Automated reasoning in normative detachment structures with ideal conditions

Tomer Libal
The American University of Paris
Paris, France
tlibal@aup.edu

Matteo Pascucci
TU Wien
Vienna, Austria
matteo.pascucci@tuwien.ac.at

ABSTRACT

In this article we introduce a logical structure for normative reasoning, called Normative Detachment Structure with Ideal Conditions, that can be used to represent the content of certain legal texts in a normalized way. The structure exploits the deductive properties of a system of bimodal logic able to distinguish between ideal and actual normative statements, as well as a novel formalization of conditional normative statements able to capture interesting cases of contrary-to-duty reasoning and to avoid deontic paradoxes. Furthermore, we illustrate how the theoretical framework proposed can be mechanized to get an automated procedure of query-answering on an example of legal text.

CCS CONCEPTS

• **Theory of computation** → Automated reasoning; Modal and temporal logics; • **Applied computing** → Law;

KEYWORDS

Deontic logic, Legal reasoning, Normative ideality

ACM Reference format:
Tomer Libal and Matteo Pascucci. 2019. Automated reasoning in normative detachment structures with ideal conditions. In *Proceedings of Seventeenth International Conference on Artificial Intelligence and Law, Montreal, QC, Canada, June 17–21, 2019 (ICAIL '19),* 10 pages.
https://doi.org/10.1145/3322640.3326707

1 INTRODUCTION

A very useful procedure toward the automated analysis of the content of a legal text is converting the *surface structure* of the text, which consists of natural language sentences often disconnected and ambiguous, to a *rigorous structure* which provides a transparent interpretation of normative concepts and clarifies the relation between the various statements. This procedure is sometimes referred to as the *normalization* of a legal text (see, e.g., [1, 2]). Once normalization has been performed, one can choose a suitable formal system to reason on scenarios regulated by the text and encode it in a theorem prover. The choice of the formal system should represent a good balance between complexity and expressiveness. For instance, it is well-known that in order to represent contrary-to-duty scenarios the system **SDL** (*Standard Deontic Logic*) is fundamentally unsatisfactory, since its language is too poor [24]. On the other hand, there are propositional extensions of **SDL** that behave significantly better.

One of such extensions, called **DL**, is proposed by Jones and Pörn in [19] and aims at a rigorous representation of the difference between the notions of *normative ideality* and *normative sub-ideality*, a difference that is not expressible in **SDL** and that turns out to be very useful to avoid certain paradoxes of contrary-to-duty reasoning; here we will consider a variation of **DL** suggested by de Boer et al. in [6], which will be called $\mathbf{DL}^*$.[1] We will show that $\mathbf{DL}^*$ can be suitably exploited to reason on normative scenarios (and, more specifically, legal texts) from which one can extract a *logical structure* involving a list of ideal normative statements, a list of normative conditionals, a list of factual relations among the various statements and some actual circumstances. Such structure will be called a *Normative Detachment Structure with Ideal Conditions.* In particular, we will see how, extracting this structure from a legal text, one can use $\mathbf{DL}^*$ to formulate normatively relevant queries about the text; moreover, we will show that these queries can be answered by a theorem prover.

The article is organized as follows. Section 2 is devoted to a thorough presentation of logics of normative ideality and sub-ideality, including detailed motivations for the choice of $\mathbf{DL}^*$. In the same section we also present the logical structure of the normative scenarios we want to deal with. In section 3 we provide an example of a legal text and formalize its core sentences within the language of the logical system employed. In section 4 we represent some normatively relevant queries as problems of derivability of formulas in $\mathbf{DL}^*$. Furthermore, we introduce a program which can answer similar queries in a fully-automated way. Finally (section 5), we conclude our work with theoretical reflections on the representation of contrary-to-duty scenarios proposed in the article, some legal applications of our framework and a comparison with related literature.

2 LOGICS OF NORMATIVE IDEALITY AND SUB-IDEALITY

The system of Standard Deontic Logic (**SDL**) is the weakest normal deontic system closed under the schema $OA \rightarrow \neg O \neg A$, namely the deontic version of the alethic system **KD**. It is a common practice to point out a list of theorems of **SDL** that are associated with

[1] It has to be remarked that **DL** was not the favourite deontic logic of its proponents, at least according to what is stated in [9], where a more sophisticated approach including different levels of normative ideality is advocated.

ICAIL '19, June 17–21, 2019, Montreal, QC, Canada

ACM ISBN 978-1-4503-6754-7/19/06…$15.00
https://doi.org/10.1145/3322640.3326707

paradoxes of deontic reasoning. For instance, the provable schema $OA \rightarrow O(A \vee B)$ gives rise to *Ross's paradox* when the formulas A and B represent, respectively, the propositions expressed by sentences like 'Mark posts the letter' and 'Mark burns the letter': the inference from 'Mark ought to post the letter' to 'Mark ought to post the letter or burn it' is at least difficult to justify. While these paradoxes are sometimes due to ambiguities in the natural language sentences to be formalized, there is a more important flaw of **SDL** which concerns the formalization of *contrary-to-duty obligations*. This problem is exemplified by *Chisholm's paradox*.[2] Consider the following set of sentences:

(1) it ought to be that Jane helps her neighbors;
(2) it ought to be that if Jane helps her neighbors, she tells them that she is coming;
(3) if Jane does not help her neighbors, then she ought not to tell them that she is coming;
(4) Jane does not help her neighbors.

Under any plausible formalization in the language of **SDL**, these sentences turn out to be either inconsistent or not logically independent and both outcomes are clearly undesirable.

On the other hand, the reasons of the aforementioned drawbacks are easily overlooked. As Jones and Pörn claim in [19], many issues arise from the interpretation of the operator O. Indeed, the semantic intuition associated with a formula of kind OA in **SDL** is that A is true in all *normatively ideal circumstances* (or worlds), namely in all those circumstances in which *every* prescription is observed. However, many sentences which describe propositions true in all normatively ideal circumstances are not normatively relevant, such as the sentence 'it either rains or does not rain'. Thus, in order to formally capture the meaning of 'ought'-sentences, one has to take into account *some criterion of normative relevance for sentences*. The proposal made in [19] consists in requiring an 'ought'-sentence to describe a proposition which not only holds in all normatively ideal circumstances, but also fails in some *normatively sub-ideal circumstance*, namely in some circumstance in which *not every* prescription is observed. For instance, if Jane ought to help her neighbors, then one can say that this happens to be the case in all normatively ideal scenarios, but fails to be the case in some normatively sub-ideal scenario.

In order to distinguish between normative ideality and normative sub-ideality, Jones and Pörn extend the language of **SDL** with an operator O' such that the formula $O'A$ means that A is true in all normatively sub-ideal worlds. Then, they propose the following formalization of 'ought'-sentences: $Ought(A) =_{def} OA \wedge \neg O'A$. The system obtained with the addition of O' is called **DL** and represents a bimodal version of **SDL** that needs to be supplemented at least with the axiom-schema $(OA \wedge O'A) \rightarrow A$. Indeed, it can be easily shown that the latter schema is required to capture the set of valid formulas in the intended semantics (see also [6] on this point; Jones and Pörn do not provide an axiomatization for their logic). Frames to interpret **DL** are structures of kind $\mathfrak{F} = \langle W, R_O, R_{O'} \rangle$, where W is a domain of worlds and R_O and $R_{O'}$ are binary relations over W satisfying the following properties:

(I) for all $w \in W$, there are $v, u \in W$ s.t. $wR_O v$ and $wR_{O'} u$;
(II) for all $w \in W$, $R_O(w) \cap R_{O'}(w) = \emptyset$;
(III) for all $w \in W$, either $w \in R_O(w)$ or $w \in R_{O'}(w)$.

