Normative Reasoning and Consequence

Jan Broersen[1], Stephen Cranefield[2], Yehia Elrakaiby[3], Dov Gabbay[4], Davide Grossi[5], Emiliano Lorini[6], Xavier Parent[3], Leendert W. N. van der Torre[3], Luca Tummolini[7], Paolo Turrini[3], and François Schwarzentruber[8]

1 **Utrecht University, The Netherlands**
2 **University of Otago, New Zealand**
3 **University of Luxembourg, Luxembourg**
4 **King's College London, United Kingdom**
5 **University of Liverpool, England**
6 **Paul Sabatier University – Toulouse, France**
7 **National Research Council (CNR) / ISIC – Rome, Italy**
8 **ENS Cachan / IRISA, France**

Abstract

In this chapter we first provide a general introduction to the research area methodology and relevance, then we discuss normative reasoning for multi-agent systems, and finally we discuss current research challenges. We cover the main issues in modern deontic logic, which is much broader than the traditional modal logic framework of deontic logic, with an emphasis to our intended audience. To emphasize this broadness, we typically refer to "deontic logic and normative systems" rather than deontic logic only.

1998 ACM Subject Classification I.2.11 Distributed Artificial Intelligence

Keywords and phrases Norms, MAS

Digital Object Identifier 10.4230/DFU.Vol4.12111.33

1 Introduction

We first give a general introduction to the research area methodology and relevance, then we discuss normative reasoning for multi-agent systems, and finally we discuss current research challenges.

The intended audience is a general multi-agent systems audience, and no previous knowledge of deontic logic is presupposed. For a more detailed and in depth discussion on deontic logic, we refer to the handbook of deontic logic and normative systems.

We cover the main issues in modern deontic logic, which is much broader than the traditional modal logic framework of deontic logic, with an emphasis to our intended audience. To emphasize this broadness, we typically refer to "deontic logic and normative systems" rather than deontic logic only.

The paper is based on discussions during Dagstuhl seminar "Normative Multi-Agent Systems"[1], and has subsequently been written by participants of the workshop, extended with a few additional authors.

[1] `http://www.dagstuhl.de/12111`

Normative Multi-Agent Systems. *Dagstuhl Follow-Ups, Vol. 4.* ISBN 978-3-939897-51-4.
Editors: Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert W.N. van der Torre; pp. 33–70
DAGSTUHL FOLLOW-UPS Dagstuhl Publishing
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Germany

2 Methodology and relevance

We explain what deontic logic and normative reasoning are, why normative reasoning is relevant for normative multi-agent systems, what the advantages of formal methods in multi-agent systems are, and whether normative reasoning is special in comparison to other kinds of reasoning.

2.1 What are deontic logic and normative reasoning?

Succinctly put, deontic logic can be defined as the formal study of normative reasoning. In this section, we explain what this definition means.

Generally speaking, one might define logic as the study of the principles of correct reasoning. It tells us whether certain conclusions follow from some given assumptions. The truth of propositions and the validity of reasoning are quite distinct. Empirical scientists, private detectives *etc.* are concerned with the first. Logicians are concerned with the second. For instance, the logic principle $\neg(A \wedge \neg A)$, where $\neg$ stands for negation and $\wedge$ stands for conjunction, is known as the principle of non-contradiction. It says nothing about the truth-value of A and $\neg A$, but only that they cannot be true at the same time.

Sentential logic (propositional logic, first-order logic, *etc*) looks at the logical relationships amongst utterances that assert that something can be judged true or false, like "the cat is on the mat." Such a sentence is usually called a declarative one. By contrast, deontic logic is the study of the logical relationships among propositions that assert that certain actions or states of affairs are obligatory, forbidden or permitted. Deontic comes from the Greek *deon* meaning "that which is binding, right." The latter sentences are usually called directives. A typical example is "you should not eat with your fingers." Such a sentence can be formalized as $\bigcirc\neg p$, where $\bigcirc$ is read as "it is obligatory that", and p is read as "eating with your fingers." One might add various levels of granularity using first-order logic, or using other modal operators (here, a modal operator for agency).

It is traditional to take the view that logic is topic neutral. The laws of biology might be true only of living creatures, and the laws of economics are only applicable to groups of agents that engage in financial transactions. But the principles of logic are universal principles which are more general than biology and economics. Thus, the principle of non-contradiction mentioned above applies to both biology and economics. The same point can be made about the principles of deontic logic. It does not matter where the directives come from. These can be part of a moral code, or a legal system, or a set of traffic regulations, *etc.*

There are two main types of directive sentences that are studied in deontic logic: regulative norms and constitutive norms. The first are obligations, prohibitions and permissions part of a normative system. The second say what counts as what within a normative system. In the example below, the first premise is a constitutive norm, while the second premise and the conclusion are regulative norms:

- Bikes count as vehicles.
- Vehicles are not allowed to access public parks.
- Therefore, bikes are not allowed to access public parks.

Deontic logic studies the two, and their interaction.

2.2 Why is normative reasoning relevant for NorMAS?

In recent years, the study of Multi-Agent Systems (MAS) has undergone what can be called a normative turn, shifting the emphasis on normative issues in agent organizations. The

AgentLink Roadmap, published in 2005, considers norms as key for the development of MAS. This shift in focus has given rise to a new interdisciplinary research area called "NorMAS" (Normative Multi-Agent Systems). It can be defined as the intersection of normative systems and multi-agent systems.

One motivation comes from systems where artificial and human agents interact. 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, or that are to display behavior comparable to human intelligent behavior. By integrating norms and individual intelligence normative multi-agent systems provide a promising model for human and artificial agent cooperation and coordination, group decision making, multi-agent organizations, regulated societies, electronic institutions, secure multi-agent systems, and so on.

Another key point for using normative reasoning is that it is an independent tool for building interoperability standards and for reasoning about them, a sort of lingua franca that preserves the heterogeneity of the agents and their autonomy.

Multi-agent research has been driven by the need to find a substantially more realistic model than those used thus far. It is a common assumption in many multi-agent systems that agents will behave as they are intended to behave. There are many circumstances in which such an assumption must be abandoned. In particular, in open agent societies, where agents are programmed by different parties, where there is no direct access to an agent's internal state, and where agents do not necessarily share a common goal, it cannot be assumed that all agents will behave according to the system norms that govern their behavior. Agents must be assumed to be untrustworthy because they act on behalf of parties with competing interests, and so they may fail to, or even choose not to, comply with the society's norms in order to achieve their individual goals. It is then usual to impose sanctions to discourage norm violating behavior and to provide some form of reparation when it does occur.

An alternative use of normative reasoning comes from so-called logic based agents. Agents need to represent their environment in order to be able to act intelligently. The assumption (already made by McCarthy in 1958 – see [97]) is that an efficient representation can be found using logic. The idea is that agents only need to store "basic" facts about the external world, and derive the rest using logic. This hints at what Russell and Norvig call "logic-based agents" (see also Wooldridge [148]). An agent is considered having amongst others two components:

- a knowledge base;
- a set of deduction rules (a "program").

The knowledge base stores information about the world. By applying the deduction rules to the knowledge base, the agent gets more information about the world, and can interact with it more intelligently. In the case of a normative multi-agent system, the knowledge base contains roles played by the agent, obligations and permissions associated with these roles, constitutive rules, and the like. The goal of deontic logic is to identify the relevant deduction rules used to extract information from the knowledge base, thus conceived.

2.3 Advantages of formal methods

Deontic logic and other formalisms for normative reasoning are examples of formal methods with formal semantics. Formal methods can be used as a modeling language when designing multi-agent systems, for explaining their structure to other designers, or for reasoning about the system.

In general, the advantages of using formal methods over informal ones may be seen by contrasting them with other methods, like the Unified Modeling Language (UML) – see *e.g.*

[59]. This graphical language is a *de facto* standard for object-oriented modeling. Its success within the industry has resulted in a number of attempts to adapt the UML notation for modeling agent systems. Odell and colleagues [107] have discussed several ways in which the UML notation might usefully be extended to enable the modellng of agent systems. The proposed modification include:

- support for expressing concurrent threats of interaction, thus enabling UML to model such well-known agent protocols as the Contract-Net
- a notion of "role" that extends that provided in UML, and in particular, allows the modeling of an agent playing many roles roles, which are usually associated with obligations, permissions and powers.

In spite of being urgently required, little work has been done on extending UML with such normative notions (though it has been studied in the context of so-called business rules in that community).

There are many papers that explain that UML has indeed a semantics even if it is not based on formal methods and why OMG decided this way. UML and formal methods have two completely different purposes. Many papers also proposed formal methods for UML (mainly based on Petri Nets), which is used in specific domains and applications. This it is not generally adopted, because all the proposed formal methods are too specific for the general purposes UML aims at.

A graphical language like UML is useful. For it mainly facilitates communication amongst researchers with different backgrounds. But it has a number of pitfalls, among which is the fact that it is error-prone. This is where formal methods come into the picture. They provide a mathematically rigorous framework for modeling normative multi-agent systems, so advanced tool support can be given. The modeling language is given a formal semantics, which constrains the intuitive characterization of the normative notions being used. The language also comes 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. 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. It should be remembered that in the early decades of the 20th century the advent of logic was mainly motivated by such foundational questions of mathematics as the question of how to establish the consistency of arithmetics. Logic is the only tool available for this task.

2.4 Is normative reasoning special?

Logic is a very broad field. There are many different logics around, all differing in language, ontological commitments, epistemological commitments, *etc.* One of these logics, or classes of logics, is deontic logic.

Some people make a distinction between logics that study the notion of inference itself, and logics that use logical inference to model reasoning about a phenomena. Examples of the latter are temporal logic and epistemic logic, and examples of the former are (non-classical) logics like intuitionistic logic, relevance logic and linear logic. For example, intuitionistic reasoning prescribes an *alternative way* to come from arbitrary premisses to entailed conclusions. The same holds for relevance logic, and other alternatives to classical logic. The question we raise here is whether or not deontic reasoning is special in this very same sense, that is, does (or should) deontic logic aim at systems that give an alternative way to come from *arbitrary* premises to their entailed conclusions? In this sense, are deontic logics non-classical?

Though some philosophers seem to pursue the first position where deontic logic is special, many computer scientists see deontic logic as aimed at designing formal systems for coming from *deontic* premises to entailed *deontic* conclusions. But, so the pragmatic argument goes, the logic governing these entailment relations can itself be quite classical. If that is true, then the burden of the deontic logician can shift from designing alternative deontic inference mechanisms to designing rich enough, but classical languages for specifying deontic conditions.

Let us take as an example the famous Chisholm 'paradox' [45], consisting of the sentences (1) "you should help", (2) "if you help you should tell you will", (3) "if you do not help you should not tell you will", (4) "you do not help". We can have at least two views on the way to approach it. The first is that we somehow need to find the right general deontic inference procedure to come to the right conclusion (you should not tell you will help) starting from the propositions representing the sentences. We might for instance claim that the conditionals in the example (sentences 2 and 3) have to be dealt with in a special, deontic way. The second view is that the propositions that make up the example hide a lot of structure (temporal structure, agency, intention) that should be made explicit. After this structure has been made explicit, in classical logics of time, agency or intention, then the deontically correct conclusion follows by classical reasoning.

This relates directly to another interesting issue. Sometimes, in discussions with other logicians, deontic logicians have to defend deontic logic against the claim that there is not a single principle of deontic logic that is non-disputed. Indeed, if one aims at designing a 'core' logic of deontic reasoning, one may end up with a very weak system, since for every suggested principle, some deontic logician might raise his hand and come with a concrete scenario and the claim that this is a counterexample. It may be that such counterexamples introduce context that interferes with the deontic reasoning. The solution to such situations would then not be to leave the logical language as it is and adapt (weaken) the logic, but to leave the logic intact and enrich the language to include the formerly hidden concepts.

So, deontic reasoning is special either way. For some it is special, because they are convinced deontic consequence cannot be classical. For others, especially for those adhering to the pragmatic computer scientist point of view, deontic logic is special because of the many different contexts that influence correct deontic reasoning in concrete examples: there is influence from all kinds of modalities, like belief, intentions, action, time, trust, ethical systems, and different views on the phenomenon of agency. Many researchers seem to believe that in order to be practically useful for computer science, deontic logic should incorporate more than just deontic modalities.

3 Background

We discuss research issues in deontic logic and NorMAS. We focus on two actual topics: norm change and proof methods. Subsection 3.3, titled *Norm Change*, shows the need to apply the tools of logic to reason about the dynamics of normative systems. On the one hand this part discusses the changes of a normative system in time, analyzing legal phenomena such as *ex tunc* and *ex nunc* regulations, on the other hand it presents a classical AGM like approach to norm change, treated as a special case of theory change. Subsection 3.4, titled *Proof Methods*, surveys the proof methods employed in deontic logic so far. they have been given less attention in philosophical logic, but they play a central role in computer science.

3.1 Current research trends in deontic logic

This section describes the different problems that are addressed in deontic logic, and gives a short literature survey. A more comprehensive overview of the state of the art can be found in the forthcoming *Handbook of Deontic Logic* [61].

3.1.1 Norm without truth

A first problem is to reconstruct deontic logic in accord with the idea that norms are neither true nor false. There are two approaches.

The mainstream approach is to reconstruct deontic logic as a logic of normative propositions. The idea is that, though norms are neither true nor false, one may state that (according to the norms), something ought to be done: the statement "John ought to leave the room" is, then, a true or false description of a normative situation. Such a statement is usually called a normative proposition, as distinguished from a norm. The Input/Output (I/O) framework of Makinson and van der Torre [91], and the bi-modal system NOBL due to Åqvist [11], are two different reconstructions of deontic logic as a logic of normative propositions, thus conceived.

The other approach consists in reconstructing deontic logic as a logic of imperatives. This approach is documented in Hansen [70, 71], to which the reader is referred for further details.

