, 66, 72, 76, 77

Boella, G. and van der Torre, L. (2007a). A game-theoretic approach to normative multi-agent systems. In Boella et al. (2007). 7

Boella, G. and van der Torre, L. (2007b). Norm negotiation in multiagent systems. *International Journal of Cooperative Information Systems*, 16(1):97–122. 8

Boella, G. and van der Torre, L. (2008). Substantive and procedural norms in normative multiagent systems. *Journal of Applied Logic*, 6(2):152 – 171. 165

Boella, G., van der Torre, L., and Verhagen, H. (2006). Introduction to normative multiagent systems. *Computation and Mathematical Organizational Theory, special issue on normative multiagent systems*, 12(2-3):71–79. 3

Boella, G., van der Torre, L., and Verhagen, H., editors (2007). *Normative Multi-agent Systems, 18.03. - 23.03.2007*, volume 07122 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany. 172, 176

Boella, G., van der Torre, L., and Verhagen, H. (2008b). Introduction to the special issue on normative multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 17(1):1–10. 3, 98, 132

Böhler, E., Creignou, N., Reith, S., and Vollmer, H. (2003). Playing with boolean blocks. Part I: Post's lattice with applications to complexity theory. *ACM SIGACT News*, 34(4):38–52. 119

Boman, M. (1999). Norms in artificial decision making. *Artificial Intelligence and Law*, 7(1):17–35. 3

Bonzon, E., Lagasquie-Schiex, M., Lang, J., and Zanuttini, B. (2006). Boolean games revisited. In Brewka, G., Coradeschi, S., Perini, A., and Traverso, P., editors, *ECAI 2006, 17th European Conference on Artificial Intelligence,* August 29 - September 1, 2006, Riva del Garda, Italy, pages 265–269, Amsterdam, The Netherlands. IOS Press. 134

Bonzon, E., Lagasquie-Schiex, M., Lang, J., and Zanuttini, B. (2009). Compact preference representation and boolean games. *Autonomous Agents and Multi-Agent Systems*, 18(1):1–35. 14, 132, 133, 164

Boutilier, C. (1994). Toward a logic for qualitative decision theory. In Doyle, J., Sandewall, E., and Torasso, P., editors, *Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning (KR'94)*. Bonn, Germany, May 24-27, 1994, pages 75–86, San Francisco, USA. Morgan Kaufmann Publishers Inc. 6

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge, UK. 45

Brass, S. (1993). *Deduction with supernormal defaults*, pages 153–174. Springer Berlin Heidelberg, Berlin and Heidelberg, Germany. 22

Brewka, G. (1994). Adding priorities and specificity to default logic. In MacNish, C., Pearce, D., and Pereira, L. M., editors, *Logics in Artificial Intelligence, European Workshop, JELIA '94*, York, UK, September 5-8, 1994, pages 247–260, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 10, 25, 123

Bringsjord, S., Arkoudas, K., and Bello, P. (2006). Toward a general logicist methodology for engineering ethically correct robots. *IEEE Intelligent Systems*, 21(4):38–44. 11

Broersen, J. (2003). *Modal Action Logics for Reasoning About Reactive Systems*. PhD thesis, Utrecht University. 6

Broersen, J., Cranefield, S., Elrakaiby, Y., Gabbay, D. M., Grossi, D., Lorini, E., Parent, X., van der Torre, L., Tummolini, L., Turrini, P., and Schwarzentruber, F. (2013). Normative reasoning and consequence. In Andrighetto, G., Governatori, G., Noriega, P., and van der Torre, L., editors, *Normative Multi-Agent Systems*, pages 33–70. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany. 7

Broersen, J., Dastani, M., and van der Torre, L. (2005). Beliefs, obligations, intentions, and desires as components in an agent architecture. *International Journal of Intelligent Systems*, 20(9):893–919. 12

Cariani, F., Grossi, D., Meheus, J., and Parent, X., editors (2014). *Deontic Logic and Normative Systems - 12th International Conference*, Switzerland. Springer International Publishing. 182, 186

Carnielli, W., Coniglio, M. E., and van der Torre, L. (2013). *Input/output consequence relations: reasoning with intensional contexts*. Unpublished report. ftp://ftp.cle.unicamp.br/pub/Thematic-Consrel-FAPESP/Report-04-2009/[CCvT09].pdf. 161, 162

Castro, P. and Maibaum, T. (2009). Deontic action logic, atomic boolean algebras and fault-tolerance. *Journal of Applied Logic*, 7(4):441–466. 6

Chevaleyre, Y., Dunne, P. E., Endriss, U., Lang, J., Lemaître, M., Maudet, N., Padget, J. A., Phelps, S., Rodríguez-Aguilar, J. A., and Sousa, P. (2006). Issues in multiagent resource allocation. *Informatica*, 30(1):3–31. 38, 157, 158

Chisholm, R. (1963). Contrary-to-duty imperatives and deontic logic. *Analysis*, 24:33–36. 16

Conte, R. and Dellarocas, C., editors (2001). *Social Order in Multiagent Systems*. Springer US, New York, USA. 3

Conte, R., Falcone, R., and Sartor, G. (1999). Introduction: Agents and norms: How to fill the gap? *Artificial Intelligence and Law*, 7(1):1–15. 3

Cook, S. (1971). Characterizations of pushdown machines in terms of time-bounded computers. *Journal of the ACM*, 18(1):4–18. 104

Cuppens, F., Cuppens-Boulahia, N., and Elrakaiby, Y. (2013). Formal specification and management of security policies with collective group obligations. *Journal of Computer Security*, 21(1):149–190. 158

Cuppens-Boulahia, N. and Cuppens, F. (2008). Specifying intrusion detection and reaction policies: An application of deontic logic. In van der Meyden and van der Torre (2008), pages 65–80. 158

Dantsin, E., Eiter, T., Gottlob, G., and Voronkov, A. (2001). Complexity and expressive power of logic programming. *ACM Computing Surveys*, 33(3):374–425. 143

Daskalakis, C., Goldberg, P. W., and Papadimitriou, C. H. (2009). The complexity of computing a Nash equilibrium. *Communications of the ACM*, 52(2):89–97. 38