Property (I) can be captured already by the axiomatic basis of bimodal **SDL**, property (II) requires further discussion that will be provided below and property (III) can be captured only if one extends the axiomatic basis of bimodal **SDL** with a schema such as $(OA \wedge O'A) \rightarrow A$. Thus, if one wants **DL** to be the logic characterized by the class of frames satisfying properties (I), (II) and (III), as it seems to be suggested by Jones and Pörn, then **DL** has to be a proper extension of bimodal **SDL**.

Despite its broader expressive power, **DL** still encounters some obstacles in dealing with contrary-to-duty obligations. Indeed, as observed by Prakken and Sergot in [30], it gives rise to a 'pragmatic oddity' when the sentences (1)-(4) above are formalized in the way proposed by Jones and Pörn, namely (taking P to be 'Jane helps her neighbors' and Q to be 'Jane tells her neighbors that she is coming'):

(1a) $Ought(P)$;
(2a) $O(P \rightarrow Ought(Q)) \wedge O'(P \rightarrow Ought(Q))$;
(3a) $O(\neg P \rightarrow Ought(\neg Q)) \wedge O'(\neg P \rightarrow Ought(\neg Q))$;
(4a) $\neg P$.

The oddity is hidden in the fact that in **DL** (1a)-(4a) entail both OP and $O\neg Q$, which means that in all normatively ideal worlds Jane helps her neighbors without telling them that she is coming. Furthermore, Hansson shows in [16] that certain instances of paradoxes of deontic reasoning still hold in **DL**, such as the following version of *Ross's paradox*: if Mark neither posts the letter nor burns it, while he ought to post it, then he ought to post it or burn it. Indeed, the schema $(\neg A \wedge \neg B) \rightarrow (Ought(A) \rightarrow Ought(A \vee B))$ is provable in **DL**.

The latter problem finds a remedy in [6], where de Boer et al. propose to replace the operator $Ought$ with an operator $Ought^*$ such that $Ought^*(A) =_{def} OA \wedge O'\neg A$; in this way the problematic schemata mentioned by Hansson are no longer provable. However, there is a fundamental aspect of $Ought^*$ which requires further analysis. Since $Ought^*(A)$ is true at a world only if $O'\neg A$ is true there, the meaning of O' proposed by Jones and Pörn needs to be revised. Indeed, otherwise one would have that *every* prescription is violated in all sub-ideal worlds, which is implausible, since a world can be classified as normatively sub-ideal even if *some but not all* prescriptions are violated there. Thus, in order to exploit the operator $Ought^*$ it is better to take $O'A$ as meaning that A holds in all *normatively awful* worlds. Notice that this reading allows one to get rid of the schema $(OA \wedge O'A) \rightarrow A$, as well as of the frame condition (III) associated with it, since the current world might turn out to be neither normatively ideal nor normatively awful (from the perspective of the norms currently in effect). This change is not dramatic for the notion of sub-ideality: a world is still classified as sub-ideal if and only if it is not an ideal one.[3]

Hereafter we will denote by **DL*** the logic resulting from **DL** by removing the axiom $(OA \wedge O'A) \rightarrow A$ and adding the definition of $Ought^*$. A crucial issue is whether **DL*** coincides with a bimodal

[2] For an extended discussion of the paradox we refer the reader to [9].

[3] The bimodal definition of the operator $Ought^*$ resembles the definition for the operator 'all I know' used in epistemic logic [21]. However, while according to the semantics of 'all I know' worlds have to be divided into two sets (those that are compatible with what I know and those that are not), here worlds are divided into three sets (the normatively ideal ones, the normatively awful ones and those that are neither normatively ideal nor normatively awful).

version of **SDL**. This is actually the case, since Lemma 3.5 in [6] shows that every model over a frame which violates property (II) can be transformed into an equivalent model over a frame satisfying property (II); thus, since property (I) can be captured already by the axioms of bimodal **SDL**, there is no need to add further postulates to get a characterization result for $\mathbf{DL}^*$ with respect to its intended class of frames, namely the class of frames satisfying properties (I) and (II).

It is important to remark that the 'pragmatic oddity' may affect the formalization of Chisholm's example even if one uses $Ought^*$ in place of *Ought*. For instance, in [6] de Boer et al. argue that, in the absence of property (III), the formalization of sentences (2) and (3) needs the following revision in order to allow for the detachment of $Ought^*\neg Q$, which is an intended consequence of the scenario (Jane ought not to tell her neighbors that she is coming, since she decided not to help them):

(2a') $(P \to Ought^*(Q)) \wedge O(P \to Ought^*(Q)) \wedge O'(P \to Ought^*(Q))$;
(3a') $(\neg P \to Ought^*(\neg Q)) \wedge O(\neg P \to Ought^*(\neg Q)) \wedge O'(\neg P \to Ought^*(\neg Q))$.

However, from (1a), (2a'), (3a') and (4a) one still gets both OP and $O\neg Q$ as consequences and so an implausible description of what is the case in all normatively ideal worlds (Jane helps her neighbors without telling them).

A solution to the pragmatic oddity can be formulated by combining the use of $Ought^*$ (which is beneficial anyway, since it allows one to get rid of Hansson's version of deontic paradoxes) and an intuition put forward by Jones and Pörn in [19], according to which the first sentence of a Chisholm-like scenario expresses an obligation which holds in ideal circumstances, hereafter simply called an *ideal obligation*. For instance, one can imagine that in the actual circumstance something prevents Jane from helping her neighbors and thus that she does not have an *actual obligation* to do so, while in ideal circumstances she would have an obligation to do so.

Notice that, as it is claimed by Carmo and Jones in [9], the distinction between ideal prescriptions and actual prescriptions does not coincide with the distinction between *prima facie* prescriptions and *all-things-considered* prescriptions, which is often invoked in the literature on defeasible reasoning [3]. Indeed, among the set of *prima facie* prescriptions one can have both ideal prescriptions and non-ideal ones. The notion of ideal prescription makes reference to some *normative standard* that is sometimes assumed in a scenario without being explicitly stated; it has to be made explicit via conceptual analysis of the scenario. In Chisholm's example, Jane has the *prima facie* obligation of helping her neighbors and the *prima facie* obligation of telling them that she is not coming (as soon as she decides not to help them); however, the normative standard applies only to the first prescription, since one would say that in ideal circumstances she ought to help her neighbors but not that in ideal scenarios she ought not to tell her neighbors that she is coming.[4]

The result of taking ideal obligations into account in Chisholm's scenario is the following formalization of sentence (1):

(1a') $O(Ought^*(P))$.