3.1.2 Reasoning about norm violation

The system SDL has one modality OA and one accessibility relation R, where xRy means that y is an ideal world relative to x. Unfortunately such a system is not adequate for representing contrary to duty obligations. We also note that some obligations have a temporal aspect to them and that even in the case where there is no real temporal aspect, there is nevertheless a progression along the axis of violations. The Chisholm example has both a temporal progression $\pm y_0 < ... < \pm y_n$ and a violation progression $\pm x_0 < ... < \pm x_n$. We can add the "temporal" relation R with the modalities Nec and Y (yesterday), to stand and represent any type of progression. Nec is flexible enough to do that. We now have models of the form (S,R, R') where the R' ideal worlds are dispersed among the other worlds. In such semantics and language, we can express the general Chisholm set and more. The intuitive concept is that when we have a set of obligations involving both real temporal progression and violation progression, we try to move along a path which will satisfy the obligations, by trying to pass through various ideal worlds in the correct way. Such a logic will have no paradoxes, because the facts correspond to families of paths and the contrary to duty obligations are wffs of the language.

3.1.3 Normative conflicts

There are two main questions here. The first one is: how can deontic logic accommodate possible conflicts between norms? The first systems of deontic logic precluded the possibility of any such conflict. This makes them unsuitable as a tool for analyzing normative reasoning. Different ways to accommodate normative conflicts have been studied over the last fifteen years. A comparative study of them can be found in Goble [65].

The second question is: how can the resolution of conflicts amongst norms be semantically modeled? An intuitively appealing modeling approach consists in using a priority relation defined on norms. There have been several proposals to this effect, and the reader is referred to the discussions in Boella and van der Torre [31], Hansen [70, 71], Horty [78] and Parent

[111]. An open question is whether tools developed for so-called non-monotonic reasoning are suitable for obligations and permissions.

3.1.4 Time

Most formalisms do not have temporal operators in the object language, nor do they have, in their standard formulation, an interpretation in temporal models. Yet for several scenarios and decisions involving deontic reasoning, the temporal aspect of the reasoning seems crucial, and several researchers have sought to study logics for the interactions between temporal and deontic modalities. The research question is: what is the relation between deontic conditionals and temporal deontic modalities?

Two natural concepts to be considered are 'validity time' and 'reference time' of an obligation, prohibition or permission. The validity time is the point in time where a deontic modality is true (surpassing the issue of section 3.1.1 here we simply assume normative modalities have truth values relative to some coherent body of norms that is left implicit) and the reference time is the point in time the obligation, prohibition or permission applies to. For instance, we can have the obligation now (validity time) to show up at the dentist's tomorrow (reference time).

Systems dealing with these temporal differences have been studied, for instance, in [12, 135]. Subtleties in expressing deontic temporal statements involving deontic deadlines have been studied in [40, 37].

3.1.5 Action

We often think of deontic modalities as applying to actions instead of states of affairs. The problems arising in this area are the following: how do we combine deontic modalities with logics of action? How do deontic and action modalities interact. Which action formalisms are best suited for a deontic extension?

Two approaches to deontic action logic prominent in the literature are dynamic deontic logic [100] and deontic *stit* logic [77]. In dynamic deontic logic normative modalities are reduced to dynamic logic action modalities by using violation constants. Prohibition, for instance, is modeled as the dynamic logic conditional assertion that if the action is executed, a violation will occur. In deontic *stit* logic, the perspective on action is different. Where in dynamic logic actions are objects that are given proper names in the object language, in *stit* logic actions derive their identity from the agent(s) executing them and the effect they achieve. This allows for a proper theory of agency, ability and joint ability. In [77] normativity is introduced in *stit* theory by means of a deontic ideality ordering. But the alternative of violation constants has also been used in the *stit* context [22, 38].

3.1.6 Permissive norms

For a long time, it was naively assumed that permission can simply be taken as the dual of obligation, just as possibility is the dual of necessity in modal logic. Something is permitted if its negation is not forbidden. Nowadays in deontic logic a more fine-grained notion of permission is used. The notions of explicit permission, dynamic permission, and permission as exception to a pre-existing obligation are also used. (A dynamic permission is forward-looking, and is like a constitutional right – it sets limits on what can be forbidden). One main finding is that these normative concepts can all be given a well-defined semantics in terms of Input/Output logic [93, 33, 134, 133]. The main open problem concerns their proof-theory, which is still lacking.

3.1.7 Constitutive norms

So-called regulative norms describe obligations, prohibitions and permissions. So-called constitutive norms make possible basic 'institutional' actions such as the making of contracts, the issuing of fines, the decreeing of divorces. Basically they tell us what counts as what for a given institution. As pointed out in [32], constitutive norms have been identified as the key mechanism to normative reasoning in dynamic and uncertain environments, for example to realize agent communication and electronic contracting.

The paper [82] by Jones and Sergot is often credited for having launched the area. There the counts-as relation is viewed as expressing the fact that a given action "is a sufficient condition to guarantee that the institution creates some (usually normative) state of affairs." A conditional connective $\Rightarrow_s$ is used to express the "counts-as" connection holding in the context of an institution s.

When defining constitutive norms, the main issue is in defining their relation with regulative norms. To this end, Boella and van der Torre [30] use the notion of a logical architecture combining several logics into a more complex logical system, also called logical input/output nets (or lions).

3.2 Current research trends in NorMAS

3.2.1 New standard for deontic reasoning

A normative system is used to guide, control, or regulate desired system behavior. We can distinguish four traditional ways to look at normative reasoning in the deontic logic literature. Von Wright's system KD [145] distinguishes good and bad, or right and wrong. Anderson's reduction represents norms by their violation conditions. Hansson's preference-based semantics [72] makes it possible to represent tradeoffs among norms. Makinson [90] criticizes the hegemony of modal logic and proposes an alternative iterative approach. His iterative detachment approach and alternative candidates for a new standard represent the norms explicitly. In the *Handbook of deontic logic*, which is currently in preparation, the classical modal logic framework is mainly confined to the historical chapter. A chapter presents the alternatives to the modal framework, and three concrete approaches, input/output logic, the imperativist approach, and the algebraic conceptual implication structures or cis approach. There are also other candidates for a new standard, such as nonmonotonic logic or deontic update semantics.

Deontic formalisms should be able to capture more applied scenarios. For example, input/output logic is relatively abstract, so can it solve the problems left open by SDL? Just like SDL has been extended with all kinds of things, also input/output logic or abstract normative systems can be extended with all kinds of things. For several classical problems it has been shown that the input/output logic framework can give new insights. In my order of preference. It has been shown that the input/output logic approach to permission, the most classical problem of normative reasoning, has been a big step forward. Not only conceptually distinguishing kinds of norms, but also providing proof systems. Second, it has been shown that the existing semantic solutions to contrary-to-duty reasoning and dilemmas can be reproduced, leaving to a better framework for their formal analysis and comparison (such as proof rewriting techniques). Third, it has been shown that priorities among rules can be studied more systematically. Fourth, it has been shown that reasoning about obligations and time can be done more systematically, including proof systems. Input/output logic completely ignores agent interactions (which are fundamental for – say – social norms). The same holds for all other approaches. There is a very important challenge here, but also here it is crucial to have norms explicitly in the language. There are several contributions on games and deontic logic, but they do not make norms explicit.

3.2.2 The internal and the external

When an individual or a group of individuals is confronted with a number of possible choices, often the question arises of what that individual should do. In the history of deontic logic two perspective have been taken in modeling these type of concepts:

- In the first, norms assume an internal or utilitarian character: actions that are obligatory for a player (or a group of players) are those that are best for the player itself (or, in a general sense, meet the preferences of some players).
- In the second, norms assume an external or systemic character: choices are judged against predetermined interests, specified from outside the system. This is the classical view of deontic logic, which has its roots in Anderson's work, and has been explicitly connected to agency by Meyer's Dynamic Deontic Logic framework.

3.2.3 Expectations

Much MAS research has investigated the use of *commitments* and *norms* to provide social semantics and control to interactions within societies of agents [35, 55, 142, 29]. These constructs both represent socially contextualised constraints on the future behavior of agents, with the fulfilment or violation of these constraints having significance within some formal context (a specific formalised interaction protocol or the code of conduct of a society). Stripping away the social context, we are left with a less formal type of constraint: the expectations that an agent has about the future.

Expectations represent a potential future state of affairs that an agent has an interest in tracking over time (which may be represented by an explicit goal or some computational mechanism that implicitly embodies that goal). While they can be seen as the core aspect of commitments or instantiated norms, they can also arise for less formal reasons. For example, short-term team tactics in sport are based around expectations about the behaviours of other team members. Also, an agent may plan its practical reasoning around expectations that are justified by its own experience, but which need to be tracked in case they turn out to be violated in some situations.

3.2.4 Norms and argumentation

In law, Bench-Capon *et al.* present how argumentation theory has been used in legal reasoning. For instance, legal disputes arise out of a disagreement between two parties and may be resolved by presenting arguments in favor of each party's position. These arguments are proposed to a judging entity, who will justify the choice of the arguments he accepts with an argument of his own, with the aim to convince the public. The common conclusion shared by such works is that argumentation has the potential to become a useful tool for people working in the legal field. Even if a common answer from lawyers when they are asked about what argumentation theory can do for them is that it can be used to deduce the consequences from a set of facts and legal rules, and to detect possible conflicts, there is much more in argumentation. Following the example proposed by Bench-Capon et al., a case is not a mere set of facts, but it can be seen as a story told by a client to his lawyer. The first thing the lawyer does is to interpret this story in a particular legal context. The lawyer can interpret the story in several different ways, and each interpretation will require further facts to be obtained. Then the lawyer has to select one of the possible interpretations, she has to provide arguments to persuade the judging entity of the client's position, and to rebut any further objection. The major topics that emerge as relevant in norms and argumentation include, among others, case based reasoning, arguing about conflicts and defeasibility in rule based systems, dialogues and dialectics, argument schemes, and arguing about the successfulness of the attacks.

3.2.5 Logics for MAS and NorMAS

Several logical systems have been proposed in the last twenty years to model the properties of agents, multi-agent systems (MAS) and normative multi-agent systems (NorMAS). Among them we should mention Propositional Dynamic Logic PDL, Computational Tree Logic CTL , Coalition Logic CL and Alternating-time Temporal Logic ATL, STIT logic (the logic of "seeing to it that") by Belnap, Horty and coll., Dynamic Logic of Agency DLA. Some relationships between these different logical systems have been studied. For instance, it has been shown that both CL and CTL are fragments of ATL, that the 'strategic' variant of STIT logic embeds ATL and that DLA embeds both CL and STIT.

3.2.6 Visualizing normative reasoning

Tosatto et al. [136] promote the use of visual reasoning formalisms for normative reasoning, and insist that the formalism should have a clear and unambiguous semantics. Moreover, they believe that a visual formalism is best accompanied by a logical one, and they therefore refer to visualization of normative reasoning rather than a visual reasoning formalism. There are some related approaches. For example, for business processes there is Declare by van der Aalst's research group [113] (see also Marco Montali's book [101]), and Baldoni et al.'s proposal for commitments [21].

3.3 Norm change

There are two competing theories of norm change, developed as branch of theory change, and as theory of legal dynamics.

3.3.1 Revision of a set of norms

In general, a code G of regulations is not static, but changes over time. For example, a legislative body may want to introduce new norms or to eliminate some existing ones. A different (but related) type of change is the one induced by the fusion of two (or more) codes as it is addressed in the next section.

Little work exists on the logic of the revision of a set of norms. To the best of our knowledge, Alchourrón and Makinson were the first to study the changes of a legal code [8, 9]. The addition of a new norm n causes an enlargement of the code, consisting of the new norm plus all the regulations that can be derived from n. Alchourrón and Makinson distinguish two other types of change. When the new norm is incoherent with the existing ones, we have an *amendment* of the code: in order to coherently add the new regulation, we need to reject those norms that conflict with n. Finally, *derogation* is the elimination of a norm n together with whatever part of G implies n.

In [8] a "hierarchy of regulations" is assumed. Few years earlier, Alchourrón and Bulygin [6] already considered the *Normenordnung* and the consequences of gaps in this ordering. For example, in jurisprudence the existence of precedents is an established method to determine the ordering among norms.

However, although Alchourrón and Makinson aim at defining change operators for a set of norms of some legal system, the only condition they impose on G is that it is a non-empty and finite set of propositions. In other words, a norm x is taken to be simply a formula in propositional logic. Thus, they suggest that "the same concepts and techniques may be taken up in other areas, wherever problems akin to inconsistency and derogation arise" ([8], p. 147).

This explains how their work (together with Gärdenfors' analysis of counterfactuals) could ground that research area that is now known as *belief revision*. Belief revision is the formal studies of how a set of propositions changes in view of a new information that may cause an inconsistency with the existing beliefs. Expansion, revision and contraction are the three belief change operations that Alchourrón, Gärdenfors and Makinson identified in their approach (called AGM) and that have a clear correspondence with the changes on a system of norms we mentioned above. Hence, the following question needs to be addressed:

> How to revise a set of regulations or obligations? Does belief revision offer a satisfactory framework for norms revision?

Some of the AGM axioms seem to be rational requirements in a legal context, whereas they have been criticized when imposed on belief change operators. An example is the *success* postulate, requiring that a new input must always be accepted in the belief set. It is reasonable to impose such a requirement when we wish to enforce a new norm or obligation. However, it gives rise to irrational behaviors when imposed to a belief set, as observed for instance in [62].

On the other hand, when we turn to a proper representation of norms, like in the input/output logic framework, the AGM principles prove to be too general to deal with the revision of a normative system. For example, one difference between revising a set of propositions and revising a set of regulations is the following: when a new norm is added, coherence may be restored modifying some of the existing norms, not necessarily retracting some of them. The following example will clarify this point:

▶ **Example 1.** If we have $\{(\top, a), (a, b)\}$ and we have that c is an exception to the obligation to do b, then we need to retract (c, b). Two possible solutions are $\{(\neg c, a), (a, b)\}$ or $\{(\top, a), (a \wedge \neg c, b)\}$.