Delgado, J. (2002). Emergence of social conventions in complex networks. *Artificial Intelligence*, 141(1/2):171–185. 60

Demolombe, R. and Jones, A. (1996). Integrity constraints revisited. *Logic Journal of the IGPL*, 4(3):369–383. 158

Deng, B. (2015). Machine ethics: The robot's dilemma. *Nature*, (523):24–26. 11

Dignum, F. (1999). Autonomous agents with norms. *Artificial Intelligence and Law*, 7(1):69–79. 3

Dodis, Y., Halevi, S., and Rabin, T. (2000). A cryptographic solution to a game theoretic problem. In Bellare, M., editor, *Advances in Cryptology - CRYPTO 2000, 20th Annual International Cryptology Conference*, Santa Barbara, California, USA, August 20-24, 2000, pages 112–130, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 157

Dunne, P. E., van der Hoek, W., Kraus, S., and Wooldridge, M. (2008). Cooperative boolean games. In Padgham, L., Parkes, D. C., Müller, J. P., and Parsons, S., editors, *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008),* Estoril, Portugal, May 12-16, 2008, Volume 2, pages 1015–1022, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 14, 133

Epstein, J. (1998). Zones of cooperation in demographic prisoner's dilemma. *Complexity*, 4(2):36–48. 54

Epstein, J. (2001). Learning to be thoughtless: Social norms and individual computation. *Computational Economics*, 18(1):9–24. 3

Fehr, E. and Schmidt, K. M. (2003). Theories of fairness and reciprocity: Evidence and economic applications. In Dewatripont, M., Hansen, L. P., and Turnovsky, S. J., editors, *Advances in Economics and Econometrics*, volume 1, pages 208–257. Cambridge University Press, New York, USA. 132

Fonseca dos Santos Neto, B., Torres da Silva, V., and José Pereira de Lucena, C. (2012). An architectural model for autonomous normative agents. In de Barros, L. N., Finger, M., Pozo, A. T. R., Lugo, G. A. G., and Castilho, M. A., editors, *Advances in Artificial Intelligence - SBIA 2012 - 21th Brazilian Symposium on Artificial Intelligence,* Curitiba, Brazil, October 20-25, 2012, pages 152–161, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 4

Frantz, C., Purvis, M. K., Nowostawski, M., and Savarimuthu, B. T. R. (2013). Modelling institutions using dynamic deontics. In Balke et al. (2014), pages 211–233. 7

Fudenberg, D. and Levine, D. (1998). *The Theory of Learning in Games*. MIT Press, Cambridge, USA. 53, 160

Gabbay, D., Horty, J., Parent, X., van der Meyden, R., and van der Torre, L., editors (2013). *Handbook of Deontic Logic and Normative Systems*. College Publications, London, UK. 2, 4, 6, 18, 76, 177, 179, 182

Gabbay, D., Horty, J., Parent, X., van der Meyden, R., and van der Torre, L., editors (2016). *Handbook of Deontic Logic and Normative Systems*, volume 2. College Publications, London, UK. to appear. 2, 4, 6

Garg, V. (2015). *Introduction to Lattice Theory with Computer Science Applications*. Wiley, New Jersey, USA. 99

Gelfond, M. and Lifschitz, V. (1988). The stable model semantics for logic programming. In Kowalski, R. A. and Bowen, K. A., editors, *Logic Programming, Proceedings of the Fifth International Conference and Symposium*, Seattle, Washington, August 15-19, 1988, pages 1070–1080, Cambridge, USA. MIT Press. 163

Gert, B. (2005). *Morality: Its Nature and Justification*. Oxford University Press, New York, USA. 144

Gintis, H. (2010). Social norms as choreography. *Politics, Philosophy and Economics*, 9(3):251–264. 3, 8, 9, 36, 146

Gintis, H., Bowles, S., Boyd, R., and Fehr, E., editors (2005). *Moral Sentiments and Material Interests*. MIT Press, Cambridge, USA. 132

Gips, J. (1994). Towards the ethical robot. In Ford, K. M., Glymour, C., and Hayes, P., editors, *Android Epistemology*. MIT Press, Cambridge, USA. 144, 145

Givant, S. and Halmos, P. (2009). *Introduction to Boolean Algebras*. Springer-Verlag New York, New York, USA. 82, 84, 85, 95

Glasgow, J. I., MacEwen, G. H., and Panangaden, P. (1992). A logic for reasoning about security. *ACM Transactions on Computer Systems*, 10(3):226–264. 158

Gonçalves, R. and Alferes, J. J. (2010). Parametrized logic programming. In Janhunen, T. and Niemelä, I., editors, *Logics in Artificial Intelligence - 12th European Conference, JELIA 2010*, Helsinki, Finland, September 13-15, 2010, pages 182–194, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 163

Gonçalves, R. and Alferes, J. J. (2012). An embedding of input-output logic in deontic logic programs. In Ågotnes, T., Broersen, J., and Elgesem, D., editors, *Deontic Logic in Computer Science - 11th International Conference, DEON 2012*, Bergen, Norway, July 16-18, 2012, pages 61–75, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 163

Gottlob, G. (1992). Complexity results for nonmonotonic logics. *Journal of Logic and Computation*, 2(3):397–425. 10

Governatori, G., Olivieri, F., Rotolo, A., and Scannapieco, S. (2013). Computing strong and weak permissions in defeasible logic. *Journal of Philosophical Logic*, 42(6):799–829. 6, 11, 18, 101, 135

Governatori, G. and Rotolo, A. (2007). BIO logical agents: Norms, beliefs, intentions in defeasible logic. In Boella et al. (2007). 12

Governatori, G. and Rotolo, A. (2010). Norm compliance in business process modeling. In Dean, M., Hall, J., Rotolo, A., and Tabet, S., editors, *Semantic Web Rules - International Symposium, RuleML 2010,* Washington, DC, USA, October 21-23, 2010, pages 194–209, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 4, 32

Governatori, G. and Sartor, G., editors (2010). *Deontic Logic in Computer Science, 10th International Conference*, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 185, 187

Grau, C. (2011). There is no 'I' in 'Robot': Robots and utilitarianism. In Anderson, S. and Anderson, M., editors, *Machine Ethics*, pages 451–463. Cambridge University Press, New York, USA. 144