The formula (1a') has to be read 'in ideal circumstances, it ought to be the case that P'. Thus, ideal obligations are obtained via a nesting of the operators O and $Ought^*$. Moreover, we propose here three modifications of (2a') and (3a'). First, since the reading of O' has to be changed from 'in all normatively sub-ideal worlds' to 'in all normatively awful worlds' and normatively awful worlds cannot be expected to verify conditional obligations, then the conjuncts $O'(P \to Ought^*(Q))$ and $O'(\neg P \to Ought^*(\neg Q))$ can be dropped from (2a') and (3a'). Second, we remove also the conjuncts $O(P \to Ought^*(Q))$ and $O(\neg P \to Ought^*(\neg Q))$, since in more complex scenarios conjuncts of this kind would allow one to infer an ideal obligation, $O(Ought^*(B))$, from two premises $Ought^*(A)$ and $O(A \to Ought^*(B))$; indeed, $Ought^*(A)$ entails OA and this, together with $O(A \to Ought^*(B))$, entails $O(Ought^*(B))$. The point is that it is clearly not acceptable to infer an ideal obligation from an actual one. Third, we want the formal representation of a conditional obligation to allow for the construction of a chain of statements that provide a full description of what ideally (i.e., in ideal circumstances) ought to be the case. For instance, we know that Jane *ideally* ought to help her neighbors and that the fact that she helps her neighbors entails that she ought to tell them that she is coming; from this one would like to infer that Jane *ideally* ought to tell her neighbors that she is coming. In order to get this result without affecting a *uniform rendering* of conditional obligations, we add to (2a') the conjunct $O(Ought^*(P) \to Ought^*(Q))$ and to (3a') the conjunct $O(Ought^*(\neg P) \to Ought^*(\neg Q))$. While in the former case one gets $O(Ought^*(Q))$ from (1a') and the schema $O(A \to B) \to (OA \to OB)$, in the latter case no ideal obligation can be detached, since the antecedent itself is not an ideal obligation. The result of all modifications is the following rendering of sentences (2) and (3):

(2a") $(P \to Ought^*(Q)) \wedge O(Ought^*(P) \to Ought^*(Q))$;
(3a") $(\neg P \to Ought^*(\neg Q)) \wedge O(Ought^*(\neg P) \to Ought^*(\neg Q))$.

The formula OP is not derivable from (1a'), (2a"), (3a") and (4a), so the pragmatic oddity no longer arises;[5] furthermore, the four premises are still logically independent and consistent. The key aspect of this solution is that obligations with respect to ideal circumstances (ideal obligations), such as $O(Ought^*(P))$, are kept distinct from obligations with respect to the actual circumstances (actual obligations), such as $Ought^*(\neg Q)$.

We can generalize this approach to scenarios that are more complex than Chrisholm's paradox and that include:

(i) a list of ideal normative statements;
(ii) a list of normative conditionals;
(iii) some factual relations among the statements in (i) and (ii);
(iv) some actual circumstances which trigger the antecedents of some conditionals in (ii).

[4] From a semantic point of view, it has to be remarked that, according to this suggestion by Jones and Pörn, a world w is normatively ideal with respect to a world w' if and only if:

- all prescriptions that are *actual* in w' are observed in w;
- all prescriptions that are ideal in w' apply to w.

However, it would be more appropriate to say that there are *two levels of ideality* from the perspective of a world w: the first level is that of any world w' s.t. $wR_O w'$, which represents *ideality with respect to what is actually prescribed* in w; the second level is that of any world w'' s.t. $w'R_O w''$ and $wR_O w'$, which represents *ideality in a strict sense* (everything which is ideally prescribed in w is observed in w'').

[5] This can be justified in terms of the two levels of ideality described in the previous footnote.

We can call this logical structure a *Normative Detachment Structure with Ideal Conditions* (hereafter *NDSIC*). Let us consider the case in which all normative statements involved in an *NDSIC* are obligations, then this structure can be more precisely described as follows (for some $n, k \geq 0$ and some $m \geq 1$):

(Cid_1) A_1 *ideally* ought to be the case;
...
(Cid_n) A_n *ideally* ought to be the case;
($Ccon_1$) if A'_1 then B_1 ought to be the case;
...
($Ccon_m$) if A'_m then B_m ought to be the case;
($Crel_1$) some relation Rel_1 among the statements involved in (Cid_1)-($Ccon_m$) holds;
...
($Crel_k$) some relation Rel_k among the statements involved in (Cid_1)-($Ccon_m$) holds;
($Cant$) the antecedents of some conditionals in ($Ccon_1$)-($Ccon_m$) hold.

In a structure of this kind one can detach both actual obligations (triggered by ($Cant$)) and ideal obligations (triggered by (Cid_1)-(Cid_n)) from the list of conditionals.

Notice that an *NDSIC* is not a contrary-to-duty structure on its own; a contrary-to-duty feature emerges when some clause of kind (Cid), which describes an ideal normative statement, conflicts with the clause ($Cant$), which describes what actually is the case and triggers some normative statement involved in the list of conditionals; in this situation, some of the normative statements triggered by ($Cant$) are contrary-to-duty ones. On the other hand, if there is no conflict between the ideal and the actual, then an *NDSIC* simply represents a logical structure in which normative statements can be detached from conditionals given the actual circumstances and the clauses (Cid_1)-(Cid_n).

One of the key aspects of an *NDSIC* is the representation of a conditional normative statement via a combination of two modal operators, O and O'; thus, in our approach we rely on a *multimodal translation* of a conditional statement. Similar translations are quite common in the literature on the logic of conditionals; for instance, in [7] and [8] Boutilier proposes an approach to defeasible reasoning in which conditionals are defined in terms of a language with two primitive (monadic) operators of necessity. Boutilier relies on a logical framework originally due to Humberstone [17]. The same framework has been exploited also in the context of dyadic deontic logic; see, for instance, the preference-based deontic logic discussed by van der Torre in [35].

We conclude this section with another important comment on the logic $\mathbf{DL}^*$: the definition of a plausible operator of permission. One cannot simply take the dual of $Ought^*$, since $\neg Ought^*(\neg A)$ means $\neg(O\neg A \wedge O'A)$, namely $\mathcal{P}A \vee \mathcal{P}'\neg A$ (taking $\mathcal{P}$ to be a shorthand for $\neg O \neg$), which is too weak to express permission. Also in this case, one can borrow a solution from [19] and have $Perm^*(A) =_{def} \mathcal{P}A \wedge \mathcal{P}'\neg A$. According to such definition and the revised reading of O', A is permitted iff there is a normatively ideal world in which it holds and a normatively awful world in which it does not hold. The first conjunct witnesses that A is compatible with everything which should be the case; the second conjunct that A is not trivially true in all possible scenarios.

3 REASONING IN LEGAL TEXTS

In order to see how the logic $\mathbf{DL}^*$ can be put at work, we borrow an example of a legal text from [20], *The United Nations Convention on Contracts for the International Sale of Goods*. We will use $\mathbf{DL}^*$ to represent some *normatively relevant queries* related to this text; more specifically, we will consider a situation in which an international transaction has just been concluded and either the buyer or the seller wants to navigate through the directives of the Convention to understand which are the normative consequences of the current circumstances and/or whether some violation of the Convention occurred.

3.1 The United Nations Convention on Contracts for the International Sale of Goods

What follows is a list of articles from the Convention, which describe some rights and duties of the seller and of the buyer in case of a transaction. We will then show how an *NDSIC* can be extracted from these articles.

Article 30

The seller must deliver the goods, hand over any documents relating to them and transfer the property in the goods, as required by the contract and this Convention.

Article 31

If the seller is not bound to deliver the goods at any other particular place, his obligation to deliver consists:

(a) if the contract of sale involves carriage of the goods – in handing the goods over to the first carrier for transmission to the buyer;
(b) if, in cases not within the preceding subparagraph, the contract relates to specific goods, or unidentified goods to be drawn from a specific stock or to be manufactured or produced, and at the time of the conclusion of the contract the parties knew that the goods were at, or were to be manufactured or produced at, a particular place – in placing the goods at the buyer's disposal at that place;
(c) in other cases – in placing the goods at the buyer's disposal at the place where the seller had his place of business at the time of the conclusion of the contract.