Future research must investigate whether general patterns in the revision of norms exist and how to formalize them.

3.3.2 Legal dynamics

One peculiar feature of the law is that it necessarily takes the form of a dynamic normative system [83, 73]. Despite the importance of norm-change mechanisms, the logical investigation of legal dynamics is still much underdeveloped.

As is well-known, the AGM framework distinguishes three types of change operation over theories. Contraction is an operation that removes a specified sentence ϕ from a given theory Γ (a logically closed set of sentences) in such a way as Γ is set aside in favor of another theory Γ^-_ϕ which is a subset of Γ not containing ϕ. Expansion operation adds a given sentence ϕ to Γ so that the resulting theory Γ^+_ϕ is the smallest logically closed set that contains both Γ and ϕ. Revision operation adds ϕ to Γ but it is ensured that the resulting theory Γ^*_ϕ be

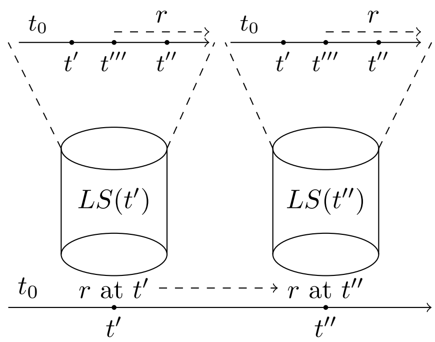

Figure 1 Legal System at t' and t''.

consistent [7]. Alchourrón, Gärdenfors and Makinson argued that, when Γ is a code of legal norms, contraction corresponds to norm derogation (norm removal) and revision to norm amendment.

AGM framework has the advantage of being very abstract but works with theories consisting of simple logical assertions. For this reason, it is perhaps suitable to capture the dynamics of obligations and permissions, not of legal norms. In fact, it is essential to distinguish norms from obligations and permissions [60, 68]: the latter ones are just possible effects of the application of norms and their dynamics do not necessarily require to remove or revise norms, but correspond in most cases to instances of the notion of *norm defeasibility* [68]. Very recently, some research has been carried out to reframe AGM ideas within rule-based logical systems, which take this distinction into account [132, 117]. However, also these attempts suffer from some drawbacks, as they fail to handle the following aspects of legal norm change:

1. the law usually regulate its own changes by setting specific norms whose peculiar objective is to change the system by stating what and how other existing norms should be modified;
2. since legal modifications are derived from these peculiar norms, they can be in conflict and so are defeasible;
3. legal norms are qualified by temporal properties, such as the time when the norm comes into existence and belongs to the legal system, the time when the norm is in force, the time when the norm produces legal effects, and the time when the normative effects hold.

Hence, legal dynamics can be hardly modeled without considering defeasibility and temporal reasoning. Some recent works (see, e.g., [68]) have attempted to address these research issues. All norms are qualified by the above mentioned different temporal parameters and the modifying norms are represented as defeasible meta-rules, *i.e.*, rules where the conclusions are temporalized rules.

If $t_0, t_1, \ldots, t_j$ are points in time, the dynamics of a legal system LS are captured by a time-series $LS(t_0), LS(t_1), \ldots, LS(t_j)$ of its versions. Each version of LS is called a *norm repository*. The passage from one repository to another is effected by legal modifications or simply by temporal persistence. This model is suitable for modeling complex modifications such as retroactive changes, *i.e.*, changes that affect the legal system with respect to legal effects which were also obtained before the legal change was done. The dynamics of norm change and retroactivity need to introduce another time-line within each version of LS (the time-line placed on top of each repository in Figure 1). Clearly, retroactivity does not imply that we can really change the past: this is "physically" impossible. Rather, we need to set a mechanism through which we are able to reason on the legal system from the viewpoint of its current version but *as if* it were revised in the past: when we change some $LS(i)$ retroactively, this does not mean that we modify some $LS(k)$, $k < i$, but that we move back from the perspective of $LS(i)$. Hence, we can "travel" to the past along this inner time-line, *i.e.*, from the viewpoint of the current version of LS where we modify norms. Figure 1 shows a case where the legal system LS and its norm r persist from time t' to time t''; however, such a norm r is in force in LS (it can potentially have effects) from time t''' (which is between t' and t'') onwards.

3.3.3 Dynamic logic approaches

Inspired by recent theoretical and technical developments in the logical study of dynamics—especially the dynamics of informational attitudes such as knowledge and belief[2]—some scholars have proposed models for the 'dynamification' of several kinds of deontic logics.

At the heart of these approaches lies the notion of structure transformation. Let us consider for instance a semantic analysis of obligations based on an ideality ordering among worlds. This semantics lends itself easily to a view of obligation dynamics based on ways of manipulating that ideality ordering. To make a simple example, following [140], the enactment of a command that ϕ be the case could be rendered by the modification of that ideality ordering in such a way that all ϕ-states are ranked as more ideal than all $\neg\phi$-states. The upshot is the modeling of different forms of norm dynamics in terms of different operations on their semantic structures. Other recent contributions along these lines, although based on different structures, are for instance [17, 15].

The advantage of this approach is to maintain a clear link with the underlying logical semantics of deontic notions. How the two perspectives can be technically bridged is very much an open issue.

3.4 Proof systems for deontic logic

This section is devoted to present the different proof systems available for different deontic logic formalisms. The first attempt to proof systems for deontic logic is probably one of Mally [94]. His formal system is based on the classical propositional calculus.

3.4.1 Standard deontic logic KD

Standard deontic logic is the monadic modal logic KD defined as the valid formulas on the class on serial frames. Sahlqvist theorem [119] gives an Hilbert style axiomatization made up with:

- all tautologies of classical propositional logic
- $O(\phi \rightarrow \psi) \rightarrow (O\phi \rightarrow O\psi)$
- $O\phi \rightarrow \neg O\neg\phi$
- modus ponens rule
- necessitation rule: from $\vdash \phi$ infer $\vdash O\phi$

A tableau system for KD exists: you extend the tableau system for K by the following rule:

$$\frac{(w\ O\phi)}{(w\ R\ v)(v\ \phi)}$$

[2] See [139] for a recent comprehensive overview

meaning that you add a successor in all labels containing a formula of the form $O\phi$. There exists an implementation of KD and variants (D4, *etc.*) in KED [14]. Such a tableau system is implemented in generic tableau provers [54], [123].

3.4.2 Dyadic deontic logic

Dyadic deontic logic has the dyadic modality $O(\psi \mid \phi)$ as primitive syntactical construct. It is read as "ψ is obligatory conditional upon ϕ being the case". Expressed in BNF notation, the syntax may be defined by

$$\phi ::= p \mid \neg\phi \mid \phi \wedge \psi \mid O(\psi \mid \phi)$$

The semantics is given in terms of Kripke models equipped with a binary relation $\geq$ defined on the universe of the model. The latter relation is used to rank possible worlds in terms of betterness. Compared to Kripke models based on a usual accessibility relation, the main novelty is that the semantics distinguishes various grades of ideality. This is needed to model the notion of CTD obligation, the antecedent of which refers to a sub-optimal situation where a primary obligation is violated. A Kripke model equipped with an accessibility relation uses a binary classification of worlds as good/bad. This binary classification is too rigid to reason about norm violation.

Formally a model becomes a triple $\mathcal{M} = (W, \geq, V)$ where:

- W is a non-empty set of worlds w, w', ...;
- $\geq$ is a binary relation on W; intuitively, $w \geq w'$ may be read as "w is at least as good as w' ";
- V is a valuation, which associates with each possible world a set of propositional formulae (intuitively, the set of those that are true at that world).

Intuitively the evaluation rule for the dyadic obligation operator puts $\bigcirc(\psi \mid \phi)$ true at a world in a model whenever ψ holds in all the best (according to ranking $\geq$) worlds where ϕ is true. This may be expressed as follows:

$w \models \bigcirc(\psi/\phi)$ iff $w' \models \psi$ for all w' such that

$$w' \models \phi \;\&\; \forall w'' \; (w'' \models \phi \Rightarrow w' \succeq w'' \,)$$

There are different systems of dyadic deontic logic depending on the constraints the relation $\geq$ satisfies. In the paper that launched the area, Hansson [72] distinguished three main systems, which he called DSDL1, DSDL2, DSDL3. They are defined as shown in the following table:

	constraints on $\geq$
DSDL1	reflexivity
DSDL2	reflexivity, and limitedness
DSDL3	reflexivity, transitivity, totalness, and limitedness

Roughly speaking, limitedness is the condition that any chain of strictly better worlds is finite. The role of the limitedness condition is, thus, to rule out infinite sequences of strictly better worlds.

It is possible to supplement the logic with additional operators. In particular, given limitedness, the universal modality $\Box$ may be introduced by means of the definition $\Box\phi \equiv$

$O(\bot \mid \neg\phi)$, where $\bot$ denotes a contradiction. It is also possible to add the unary modal operator $Q\phi$. Intuitively, $Q\phi$ says that we are in an ideal situation where ϕ holds. Note that, in a language with $\Box$ and Q has primitive syntactical constructs, $O(\psi \mid \phi)$ is equivalent to $\Box(Q\phi \to \psi)$. The latter just encodes into the syntax the truth-conditions used for the former.

Proof method for these logics is still an active research area. Known results are for DSDL3 mostly. Spohn [131] gave a weakly complete axiomatization for a fragment of DSDL3, with no iterated deontic modalities, and no truthfunctional compound propositions. Åqvist [10] extended Spohn's weak completeness result to the full language of DSDL3. Parent [109] strengthens Åqvist's result into a strong completeness one. A corollary of Spohn's weak completeness result is decidability of the system.

For DSDL2, a partial axiomatization result is available only. Parent [110] provides a strongly complete axiomatization using a language with Q and $\Box$ as primitive syntactical constructs. Åqvist [10] conjectured an alternative axiomatization using the dyadic obligation operator as primitive syntactical construct. This is his system F. His conjecture has not been settled yet.

The axiomatization problem for DSDL1 is still an open one. Åqvist [10] conjectured an axiomatization, which he called E. It is obtained from F by leaving out a suitable axiom. Åqvist's conjecture has not been settled yet.

Rönnedal gave 16 different tableau proof systems for different variants of dyadic deontic logic [116] where he changes the semantics: he works with a usual accessibility relation indexed by sentences. It remains to be seen what a tableau construction would be like in the original setting. Furthermore, the termination problem is not discussed.

3.4.3 See-to-it-that logic

See-to-it-that logic (*stit*) is a framework to deal with agency. Although it can be (and often is) studied completely independent of any normative context, the connection with normative issues is never far away. Agency and normativity are so closely entangled that this justifies looking at proof systems for *stit* in the context of this chapter.

See-to-it-that logic typically provides modal constructions $[J : stit]\phi$ meaning that the group of agents J sees to it that ϕ is true. The construction $[\emptyset : stit]\phi$ for empty set coalition stands for 'ϕ is necessarily true' and the operator $[\emptyset : stit]$ is called historical necessity. The semantics is given in terms of branching time and choice structures and the reader may refer to [23] for details. We give here the semantics of *stit* without time operator given in terms of Kripke structure [74].

Let AGT be a finite set of agents. A Kripke model for the logic *stit* is a tuple $\mathcal{M} = (W, R, V)$ where:

- W is a set of points;
- R is a mapping associating to every agent $i \in AGT$ an equivalence relation R_i on W such that for all $(w_1, w_2, \ldots, w_n) \in W^n$, $\bigcap_{i \in AGT} R_i(w_i) \neq \emptyset$;
- V is a valuation function.

Intuitively, R_i is nothing more than the equivalence relation corresponding to the choice of agent i. When $u \in R_i(w)$ then agent i's current choice at w cannot distinguish between w and u. The truth conditions are given by:

- $\mathcal{M}, w \models [J : stit]\phi$ iff for all $u \in \bigcap_{i \in J} R_i(w)$, we have $\mathcal{M}, u \models \phi$.

There exists a variant of *stit* called deliberate *stit* providing only the historical necessity and constructions $[J : dstit]\phi$ standing for 'J deliberatively sees to it that ϕ is true'. The construction $[J : dstit]\phi$ is defined as a macro of $[J : stit]\phi \wedge \neg[\emptyset : stit]\phi$.

A Hilbert axiomatization is said to be orthodox if the axiomatization is defined by a finite set of axioms schemas (then all instances of the axioms where we substitute any proposition by a formula are theorems) and the two following rules: modus ponens and necessitation rule. Unfortunately there is no so called orthodox finite Hilbert axiomatization for Chellas' *stit* if there are more than 3 agents in the system [74].

Nevertheless when we consider syntactic fragments of Chellas' *stit*, there may exist some Hilbert axiomatizations. For instance, there exists an orthodox axiomatization for the individual *stit* that is the fragment where we only allow construction $[\emptyset : stit]\phi$ and constructions $[\{i\} : stit]\phi$ for individual agents. For individual deliberative *stit* ($[\emptyset : stit]\phi$ and $[\{i\} : dstit]\phi$ for all agents i), Xu's gave an orthodox finite Hilbert axiomatization in [150]. A finite axiomatization for individual Chellas' *stit* and an alternative axiomatization for deliberative individual *stit* may be found in [20]. The axiomatization of individual Chellas' *stit* is given by:

- $S5$ axioms for each modality $[\emptyset : stit]$ and $[\{i\} : stit]$;
- $[\emptyset : stit]\phi \rightarrow [\{i\} : stit]\phi$;
- $\Diamond[\{1\} : stit]\phi_1 \wedge \ldots \Diamond[\{n\} : stit]\phi_n \rightarrow \Diamond([\{1\} : stit]\phi_1 \wedge \ldots [\{n\} : stit]\phi_n)$.

where $\Diamond$ is the dual operator of $[\emptyset : stit]$. The last axiom is interesting and states the independence of agents.