Greenwald, A. and Hall, K. (2003). Correlated Q-learning. In Fawcett, T. and Mishra, N., editors, *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003),* August 21-24, 2003, Washington, DC, USA, pages 242–249, San Francisco, USA. AAAI Press. 36, 39, 43, 46

Grossi, D. and Jones, A. (2013). Constitutive norms and counts-as conditionals. In Gabbay et al. (2013), pages 407–441. 76

Guarini, M. (2006). Particularism and the classification and reclassification of moral cases. *IEEE Intelligent Systems*, 21(4):22–28. 11

Halpern, J. (1995). The effect of bounding the number of primitive propositions and the depth of nesting on the complexity of modal logic. *Artificial Intelligence*, 75(2):361–372. 106

Halpern, J. and Hitchcock, C. (2015). Graded causation and defaults. *British Journal for the Philosophy of Science*, 66(2):413–457. 165

Hansen, J. (2004). Problems and results for logics about imperatives. *Journal of Applied Logic*, 2(1):39–61. 24

Hansen, J. (2005). Conflicting imperatives and dyadic deontic logic. *Journal of Applied Logic*, 3(3-4):484–511. 24

Hansen, J. (2006). Deontic logics for prioritized imperatives. *Artificial Intellligence and Law*, 14(1-2):1–34. 24

Hansen, J. (2008). Prioritized conditional imperatives: problems and a new proposal. *Autonomous Agents and Multi-Agent Systems*, 17(1):11–35. 6, 16, 24, 25, 126

Hansen, J. (2014). Reasoning about permission and obligation. In Hansson (2014), pages 287–333. 6

Hansson, B. (1969). An analysis of some deontic logics. *Noûs*, 3(4):373–398. 6

Hansson, S. O., editor (2014). *David Makinson on Classical Methods for Non-Classical Problems*. Springer Netherlands, Dordrecht, the Netherlands. 177, 182

Harrenstein, P. (2004). *Logic in conflict*. PhD thesis, Utrecht University. 14, 133

Harrenstein, P., van der Hoek, W., Meyer, J.-J., and Witteveen, C. (2001). Boolean games. In van Benthem, J., editor, *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge*, pages 287–298, San Francisco, USA. Morgan Kaufmann Publishers Inc. 14, 132, 133

Haynes, C., Miles, S., and Luck, M. (2013). Monitoring the impact of norms upon organisational performance: A simulation approach. In Balke et al. (2014), pages 103–119. 7

Herzig, A., Lorini, E., Moisan, F., and Troquard, N. (2011). A dynamic logic of normative systems. In Walsh (2011), pages 228–233. 132, 165

Heyting, A. (1930). *Die formalen Regeln der intuitionistischen Logik*. Berlin: Verlag der Akademie der Wissenschaften, Berlin, Germany. 92

Horty, J. (2001). *Agency and Deontic Logic*. Oxford University Press, New York, USA. 6

Horty, J. (2003). Reasoning with moral conflicts. *Noûs*, 37(4):557–605. 6, 25

Horty, J. (2007). Defaults with priorities. *Journal of Philosophical Logic*, 36(4):367–413. 6, 25, 27, 127

Horty, J. (2012). *Reasons as Defaults*. Oxford University Press, New York, USA. 6, 25

Horty, J. (2014). Deontic modals: Why abandon the classical semantics? *Pacific Philosophical Quarterly*, 95(4):424–460. 6, 25

Huberman, B. A. and Hogg, T. (2003). Quantum solution of coordination problems. *Quantum Information Processing*, 2(6):421–432. 157

Jones, A. and Sergot, M. (1992). Formal specification of security requirements using the theory of normative positions. In Deswarte, Y., Eizenberg, G., and Quisquater, J., editors, *Computer Security - ESORICS 92, Second European Symposium on Research in Computer Security*, Toulouse, France, November 23-25, 1992, pages 103–121, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 158

Jones, A. and Sergot, M. (1996). A formal characterization of institutionalised power. *Logic journal of the IGPL*, 4(3):427–443. 67, 69

Jörgensen, J. (1938). Imperatives and logic. *Erkenntnis*, 7(1):288–296. 16

Kanger, S. (1971). New foundations for ethical theory. In Hilpinen, R., editor, *Deontic Logic: Introductory and Systematic Readings*, pages 95–124, Dordrecht, the Netherlands. Springer Netherlands. 24

Keogh, K. and Sonenberg, L. (2013). Coordination using social policies in dynamic agent organizations. In Balke et al. (2014), pages 83–102. 7

Kooi, B. and Tamminga, A. (2008). Moral conflicts between groups of agents. *Journal of Philosophical Logic*, 37(1):1–21. 6

Krentel, M. W. (1992). Generalizations of opt P to the polynomial hierarchy. *Theoretical Computer Science*, 97(2):183–198. 128

La Mura, P. (2005). Correlated equilibria of classical strategic games with quantum signals. *International Journal of Quantum Information*, 3(01):183–188. 157

Levin, L. (1975). Universal sequential search problems. *Problems of Information Transmission*, 9(3):265–266. Translation into English of Russian article originally published in 1973. 104

Lewis, D. (1969). *Conventions: a Philosophical Study*. Harvard University Press, Cambridge, USA. 7, 52

Lewis, H. (1979). Satisfiability problems for propositional calculi. *Mathematical Systems Theory*, 13(1):45–53. 119

Lewontin, R. (1961). Evolution and the theory of games. *Journal of theoretical biology*, 1(3):382–403. 52

Lindahl, L. and Odelstad, J. (2000). An algebraic analysis of normative systems. *Ratio Juris*, 13(3):261–278. 10, 81, 82, 152

Lindahl, L. and Odelstad, J. (2008). Intermediaries and intervenients in normative systems. *Journal of Applied Logic*, 6(2):229–250. 10, 81, 152

Lindahl, L. and Odelstad, J. (2013). TJS: a formal framework for normative systems with intermediaries. In Gabbay et al. (2013). 10, 81, 82, 83, 152

Litak, T. (2006). Algebraization of hybrid logic with binders. In Schmidt, R. A., editor, *Relations and Kleene Algebra in Computer Science, 9th International Conference on Relational Methods in Computer Science and 4th International Workshop on Applications of Kleene Algebra, RelMiCS/AKA 2006,*