Article 32

(1) If the seller, in accordance with the contract or this Convention, hands the goods over to a carrier and if the goods are not clearly identified to the contract by markings on the goods, by shipping documents or otherwise, the seller must give the buyer notice of the consignment specifying the goods.
(2) If the seller is bound to arrange for carriage of the goods, he must make such contracts as are necessary for carriage to the place fixed by means of transportation appropriate in the circumstances and according to the usual terms for such transportation.
(3) If the seller is not bound to effect insurance in respect of the carriage of the goods, he must, at the buyer's request, provide

him with all available information necessary to enable him to effect such insurance.

Article 45

(1) If the seller fails to perform any of his obligations under the contract or this Convention, the buyer may:
  (a) exercise the rights provided in articles 46 to 52;
  (b) claim damages as provided in articles 74 to 77.
(2) The buyer is not deprived of any right he may have to claim damages by exercising his right to other remedies.
(3) No period of grace may be granted to the seller by a court or arbitral tribunal when the buyer resorts to a remedy for breach of contract.

Article 53

The buyer must pay the price for the goods and take delivery of them as required by the contract and this Convention.

Let us first provide a brief and informal reconstruction of the relations among the various Articles in this part of the Convention, according to the conceptual tools available in $\mathbf{DL}^*$.

Article 30 in the Convention specifies some default duties of the seller after a transaction has been made: he/she is required to deliver the goods, hand over the documents and transfer the property in the goods. These duties represent some ideal outcomes of the transaction, so we can consider them as ideal obligations. Article 31 specifies some conditional obligations: if the seller is committed to carriage of the goods, then he/she must hand the goods over to the first carrier; in other cases, he/she needs to place the goods at the buyer's disposal at a specified place: either the place of production or his/her own place of business at the time of the conclusion of the contract. Article 32 includes further conditional obligations triggered by the situation in which the seller hands the goods over to the first carrier available (i.e., by one of the scenarios illustrated in Article 31). Article 45 introduces a conditional permission depending on a contrary-to-duty circumstance: if the seller does not fulfill his/her duties described in the previous Articles, then the buyer can exercise some right and claim damage. Article 53 describes some further duties, this time concerning the buyer (cf. Article 30): the buyer is required to pay and take delivery of the goods. Both duties represent ideal outcomes of the transaction; however, here we choose to represent the second obligation as a conditional one, depending on a proper delivery of the goods, and the first obligation as an unconditional one. Our choice is motivated by the fact that sometimes (e.g., in the case of on-line transactions) the buyer is required to pay in advance, so that his/her duty to pay cannot depend on an appropriate delivery.

3.2 A formal representation of the Convention

We introduce some formal notation to simplify the normalization of the portion of the Convention considered. For the sake of clarity, we want to stress that our focus here is on the methodology, rather than on the degree of faithfulness of the normalized text, which can be revised at later stages. For instance, one might want to consider only transactions in which payment depends on appropriate delivery of the goods.

Definition 3.1 (Language of the Convention). We start by codifying some statements from the Convention as propositional symbols (constants):

- D - the seller delivers the goods, hands over the documents and transfers the property according to the procedure described in the contract;
- $D0.1$ - the contract requires the seller to take care of the carriage of goods;
- $D0.2$ - the contract relates to goods to be produced at a particular place;
- $D1$ - the goods are handed over to the first carrier;
- $D1.1$ the goods are clearly identified by markings, shipping documents, etc.;
- $D1.2$ the seller notifies the buyer of the consignment;
- $D1.3$ the seller makes contracts necessary for carriage;
- $D1.4$ the seller is bound to effect insurance in respect of the carriage;
- $D1.5$ the seller provides the buyer information to effect the insurance for carriage;
- $D2$ - the goods are disposed at the place of production;
- $D3$ - the goods are disposed at the business address of the seller;
- $E1$ - the buyer exercises rights;
- $E2$ - the buyer claims damages;
- G - the buyer takes delivery of the goods;
- P - the buyer pays for the goods.

Then, we introduce deontic operators and normative conditionals that can be expressed in $\mathbf{DL}^*$; we employ here a different and simplified notation which points out their reading in a more explicit way. Let A and B be propositional formulas, then:

- $Id(A)$ - A holds in all normatively ideal circumstances (this corresponds to the formula OA in $\mathbf{DL}^*$);
- $Aw(A)$ - A holds in all normatively awful circumstances (this corresponds to the formula $O'A$ in $\mathbf{DL}^*$);
- $Ob(A)$ - A ought to be the case (this corresponds to the formula $Ought^*(A)$ in $\mathbf{DL}^*$);
- $Pm(A)$ - A can be the case (this corresponds to the formula $Perm^*$ in $\mathbf{DL}^*$)
- $A \Rightarrow_{Ob} B$ - B is an obligation triggered by condition A (this corresponds to the formula $(A \rightarrow Ought^*(B)) \wedge O(Ought^*(A) \rightarrow Ought^*(B))$ in $\mathbf{DL}^*$);
- $A \Rightarrow_{Pm} B$ - B is a permission triggered by condition A (this corresponds to the formula $(A \rightarrow Perm^*(B)) \wedge O(Perm^*(A) \rightarrow Perm^*(B))$ in $\mathbf{DL}^*$).

It is interesting to remark that the identity principles $A \Rightarrow_{Ob} A$ and $A \Rightarrow_{Pm} A$ are not derivable in $\mathbf{DL}^*$, whereas analogous principles usually hold for multimodal translations of conditional statements discussed in the literature (see, e.g., [7] and [8]). The reason is that $\mathbf{DL}^*$ is not closed under the schemata $A \rightarrow Ob(A)$ and $A \rightarrow Pm(A)$, which have implausible consequences for normative reasoning. Furthermore, notice that one cannot infer B under the assumptions A and $A \Rightarrow_{Ob} B$, so $\Rightarrow_{Ob}$ (and, similarly, $\Rightarrow_{Pm}$) does not obey Modus Ponens. Finally, $\Rightarrow_{Ob}$ does not tolerate contradicting succedents of a true antecedent, since $\mathbf{DL}^*$ is closed under the schema $A \rightarrow \neg((A \Rightarrow_{Ob} B) \wedge (A \Rightarrow_{Ob} \neg B))$.

Definition 3.2 (The Convention). The following is the formal description of a relevant set of statements from the Convention.

(1) $Id(Ob(D))$;
(2) $Id(Ob(P))$;
(3) $D0.1 \Rightarrow_{Ob} D1$;
(4) $D0.2 \Rightarrow_{Ob} D2$;
(5) $(\neg D0.1 \wedge \neg D0.2) \Rightarrow_{Ob} D3$;
(6) $D \Rightarrow_{Ob} G$;
(7) $D1 \Rightarrow_{Ob} D1.3$;
(8) $(D1 \wedge \neg D1.1) \Rightarrow_{Ob} D1.2$;
(9) $(D1 \wedge \neg D1.2) \Rightarrow_{Ob} D1.1$;
(10) $(D1 \wedge \neg D1.4) \Rightarrow_{Ob} D1.5$;
(11) $(D1 \wedge \neg D1.5) \Rightarrow_{Ob} D1.4$;
(12) $V1 \Rightarrow_{Pm} E1$;
(13) $V1 \Rightarrow_{Pm} E2$;
(14) $D0.1 \rightarrow \neg D0.2$;
(15) $D1 \rightarrow \neg D2$;
(16) $D1 \rightarrow \neg D3$;
(17) $D2 \rightarrow \neg D3$;
(18) $D \rightarrow [(D0.1 \rightarrow (D1 \wedge (D1.1 \equiv \neg D1.2) \wedge D1.3 \wedge (D1.4 \equiv \neg D1.5))) \wedge (D0.2 \rightarrow D2) \wedge ((\neg D0.1 \wedge \neg D0.2) \rightarrow D3)]$.