Then, if the set of allowed coalitions in the language are $\emptyset$, $\{i\}$ then there is an axiomatization. Schwarzentruber [122] exhibits a bigger set of allowed coalitions in the language such that there exists a finite Hilbert axiomatization. Lorini and Schwarzentruber [89] exhibit a fragment of Chellas' *stit* logic, with a restriction on the modal depth and an axiomatization for it is given. An axiomatization of *stit* plus the linear temporal logic operators is given by Lorini [88]. Non-terminating tableaux for deliberative individual *stit* are given by Wansing [147].

3.4.4 Other formalisms

In general methods are imported from other areas, in particular from conditional logic and the logic of counterfactuals [86], and van der Torre and Tan [143] uses Boutilier's axiomatization in modal logic [36] to represent DSDL3 and several other logics such as Prohairetic Deontic Logic.

Meyer [100] proposes an approach based on Propositional Dynamic Logic PDL with an atom to designate a violation situation. He gives the Hilbert style axiomatization of PDL. Balbiani [19] also proposes an alternative deontic logic based on PDL.

Goble [66] gave strongly complete axiomatizations for logics incorporating multiple accessibility relations and multiple betterness ranking on alternative worlds to represent distinct normative standards The language uses two monadic modal operators: $O_e\phi$ and $O_a\phi$. The first says that there exists a normative standard in which ϕ is obligatory, and the second says that it is so according to all the normative standards. There are open axiomatization problems for the dyadic counterparts to these modalities.

Alternative kinds of proof systems are developed in input/output logic [91, 92].

4 Research challenges

This section presents a number of fundamental challenges for deontic logic and normative systems. They are represented by research areas with contiguous interests – such as the study of norms – that we believe constitute a frontier where the machinery of deontic logic and normative systems can show its added value.

Subsection 4.1, titled *Norms and Games*, presents a summary of game-theoretical approaches to norms. In particular it distinguishes between two understanding of norms in the field: norms as rules of the game, the so-called mechanism design perspective; and norms as equilibria, the so-called stable state perspective. We believe that deontic logic and normative systems, applying their reasoning tools, can make explicit several foundational issues in norms and games.

Subsection 4.2, titled *Norms and Responsibility*, introduces two dimensions of responsibility in multi-agent systems, namely: (i) responsibility as in 'who did it?', which refers to agents performing an action violating some prescription, (ii) responsibility as in 'who is to blame?' referring to the agents who are instead to be accountable for the damage brought about. The two dimensions do not necessarily coincide, but only a precise modelling of their properties and their consequences can help establishing responsibility in a normative system.

Subsection 4.3, titled *Abstract and Concrete*, shows that normative concepts do not share the same level of detail. We go from extremely general laws, such as constitutional rights, to extremely concrete ones, such as civil law regulations. This part presents an interesting connections between the level of concreteness of regulations and the type of actions that they recommend and sketches an interesting similarity with the logics for ability needed to reason about them.

Subsection 4.5, titled *Visualization*, argues that unlike several important areas of multi-agent systems – such as argumentation – deontic logic and normative systems still lack a representation that is easy to visualize and work with – such as Dung graphs for the case of argumentation.

Subsection 4.6, titled *Proof Methods*, argues about the need to have more general results in the field, such as systematic tableau methods and correspondence results. For instance several important formalism employed in the area of deontic logic, such as dyadic deontic logic, are still not well understood in terms of better-behaved normal modal logics. Bridging the gap would allow to transfer the variety of results available for the latter – such as Salqvist completeness – to the former.

Subsection 4.7, titled *From Deontic Logic to Norms and Policies*, discusses the relation between the formalisms employed to reason about computer systems and their actual implementation. The chapter touches upon subjects such as policies, that are widespread in the practice of disciplines such as security and software engineering.

Subsection 4.8, titled *Expectations*, starts from the observation that norms can have an impact on the practical reasoning of individual agents. Then it goes on arguing that a norm-aware agent must consider the constraints that the prevailing norms impose on its own future behaviour, but it can also benefit by considering how those norms constrain the behaviour of other agents.

Subsection 4.9, titled *Agreement Technologies* presents challenges for Agreement Technologies. In particular it revisits the metaphor of sandbox, which sees methods and mechanisms from the fields of semantic alignment, norms, organization, argumentation and negotiation, as well as trust and reputation are part of a "sandbox" to build software systems based on a technology of agreement. This parts presents an input/output perspective on it.

	C	D
C	2, 2	0, 3
D	3, 0	1, 1

	L	R
L	1, 1	0, 0
R	0, 0	1, 1

Figure 2 Prisoner's dilemma (with C=cooperate and D=defect) and Coordination game (with L=left and R=right).

4.1 Norm and games

Generally speaking, the contributions in the literature in the intersection between games and norms[3] can be divided into two main branches: the first, mostly originating from economics and game theory [47, 79, 80], exploits normative concepts, such as institutions or laws, as *mechanisms* that enforce desirable properties of strategic interactions; the second, that has its roots in the social sciences and evolutionary game theory [138, 48] views norms as *equilibria* that result from the interaction of rational individuals.

4.1.1 Norms as mechanisms

This section presents the view of norms as constraints that, once imposed on players' behaviour, enforce desirable social outcomes in games. In this view, norms are conceived as the rules of the game[4], and it is the most common approach to norms within the so-called New Institutional Economics[5]. An interpretation of this view from the standpoint of game theory is developed in [79], which models the rules of the game in terms of the theory of mechanism design.

In brief, institutions are seen as collective procedures geared towards the achievement of some desirable social outcomes[79]. An example of them are auctions, *viz.* mechanisms to allocate resources among self-interested players. In many auctions goods are not assigned to the bidder valuing them most as bidders might find it convenient to misrepresent their preferences. In such situations mechanism design can be used to enforce the desirable property of truth telling. For instance, when the bidders submit independently and anonymously and the winner pays an amount equivalent to the bid of the runner-up, truth telling is a dominant strategy.[6] In other words, in a second-price sealed bid auction, independently of the way bidders value the auctioned good, they cannot profitably deviate from stating their preferences truthfully.

Viewing norms as mechanisms assigns to norms the same role as auctions. Just like in auctions, norms are supposed to make no assumptions on the preferences of the participating agents. They merely define the possible actions that participants can take, and their consequences. Slightly more technically, they are *game forms* (or mechanisms), *viz.* games without preferences.

Two aspects of this view are worth stressing. First, it clearly explains the rationale for norms and institutions: they exist to guarantee that socially desirable outcomes are realized as equilibria of the possible games that they support (*implementation*). Second, it

3 For a more detailed discussion on these topics see [69].
4 As far as we know, this locution has been introduced in [106].
5 New Institutional Economics has brought institutions and norms to the agenda of modern economics, viewing them as the social and legal frameworks of economic behaviour. See [47] for a representative paper.
6 This is the so-called Vickrey auction. See [125, Ch. 11] for a neat exposition.

presupposes some sort of infallible enforcement: implementation can be obtained only by assuming that players play within the space defined by the rules, which represents a strong idealization of how institutions really work.[7]

The view of norms as mechanisms is by no means limited to economic analysis of interaction, but it has been also successfully applied in computer science to regulate the behaviour of computer systems. A game-transformation approach has been pioneered by [126] in order to engineer laws which guarantee the successful coexistence of multiple programs and programmers. It has been further explored in the multi-agent systems community in [141], to study temporal structures obeying systemic requirements, and [41] , which has made the role of norms explicit in leading players' behaviour to a desirable outcome.

4.1.2 Norms as equilibria

Starting from the classical problem of the spontaneous emergence of social order, the game-theoretic analysis of norms has focused in particular on informal norms enforced by a community of agents, *i.e. social* norms. From this perspective, the view of norms as Nash equilibria has been first suggested by Schelling [121], Lewis [85] and Ullmann-Margalit [138]. A Nash equilibrium is a combination of strategies, one for each individual, such that each player's strategy is a best reply to the strategies of the other players. Since each player's beliefs about the opponent's strategy are correct when part of an equilibrium, this view of norms highlights the facts that a norm is supported by self-fulling expectations.

However, not every Nash equilibrium seems like a plausible candidate for a norm. In the Prisoner's Dilemma (see Figure 2) mutual defection is a Nash equilibrium of the game without being plausibly considered a norm-based behavior. In fact, the view of norms as Nash equilibria has been refined by several scholars. Bicchieri [24], for instance, has suggested that, in the case of norms conformity is always *conditional* upon expectations of what other players will do. Moreover, in this model, norms are different from mere conventions, in that norms are peculiar of mixed-motives games (e.g. the Prisoner's Dilemma) and operate by transforming the original games into coordination ones.

Another influential view of norms characterized them as devices that solve equilibrium selection problems. A comprehensive and concise articulation of this view can be found in [26] which emphasizes two key features of norms. First, as equilibria, they determine self-enforcing patterns of collective behavior[8], e.g., making cooperation an equilibrium of the (infinitely iterated) Prisoner's Dilemma. Second, since repeated interaction can create a large number of efficient and inefficient equilibria, a norm is viewed as a device to select among them—a paradigmatic example of a game with multiple equilibria is the game on the right in Figure 2, known as the coordination game.

Finally, it has been recently suggested that a norm is best captured as a correlating device that implements a correlated equilibrium of an original game in which all agents play strictly pure strategies [64]. A correlated equilibrium is a generalization of the Nash equilibrium concept in which the assumption that the players' strategies are probabilistically independent is dropped. When playing their part on a correlated equilibrium the players condition their choice on the same randomizing device [18]. Since the conditions under which a correlated

[7] This problematic assumption has been put under discussion extensively in [80].

[8] Self-enforcement is the type of phenomenon captured by the so-called *folk theorem*. The theorem roughly says that, given a game, any outcome which guarantees to each player a payoff at least as good as the one guaranteed by her minimax strategy is a Nash equilibrium in the infinite iteration of the initial game (cf. [108, Ch. 8]).

equilibrium is played are less demanding than those characterizing Nash equilibria, the view of norms as a correlating device seems more plausible. Moreover, the correlating device is seen as a device that suggests separately to each player what she is supposed to do and thus seems to better characterize the prescriptive nature of norms [49]. On the other hand, since such correlating devices are viewed as an emergent property of a complex social system, their origins is left unclear.

Although an equilibrium-based analysis of norms might provide a rationale for compliance, it does not explain how such norms can possibly arise in strictly competitive situations—like the Prisoner's Dilemma. Such explanation can be obtained, on the other hand, by adding an evolutionary dimension to the standard game-theoretic framework, as studied for instance in [129].

4.2 Norms and responsibility

The notion of responsibility has two connotations. On the one hand, there is responsibility as in 'who did it?'. But another use of the term responsibility identifies it with 'who is to blame?'. These two forms of responsibility are closely related, but do not coincide; an employee can be responsible for something that happened in the sense that he/she intentionally conducted the activity, without the employee being responsible for what happened in the legal or moral sense; it might be, for instance, that according to the regulations his/her superior is to blame.

Normative systems define who is to blame for what circumstances under which conditions. A standard example is formed by our systems of law. Other examples are systems of moral values, religious commandments, social conventions, *etc.* A normative system defines (either explicitly, as in the law, or implicitly through the collective beliefs of some society, as in social conventions) if an agent that is responsible for something that occurred or might occur (maybe due to some other agent) is in violation from the point of view of that system. Deontic logic models the reasoning of agents having to make decisions and draw normative conclusions in the context of a normative system (What do I have to do? What am I allowed to do? What am I forbidden to do?). Although deontic logic has now been studied for over 60 years, only fairly recently a connection with agency and different forms of responsibility was made [77][23]. Exploring the logical connections between agency, responsibility and normative systems is one of the challenges for the theory of normative systems in the near future.

Responsibilities can be the result of commitments agents made to other agents (or themselves). Baldoni et al. [96] face the problem of defining control, safety, and responsibility in a chain of commitments.

Part of the challenge is the search for logical theories about *degrees* of responsibility and their associated *degrees* of blame relative to a normative system. In this context it is natural to look at probabilities. When we think of responsibilities, probabilistic action plays a central role [149]. For instance, the responsibility for an action may be related to the (subjective) chance of success for that action; if an agent does not have full control over the outcome of an action it can only be partly responsible for bad outcomes. Also, it is very natural to think of having responsibilities relative to a normative system as having to optimize the chance to obey obligations and having to avoid the risk of violations. With the exception of [39] very little is known about logical models relating probability, agency and normative systems.

4.3 Abstract versus concrete norms

In computer science, abstraction is one of the techniques that can be found in almost any sub-area. For instance, in model checking, abstraction is used to get a handle on the complexity of search spaces. In software design abstraction is used for the traceability of requirements. In planning theory, abstraction is used in HTN planning [57]. In logic, abstraction is used in generalized logic.

It makes sense then to assume that one of the new challenges for normative systems in computer science is to integrate them with abstraction techniques. We briefly discuss three sub-areas where abstraction already plays a modest role and that are promising candidates for coming to a more general view on abstraction and norms.

Abstraction of actions. First, in the area of deontic action logic, two views emerged. The first is Meyer's work on dynamic deontic logic [100]. Here the idea is that deontic action logic can be reduced to dynamic logic plus a violation constant. Dynamic logic is a formalism designed for reasoning about pre- and postconditions of basic and complex programs. So the central element of Meyer's contribution is the claim that reasoning about agentive action can be modeled as reasoning about programs. The second view is the *stit* (seeing to it that) view, and it has its background in philosophy [23]. Here the view on action itself is more abstract; an act is a relation between an agent and what this agent achieves. Now we can see *stit* actions as abstractions of dynamic logic actions. The *stit* view on action is close to the view in HTN planning: using *stit* operators we can reason about abstract action and about how they can be refined into more concrete action. In dynamic logic, this is exactly the other way around: we can reason about concrete basic action and how they relate to more complex action expressed in terms of them.

Abstraction of normative systems. Second, also the difference between, for instance, constitutional law and normal law can be seen as a matter of abstraction. Constitutional law might be seen as setting the general, more abstract stage for normal law to take effect. This is a different form of abstraction, that applies to the normative systems themselves and not to the actions that are regulated by such systems. Whether abstraction of normative systems takes the form of orderings over such systems [8], of meta-level descriptions, or of any other relation between them, is one of the challenges.