Manchester, UK, August 29-September 2, 2006, pages 281–295, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 164

López y López, F., Luck, M., and d'Inverno, M. (2002). Constraining autonomy through norms. In *The First International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS 2002,* July 15-19, 2002, Bologna, Italy, pages 674–681, New York, USA. ACM Press. 4

López y López, F. and Marquez, A. A. (2004). An architecture for autonomous normative agents. In *5th Mexican International Conference on Computer Science (ENC 2004),* 20-24 September 2004, Colima, Mexico, pages 96–103, Los Alamitos, USA. IEEE Computer Society. 4

Lorini, E. (2015). A logic for reasoning about moral agents. *Logique et Analyse*, 58(230). http://virthost.vub.ac.be/lnaweb/ojs/index.php/LogiqueEtAnalyse/article/view/1700. 12, 137

Mahmoud, S., Griffiths, N., Keppens, J., and Luck, M. (2011). Overcoming omniscience for norm emergence in Axelrod's metanorm model. In Cranefield, S., van Riemsdijk, M. B., Vázquez-Salceda, J., and Noriega, P., editors, *Coordination, Organizations, Institutions, and Norms in Agent System VII, COIN 2011 International Workshops, COIN@AAMAS 2011,* Taipei, Taiwan, May 3, 2011, COIN@WI-IAT 2011, Lyon, France, August 22, 2011, pages 186–202, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 31

Mahmoud, S., Griffiths, N., Keppens, J., and Luck, M. (2012). Norm emergence through dynamic policy adaptation in scale free networks. In Aldewereld, H. and Sichman, J. S., editors, *Coordination, Organizations, Institutions, and Norms in Agent Systems VIII - 14th International Workshop, COIN 2012, Held Co-located with AAMAS 2012,* Valencia, Spain, June 5, 2012, pages 123–140, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 31

Makinson, D. (1993). Five faces of minimality. *Studia Logica*, 52(3):339–380. 165

Makinson, D. (1999). On a fundamental problem of deontic logic. In McNamara, P. and Prakken, H., editors, *Norms, Logics and Information Systems. New Studies in Deontic Logic and Computer Science*, pages 29–53. IOS Press, Amsterdam, The Netherlands. 157

Makinson, D. and van der Torre, L. (2000). Input-output logics. *Journal of Philosophical Logic*, 29(4):383–408. 4, 6, 10, 18, 19, 20, 66, 98, 107, 108, 109, 112

Makinson, D. and van der Torre, L. (2001). Constraints for input/output logics. *Journal of Philosophical Logic*, 30(2):155–185. 6, 18, 22

Makinson, D. and van der Torre, L. (2003). Permission from an input/output perspective. *Journal of Philosophical Logic*, 32(4):391–416. 6, 17, 18, 21, 23, 24, 135

McLaren, B. M. (2006). Computational models of ethical reasoning: Challenges, initial steps, and future directions. *IEEE Intelligent Systems*, 21(4):29–37. 11

Meyer, J. J. (1988). A different approach to deontic logic: deontic logic viewed as a variant of dynamic logic. *Notre Dame Journal of Formal Logic*, 29(1):109–136. 6

Morales, J., López-Sánchez, M., and Esteva, M. (2011). Using experience to generate new regulations. In Walsh (2011), pages 307–312. 4, 9, 35, 51

Morales, J., López-Sánchez, M., Rodríguez-Aguilar, J. A., Wooldridge, M., and Vasconcelos, W. W. (2014). Minimality and simplicity in the on-line automated synthesis of normative systems. In Bazzan, A. L. C., Huhns, M. N., Lomuscio, A., and Scerri, P., editors, *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '14*, Paris, France, May 5-9, 2014, pages 109–116, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 48, 49

Morales, J., López-Sánchez, M., Rodríguez-Aguilar, J. A., Wooldridge, M., and Vasconcelos, W. W. (2015). Synthesising liberal normative systems. In Weiss, G., Yolum, P., Bordini, R. H., and Elkind, E., editors, *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015*, Istanbul, Turkey, May 4-8, 2015, pages 433–441, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 4, 9, 35, 48, 49, 51

Nagenborg, M. (2007). Artificial moral agents: An intercultural perspective. *International Review of Information Ethics*, 7(9):129–133. 12

Nash, J. (1950). The bargaining problem. *Econometrica*, 18(2):155–162. 44

Niiniluoto, I. (1986). Hypothetical imperatives and conditional obligations. *Synthese*, 66(1):111–133. 24

Nisan, N. and Ronen, A. (2001). Algorithmic mechanism design. *Games and Economic Behavior*, 35(1-2):166 – 196. 159

Nowak, M. and May, R. (1992). Evolutionary games and spatial chaos. *Nature*, 359:826–829. 54

Odelstad, J. and Boman, M. (2004). Algebras for agent norm-regulation. *Annals of Mathematics and Artificial Intelligence*, 42(1):141–166. 83

Odelstad, J. and Lindahl, L. (2000). Normative systems represented by Boolean quasi-orderings. *Nordic Journal of Philosophical logic*, 5(2):161–174. 10, 81, 152

Odelstad, J. and Lindahl, L. (2002). The role of connections as minimal norms in normative systems. In Bench-Capon, T., Daskalopulu, A., and Winkels, R., editors, *Legal Knowledge and Information Systems*. IOS Press, Amsterdam, The Netherlands. 97

Ossowski, S., editor (2013). *Agreements Technologies*. Springer Netherlands, Dordrecht, the Netherlands. 32

Papadimitriou, C. H. and Roughgarden, T. (2008). Computing correlated equilibria in multi-player games. *Journal of the ACM*, 55(3):1–29. 38

Parent, X. (2008). On the strong completeness of Åqvist's dyadic deontic logic G. In van der Meyden and van der Torre (2008), pages 189–202. 6

Parent, X. (2011). Moral particularism in the light of deontic logic. *Artificial Intelligence and Law*, 19(2-3):75–98. 6, 18, 22, 23, 25

Parent, X., Gabbay, D., and van der Torre, L. (2014). Intuitionistic basis for input/output logic. In Hansson (2014), pages 262–286. 90, 91, 92, 152, 161