We illustrate the reading of (8) as an example: 'if the goods are handed over to the first carrier and they are not clearly identified by markings, shipping documents, etc, then it is obligatory that the seller notifies the buyer of consignment'.

Let UN be the conjunction of formulas (1)-(18) above. By extending UN with a set of actual circumstances playing the role of (*Cant*), one gets an *NDSIC*, where (1)-(2) stand for clauses of kind (*Cid*), (3)-(13) for clauses of kind (*Ccon*) and (14)-(18) for clauses of kind (*Crel*). In order to prepare the ground for a mechanization of the query-answering procedure, it is convenient to exploit also the following definitions, where $V1$ means that the seller violated the Convention and $V2$ that the buyer violated the Convention: $V1 =_{def} \neg D$ and $V2 =_{def} (\neg P \vee \neg G)$. Indeed, in a query a subject usually simply asks whether a violation of the Convention (by some of the parties involved) occurred, rather than whether some specific norm was not fulfilled. Notice that any obligation not observed by the seller in a given circumstance leads to $\neg D$, due to (18).

We will next consider some normatively relevant queries concerning this scenario, that is queries whose answer can help the buyer and the seller in understanding the normative consequences of the Convention.

4 NORMATIVE QUERIES ABOUT THE CONVENTION

In this section we provide some examples of queries concerning the Convention that can be formally represented within the language of $\mathbf{DL}^*$, in accordance with Definition 3.1 and Definition 3.2. These queries might be formulated by a buyer or a seller who has just concluded a transaction and wants to explore its normative consequences given some actual circumstances.

The first group of problems (Queries 1-2) concerns the compatibility of a given scenario with the Convention. Here a subject asks whether, given some actual circumstances, a violation of the Convention occurred.

Query 1. Is there any violation of the Convention by the seller if the contract requires the seller to take care of the carriage of the goods and they are placed at the seller's address of business?

This problem can be expressed as a question about the derivability of the following conditional in $\mathbf{DL}^*$: $(UN \wedge D0.1 \wedge D3) \rightarrow V1$. If this conditional is derivable in $\mathbf{DL}^*$, then the situation described in Query 1 actually represents a violation of the Convention by the seller.

Query 2. Is there any violation of the Convention by the seller if the contract neither requires the seller to take care of the carriage of the goods nor refers to goods that have to be produced at a particular place and the seller does not dispose the goods at his/her place of business?

This problem can be expressed as a question about the derivability of the following conditional in $\mathbf{DL}^*$: $(UN \wedge \neg D0.1 \wedge \neg D0.2 \wedge \neg D3) \rightarrow V1$. If this conditional is derivable in $\mathbf{DL}^*$, then the situation described in Query 2 actually represents a violation of the Convention by the seller.

The second group of problems (Queries 3-6) concerns the detachment of normative statements from given scenarios. The last three problems in this group (Queries 4-6) make reference to contrary-to-duty scenarios.

Query 3. In case the contract requires the seller to take care of the carriage of the goods, does the seller have to notify the buyer of consignment if he/she hands them over to the first carrier but cannot identify them with appropriate markings, shipping documents, etc.?

This problem can be formulated as a question concerning the derivability of the following conditional: $(UN \wedge D0.1 \wedge D1 \wedge \neg D1.1) \rightarrow Ob(D1.2)$. The answer to Query 3 is positive if and only if the conditional is provable in $\mathbf{DL}^*$.

Query 4. Is the buyer allowed to claim for damage in case the contract requires the seller to take care of the carriage and the seller neither effects a carriage insurance nor provides the buyer information to effect such an insurance?

This problem can be expressed as a question concerning the derivability of the following conditional in $\mathbf{DL}^*$: $(UN \wedge D0.1 \wedge \neg D1.4 \wedge \neg D1.5) \rightarrow Pm(E2)$. The answer is positive if and only if the conditional is provable.

Query 5. Is the buyer allowed to exercise rights in case the contract makes reference to goods to be produced at a particular place and the seller disposes them at his/her address of business?

This problem can be expressed as a question concerning the derivability of the following conditional in $\mathbf{DL}^*$: $(UN \wedge D0.2 \wedge D3) \rightarrow Pm(E1)$. The answer is positive if and only if the conditional is provable.

Query 6. Is the buyer allowed to exercise rights in case the goods are not delivered according to the procedure described in the contract?

This problem can be expressed as a question concerning the derivability of the following conditional in **DL***: $(UN \wedge \neg D) \rightarrow Pm(E1)$. The answer is positive if and only if the conditional is provable.

4.1 Using MleanCoP

In order to automate the answering of such questions, we need an efficient implementation of a proof-calculus for **DL***. Several proof-calculi and implementations for multimodal versions of **SDL** exist and, according to a result in [6] mentioned in section 2, the logic **DL*** turns out to be a bimodal version of **SDL** exploiting a non-primitive modal operator, *Ought**, to express obligations. This fact allows us to use standard theorem-provers for normal multimodal logic in order to check the derivability of a formula in **DL***. We also want to mention that a proof-calculus for the related system **DL** is developed in [12]; however, such calculus has no implementation and, as we argued in section 2, there are many theoretical reasons to prefer **DL***over **DL** in order to represent normative scenarios involving contrary-to-duty reasoning.

Among the various systems implementing proof-calculi for normal multimodal logic two are prominent. The first, MleanCoP [28], is a native theorem-prover for various systems of normal modal logic, among which multimodal versions of **DL***.[6] The second, Leo3 [33], is a theorem-prover for higher-order logic that can be exploited by translating formulas of a modal language into formulas of a higher-order non-modal language. Each method has its benefits and limitations. Here we choose to employ MleanCoP, which provides an efficient method and builds proofs directly within the modal language, so no backward translation from higher-order logic is needed. In order to ask MleanCoP the queries in section 3, we have to translate them into a format which MleanCoP can understand. Currently, MleanCoP supports two different formats to codify logical languages, the general format TPTP[7] and its own specific one. We will use MleanCoP's own format, since it is more concise.

Definition 4.1 (MleanCoP's syntax). An MleanCoP problem is a predicate of the form $f(G)$. where G stands for an arbitrary formula. Formulas are constructed from atoms, whose name must start with a lowercase letter, and the following operators:

- The standard propositional operators `'~'` (negation), `';'` (disjunction), `','` (conjunction), `'=>'` (implication) and `'<=>'` (equivalence).
- The modal box operators `'# 1^d: G'` (`G` holds in all ideal worlds) and `'# 2^d: G'` (`G` holds in all awful worlds).
- The modal diamond operators `'* 1^d: G'` (`G` holds in some ideal worlds) and `'* 2^d: G'` (`G` holds in some awful worlds).

A comparison between our formalization of the Convention in section 3.2 and MleanCop's syntax points out that the encoding of queries in MleanCoP is very laborious. In order to make the theorem-prover more user-friendly, we developed a program capable of executing queries written in our encoding. This program translates the encoding into the syntax of MleanCoP and then executes the prover. The programs discussed in this paper as well as the latest version of MleanCoP can be downloaded from Zenodo.[8]

Definition 4.2 (The program syntax). The input to the program consists of a predicate of the form `([ls],F)` where `[ls]` is a list of assumptions separated by commas and F is a consequence (or goal) that we want to check. Both assumptions and goals are constructed using the following operators, which imitate those in Definition 3.1.