Abstraction of norms and normative contexts. Third, the difference between general norms, independent of a particular moment, a particular agent, and a particular choice situation, and specific obligations relative to a point in time, an agent and the action it chooses, can also be easily viewed as a form of abstraction. Here the difference is between concrete norms (obligations) and abstract, more general norms. This form of abstraction does not concern normative systems as a whole and also does also not concern regulated actions. What the relation with these other forms of abstraction is, is again one of the challenges.

4.4 Visualization of normative reasoning

Successful reasoning formalisms in Artificial Intelligence such as Bayesian networks, causal networks, belief revision, dependence networks, CP-nets, Dung's abstract argumentation systems, come with intuitive and simple visualizations. Traditionally deontic logic has been associated with preference orders [72], which have an intuitive visualization too. However, it is less clear how to extend this visualization of pre-orders to other aspects of normative reasoning.

In general, we see two approaches to visualization, depending on the audience for which the visualization is developed. On the one hand, we may aim to illustrate a derivation in all its details, and on the other hand, we may look for an abstract approach that visualizes the rough structure of normative reasoning, hiding the more detailed structure. Such an abstract approach may also be used to summarize a more complex derivation. In this paper we follow the latter approach. We thus aim at a visualization that can be understood by non-experts in normative reasoning.

An intuitive and simple visualization for abstract normative systems is important to make them adopted in real applications. The idea shares the motivation with Dung's argumentation networks for non-monotonic reasoning [56], with visual languages such as UML for object-oriented software engineering[9] [118], and TROPOS-like visual representation of early and late requirements[10] [102, 103], *etc.*

Though we promote the use of visual reasoning formalisms for normative reasoning, we insist that the formalism should have a clear and unambiguous semantics. Moreover, we believe that a visual formalism is best accompanied by a logical one, and we therefore refer to visualization of normative reasoning rather than a visual reasoning formalism.

It would be beneficial if the notation were suited both for printed documents and hand-written notes, or machine-processable and paper-and-pencil, which might mean that shading and dashed circles are best avoided. However, we expect that we need more advanced techniques than just diagrams that can be printed, for example using interactive visualizations.

4.5 Proof methods

4.5.1 Dyadic deontic logic

One first important issue is to resolve the axiomatization problem and the decidability problem for DSDL1 and DSDL2 proposed by Hansson.

In dyadic deontic logic there is no systematic way to obtain an axiomatization from a specific class of frames. This is because of the form the evaluation rule for the dyadic obligation operator has.

Another important issue would be to obtain a general correspondence between axiomatization and constraints on the semantics. The idea is not only to focus on DSDL1, DSDL2 and DSDL3 proposed by Hansson but also new systems. One may wonder what the axiomatization of dyadic deontic modal would be like when the relation $\leq$ is reflexive and euclidian. The aim is to obtain a general result in the same flavour that the Sahlqvist theorem gives such a correspondence in normal modal logic [119].

Another main concern is automated reasoning and especially addressing the satisfiability problem. Among all the existing type of proof systems, tableaux are good candidate for providing algorithms. This is especially true for non-classical logics as modal logic, intuitionistic logic and description logic. Indeed, in tableaux contrary to Hilbert or sequent calculus, most of the rules are deterministic in such that a way that the calculus is guided. Furthermore rules are often designed so that terms that are generated are getting strictly smaller and smaller so that it guarantees that the rewriting process of the proof system is terminating. In that case, we say than tableaux rules are strictly analytic according to Fitting's terminology [58]. If some rules are not strictly analytic the termination is no more guaranteed and there are some loop-check techniques to enforce termination without loosing

[9] http://www.uml.org/
[10] http://www.troposproject.org/node/120

soundness and completeness of the algorithm. This is classical for modal logic S4 of reflexive and transitive frames [75].

Here an important problem is to devise proof systems – for instance tableaux – that can be turned into an algorithm. For this, it would also be interesting to have a generic result: for instance we may treat decidability at once for DSDL1, DSDL2, DSDL3 by providing a set of common tableaux rules extended with specific rules for each logic DSDL1, DSDL2, DSDL3.

4.5.2 *stit* challenges

No proof-theory for *stit* logics integrating the deontic modalities is available. A first challenge would be to devise one such.

In Chellas' *stit* logic, we conjecture that a syntactic fragment of it does not distinguish *stit* models from super additive ones if, and only if there exists an orthodox Hilbert finite axiomatization that generates all the validities of the fragment.

There is also a long avenue concerning the proof system – such as tableaux – that may be transformed into an algorithm for providing effective procedures to deal with the satisfiability problem of some fragments of *stit* logic.

4.6 From deontic logic to norms and policies

The deontic concepts of permission, prohibition and obligation have received attention in disciplines other than deontic logic. This is primarily due to the practicality of the notions studied by deontic logic as these notions regulate and coordinate our lives together, making deontic logic valuable for the study of topics of considerable practical significance such as morality, law, social and business organizations (their norms, as well as their normative constitution), and security systems [98].

Among disciplines where deontic logic is relevant is research on security policy languages and models. Security policy languages aim at the practical specification of security requirements in information systems, whereas security models identify the basic concepts and elements necessary to study and analyze these requirements. Security policies and models are not separated concepts as policy languages are often underpinned by a security model. In security policy languages, permissions and prohibitions are used to specify access control requirements [120, 81, 2] whereas obligations have been used to specify requirements such as availability [53], privacy [104] and usage control [112, 152].

Although studying the same concepts, deontic logic and research on security policy languages and models have different research objectives. As a branch of symbolic logic, deontic logic is primarily interested in the study of *valid* inferences when deontic concepts are considered. More specifically, it is often the case that modal logics and Kripke's possible worlds semantics are used to provide a formal view of the semantics of deontic concepts. This allows the proof of the completeness and soundness of the axiomatic with respect to the possible world semantics and the analysis of various deontic puzzles.

On the other hand, security policy languages are more concerned with the clear specification, analysis and enforcement of a system's security requirements, rather than studying valid inferences from deontic concepts. For this reason, norms in policy languages are generally expressed using less abstract constructs than those used in deontic logics for a clear and intuitive representation of requirements. Policy languages also consider the computational aspects of the language to allow efficient analysis and enforcement of system requirements. For this reason, policy languages are generally formalized using tractable fragments of first-order predicate logic such as Datalog [137, 1].

Another possible way to compare deontic logic and security policies is to consider them as models of deontic concepts. A *model* is typically a symbolic or physical representation of a concept or an object that is intended to simplify the understanding, validation or analysis of the modeled concept or object. A model should be simple with great explanatory power. Since these two requirements are typically contradictory, a trade-off often occurs between the simplicity and efficiency of the model on one hand and its explanatory power on the other hand. Since security models (associated with policy languages) and deontic logics are symbolic representations of the deontic concepts of permission, prohibition and obligation, they may be thought of as models which provide the necessary elements and tools to understand, analyze, and/or enforce deontic concepts. In this context, deontic logics are general models for reasoning about deontic concepts whereas security models are more application-oriented but also more efficient and allow clearer specification of norms.

One research challenge is therefore bringing together research on deontic logic and security policy languages and models. This should be beneficial to both communities as it would add to the practicality of deontic logic and provide a more rigorous and formal foundation to research on security policies and languages. Representative research work that considers the use of deontic logic reasoning style and the specification of practical security requirements are [63, 16].

4.7 Norms and expectations

Norms can have an impact on the practical reasoning of individual agents. A norm-aware agent must consider the constraints that the prevailing norms impose on its own future behaviour, but it can also benefit by considering how those norms constrain the behaviour of other agents. If the agent has reason to be confident that other agents will follow the norms, then its own planning can be simplified by adopting this assumption. However, the agent has an interest in knowing whether the norms are indeed followed by the other agents that it interacts with. If its assumption of their compliance turns out to be unwarranted, then its intentions and plans must be reconsidered. Thus, norms induce *expectations*: anticipations of the future course of events or states of affairs, together with a goal to know whether the anticipated future eventuates. This goal may be represented explicitly [42] or by some monitoring mechanism that implicitly embodies it [51].

An activated norm gives rise to an expectation that represents the core temporal content of an activated norm, where the key concern is under what conditions the expectation becomes fulfilled or violated, and in what form the expectation persists from one state to the next if it has not been fulfilled or violated, and other issues such as the source of the norm and the consequences of violation are left to a more specialised and contextual layer of social reasoning. While the definition of norm fulfilment, violation and persistence is often trivial in many normative languages, especially those in which norms express predicates that should hold in some ideal state, these concepts are more subtle when expectations can have a complex temporal structure, e.g. after paying a magazine subscription I expect to receive an issue each month for a year [51].

While expectations can arise from norms, they can also arise from contracts, from commitments resulting from agent interaction, from joint plans (e.g. in team plays in sport), or simply from agents' own observations of the regularities in their environments. Thus, studying techniques for formally modelling and reasoning about expectations promises to lead to a unified treatment of commitments, norms and other constructs, with the notions of future expectation, fulfilment and violation as a core module. Furthermore, while much work on deontic logic has focused on norms with a propositional content, sometimes with the additional of a deadline, expectations arising for non-normative reasons (e.g. joint plans) may lead to requirements for greater temporal expressiveness. The development of expectation languages and reasoning techniques that meet these requirements can, in a modular account of norms and expectations, also lead to richer representation of the temporal aspects of norms.

4.7.1 Current understanding

Expectations have been studied in a number of different settings, as outlined below.

Castelfranchi and colleagues have studied the role that expectation plays as a form of mental anticipation and its relationship with conventions, commitments, obligations, emotions and trust [43, 42]. In their approach, an expectation combines a belief, representing a mental anticipation of a future state or event, with a goal to know whether the anticipated future occurs as predicted. This can be seen as a precursor to the development of social norms, conventions, and commitments. This work has been applied in the development of a computational model for agents that combines a BDI engine with expectation-driven deliberation and an affective control mechanism based on expectations [114].

Alberti et al. have proposed modelling agent interaction protocols using an explicit representation of expectations [4]. In this approach, protocols are expressed using logical rules defining how future expectations on agents' communicative acts arise from observations of current and past communicative acts. Time is treated using explicit time variables and comparisons between the times of different events. The abductive proof procedure $\mathcal{S}$CIFF [3], which includes positive and negative expectation predicates, is used to verify agents' compliance with protocols. An extension of the event calculus based on $\mathcal{S}$CIFF has been proposed for the specification of the social semantics of agent interactions in terms of commitments, and their run-time verification [44]. It has also been shown how the deontic logic concepts of obligation, prohibition and permission can be mapped to the notion of expectation used in this line of research [5].

Cranefield and colleagues have focused on the temporal aspect of expectations, and developed a logic for modelling the activation, fulfilment and violation of rules of expectations with a rich temporal structure, as well as an associated model checking technique for monitoring expectations [50, 51, 52]. Initial work used a first order metric interval temporal logic (with guarded quantification) [50], but more recent work has focused on a propositional linear temporal logic with future and past time operators and some hybrid logic features. The work has been applied in monitoring agents' expectations in the Second Life virtual world[11] [51, Section 6.2], and the expectation checker has been integrated with the Jason BDI interpreter [115].

Nickles et al. [105] introduced the notion of *expectation-oriented modelling*, in which explicit representations of agent expectations are used both as part of the agent design process and agents' run-time execution. In this approach, agent interactions are specified using a graph-based formalism called *expectation networks* in which nodes represent event occurrences and annotated edges encode information about how the occurrence of events result in expectations of other subsequent events.

Wallace and Rovatsos [146] defined an approach for specifying and executing an agent's

[11] `http://secondlife.com`

practical social reasoning in terms of expectations. They consider an expectation to be a conditional belief that is associated with a specific test condition that will (eventually) confirm or refute the expectation. The agent's pre-existing knowledge of its social context is encoded by defining, for each expectation, how positive and negative test results will result in the activation and deactivation of expectations. From this, an "expectation graph" is derived, representing the possible transitions between sets of active expectations. Another set of rules is defined to specify when agent actions should be generated based on the agents' current beliefs and queries on the expectation graph.

4.7.2 Open research questions

Some open research questions related to expectations are listed below. Some of these questions apply equally well to norms, but may be usefully addressed by focusing on expectations in the first instance, without the added complications of social context and the questions of how the expectations are initiated and what the consequences of fulfilment and violation are.

- What is the relationship between expectations, commitments and norms? Can existing approaches to modelling commitments and norms be expressed in terms of expectations plus additional social context? Is there a common formal model of expectation that can be seen to underlie a range of approaches?
- How expressive can formal models of expectation be while still allowing tractable run-time monitoring? For example, can an appropriate guarded fragment of first order predicate logic be combined with a temporal logic for convenience of encoding complex expectations? What monitoring techniques can be used with different restrictions on expectation expressiveness?
- What techniques can be used to answer questions such as the following, for different levels of expectation expressiveness? (i) Is a set of expectations consistent? (ii) What outcomes are possible for an agent assuming that a given a set of expectations will be fulfilled? (iii) What plans are consistent with a set of expectations? (iv) How can an agent plan to meet its goals and fulfil any expectations applying to its own behaviour, given a set of expectations?
- What are desirable properties for the semantics of expectation, for example, what closure conditions should apply to a set of expectations? If an agent expects ϕ and also expects ψ, should it also expect $\phi \land \psi$? While this might seem plausible, if expectations consist of expected beliefs and goals to track them, as proposed by Castelfranchi, then should the agent have a goal to track the truth of $\phi \land \psi$?
- Can existing approaches for defining the model-theoretic semantics for commitments (such as the work of Singh [127]), or normative concepts such as obligation, be adapted to include expectations as a core module?
- How can the connection between expectations and practical reasoning architectures, such as the BDI approach, be formalised?

Below, we sketch out one possible way that the first question above might be addressed.