Parent, X. and van der Torre, L. (2014a). Input/output logic. In Gabbay et al. (2013), pages 499–544. 6, 10, 18, 23, 124

Parent, X. and van der Torre, L. (2014b). "Sing and dance!" - input/output logics without weakening. In Cariani et al. (2014), pages 149–165. 21, 66

Pearl, J. (2000). *Causality: models, reasoning and inference*. Cambridge University Press, New York, USA. 165

Pereira, L. M. and Saptawijaya, A. (2009). Modelling morality with prospective logic. *International Journal of Reasoning-based Intelligent Systems*, 1(3-4):209–221. 11

Peterson, C. (2014). The categorical imperative: Category theory as a foundation for deontic logic. *Journal of Applied Logic*, 12(4):417–461. 6

Peterson, C. (2015). Contrary-to-duty reasoning: A categorical approach. *Logica Universalis*, 9(1):47–92. 6

Posner, E. (2002). *Law and Social Norms*. Harvard University Press, Cambridge, USA. 2

Post, E. (1941). *The two-valued iterative systems of mathematical logic*. Princeton University Press, New Jersey, USA. 119

Powers, T. M. (2006). Prospects for a Kantian machine. *IEEE Intelligent Systems*, 21(4):46–51. 145

Prakken, H. and Sergot, M. (1996). Contrary-to-duty obligations. *Studia Logica*, 57(1):91–115. 22

Putte, F. V. D. and Straßer, C. (2013). A logic for prioritized normative reasoning. *Journal of Logic and Computation*, 23(3):563–583. 6

Rawls, J. (1971). *A Theory of Justice*. Oxford University Press, New York, USA. 43

Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence*, 13(1-2):81–132. 25

Rintanen, J. (1998a). Complexity of prioritized default logics. *Journal of Artificial Intelligence Research*, 9(1):423–461. 10, 123

Rintanen, J. (1998b). Lexicographic priorities in default logic. *Artificial Intelligence*, 106(2):221–265. 10, 25, 123

Ross, A. (1944). Imperatives and logic. *Philosophy of Science*, 11(1):30–46. 20

Sági, G. (2013). Polyadic algebras. In Andréka, H., Ferenczi, M., and Németi, I., editors, *Cylindric-like Algebras and Algebraic Logic*, pages 367–389, Berlin and Heidelberg,Germany. Springer Berlin Heidelberg. 97

Sangal, R., Mehta, H., and Bagga, R. K., editors (2007). *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence,* Hyderabad, India, January 6-12, 2007, San Francisco, USA. AAAI Press. 167, 184

Savarimuthu, B. T. R. and Cranefield, S. (2011). Norm creation, spreading and emergence: A survey of simulation models of norms in multi-agent systems. *Multiagent and Grid Systems*, 7(1):21–54. 4, 7, 63

Segerberg, K. (1982). A deontic logic of action. *Studia Logica*, 41(2):269–282. 6

Sen, O. and Sen, S. (2009). Effects of social network topology and options on norm emergence. In Padget, J. A., Artikis, A., Vasconcelos, W. W., Stathis, K., da Silva, V. T., Matson, E. T., and Polleres, A., editors, *Coordination, Organizations, Institutions and Norms in Agent Systems V, COIN 2009 International Workshops. COIN@AAMAS 2009,* Budapest, Hungary, May 2009, COIN@IJCAI 2009, Pasadena, USA, July 2009, COIN@MALLOW 2009, Turin, Italy, September 2009, pages 211–222, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 7, 62

Sen, S. and Airiau, S. (2007). Emergence of norms through social learning. In Sangal et al. (2007), pages 1507–1512. 4, 7, 9, 35, 51, 62

Shoham, Y. and Leyton-Brown, K. (2009). *Multiagent Systems - Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, Cambridge, UK. 3

Shoham, Y. and Tennenholtz, M. (1992). On the synthesis of useful social laws for artificial agent societies (preliminary report). In Swartout (1992), pages 276–281. 3, 133

Shoham, Y. and Tennenholtz, M. (1996). On social laws for artificial agent societies: Off-line design. *Artificial Intelligence*, 73(1-2):231–252. 4, 35, 133

Shoham, Y. and Tennenholtz, M. (1997). On the emergence of social conventions: Modeling, analysis, and simulations. *Artificial Intelligence*, 94(1-2):139–166. 3, 4, 7, 9, 35, 51, 62

Sinnott-Armstrong, W. (2015). Consequentialism. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Winter 2015 edition. 36

Sipser, M. (2012). *Introduction to the theory of computation*. Cengage Learning, Boston, USA, 3 edition. 102, 103, 104

Skyrms, B. (2014). *Evolution of the Social Contract*. Cambridge University Press, Cambridge, UK. 7, 10, 48, 51

Smith, J. M. and Price, G. (1973). The logic of animal conflict. *Nature*, 246:15–18. 52

Sondrol, P. C. (2009). Totalitarian and authoritarian dictators: A comparison of fidel castro and alfredo stroessner. *Journal of Latin American Studies*, 23(3):599–620. 96

Stenius, E. (1971). The principles of a logic of normative systems. *Journal of Symbolic Logic*, 36(3):519–520. 24

Stillman, J. (1992). The complexity of propositional default logics. In Swartout (1992), pages 794–799. 10

Stolpe, A. (2008a). Normative consequence: The problem of keeping it whilst giving it up. In van der Meyden and van der Torre (2008), pages 174–188. 18, 20, 21, 66, 71, 151

Stolpe, A. (2008b). *Norms and norm-system dynamics*. PhD thesis, University of Bergen. 6, 10, 18, 74

Stolpe, A. (2010a). Norm-system revision: theory and application. *Artificial Intelligence and Law*, 18(3):247–283. 98, 165

Stolpe, A. (2010b). Relevance, derogation and permission. In Governatori and Sartor (2010), pages 98–115. 18

Stolpe, A. (2010c). A theory of permission based on the notion of derogation. *Journal of Applied Logic*, 8(1):97–113. 18, 135, 164

Stolpe, A. (2015). A concept approach to input/output logic. *Journal Applied Logic*, 13(3):239–258. 99