- The standard propositional operators as defined by MleanCoP;
- The ideality operator `Id(F)` and the awfulness operator `Aw(F)` applied to a formula `F`;
- The obligation operator `Ob(F)` and the permission operator `Pm(F)` applied to a formula `F`;
- The conditional obligation `NO(F,G)` and the conditional permission `NP(F,G)` applied to a pair of formulas `F` and `G`.

In addition, in order to simplify the process of asking questions, we added a constant `un` which codifies the conjunction of the formulas (1)-(18) in Definition 3.2, namely UN.

MleanCoP is written in Prolog and an installation of one of the supported distributions of Prolog is required to run it. Our program comes bundled with MleanCoP version 1.3 and requires Ruby version 2.5.1 and Ruby's gem 'bundler' version 1.16.2. We tested our program on Debian 9 using ECLiPSe version 5.10 #147 and SWI-prolog version 7.2.3. Let A be a formula in the language of **DL*** given as an input: the Ruby program translates A into the format compatible with MleanCoP and executes the latter. The possible answers are 'Theorem' and 'Non-theorem'. The answer 'Theorem' means that the formula given as an input is provable in **DL***; the answer 'Non-theorem' means that the formula given as an input is not provable in **DL***.

In addition, an answer 'Theorem' is accompanied by a proof of derivability in the modal connection calculus employed by MleanCoP [27]. As an example of the procedure described so far, we show how one can formulate Query 1 from section 4. In order to adhere to the syntax of MleanCoP, we use lower case letters to encode the propositional constants from Definition 3.2 and we simplify a bit the names of formulas (e.g., $D0.1$ becomes $d01$).

```
ruby prove1.rb \
  "([un,d01,d3],((~ d);((~ p);(~ g))))"
problem is a modal (multi/const) Theorem
Start of proof for problem
...
End of proof for problem
```

Once the above is executed, MleanCoP states that this is a Theorem and returns a proof. The commands to execute the program on the remaining queries can be found in Figure 2. The method just described is efficient. The total running time of all involved programs on each query takes about a second.[9]

We now show the executions of our Ruby program which correspond to the six queries of section 4. Each line in the table in Figure 1 displays one of the relevant executions. Queries are numbered

[6] http://www.leancop.de/mleancop/
[7] http://www.tptp.org/
[8] https://zenodo.org/record/2544841#.XEjCVy6YUWN
[9] The program was executed on a laptop with an Intel Core i7-5600U processor.

	Facts	Goal	Type
1	$D0.1$, $D3$	$V1$	Violation
2	$\neg D0.1$, $\neg D0.2$, $\neg D3$	$V1$	Violation
3	$D0.1$, $D1$, $\neg D1.1$	$D1.2$	Obligation
4	$D0.1$, $\neg D1.4$, $\neg D1.5$	$E1$	Permission
5	$D0.2$, $D3$	$E2$	Permission
6	$\neg D$	$E2$	Permission

Figure 1: The logical representation of the six queries

	Command	Result
1	`([un,d01,d3],v1)`	Theorem
2	`([un,(~ d01),(~ d02),(~ d3)],v1)`	Theorem
3	`([un, d01,d1,(~ d11)],(Ob d12))`	Theorem
4	`([un,d01,(~ d14),(~ d15)],(Pm e1))`	Theorem
5	`([un,d02,d3],(Pm e2))`	Theorem
6	`([un,(~ d)],(Pm e2))`	Theorem

Figure 2: The commands for the queries in Figure 1 and their result

(first column) and involve a set of actual circumstances (second column) as well as a goal (third column). The fourth column of the table specifies the type of the query (e.g., a violation checking, an obligation checking, etc.). The table in Figure 2 displays the answer of the theorem prover for each of the queries.

We developed also a second program which generates a graph displaying some possible *normative consequences* of a given a set of actual and possible circumstances. Figure 3 shows the execution of the second program given the set of actual circumstances $\{D0.1\}$, the set of possible circumstances $\{D1.1, D1.2, D1.3\}$ and the set of possible normative outcomes $\{Ob(D1.1), Ob(D1.2), Ob(D1.3)\}$. To obtain the image of the graph in dot format,[10] a user should execute the following command:

```
> ruby tree.rb "d01" "d11,d12,d13" \
  "(Ob d11),(Ob d12),(Ob d13)" > mytree.dot
```

5 FINAL DISCUSSION

5.1 A novel representation of contrary-to-duty scenarios

In this article we provided an automatic procedure for reasoning on logical structures extracted from legal texts. We relied on a logic of normative ideality and sub-ideality called **DL***, presented in [6], and focused on problems that can be represented in terms of a Normative Detachment Structure with Ideal Conditions (*NDSIC*). Our theoretical framework provides also some ground for reflection on the problem of the formal rendering of contrary-to-duty scenarios in the style of Chisholm's paradox. In section 2 we proposed to formalize sentences (1)-(4) as follows (here we employ again the notation used in the literature):

(1a') $O(Ought^*(P))$;
(2a") $(P \rightarrow Ought^*(Q)) \wedge O(Ought^*(P) \rightarrow Ought^*(Q))$;
(3a") $(\neg P \rightarrow Ought^*(\neg Q)) \wedge O(Ought^*(\neg P) \rightarrow Ought^*(\neg Q))$
(4a) $\neg P$.

Formula (1a') is suggested by Jones and Pörn in [19] and wants to stress that Jane ought to help her neighbors in normatively ideal circumstances (though, not necessarily in the actual circumstance). Formula (4a) is the obvious way of rendering the fact that Jane does

[10] https://www.graphviz.org/doc/info/lang.html

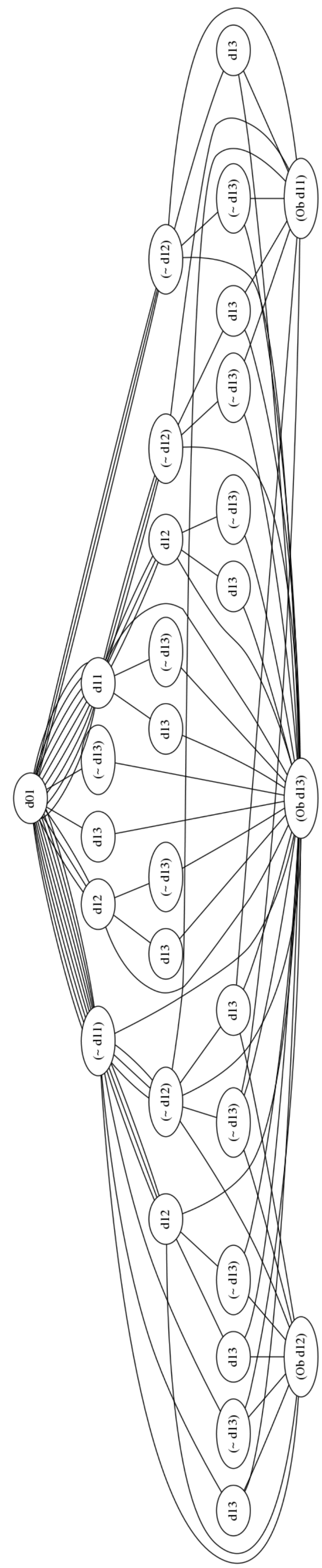

Figure 3: A graph representing possible normative outcomes of a given scenario

not help her neighbors. Formulas (2a") and (3a") are a novelty to represent conditional obligations. They were obtained by putting together some intuitions in [6] and the possibility of building a chain of ideal obligations; in particular, since Jane ideally ought to help her neighbors, then she ideally ought to tell them. While being more sophisticated than alternative logical renderings of Chisholm's scenario, this solution has several advantages which can be highlighted by making reference to some discussion in [9]. Indeed, in the latter work Carmo and Jones list *eight criteria* that should be met by a logical representation of contrary-to-duty scenarios:

(i) consistency;
(ii) logical independence of the members;
(iii) applicability to (at least apparently) timeless and actionless contrary-to-duty examples;
(iv) analogous logical structures for conditional sentences;
(v) capacity to derive actual obligations;
(vi) capacity to derive ideal obligations;
(vii) capacity to represent the fact that a violation of an obligation has occurred;
(viii) capacity to avoid the pragmatic oddity.