4.7.3 Towards a formalism linking expectation, commitments and norms

This section briefly (and somewhat informally) illustrates how a logic of expectations can be used to provide common semantics for the fulfilment and violation of commitments and norms with rich temporal content. The presentation is based on the logic of Cranefield and

Winikoff [51]. A shorter overview of the key aspects of the logic is presented by Cranefield et al. [52, Section 3].

Commitments and norms have been formalised and operationalised using a wide range of formalisms and computational mechanisms.[12] This section adopts a combination of the event calculus [84, 124] and the logic of expectations.[13]

The event calculus is a formalism for specifying the effects of actions. Amongst many other uses, it has been used directly or in a modified form to specify and reason about agent interaction protocols in terms of commitments [151] and to define norm-governed multi-agent systems [13, 46].

4.7.3.1 Expectations and commitments

We begin by considering the specification of agent communication acts in terms of commitments, following the ideas (but not the formalism) of Verdicchio and Colombetti [144]. A partial specification in terms of the event calculus might look like this:

$$\begin{aligned}
&\mathit{initiates}(\mathit{inform}(x,y,\phi),\ \mathit{comm}(t,x,y,\phi),\ t) && (1)\\
&\mathit{initiates}(\mathit{request}(x,y,\phi),\ \mathit{precomm}(t,y,x,\phi),\ t) && (2)\\
&\mathit{terminates}(\mathit{accept}(x,y,\phi,t_1),\ \mathit{precomm}(t_1,x,y,\phi),\ t_2) && (3)\\
&\mathit{initiates}(\mathit{accept}(x,y,\phi,t_1),\ \mathit{comm}(t_1,x,y,\phi),\ t_2) && \\
&\qquad \leftarrow \mathit{holds_at}(\mathit{precomm}(t_1,x,y,\phi),\ t_2) && (4)\\
&\mathit{terminates}(\mathit{refuse}(x,y,\phi,t_1),\ \mathit{precomm}(t_1,x,y,\phi),\ t_2) && (5)
\end{aligned}$$

This states that the sending of an inform message from x to y with content ϕ establishes a fluent (dynamic predicate) expressing that a commitment holds from x to y that ϕ is true. A request initiates a precommitment, which is terminated when the request is accepted or refused (the *request* and *refuse* communicative acts include the time that the precommitment was established in order to disambiguate different requests with the same content). If the request is accepted, a commitment is established. The time at which the commitment (or precommitment, if applicable) was established is recorded in the *comm* fluent. This is important for linking commitments (with their additional social context, x and y) to expectations.

We now model the relationship between commitments and expectations:

$$\begin{aligned}
&\mathit{holds_at}(\boldsymbol{exp}(\mathit{true},\phi,t,\phi),\ t) \leftarrow \mathit{holds_at}(\mathit{comm}(t,x,y,\phi),\ t) && (6)\\
&\mathit{holds_at}(\mathit{fulf_comm}(t_1,x,y,\phi),\ t_2) && \\
&\qquad \leftarrow \mathit{holds_at}(\mathit{comm}(t_1,x,y,\phi),\ t_2) \wedge \mathit{holds_at}(\boldsymbol{fulf}(\mathit{true},\phi,t_1,_),\ t_2) && (7)\\
&\mathit{holds_at}(\mathit{viol_comm}(t_1,x,y,\phi,\psi),\ t_2) && \\
&\qquad \leftarrow \mathit{holds_at}(\mathit{comm}(t_1,x,y,\phi),\ t_2) \wedge \mathit{holds_at}(\boldsymbol{viol}(\mathit{true},\phi,t_1,\psi),\ t_2) && (8)
\end{aligned}$$

Here, constructs from the logic of expectations are written in bold. $\boldsymbol{exp}(\lambda,\rho,t,\phi)$ states that ϕ *would be* expected to hold currently if there were a rule of expectation with condition λ and content ρ, due to that rule having fired in the (possibly prior) state at time t. Similarly, ***fulf*** and ***viol*** express the fulfilment or violation of a current expectation. Note that these formulae define expectation, fulfilment and violation *relative* to the rule represented by the first two

[12] An partial survey is given by Cranefield et al. [52].
[13] This combination has not yet been formalised or implemented.

arguments. In any state, a countably infinite number of instances of these formulae will hold (e.g. all possible unconditional rules would result in a current expectation). It is therefore up to an implemented system to only track the expectations that are of relevance.[14] The formulae λ, ρ and ϕ are expressed using a form of linear temporal logic with past operators, but any future states available in the model are ignored when evaluating the condition λ. Once a rule has fired, the formula that is expected to hold (the last argument of ***exp***) is initially ρ. However, this may refer to the future, and thus as long as it is not fufilled or violated in any state, it is partially evaluated and 'progressed' to the following state (e.g. "In the next state, ϕ should hold" progresses to "ϕ should hold"). This means that the expected formula is always expressed from the viewpoint of the present—not the time at which the rule was initially triggered.

Clause 6 states that an unconditional rule of expectation is triggered at the time at which a commitment is created (note that t appears twice in the right hand side). Clauses 7 and 8 state that a commitment is fulfilled (respectively violated) if it currently exists (having been established at some time t) and the corresponding unconditional rule of expectation would have resulted in a current fulfilment (respectively violation) if it had fired at time t. The predicate *viol_comm* has an additional final argument (compared to *fulf_comm*) that encodes the residual formula ψ that was violated in the current state. This is likely to differ from the original commitment after partial evaluation and progression across a number of states occurring between t_1 and the present.

We assume the use of an extended version of the event calculus that incorporates the semantics of ***exp***, including the progression of expectations that are not fulfilled or violated, and can perform on-line and/or off-line determination of fulfilment and violation, e.g. by using the technique of Cranefield and Winikoff [51]. This means we need not explicitly use *holds_at* or *initiates* to define how expectations are progressed. However, we use *holds_at* to define the *first* time at which a rule of expectation becomes relevant to the system, to indicate that it should be tracked starting at that time.

4.7.3.2 Expectations and norms

The approach sketched out above can also be used to express protocol-based norms of institutional power, permission and obligation in terms of expectations. For example, the event calculus based approach of Artikis and Sergot [13] and the related work by Cliffe et al. [46] could be adapted to make use of the ***exp***, ***fulf*** and ***viol*** operators.

In this section we show how the logic of expectations can also be used to define the fulfilment and violation of conditional rule-based norms. We assume that norms are encoded by propositions of the form $norm(\lambda, \rho, sanction)$, where λ is the condition under which the norm holds, ρ is a linear temporal logic formula encoding the norm as a constraint on the present and future states of the world, and *sanction* encodes a sanction to be applied if the norm is violated. The sanction is an example of the additional contextual information that might be associated with a norm in contrast to the strictly temporal focus of an expectation. This example uses an additional operator from the logic of expectations: ***truncs***. A formula ***truncs***(ϕ) ("truncate model and evaluate with strong finite model semantics") is true when ϕ can be determined to hold without the use of any future information that might be available in the trace under consideration. This is necessary when checking for fulfilment or violation

[14] In the context of the event calculus, this could be done implicitly by a hypothetical combination of the event calculus with the logic of expectations. Alternatively, explicit fluents could be used in the clauses above to record the currently relevant rules of expectation.

of expectations that might involve future-oriented temporal operators (which may occur nested inside past-oriented operators).[15]

$$holds\_at(\boldsymbol{exp}(\lambda, \rho, t, \rho), t) \leftarrow norm(\lambda, \rho, sanction) \wedge holds\_at(\boldsymbol{truncs}(\lambda), t) \tag{9}$$

$$holds\_at(fulf\_norm(\lambda, \rho, t_1, sanction), t_2) \leftarrow norm(\lambda, \rho, sanction) \wedge holds\_at(\boldsymbol{fulf}(\lambda, \rho, t_1, \_), t_2) \tag{10}$$

$$holds\_at(viol\_norm(\lambda, \rho, t_1, \phi, sanction), t_2) \leftarrow norm(\lambda, \rho, sanction) \wedge holds\_at(\boldsymbol{viol}(\lambda, \rho, t_1, \phi), t_2) \tag{11}$$

These clauses state that when a conditional norm is triggered, a corresponding expectation is triggered. The fulfilment or violation of an expectation that corresponds to a triggered norm results in the fulfilment or violation of the norm. We assume that norms are static, so do not use *holds_at* to check for their existence. Note that the time the norm was triggered serves to distinguish different fulfilments or violations of the same norm—it appears as the second argument in *fulf_norm* and *viol_norm*. As in the commitments case, we add an additional argument (ϕ) to the predicate *viol_norm* to record the residual violated formula derived from the right hand side of the norm after partial evaluation and progression since the norm was triggered.

4.8 Agreement technologies

Billhardt *et al.* [25] envision that methods and mechanisms from the fields of semantic alignment, norms, organization, argumentation and negotiation, as well as trust and reputation are part of a "sandbox" to build software systems based on technologies of agreement. Based on a well known definition of coordination as management of dependencies between organisational activities [95], they distinguish the detection of dependencies from taking a decision on which coordination action to apply. Their call-by-agreement interaction method first establishes an agreement for action, and the actual enactment of the action is requested thereafter. The normative context determines rules of the game, *i.e.* interaction patterns and additional restrictions. The so-called agreement technologies "tower" or stack of semantic alignment, norms, organization, argumentation, negotiation, trust and reputation is visualized in Figure 3.

Semantic technologies form the basis to deal with semantic mismatches and alignment of ontologies to give a common understanding of norms or agreements, defining the set of possible agreements. Norms and organizations determine constraints that the agreements, and the processes to reach them, have to satisfy. Organisational structures define the capabilities of the roles and the power and authority relationships among them. Argumentation and negotiation methods are used to make agents reach agreements. The agents use trust mechanisms that summarise the history of agreements and subsequent agreement executions in order to build long-term relationships between the agents. Billhardt *et al.* emphasize that these methods should not be seen in isolation, as they may well benefit from each other.

[15] Formulae of this type might be unlikely to appear as norm conditions, but for completeness we allow for this possibility rather than imposing syntactic restrictions on λ.

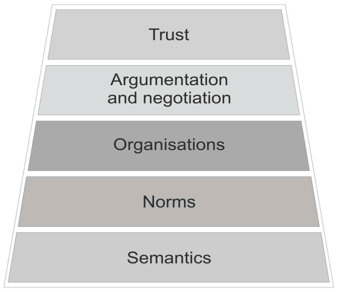

Figure 3 Agreement Technologies Tower [25].

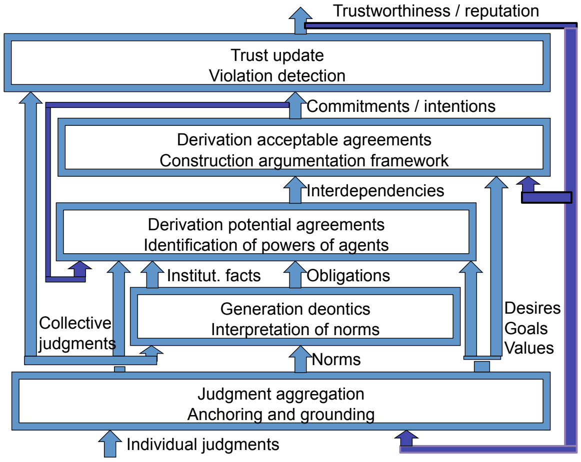

Figure 4 Architecture of Agreement Process.

4.8.1 Agreement process

Instead of combining the technologies in a sandbox, Boella and van der Torre [34] introduce a combined agreement process, whose architecture is visualized in Figure 4.

The individual judgments and preferences are grounded in observations and opinions, and aggregated into collective judgments, norms, desires, values and goals. The collective judgments and the norms in force are interpreted [28], and used to generate institutional facts, obligations and permissions. The collective judgments, institutional facts and obligations are used to identify the actions the agents can perform, and their power to satisfy the desires and goals of themselves as well as of other agents. This creates a network of dependencies among the agents. The dependencies among the agents can be used to construct an argumentation framework. Based on the desires and goals of the agents, they negotiate and commit to acceptable agreements. The resulting intentions are fed back into the argumentation and negotiation component, when new agreements are negotiated. The behavior of agents and their commitments is monitored, and in case of detection of violations of agreements the trustworthiness and reputation of the involved agents is updated. The trustworthiness of agents is fed back into the judgment aggregation operator, as well as in the argumentation and negotiation component.

4.8.2 Normative reasoning

The agreement technologies sandbox suggests a bottom-up approach, in the sense that each reasoning technique is studied in its own community, with its own conferences and its own journals. There is a semantic web conference and journal, a deontic logic in computer science and normative multi-agent systems conference, an argumentation conference and journal, and so on. The challenge of reasoning for agreement technologies is to define the relations among them, such that a coherent framework arises. In that sense, the architecture of the agreement process introduced in this paper is more top down. We now discuss how these reasoning techniques can be combined.

4.8.3 Input/output perspective

The input/output perspective on the architecture of the agreement process considers each individual reasoning method as a black box, defined by its input/output behavior, and studies their interaction. Makinson and van der Torre [91] introduce input/output logic for norms to generate institutional facts, obligations and permissions. Bochman [27] uses it to define argumentation in a causal framework. The normative theory can be used to formalize Castelfranchi's theory of dependence networks and social commitments, which has to be extended with a theory or roles. Singh [128] specializes the general way to treat conditionalization in input/output logic for the setting of trust with inferences for completion, commitments, and teamwork that do not arise with conditionals in general, but are important for an understanding of trust.

Missing is an input/output perspective on semantic alignment. Fragments of classical logic such as description logics are used to reason about ontologies, but have less to say about aggregation and alignment. We propose to adopt a judgment aggregation perspective for this step [87].

5 Concluding remarks

For further information, consult the upcoming deontic logic handbook, and the proceedings of the DEON conference series.

References

1 S. Abiteboul, R. Hull, and V. Vianu, editors. *Foundations of Databases: The Logical Level*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1995. ISBN 0201537710.