Straßer, C., Beirlaen, M., and Van De Putte, F. (2016). Adaptive logic characterizations of input/output logic. *Studia Logica*, 104(5):1–48. 6

Sugden, R. (1989). Spontaneous order. *Journal of Economic Perspectives*, 3(4):85–97. 7

Sun, X. (2011). Conditional ought, a game theoretical perspective. In van Ditmarsch, H., Lang, J., and Ju, S., editors, *Logic, Rationality, and Interaction - Third International Workshop, LORI 2011,* Guangzhou, China, October 10-13, 2011, pages 356–369, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 6

Sun, X. (2013). Proof theory, semantics and algebra for normative systems. In Grossi, D., Roy, O., and Huang, H., editors, *Logic, Rationality, and Interaction - 4th International Workshop, LORI 2013,* Hangzhou, China, October 9-12, 2013, pages 228–238, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 33

Sun, X. (2014). How to build input/output logic. In Bulling, N., van der Torre, L., Villata, S., Jamroga, W., and Vasconcelos, W., editors, *Computational Logic in Multi-Agent Systems - 15th International Workshop, CLIMA XV,* Prague, Czech Republic, August 18-19, 2014, pages 123–137, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 18, 33

Sun, X. (2015a). Boolean game with prioritized norms. In van der Hoek et al. (2015), pages 341–352. 33

Sun, X. (2015b). Boolean games with norms. In Bikakis and Zheng (2015), pages 61–71. 33

Sun, X. (2015c). Input/output STIT logic for normative systems. In Bassiliades, N., Gottlob, G., Sadri, F., Paschke, A., and Roman, D., editors, *Rule Technologies: Foundations, Tools, and Applications - 9th International Symposium, RuleML 2015,* Berlin, Germany, August 2-5, 2015, pages 347–359, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 18, 161

Sun, X. (2015d). Proof theory, semantics and algebra for normative systems. *Journal of Logic and Computation*. first published online, doi: 10.1093/logcom/exv022. 18, 33

Sun, X. and Ambrossio, D. A. (2015a). Computational complexity of input/output logic. In Bikakis and Zheng (2015), pages 72–79. 33

Sun, X. and Ambrossio, D. A. (2015b). On the complexity of input/output logic. In van der Hoek et al. (2015), pages 429–434. 33

Sun, X. and Baniasadi, Z. (2014). STIT based deontic logics for the miners puzzle. In Bulling, N., editor, *Multi-Agent Systems - 12th European Conference, EUMAS 2014,* Prague, Czech Republic, December 18-19, 2014, pages 236–251, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 6

Sun, X. and Robaldo, L. (2015). Logic and games for ethical agents in normative multi-agent systems. In Rovatsos, M., Vouros, G. A., and Julián, V., editors, *Multi-Agent Systems and Agreement Technologies - 13th European Conference, EUMAS 2015, and Third International Conference, AT 2015, Athens, Greece, December 17-18, 2015, Revised Selected Papers*, pages 367–375, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 33

Sun, X. and van der Torre, L. (2014). Combining constitutive and regulative norms in input/output logic. In Cariani et al. (2014), pages 241–257. 18

Swartout, W., editor (1992). *Proceedings of the 10th National Conference on Artificial Intelligence*, San Francisco, USA. AAAI Press. 184

Tarski, A. (1955). A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309. 74

Taylor, M. (1976). *Anarchy and cooperation*. John Wiley, London, UK. 7

Testerink, B., Dastani, M., and Meyer, J. C. (2013). Norms in distributed organizations. In Balke et al. (2014), pages 120–135. 7

Thoma, J. (2015). Bargaining and the impartiality of the social contract. *Philosophical Studies*, 172(12):3335–3355. 32

Thomason, R. H. (1968). On the stong semantical completeness of the intuitionistic predicate calculus. *Journal of Symbolic Logic*, 33(1):1–7. 91

Troelstra, A. S. and van Dalen, D. (1988). *Constructivism in Mathematics: An Introduction*. Elsevier Science Publisher, Amsterdam, The Netherlands. 93

Trypuz, R. and Kulicki, P. (2015). On deontic action logics based on boolean algebra. *Journal of Logic and Computation*, 25(5):1241–1260. 6

van Benthem, J., Grossi, D., and Liu, F. (2014). Priority structures in deontic logic. *Theoria*, 80(2):116–152. 6

Van Dalen, D. (1986). Intuitionistic logic. In Gabbay, D. and Guenthner, F., editors, *Handbook of Philosophical Logic: Volume III: Alternatives in Classical Logic*, pages 95–124, Dordrecht, the Netherlands. Springer Netherlands. 90

van der Hoek, W., Holliday, W., and Wang, W., editors (2015). *Logic, Rationality, and Interaction - 5th International Workshop, LORI 2015* Taipei, Taiwan, October 28-31, 2015, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 185, 186

van der Hoek, W., Roberts, M., and Wooldridge, M. (2007). Social laws in alternating time: effectiveness, feasibility, and synthesis. *Synthese*, 156(1):1–19. 4, 35

van der Meyden, R. (1996). The dynamic logic of permission. *Journal of Logic and Computation*, 6(3):465–479. 6

van der Meyden, R. and van der Torre, L., editors (2008). *Deontic Logic in Computer Science, 9th International Conference*, Berlin and Heidelberg, Germany. Springer-Verlag Berlin Heidelberg. 174, 182, 184

van der Torre, L. (1997). *Reasoning about obligations: defeasibility in preference-based deontic logic*. PhD thesis, Erasmus University of Rotterdam, Rotterdam. 6

van der Torre, L. (2010a). Deontic redundancy: A fundamental challenge for deontic logic. In Governatori and Sartor (2010), pages 11–32. 98

van der Torre, L. (2010b). Violation games: a new foundation for deontic logic. *Journal of Applied Non-Classical Logics*, 20(4):457–477. 8

van Eijck, J. and Verbrugge, R. (2015). Formal approaches to social procedures. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Summer 2015 edition. 166

van Fraassen, B. (1973). Values and the heart's command. *Journal of Philosophy*, 70(1):5–19. 6, 24