We have already shown that our formalization meets the requirement (viii) and the reader can easily check that (i) and (ii) are met as well. Criterion (iii) is also satisfied, since this formalization is not time-dependent (the language of $\mathbf{DL}^*$ cannot make temporal distinctions). The fact that conditional obligations always have the same logical rendering is evident, so we can mark with a tick criterion (iv) too. Criteria (v), (vi) and (vii) require further analysis. Actual obligations are those of kind $Ought^*(A)$ and in our representation of Chisholm's scenario one can surely infer $Ought^*(\neg Q)$: Jane actually ought not to tell her neighbors that she is coming (since she decided not to help them). Ideal obligations are those of kind $O(Ought^*(A))$ and in our representation of Chisholm's scenario one can infer $O(Ought^*(Q))$ from $O(Ought^*(P))$ and $O(Ought^*(P) \to Ought^*(Q))$, since the schema $O(A \to B) \to (OA \to OB)$ is provable in $\mathbf{DL}^*$: Jane ideally ought to tell her neighbors that she is coming because she ideally ought to help them. Thus, both (v) and (vi) are met. Finally, concerning criterion (vii), we can say that a violation of A occurs when A ideally ought to be the case, it is currently not the case but it could have been the case in normatively ideal circumstances. Now, in our scenario we have both $O(Ought^*(A))$ and $\neg A$ as premises; let us take these formulas as true in the actual world, call it w_a. From $O(Ought^*(A))$ one can infer $O\neg O\neg A$, which means that all worlds which are normatively ideal with respect to w_a have access to a (normatively ideal) world in which A is the case. We can paraphrase this as follows: in all ideal circumstances with respect to w_a it is possible to bring about A, despite A not being the case in w_a. This is the way in which our approach can represent the fact that a violation of an (ideal) obligation occurred. In conclusion, all eight criteria listed in [9] are met by the present approach.

5.2 Legal applications of our framework

A framework for normative reasoning as the one discussed in this paper can be used for various applications. In this section we discuss two of them. The first application is a tool for helping a subject to navigate a legal text like the Convention. Such a tool can not only make a legal text more accessible to non-experts but, in principle, also facilitate the task of a lawyer asked to express an opinion about a legal text he/she is not directly familiar with. Indeed, once a logical structure like an *NDSIC* has been extracted from a text, it is quicker to draw inferences directly from the logical structure than from the surface structure of the text. Thus, finding an appropriate logical structure to represent the content of a legal text requires some time at the beginning, due to the need of correcting possible mistakes in the formalization; however, once the logical structure has been sufficiently refined, it can be exploited to save a lot of time. A subject willing to benefit from the procedure for automated reasoning described in section 4.1 only needs to install the required software and to build queries; however, a web application and a user interface which takes sentences as inputs and translates them into the intended expressions to be checked could make the whole process even more human-friendly. A web application based on the results presented in this paper and on the ideas just mentioned is currently being developed.[11] This application allows legal practitioners to normalize the content of a text and then to perform automated reasoning on the logical structure extracted.

Our framework could also find an application as a courtroom decision-supporting system. These systems are widely used in many fields, such as management [22], medicine [34] and civil engineering [10], and help subjects in making appropriate decisions by excluding from the set of all possible decisions those which do not comply with a given set of actual circumstances. On the other hand, decision-supporting systems in the courtroom, which are specifically intended to help judges in deciding cases, are quite scarce. In fact, to the best of our knowledge, just a handful of such systems exist. Winjuris[12] and Forecourt[13] can be better classified as management systems rather than as decision-supporting systems. More relevant examples are an Israelian system for the evaluation of criminal records [31], the Australian 'Split Up' system [37], which assists in property splitting during divorce trials, and a case-based reasoning system for a 'virtual courtroom' [36]. As we previously mentioned, one of the programs described in section 4.1 can be used to display all possible normative outcomes for a given set of actual circumstances in the form of a graph (see Figure 3). Thus, a judge could exploit this program to check which decisions are supported by the information available in a given court case.

5.3 Related literature and future directions

Several works in the literature address the issue of an automatic analysis of the content of a legal text, such as [18] and [14] for business compliance checking and [29] for GDPR compliance checking. These works differ from ours in several aspects. First, only few of them go beyond the theoretical foundations and provide a software for users (see, e.g., the Regorous software in [15]). Second, most relevant works rely on the use of a logical language that is formulated specifically for the text to be analysed. For example, variants of the business contract language which was introduced in [13]; our approach is based, instead, on a general language of bimodal logic.

[11] https://nai.uni.lu
[12] http://softpert.com/legal/court-management/winjuris
[13] https://www.rsi.com/products/forecourt/

An obvious advantage of such a general language is its applicability to legal texts of various domains, i.e., also quite different from the Convention analysed here. Furthermore, since we exploited a normal system of bimodal logic, we were able to encode it in state-of-the-art and efficient theorem provers [32]. It should be also noted that all the operations involved in the automation procedure described in this paper are known to be PSPACE-complete [25]; on the other hand, complexity results of this kind are typically not available for logical systems introduced for specific texts.

It can be argued that the closest work to ours is [4], where the authors translate the language of deontic logics significantly more complex than $\mathbf{DL}^*$ into higher-order languages and then encode the latter into automatic theorem provers such as Isabelle/HOL [26] and LEO-3 [33]. The major difference with our approach turns out to be a matter of complexity in derivability-checking.

While the present work provides both theoretical grounds and a concrete tool for normative reasoning, it still covers a quite limited range of normative scenarios and we would like to extend this range in the future. For instance, since $\mathbf{DL}^*$ is a monotonic system, it is not obvious how to modify it to deal with *iterated contrary-to-duty reasoning* and *defeasible norms*. To this aim, it would be useful to support the base logical system with an external mechanism for conflict resolution, such as a mechanism applying the *principle of specificity* (see, e.g., [3] and [11]) to the antecedents of normative conditionals occurring in an *NDSIC*. Specificity would make it possible to transform an *NDSIC* by removing a conditional $A \Rightarrow_{Ob} B$ whenever there is another conditional $C \Rightarrow_{Ob} D$ s.t. C logically entails A (but not vice versa) and D logically entails $\neg B$. Furthermore, we would like to extend the language of $\mathbf{DL}^*$ with symbols to make explicit reference to *sanctions* and, in particular, to have sanctions of a different degree. This would allow us to keep formal track of cases in which a violation can be compensated incurring in a smaller sanction or in no sanction at all. Finally, we would like to extend the approach presented in this manuscript by taking into account also abductive reasoning [23].