2 A. Abou El Kalam, S. Benferhat, A. Miège, R. El Baida, F. Cuppens, C. Saurel, P. Balbiani, Y. Deswarte, and G. Trouessin. Organization based access control. *Policies for Distributed Systems and Networks, IEEE International Workshop on*, 0:120, 2003. ISBN 0-7695-1933-4.

3 M. Alberti, F. Chesani, M. Gavanelli, E. Lamma, P. Mello, and P. Torroni. Verifiable agent interaction in abductive logic programming: the sciff framework. *ACM Transactions on Computational Logic*, 9(4), 2008.

4 M. Alberti, M. Gavanelli, E. Lamma, F. Chesani, P. Mello, and P. Torroni. Compliance verification of agent interaction: a logic-based software tool. *Applied Artificial Intelligence*, 20(2):133–157, 2006.

5 M. Alberti, M. Gavanelli, E. Lamma, P. Mello, P. Torroni, and G. Sartor. Mapping deontic operators to abductive expectations. *Computational & Mathematical Organization Theory*, 12:205–225, 2006.

6 C. Alchourrón and E. Bulygin. The expressive conception of norms. in [76] 95–124.

7 C. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50:510–530, 1985.

8 C. Alchourron and D. Makinson. Hierarchies of regulations and their logic. In R. Hilpinen, editor, *New Studies in Deontic Logic*, pages 125–148. Reidel, Dordrecht., 1981.

9 C. Alchourrón and D. Makinson. On the logic of theory change: Contraction functions and their associated revision functions. *Theoria*, 48:14–37, 1982.

10 L. Åqvist. *Introduction to deontic logic and the theory of normative systems.* Bibliopolis Napoli,, Italy, 1987. This paper gives an alternative proof of weak completeness of DSDL3 given by Spohn in 1975.

11 L. Åqvist. Alchourrón and bulygin on deontic logic and the logic of norm-propositions: axiomatization and representability results. *Logique et Analyse*, 51(203):225–261, 2008.

12 L. Åqvist and J. Hoepelman. Some theorems about a tree system of deontic tense logic. In R. Hilpinen, editor, *New Studies in Deontic Logic*, pages 187–221. Reidel, 1981.

13 A. Artikis and M. Sergot. Executable specification of open multi-agent systems. *Logic Journal of the IGPL*, 18(1):31–65, 2010.

14 A. Artosi, P. Cattabriga, and G. Governatori. Ked: A deontic theorem prover. In *Workshop on Legal Application of Logic Programming*, pages 60–76, 1994. This is a prover for "deontic" SDL but in fact for several modal logics.

15 G. Aucher, G. Boella, and L. van der Torre. Prescriptive and descriptive obligations in dynamic epistemic deontic logic. In G. Governatori and G. Sartor, editors, *Proceedings of the 10th International Conference on Deontic Logic in Computer Science (DEON 2010)*, volume 6181 of *LNAI*, pages 150–161, 2010.

16 G. Aucher, G. Boella, and L. van der Torre. A dynamic logic for privacy compliance. *Artif. Intell. Law*, 19(2-3):187–231, 2011.

17 G. Aucher, D. Grossi, A. Herzig, and E. Lorini. Dynamic context logic. In X. He, J. Horty, and E. Pacuit, editors, *Proceedings of LORI 2009*, volume 5834 of *LNAI*. Springer, 2009.

18 R. Aumann. Correlated equilibrium as an expression of bayesian rationality. *Econometrica*, 55:1–18, 1987.

19 P. Balbiani. Logical approaches to deontic reasoning: From basic questions to dynamic solutions. *International Journal of Intelligent Systems*, 23(10):1021–1045, 2008. This article contains a deontic logic made up with PDL.

20 P. Balbiani, A. Herzig, and N. Troquard. Alternative axiomatics and complexity of deliberative stit theories. *Journal of Philosophical Logic*, 37(4):387–406, 2008. This paper gives an axiomatization for both individual Chellas' STIT and individual deliberative STIT.

21 M. Baldoni, C. Baroglio, and E. Marengo. Behavior-oriented commitment-based protocols. In H. Coelho, R. Studer, and M. Wooldridge, editors, *ECAI*, volume 215 of *Frontiers in Artificial Intelligence and Applications*, pages 137–142. IOS Press, 2010. ISBN 978-1-60750-605-8.

22 P. Bartha. Conditional obligation, deontic paradoxes, and the logic of agency. *Annals of Mathematics and Artificial Intelligence*, 9(1-2):1–23, 1993.

23 N. Belnap, M. Perloff, and M. Xu. *Facing the future: agents and choices in our indeterminist world.* Oxford, 2001.

24 C. Bicchieri. *The Grammar of Society: The Nature and Dynamics of Social Norms.* Cambridge University Press, 2006.

25 H. Billhardt, R. Centeno, C. E. Cuesta, A. Fernández, R. Hermoso, R. Ortiz, S. Ossowski, J. S. Pérez-Sotelo, and M. Vasirani. Organisational structures in next-generation distributed systems: Towards a technology of agreement. *Multiagent and Grid Systems*, 7(2-3): 109–125, 2011.

26 K. Binmore. The origins of fair play. *Proceedings of the British Academy*, 151:151–193, 2007.

27 A. Bochman. *Explanatory Nonmonotonic Reasoning*. World Scientific Publishing Company, New York, London, 2005.

28 G. Boella, G. Governatori, A. Rotolo, and L. van der Torre. A logical understanding of legal interpretation. In F. Lin, U. Sattler, and M. Truszczynski, editors, *KR*. AAAI Press, 2010.

29 G. Boella, G. Pigozzi, M. Singh, and H. Verhagen, editors. Special issue on normative multiagent systems. *Logic Journal of the IGPL*, 18(1), 2010.

30 G. Boella and L. van der Torre. A logical architecture of a normative system. In [67] 24–35.

31 G. Boella and L. van der Torre. Permissions and obligations in hierarchical normative systems. In *Proceedings of the 9th International Conference on Artificial Intelligence and Law, ICAIL 2003, June 24-28, Edinburgh, Scotland, UK*. ACM, 2003. Revised version to appear in *Artificial Intelligence and Law*.

32 G. Boella and L. van der Torre. Constitutive norms in the design of normative multiagent systems. In *Computational Logic in Multi-Agent Systems, 6th International Workshop, CLIMA VI*, LNCS 3900, pages 303–319. Springer, 2006.

33 G. Boella and L. van der Torre. Institutions with a hierarchy of authorities in distributed dynamic environments. *Artificial Intelligence and Law*, 16(1):53–71, 2008.

34 G. Boella and L. van der Torre. Reasoning for agreement technologies. Submitted.

35 G. Boella, L. van der Torre, and H. Verhagen, editors. Special issue on normative multiagent systems. *Computational & Mathematical Organization Theory*, 12(2–3), 2006.

36 C. Boutilier. Conditional logics of normality: A modal approach. *Artif. Intell.*, 68(1): 87–154, 1994.

37 J. Broersen. Strategic deontic temporal logic as a reduction to ATL, with an application to Chisholm's scenario. In L. Goble and J.-J. Meyer, editors, *Proceedings 8th International Workshop on Deontic Logic in Computer Science (DEON'06)*, volume 4048 of *Lecture Notes in Computer Science*, pages 53–68. Springer, 2006.

38 J. Broersen. Deontic epistemic *stit* logic distinguishing modes oof *Mens Rea*. *Journal of Applied Logic*, 9(2):127–152, 2011.

39 J. Broersen. Probabilistic action and deontic logic. In *Proceedings of the 12th International Workshop on Computational Logic in Multi-Agent Systems*, volume 6814 of *Lecture Notes in Artificial Intelligence*, pages 293–294. Springer, 2011.

40 J. Broersen, F. Dignum, V. Dignum, and J.-J. Meyer. Designing a deontic logic of deadlines. In A. Lomuscio and D. Nute, editors, *Proceedings 7th International Workshop on Deontic Logic in Computer Science (DEON'04)*, volume 3065 of *Lecture Notes in Computer Science*, pages 43–56. Springer, 2004.

41 N. Bulling and M. Dastani. Verifying normative behaviour via normative mechanism design. In T. Walsh, editor, *IJCAI*, pages 103–108. IJCAI/AAAI, 2011. ISBN 978-1-57735-516-8.

42 C. Castelfranchi. For a systematic theory of expectations. In *Proceedings of the European Cognitive Science Conference*, 2007.

43 C. Castelfranchi, F. Giardini, E. Lorini, and L. Tummolini. The prescriptive destiny of predictive attitudes: From expectations to norms via conventions. In *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*, pages 222–227, 2003. URL `http://csjarchive.cogsci.rpi.edu/proceedings/2003/pdfs/61.pdf`.

44 F. Chesani, P. Mello, M. Montali, and P. Torroni. Commitment tracking via the reactive event calculus. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 91–96. Morgan Kaufmann, 2009.

45 R. Chisholm. Contrary-to-duty imperatives and deontic logic. *Analysis*, 24:33–36, 1963.

46 O. Cliffe, M. De Vos, and J. Padget. Modelling normative frameworks using answer set programing. In *Proceedings of the 10th International Conference on Logic Programming* *and Nonmonotonic Reasoning*, volume 5753 of *Lecture Notes in Computer Science*, pages 548–553. Springer, 2009. URL `http://dx.doi.org/10.1007/978-3-642-04238-6_56`.

47 R. Coase. The problem of social cost. *Journal of Law and Economics*, 1, 1960.

48 J. Coleman. *Foundations of Social Theory.* Belknap Harvard, 1990.

49 R. Conte and C. Castelfranchi. *Cognitive and Social Action.* UCL Press, 1995.

50 S. Cranefield. A rule language for modelling and monitoring social expectations in multi-agent systems. In *Coordination, Organizations, Institutions, and Norms in Multi-Agent Systems*, volume 3913 of *Lecture Notes in Artificial Intelligence*, pages 246–258. Springer, 2006.

51 S. Cranefield and M. Winikoff. Verifying social expectations by model checking truncated paths. *Journal of Logic and Computation*, 21(6):1217–1256, 2011.

52 S. Cranefield, M. Winikoff, and W. Vasconcelos. Modelling and monitoring interdependent expectations. In S. Cranefield, M. B. van Riemsdijk, J. Vázquez-Salceda, and P. Noriega, editors, *Coordination, Organizations, Institutions, and Norms in Agent System VII*, volume 7254 of *Lecture Notes in Artificial Intelligence*, pages 149–166. Springer, 2012.

53 F. Cuppens, N. Cuppens-Boulahia, and T. Ramard. Availability enforcement by obligations and aspects identification. In *ARES '06: Proceedings of the First International Conference on Availability, Reliability and Security*, pages 229–239. IEEE Computer Society, Washington, DC, USA, 2006. ISBN 0-7695-2567-9.

54 L. del Cerro, D. Fauthoux, O. Gasquet, A. Herzig, D. Longin, and F. Massacci. Lotrec: the generic tableau prover for modal and description logics. *Automated Reasoning*, pages 453–458, 2001. This is the first paper about the generic tableau prover Lotrec developped in Toulouse. SDL is implemented.

55 F. Dignum and R. van Eijk, editors. Special issue on agent communication. *Autonomous Agents and Multi-Agent Systems*, 14(2), 2007.

56 P. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

57 K. Erol, J. Hendler, and D. S. Nau. Htn planning: Complexity and expressivity. In *In Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, pages 1123–1128. AAAI Press, 1994.

58 M. Fitting. *Proof methods for modal and intuitionistic logics.* D. Reidel, Dordrecht, 1983. This book provides proof methods and define "strictly analytic".

59 M. Fowler. *UML Distilled: A Brief Guide to the Standard Object Modeling Language.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3 edition, 2003. ISBN 0321193687.

60 G. P. G. Boella and L. van der Torre. A normative framework for norm change. In *Proc. AAMAS 2009.* ACM, 2009.

61 D. Gabbay, J. Horty, R. van der Meyden, X. Parent, and L. van der Torre (eds). *Handbook of Deontic Logic and Normative Systems.* To appear with College Publications, London.

62 D. Gabbay, G. Pigozzi, and J. Woods. Controlled revision — an algorithmic approach for belief revision. *Journal of Logic and Computation*, 13:3–22, 2003.

63 V. Genovese, D. Garg, and D. Rispoli. Labeled sequent calculi for access control logics: Countermodels, saturation and abduction. In *In Proceedins of the 25th IEEE Computer Security Foundations Symposium*, 2012.

64 H. Gintis. *The Bounds of Reason.* Princenton University Press, 2010.

65 L. Goble. Prima facie norms, normative conflicts and dilemmas. Forthcoming in [61].

66 L. Goble. Preference semantics for deontic logics. part i – Simple models. *Logique et Analyse*, 46:383–418, 2008.

67 L. Goble and J.-J. Meyer, editors. *Deontic Logic and Artificial Normative Systems. 8th International Workshop on Deontic Logic in Computer Scicence, DEON 2006, Utrecht, July 2006, Proceedings.* Springer, Berlin, 2006.

68 G. Governatori and A. Rotolo. Changing legal systems: Legal abrogations and annulments in defeasible logic. *The Logic Journal of IGPL*, 18(1):157–194, 2010.

69 D. Grossi, L. Tummolini, and P. Turrini. Norms in game theory. In *Handbook of Agreement Technologies*. 2012.

70 J. Hansen. Deontic logics for prioritized imperatives. *Artificial Intelligence and Law, forthcoming*, 2005.

71 J. Hansen. Prioritized conditional imperatives: problems and a new proposal. *Journal of Autonomous Agents and Multi-Agent Systems*, 17(1):11–35, 2008.

72 B. Hansson. An analysis of some deontic logics. *Nous*, 3(4):373–398, 1969. This paper presents variant of dyadic deontic logic differing by the constraints in the model, especially DSDL1, DSDL2, DSDL3.

73 H. Hart. *The Concept of Law.* Clarendon, Oxford, 1994.

74 A. Herzig and F. Schwarzentruber. Properties of logics of individual and group agency. *Advances in modal logic*, 7:133–149, 2008. This paper recalls the result for individual Chellas' STIT (axiomatization and complexity results) and states that group Chellas' STIT is undecidable and not finitely axiomatizable.