Vidal, J. M. (2006). *Fundamentals of Multiagent Systems: Using NetLogo Models*. Unpublished. http://www.multiagent.com. 27, 53

von Neumann, J. and Morgenstern, O. (1944). *Theory of Games and Economic Behavior (60th-Anniversary Edition)*. Princeton University Press, New Jersey, USA. 27

von Wright, G. (1951). Deontic logic. *Mind*, 60(237):1–15. 5

Wagner, K. W. (1986). More complicated questions about maxima and minima, and some closures of NP. In Kott, L., editor, *Automata, Languages and Programming, 13th International Colloquium, ICALP86,* Rennes, France, July 15-19, 1986, pages 434–443, Berlin and Heidelberg, Germany. Springer Berlin Heidelberg. 104

Walsh, T., editor (2011). *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence,* Barcelona, Catalonia, Spain, July 16-22, 2011, San Francisco, USA. AAAI Press. 178, 181

Weiss, G., editor (2013). *Multiagent systems*. MIT press, Cambridge, USA, 2 edition. 3

Wójcicki, R. (1988). *Theory of Logical Calculi: Basic Theory of Consequence Operations*. Springer Netherlands, Dordrecht, the Netherlands. 162

Wooldridge, M. (2009). *An Introduction to MultiAgent Systems (2. ed.)*. Wiley, London, UK. 3, 132

Yu, C., Zhang, M., Ren, F., and Luo, X. (2013). Emergence of social norms through collective learning in networked agent societies. In Gini, M. L., Shehory, O., Ito, T., and Jonker, C. M., editors, *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13,* Saint Paul, MN, USA, May 6-10, 2013, pages 475–482, Richland, USA. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS). 63

Zeleznikow, J. and Sartor, G., editors (2003). *Proceedings of the 9th International Conference on Artificial Intelligence and Law, ICAIL 2003,* Edinburgh, Scotland, UK, June 24-28, 2003, New York, USA. ACM Press. 171

Curriculum Vitae 个人简历

Personal details 个人信息

- Surname: Sun
- 姓: 孙
- First name: Xin
- 名: 鑫
- Title: Mr.
- 称呼: 先生
- Email: xin.sun.logic@gmail.com
- 邮箱: xin.sun.logic@gmail.com
- Language skills: Chinese (native), English (fluent)
- 语言技能: 中文(母语), 英语(流畅)
- Programming skills: Netlogo, Java, Prolog
- 编程技能: Netlogo, Java, Prolog

Education 教育背景

2012-now: PhD student in computer science, Faculty of Science, Technology and Communication, University of Luxembourg, Luxembourg (defended in July, 2016). PhD supervisor: Leendert van der Torre.

2012至今, 计算机科学博士, 卢森堡大学计算机系, 答辩时间: 2016年6月. 指导老师: Leendert van der Torre.

2009-2012: Master in Philosophy, Tsinghua University, Beijing, China. Master supervisor: Fenrong Liu.

2009-2012, 哲学硕士, 清华大学哲学系, 指导老师: 刘奋荣.

2005-2009: Bachelor in Philosophy, Southwest University, Chongqing, China.

2005-2009, 哲学学士, 西南大学哲学系.

Research interests 研究兴趣

- logic
- 逻辑
- game theory
- 博弈论
- computational complexity
- 计算复杂性

Awards 所获奖项

2009, Distinguished graduation award, Southwest University, Chongqing, China

2009, 西南大学优秀毕业生

2009, Distinguished graduation award, Chongqing Municipality, Chongqing, China

2009, 重庆市优秀毕业生

2011, Guanghua scholarship, Tsinghua University, Beijing, China

2011, 清华大学光华奖学金

Publications 科研论文

Articles in handbooks 工具书章节

2017

1. Xin Sun, Xavier Parent, Livio Robaldo. 'Computational deontic logic'. In Amit Chopra, Leendert van der Torre, Harko Verhagen and Serena Villata, editors, *Handbook of normative multiagent systems*. College Publications. http://normativemas.org/ To appear, 2016.

Papers in journals 期刊论文

2015

1. Xin Sun. 'Proof theory, semantics and algebra for normative systems'. In *Journal of Logic and Computation*. First published online: May 29, 2015. doi: 10.1093/logcom/exv022

2014

2. Dov Gabbay, Loic Gammaitoni, Xin Sun. 'The paradoxes of permission, an action based solution'. In *Journal of Applied Logic*. Volume 12, Issue 2, June 2014, Pages 179-191.

Papers in conference proceedings 会议论文

2016

1. Xin Sun, Livio Robaldo. 'Norm creation in proposition control games'. accepted by *Chinese Conference on Logic and Argumentation*, Hangzhou, China, April 02-03, 2016.

2015

2. Xin Sun. 'Input/Output STIT Logic for Normative Systems'. In Nick Bassiliades, Georg Gottlob, Fariba Sadri, Adrian Paschke and Dumitru Roman, editors, *Rule Technologies: Foundations, Tools, and Applications - 9th International Symposium, RuleML 2015, Berlin, Germany, August 2-5, 2015, Proceedings*, volume 9202 of *Lecture Notes in Computer Science*, pages 347-359. Springer, 2015.

3. Xin Sun. 'Preference refinement in normative multi-agent system'. In Julian Gutierrez, Fabio Mogavero, Aniello Murano, Michael Wooldridge, editors, *3rd International Workshop on Strategic Reasoning 2015 (SR15), Oxford, England, September 21-22, 2015, Proceedings*.

4. Xin Sun. 'Boolean game with prioritized norms'. In Wiebe van der Hoek, Wesley Holliday and Wen-fang Wang, editors, *Logic, Rationality, and Interaction - 5th International Workshop, LORI 2015, Taipei, Taiwan, October 28-21, 2015, Proceedings*, volume 9394 of *Lecture Notes in Computer Science*, Springer, 2015.

5. Xin Sun, Diego Agustin Ambrossio. 'On the complexity of input/output logic'. In Wiebe van der Hoek, Wesley Holliday and Wen-fang Wang, editors, *Logic, Rationality, and Interaction - 5th International Workshop, LORI 2015, Taipei, Taiwan, October 28-21, 2015, Proceedings*, volume 9394 of *Lecture Notes in Computer Science*, Springer, 2015.