ACKNOWLEDGEMENTS

Matteo Pascucci was supported by the WWTF project MA16–028.

REFERENCES

[1] L.E. Allen. 1957. Symbolic logic: a razor-edged tool for drafting and interpreting legal documents. *Yale Law J.* 66 (1957), 833–879.

[2] L.E. Allen. 1982. Towards a normalized language to clarify the structure of legal discourse. In *Deontic Logic, Computational Linguistics, and Legal Information Systems*, A.A. Martino (Ed.), North Holland, Amsterdam, 349–407.

[3] N. Asher and D. Bonevac. 1996. Prima facie obligation. *Studia Logica* 57, 1 (1996), 19–45.

[4] C. Benzmüller, A. Farjami and X. Parent. 2018. A dyadic deontic logic in HOL. In *Proceeedings of DEON 2018*, J.M. Broersen, C. Condoravdi, N. Shyam and G. Pigozzi (Eds.), College Publications, London, 33–50.

[5] G. Boella. L. Di Caro, L. Humphreys, L. Robaldo, P. Rossi and L. van der Torre. 2016. Eunomos, a legal document and knowledge management system for the web to provide relevant, reliable and up-to-date information on the law. *Artificial Intelligence and Law* 24, 3 (2016), 245–283.

[6] M. de Boer, D.M. Gabbay, X. Parent and M. Slavkovic. 2012. Two dimensional standard deontic logic [including a detailed analysis of the 1985 Jones-Pörn deontic logic system]. *Synthese* 187, 2 (2012), 623–660.

[7] C. Boutilier. 1994. Unifying default reasoning and belief revision in a modal framework. *Artificial Intelligence* 68, 1 (1994), 33–85.

[8] C. Boutilier. 1994. Conditional logics of normality: a modal approach. *Artificial Intelligence* 68, 1 (1994), 87–154.

[9] J. Carmo and A. Jones. 2002. Deontic logic and contrary to duties. In *Handbook of Philosophical Logic*, 2nd edition, D.M. Gabbay and F. Guenthner (Eds.), 256–344.

[10] D.J. Fonseca, E. Richards, D. Williamson and G.P. Moynihan. 2005. A knowledge-based system for the recycling of non-hazardous industrial residuals in civil engineering applications. *Expert Systems* 22, 1 (2005), 1–11.

[11] L. Goble. Prima facie norms, normative conflicts, and dilemmas. 2013. In D. Gabbay, J. Horty, X. Parent, R. van der Meyden, and L. van der Torre (Eds.), *Handbook of Deontic Logic and Normative Systems*, 241–352.

[12] G. Governatori. 1996. Labelling ideality and subideality. In *Proceedings of FAPR 1996*, D.M. Gabbay and H.J. Ohlbach (Eds.), 291–304.

[13] G. Governatori and Z. Milosevic. 2006. A formal analysis of a business contract language. *International Journal of Cooperative Information Systems* 15, 4 (2006), 659–685.

[14] G. Governatori and A. Rotolo. 2010. A conceptually rich model of business process compliance. In *Proceedings of APCCM 2010*, S. Link, A. Ghose (Eds.), 3–12.

[15] G. Governatori and S. Shek. 2013. Regorous: a business process compliance checker. In *Proceedings of ICAIL 2013*, E. Francesconi and B. Verheij (Eds.), 245–246.

[16] S.O. Hansson. 1989. A note on the deontic system DL of Jones and Pörn. *Synthese* 80, 3 (1989), 427–428.

[17] I.L. Humberstone. 1983. Inaccessible worlds. *Notre Dame Journal of Formal Logic* 24, 3 (1983), 346–352.

[18] J. Jiang, H. Aldewereld, V. Dignum and Y.H. Tan. 2015. Compliance checking of organizational interactions. *ACM Transactions on Management Information Systems* 5, 4 (2015), 1–24.

[19] A. Jones and I. Pörn. 1985. Ideality, sub-ideality and deontic logic. *Synthese* 65, 2 (1985), 275–290.

[20] A. Jones and M. Sergot. 1992. Deontic logic in the representation of law: towards a methodology. *Artificial Intelligence and Law* 1, 1 (1992), 45–64.

[21] H.J. Levesque. 1990. All I know: a study in autoepistemic logic. *Artificial Intelligence* 42, 2–3 (1990), 263–309.

[22] C.A. Magni, S. Malagoli and G. Mastroleo. 2006. An alternative approach to firms' evaluation: expert systems and fuzzy logic. *International Journal of Information Technology and Decision Making* 5, 1 (2006), 195–225.

[23] M.C. Mayer and P. Fiora. 1995. Propositional abduction in modal logic. *Logic Journal of the IGPL* 3, 6 (1995), 907–919.

[24] P.E. Navarro and J.L. Rodriguez. 2014. *Deontic logic and legal systems*. Cambridge University Press, Cambridge.

[25] L.A. Nguyen. 2004. On the complexity of fragments of modal logics. In *Proceedings of AIML 2004*, R.A. Schmidt, I. Pratt-Hartmann, M. Reynolds and H. Wansing (Eds.), 249–268.

[26] T. Nipkow, L.C. Paulson and M. Wenzel. 2002. *Isabelle/HOL. A Proof Assistant for Higher-Order Logic*. Springer, Berlin.

[27] J. Otten. 2012. Implementing connection calculi for first-order modal logics. In *Proceedings of IWIL 2012*, E. Ternovska, K. Korovin, S. Schulz (Eds.), 18–32.

[28] J. Otten. 2014. MleanCoP: a connection prover for first-order modal logic. In *Proceedings of IJCAR 2014*, S. Demri, D. Kapur, C. Weidenbach (Eds.), 269–276.

[29] M. Palmirani and G. Governatori. 2018. Modelling legal knowledge for GDPR compliance checking. In *Proceedings of JURIX 2018*, V. Rodríguez-Doncel, P. Casanovas, J. González-Conejero, E. Montiel-Ponsoda (Eds.), 101–110.

[30] H. Prakken and M. Sergot. 1996. Contrary to duty obligations. *Studia Logica* 57, 1 (1996), 91–105.

[31] U. Schild and R. Kannai. 2005. Intelligent computer evaluation of offender's previous record. *Artificial Intelligence and Law*. 13, 3-4 (2005), 373–405.

[32] R.A. Schmidt. 2006. Developing modal tableaux and resolution methods via first-order resolution. In *Proceedings of AIML 2006*, G. Governatori, I.M. Hodkinson and Y. Venema (Eds.), 1–26.

[33] A. Steen and C. Benzmüller. 2018. The higher-order prover Leo-III. In *Proceedings of IJCAR 2018*. D. Galmiche, S. Schulz, R. Sebastiani (Eds.), 108–116.

[34] E.D. Übeyli. 2009. Advances in medical decision support systems. *Expert Systems* 26, 1 (2009), 3–7.

[35] L. van der Torre. 1997. Reasoning about obligations. PhD Thesis, Erasmus Universiteit Rotterdam.

[36] T.K. Wah and M. Muniandy. 2014. Courtroom decision support system using case based reasoning. *Procedia: Social and Behavioral Sciences* 129 (2014), 489–495.

[37] J. Zeleznikow and A. Stranieri. 1995. The Split-Up System: integrating neural networks and rule-based reasoning in the legal domain. *Proceedings of ICAIL 2005*, L. Thorne McCarty (Eds.), 185–194.