75 A. Heuerding, M. Seyfried, and H. Zimmermann. Efficient loop-check for backward proof search in some non-classical propositional logics. In *TABLEAUX '96: Proceedings of the 5th International Workshop on Theorem Proving with Analytic Tableaux and Related Methods*, pages 210–225. Springer-Verlag, London, UK, 1996. This article presents the technique of loop-check.

76 R. Hilpinen, editor. *New Studies in Deontic Logic.* Reidel, Dordrecht, 1981.

77 J. Horty. *Agency and Deontic Logic.* 2001.

78 J. Horty. Defaults with priorities. *Journal of Philosophical Logic*, 36:367–413, 2007.

79 L. Hurwicz. Institutions as families of game forms. *Japanese Economic Review*, 47(2): 113–132, 1996.

80 L. Hurwicz. But who will guard the guardians? *American Economic Review*, 98(3):577–585, 2008.

81 S. Jajodia, P. Samarati, and V. S. Subrahmanian. A logical language for expressing authorizations. *Security and Privacy, IEEE Symposium on*, 0:0031, 1997. ISSN 1540-7993.

82 A. Jones and M. Sergot. A formal characterisation of institutionalised power. *Journal of IGPL*, 3:427–443, 1996.

83 H. Kelsen. *General Theory of Norms.* Clarendon, Oxford, 1991.

84 R. Kowalski and M. Sergot. A logic-based calculus of events. *New Generation Computing*, 4:67–69, 1986.

85 D. Lewis. *Convention: A Philosophical Study.* Cambridge University Press, 1969.

86 D. Lewis. Semantic analyses for dyadic deontic logic. In S. Stenlund, editor, *Logical Theory and Semantic Analysis*, pages 1 – 14. Reidel, Dordrecht, 1974.

87 C. List and C. Puppe. Judgment aggregation: A survey. In P. Anand, C. Puppe, and P. Pattanaik, editors, *Oxford handbook of rational and social choice.* Oxford University Press, New York, 2009.

88 E. Lorini. A stit logic analysis of commitment and its dynamics (to appear). *Journal of Applied Logic*, 2012. This article contains the axiomatization of STIT + the future operator.

89 E. Lorini and F. Schwarzentruber. A logic for reasoning about counterfactual emotions. *Artificial Intelligence*, 175(3):814–847, 2011. This paper presents a STIT logic for coutnerfactual emotions. It presents a decidable fragment of group Chellas' STIT and a Hilbert axiomatization.

90 D. Makinson. On a fundamental problem of deontic logic. In [99], 29–53.

91 D. Makinson and L. van der Torre. Input-output logics. *Journal of Philosophical Logic*, 29 (4):383–408, 2000.

92 D. Makinson and L. van der Torre. Constraints for input-output logics. *Journal of Philosophical Logic*, 30(2):155–185, 2001.

93 D. Makinson and L. van der Torre. Permissions from an input-output perspective. *Journal of Philosophical Logic*, 32(4):391–416, 2003.

94 E. Mally. Grundgesetze des sollens. *Mind*, 36(141):124–b, 1927. This paper is the first approach to deontic logic. It is based on classical propositional logic.

95 T. Malone and K. Crowston. The interdisciplinary study of coordination. *Computing Surveys*, 26 (1):87–119, 1994.

96 E. Marengo, M. Baldoni, C. Baroglio, A. Chopra, V. Patti, and M. Singh. Commitments with regulations: reasoning about safety and control in regula. In Sonenberg et al. [130], pages 467–474.

97 J. McCarthy. Programs with common sense. In *Proceedings of the Teddington Conference on the Mechanization of Thought Processe*, pages 756–91. Her Majesty's Stationery Office, London: Her Majesty's Stationery Office, 1959.

98 P. McNamara. Deontic logic. In D. Gabbay and J. Woods, editors, *Handbook of the History of Logic*, volume 7, pages 197–289. North-Holland Publishing, Amsterdam, 2006.

99 P. McNamara and H. Prakken, editors. *Norms, Logics and Information Systems.* IOS, Amsterdam, 1999.

100 J.-J. Meyer. A different approach to deontic logic: Deontic logic viewed as a variant of dynamic logic. *Notre Dame Journal of Formal Logic*, 29:109–136, 1988.

101 M. Montali. *Specification and Verification of Declarative Open Interaction Models - A Logic-Based Approach*, volume 56 of *Lecture Notes in Business Information Processing.* Springer, 2010. ISBN 978-3-642-14537-7. 1-383 pp.

102 D. Moody. The physics of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on Software Engineering*, 35(6):756–779, 2009.

103 D. Moody and J. van Hillegersberg. Evaluating the visual syntax of uml :an analysis of the cognitive effectiveness of the uml family of diagrams. In *Software Language Engineering*, volume 5452 of *Lecture Notes in Computer Science*, pages 16–34. . Springer, 2009.

104 Q. Ni, E. Bertino, and J. Lobo. An obligation model bridging access control policies and privacy policies. In *SACMAT '08: Proceedings of the 13th ACM symposium on Access control models and technologies*, pages 133–142. ACM, New York, NY, USA, 2008. ISBN 978-1-60558-129-3.

105 M. Nickles, M. Rovatsos, and G. Weiss. Expectation-oriented modeling. *Engineering Applications of Artificial Intelligence*, 18:891–918, 2005. URL `http://dx.doi.org/10.1016/j.engappai.2005.05.002`.

106 D. North. *Institutions, Institutional Change and Economic Performance.* Cambridge University Press, Cambridge, 1990.

107 J. Odell, H. Van Dyke Parunak, and B. Bauer. Representing agent interaction protocols in UML. In *IN OMG DOCUMENT AD/99-12-01. INTELLICORP INC*, pages 121–140. Springer-Verlag, 2001.

108 M. Osborne and A. Rubinstein. *A Course in Game Theory.* MIT Press, 1994.

109 X. Parent. On the strong completeness of Åqvist's dyadic deontic logic G. *Deontic Logic in Computer Science*, pages 189–202, 2008. This article contains the strong completeness of an axiomatization of dyadic deontic logic DSDL3.

110 X. Parent. A complete axiom set for Hansson's deontic logic DSDL2. *Logic Journal of IGPL*, 18(3):422–429, 2010. Axiomatization for DSDL2.

111 X. Parent. Moral particularism in the light of deontic logic. *Artif. Intell. Law*, 19(2-3): 75–98, 2011.
112 J. Park and R. Sandhu. The $UCON_{ABC}$ Usage Control Model. *ACM Transactions on Information and System Security (TISSEC)*, 7(1):128 – 174, February 2004.
113 M. Pesic, H. Schonenberg, and W. M. P. van der Aalst. The declare service. In A. H. M. ter Hofstede, W. M. P. van der Aalst, M. Adams, and N. Russell, editors, *Modern Business Process Automation*, pages 327–343. Springer, 2010. ISBN 978-3-642-03120-5.
114 M. Piunti, C. Castelfranchi, and R. Falcone. Expectations driven approach for situated, goal-directed agents. In *Proceedings of the 8th AI*IA/TABOO Joint Workshop "From Objects to Agents": Agents and Industry: Technological Applications of Software Agents*, pages 104–111. Seneca Edizioni Torino, 2007.
115 S. Ranathunga, S. Cranefield, and M. Purvis. Integrating expectation handling into BDI agents. In *Programming Multi-Agent Systems*, volume 7217 of *Lecture Notes in Artificial Intelligence*, pages 74–91. Springer, 2012.
116 D. Rönnedal. Dyadic deontic logic and semantic tableaux. *Logic and Logical Philosophy*, 18(3-4):221–252, 2009. TODO.
117 A. Rotolo. Retroactive legal changes and revision theory in defeasible logic. In G. Governatori and G. Sartor, editors, *Proceedings of the 10th International Conference on Deontic Logic in Computer Science (DEON 2010)*, volume 6181 of *LNAI*, pages 116–131. Springer, 2010. ISBN 978-3-642-14182-9.
118 J. Rumbaugh. Notation notes: Principles for choosing notation. *Journal of Object-Oriented Programming*, 8(10):11–14, 1996.
119 H. Sahlqvist. Completeness and correspondence in the first and second order semantics for modal logic. *Studies in Logic and the Foundations of Mathematics*, 82:110–143, 1975. This paper explains a correspondance between constraints on Kripke models and axioms.
120 R. Sandhu, E. Coyne, H. Feinstein, and C. Youman. Role-based access control models. *IEEE Computer*, 29(2):38–47, 1996.
121 T. Schelling. *The Strategy of Conflict.* Oxford University Press, 1966.
122 F. Schwarzentruber. Complexity results of stit fragments (to appear). *Studia logica*, 2011. This paper exhibits synctatic fragments of Chellas' group STIT by restricting the coalitions allowed in the language. Those fragments are finitely axiomatizable.
123 F. Schwarzentruber. Lotrecscheme. *Electronic Notes in Theoretical Computer Science*, 278: 187–199, 2011. This is a paper about some improvements of the tableau prover Lotrec and presents a prototype for it. SDL is implemented.
124 M. Shanahan. The event calculus explained. In M. Wooldridge and M. Veloso, editors, *Artificial Intelligence Today: Recent Trends and Developments*, volume 1600 of *Lecture Notes in Artificial Intelligence*, pages 409–430. Springer, 1999.
125 Y. Shoham and K. Leyton-Brown. *Multiagent Systems: Algorithmic, Game-Theoretic and Logical Foundations.* Cambridge University Press, 2008.
126 Y. Shoham and M. Tennenholtz. Social laws for artificial agent societies: Off-line design. *Artificial Intelligence*, 73(12):231–252, 1995.
127 M. Singh. Semantical considerations on dialectical and practical commitments. In *Proceedings of the 23rd National Conference on Artificial Intelligence (AAAI)*, pages 176–181. AAAI Press, 2008.
128 M. Singh. Trust as dependence: a logical approach. In Sonenberg et al. [130], pages 863–870.
129 B. Skyrms. *Evolution of the Social Contract.* Cambridge University Press, 1996.
130 L. Sonenberg, P. Stone, K. Tumer, and P. Yolum, editors. *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3.* IFAAMAS, 2011.

131 W. Spohn. An analysis of Hansson's dyadic deontic logic. *Journal of Philosophical Logic*, 4(2):237–252, 1975. This paper contains a weak completeness result for DSDL3.

132 A. Stolpe. Norm-system revision: theory and application. *Artif. Intell. Law*, 18(3):247–283, 2010.

133 A. Stolpe. Relevance, derogation and permission. In G. Governatori and G. Sartor, editors, *DEON*, volume 6181 of *Lecture Notes in Computer Science*, pages 98–115. Springer, 2010.

134 A. Stolpe. A theory of permission based on the notion of derogation. *J. Applied Logic*, 8 (1):97–113, 2010.

135 R. Thomason. Deontic logic as founded on tense logic. In R. Hilpinen, editor, *New Studies in Deontic Logic*, pages 165–176. Reidel, 1981.

136 S. Tosatto, G. Boella, L. van der Torre, and S. Villata. Visualizing normative systems. In *Procs. of DEON 2012*, LNCS. Springer, 2012.

137 D. Ullman. *Principles of database and knowledge-base systems, Vol. I.* Computer Science Press, Inc., New York, NY, USA, 1988. ISBN 0-88175-188-X.

138 E. Ulmann-Margalit. *The Emergence of Norms.* Oxford: Clarendon Press, 1977.

139 J. van Benthem. *Logical Dynamics of Information and Interaction.* Cambridge University Press, 2011.

140 J. van Benthem, D. Grossi, and F. Liu. Deontics = betterness + priority. In G. Governatori and G. Sartor, editors, *Proceedings of the 10th International Conference on Deontic Logic in Computer Science (DEON 2010)*, volume 6181 of *LNAI*, pages 50–65. Springer, 2010.

141 W. van der Hoek, M. Roberts, and M. Wooldridge. Social laws in alternating time: Effectiveness, feasibility, and synthesis. *Synthese*, 156:1:1 – 19, 2007.

142 L. van der Torre, G. Boella, and H. Verhagen, editors. Special issue on normative multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 17(1), 2008.

143 L. van der Torre and Y.-H. Tan. Contrary-to-duty reasoning with preference-based dyadic obligations. *Ann. Math. Artif. Intell.*, 27(1-4):49–78, 1999.

144 M. Verdicchio and M. Colombetti. A commitment-based communicative act library. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 755–761. ACM, 2005.

145 G. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.

146 I. Wallace and M. Rovatsos. Bounded practical social reasoning in the ESB framework. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, pages 1097–1104. IFAAMAS, 2009. URL `http://dl.acm.org/citation.cfm?id=1558109.1558166`.

147 H. Wansing. Tableaux for multi-agent deliberative-stit logic. *Advances in modal logic*, 6: 503–520, 2006. This paper presents a non-terminating tableau for individual deliberative-STIT logic.

148 M. Wooldridge. *An Introduction to MultiAgent Systems (2. ed.).* Wiley, 2009. 1-461 pp.

149 R. Wright. Causation, responsibility, risk, probability, naked statistics, and proof: Pruning the bramble bush by clarifying the concepts. *Iowa Law Review*, 73:1001–1077, 1988.

150 M. Xu. Axioms for deliberative stit. *Journal of Philosophical Logic*, 27(5):505–552, 1998. This paper gives an axiomatization for individual deliberative STIT.

151 P. Yolum and M. Singh. Reasoning about commitments in the event calculus: An approach for specifying and executing protocols. *Annals of Mathematics and Artificial Intelligence*, 42:227–253, 2004. URL `http://dx.doi.org/10.1023/B:AMAI.0000034528.55456.d9`.

152 X. Zhang, F. Parisi-Presicce, R. Sandhu, and J. Park. Formal model and policy specification of usage control. *ACM Trans. Inf. Syst. Secur.*, 8(4):351–387, 2005. ISSN 1094-9224.