6. Xin Sun. 'Boolean game with norms'. In Xianghan Zheng and Antonis Bikakis, editors, *the 9th Multi- disciplinary International Workshop on Artificial Intelligence, MIWAI 2015, Fuzhou, China, November 13-15, 2015, Proceedings*, volume 9426 of *Lecture Notes in Artificial Intelligence*, Springer, 2015.

7. Xin Sun, Diego Agustin Ambrossio. 'Computational complexity of input/output logic'. In Xianghan Zheng and Antonis Bikakis, editors, *the 9th Multi- disciplinary International Workshop on Artificial Intelligence, MIWAI 2015, Fuzhou, China, November 13-15, 2015, Proceedings*, volume 9426 of *Lecture Notes in Artificial Intelligence*, Springer, 2015.

8. Livio Robaldo, Llio Humphreys, Xin Sun, Loredana Cupi, Cristiana Teixeira Santos, Robert Muthuri. 'The ProLeMAS project: representing natural language norms in input/output logic'. In Takehiko Kasahara, and Ken Satoh, editors, *the 9th International Workshop on Juris-informatics, JURISIN 2015, Kanagawa, Japan, November 17-18, 2015, Proceedings*, volume 9067 of *Lecture Notes in Artificial Intelligence*, Springer, 2015.

9. Xin Sun, Livio Robaldo. 'Logic and games for ethical agents in normative multi-agent systems'. to appear in Michael Rovatsos, George Vouros, Vicente Julian, editors, *13th European Conference on Multi-Agent Systems/3rd International Conference on Agreement Technologies*, December 17-18, 2015. LNAI vol. 9571

10. Xin Sun, Beishui Liao. 'Probabilistic argumentation, a small step for uncertainty, a giant step for complexity'. to appear in Michael Rovatsos, George Vouros, Vicente Julian, editors, *13th European Conference on Multi-Agent Systems/3rd International Conference on Agreement Technologies*, December 17-18, 2015. LNAI vol. 9571

2014

11. Xin Sun, Leendert van der Torre. 'Combining constitutive and regulative norms in input/output logic'. In Fabrizio Cariani, Davide Grossi, Joke Meheus and Xavier Parent, editors, *Deontic Logic and Normative Systems - 12th International Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014. Proceedings*, volume 8554 of *Lecture Notes in Computer Science*, pages 241-257. Springer, 2014.

12. Dov Gabbay, Livio Robaldo, Xin Sun, Leendert van der Torre, Zohreh Baniasadi. 'Toward a Linguistic Interpretation of Deontic Paradoxes - Beth-Reichenbach Semantics Approach for a New Analysis of the Miners Scenario'. In Fabrizio Cariani, Davide Grossi, Joke Meheus and Xavier Parent, editors, *Deontic Logic and Normative Systems - 12th International Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014. Proceedings*, volume 8554 of *Lecture Notes in Computer Science*, pages 108-123. Springer, 2014.

13. Xin Sun. 'How to build input/output logic'. In Nils Bulling, Leendert van der Torre, Serena Villata, Wojtek Jamroga, and Wamberto Vasconcelos, editors, *Computational Logic in Multi-Agent Systems - 15th International Workshop, CLIMA XV, Prague, Czech Republic, August 18-19, 2014. Proceedings*, volume 8624 of *Lecture Notes in Computer Science*, pages 123-137. Springer, 2014.

14. Xin Sun, Huimin Dong. 'Stratied action negation, a logic about travel'. In Franc Grootjen, Maria Otworowska and Johan Kwisthout, editors, *26th Benelux Conference on Artificial Intelligence, BNAIC 2014, Nijmegen, the Netherlands, November 6-7, 2014, Proceedings*. pages 81-87, 2014.

15. Xin Sun, Zohreh Baniasadi, Shuwen Zhou. 'How do pessimistic agents save miners? A STIT based approach'. In Franc Grootjen, Maria Otworowska and Johan Kwisthout, editors, *26th Benelux Conference on Artificial Intelligence, BNAIC 2014, Nijmegen, the Netherlands, November 6-7, 2014, Proceedings*. pages 88-94, 2014.

16. Xin Sun, Zohreh Baniasadi. 'STIT Based Deontic Logics for the Miners Puzzle'. In Nils Bulling, editors, *Multi-Agent Systems - 12th European Conference, EUMAS 2014, Prague, Czech Republic, December 18-19, 2014, Revised Selected Papers*. volume 8953 of *Lecture Notes in Computer Science*, pages 236–251. Springer, 2014.

2013

17. Xin Sun. 'Proof theory, semantics and algebra for normative systems'. In Davide Grossi, Olivier Roy and Huaxin Huang, editors, *Logic, Rationality, and Interaction - 4th International Workshop, LORI 2013, Hangzhou, China, October 9-12, 2013, Proceedings*, volume 8196 of *Lecture Notes in Computer Science*, pages 228-238, Springer, 2013.

18. Xin Sun. ' "To be or not to be" $\neq$ "to kill or not to kill": a logic on action negation'. In Tomoyuki Yamada, editors, *3rd International Workshop on Philosophy and Ethics of Social Reality, SOCREAL 2013, Sapporo, Japan, October 25-27, 2013,Proceedings*. Pages 36-40, 2013.

2011

19. Xin Sun. 'Conditional ought, a game theoretical perspective'. In Hans van Ditmarsch, Jerome Lang and Shier Ju, editors, *Logic, Rationality, and Interaction - Third International Workshop, LORI 2011, Guangzhou, China, October 10-13, 2011. Proceedings*, volume 6953 of *Lecture Notes in Computer Science*, pages 356-369, Springer, 2011.

20. Xin Sun, Fenrong Liu. 'Consequentialist deontic logic for decisions and games'. In Alexandru Baltag, Davide Grossi, Alexandru Marcoci, Ben Rodenhauser and Sonja Smets, editors, *Logic and Interactive Rationality Yearbook 2011*. Pages 374-397, 2011

Thesis 毕业论文

- Master thesis (in Chinese): 'Deontic logic, a game theoretical perspective'. Tsinghua University. 2012.

- 硕士论文: 博弈论道义逻辑研究. 清华大学. 2012.
- PhD thesis: 'Logic and games of norms: a computational perspective'. University of Luxembourg. 2016.
- 博士论文: 'Logic and games of norms: a computational perspective'. 卢森堡大学. 2016.