A Defeasible Deontic Logic for Pragmatic Oddity

Guido Governatori

Data61, CSIRO, Australia

Silvano Colombo Tosatto

Data61, CSIRO, Australia

Antonino Rotolo

Alma Human AI, University of Bologna, Italy

Abstract

We introduce a variant of Deontic Defeasible Logic to handle the issue of Pragmatic Oddity. The key idea is that a conjunctive obligation is allowed only when each individual obligation is independent from the violation of the other obligations. The solution makes essential use of the constructive proof theory of the logic while maintaining a feasible computational complexity.

Keywords: Pragmatic Oddity, Defeasible Deontic Logic

1 Introduction

A differentiator between norms and other constraints is that, typically, (legal) norms can be violated. Moreover, normative systems contain provisions about other norms that become effective when violations occur. Since the seminal work by Chisholm [3] the obligations in force triggered by violations have been dubbed contrary-to-duty obligations (CTDs). The treatment of CTDs has proven problematic for formal (logical) representations of normative systems. Accordingly, CTDs are the source for many paradoxes and the driver for the development of many formalisms and deontic logics. The contribution in this paper follows the tradition: we are going to propose an extension of a logic (Defeasible Deontic Logic) that addresses the Pragmatic Oddity CTD paradox.

The problem of Pragmatic Oddity, introduced by Prakken and Sergot [11], is illustrated by the scenario that when you make a promise, you have to keep it. But if you do not, then you have to apologise. The oddity is that when you fail to keep your promise, you have the obligation to keep the promise and the obligation to apologise. In our view, what is odd, is not that the two obligations are in force at the same time, but that if one admits for form a conjunctive obligation from the two individual obligations then we get an obligation that is impossible to comply with. In the scenario, when the promise is broken, we have the conjunctive obligation obligation to keep the promise and to apologise for not having kept the promise.

The Pragmatic Oddity arises when we have a conjunctive obligation, i.e., $\mathsf{O}(a \wedge b)$, derived from the two individual obligations ($\mathsf{O}a$ and $\mathsf{O}b$) where one of the conjuncts is a contrary-to-duty obligation triggered by the violation of the other individual obligation, for example when $\neg a$ entails that $\mathsf{O}b$ is in force.

Most of the work on Pragmatic Oddity (e.g., [11,2]) focuses on the issue of how to distinguish the mechanisms leading to the derivation of the two individual obligations, and create different classes of obligations. Consequently, the solution to the Pragmatic Oddity problem is to prevent the conjunction when the obligations are from different classes. Accordingly, if the problem is to prevent that a conjunctive obligation is in force when the individual obligations are in force themselves, the simplest solution is to have a deontic logic that does not support the aggregation axiom [1] :

$$(\mathsf{O}a \wedge \mathsf{O}b) \rightarrow \mathsf{O}(a \wedge b)$$

However, a less drastic solution, advocated by Parent and van der Torre [9,10], is to restrict the aggregation axiom to independent obligations (meaning that one obligation should not depend on the violation of the other obligation).

We are going to take Parent and van der Torre's suggestion and propose a simple mechanism in Defeasible Deontic Logic to guard the derivation of conjunctive obligations. The mechanism guarantees that the obligations of a conjunctive obligation are independent from the violations of the individual obligations. The mechanism is founded on the proof theory of the logic.

2 Defeasible Deontic Logic

Defeasible Deontic Logic [5] is a sceptical computationally oriented rule-based formalism designed for the representation of norms. The logic extends Defeasible Logic [1] with deontic operators to model obligations and (different types of) permissions and provides an integration with the logic of violation proposed in [7]. The resulting formalism offers features for the natural and efficient representation of exceptions, constitutive and prescriptive rules and of compensatory norms. The logic is based on a constructive proof theory that allows for full traceability of the conclusions, and flexibility to handle and combine different facets of non-monotonic reasoning. In the rest of this section we are going to show how the proof theory can be used to propose a simple and (arguably) elegant treatment of the issue of Pragmatic Oddity.

We restrict ourselves to the fragment of Defeasible Deontic Logic that excludes permission and permissive rules, since they do not affect the way we handle Pragmatic Oddity: Definitions 2.12 and 2.13, the definitions that describe the mechanisms we adopt for a solution to Pragmatic Oddity, are independent

[1] See, among others, [4].

from any issue related to permission. The definitions be used directly in the full version of the logic. Accordingly, we consider a logic whose language is defined as follows.

Definition 2.1 Let PROP be a set of propositional atoms, O the modal operator for obligation.

- The set $\mathrm{Lit} = \mathrm{PROP} \cup \{\neg p \mid p \in \mathrm{PROP}\}$ is the set of *literals*.
- The *complement* of a literal q is denoted by $\sim q$; if q is a positive literal p, then $\sim q$ is $\neg p$, and if q is a negative literal $\neg p$, then $\sim q$ is p.
- The set of *deontic literals* is $\mathrm{DLit} = \{\mathsf{O}l, \neg\mathsf{O}l \mid l \in \mathrm{Lit}\}$.
- If $c_1, \ldots, c_n \in \mathrm{Lit}$, then $\mathsf{O}(c_1 \wedge \cdots \wedge c_n)$ is a *conjunctive obligation*.

In the rest of the paper, when relevant to the discussion, we will refer to elements of Lit as plain literals, and often we will use the unmodified term 'literal' to indicate either a plain literal or a deontic literal.

We introduce the compensation operator $\otimes$. This operator is used to build chains of compensation called $\otimes$-expressions. The formation rules for well-formed $\otimes$-expressions are:

(i) every literal $l \in \mathrm{Lit}$ is an $\otimes$-expression;

(ii) if $c_1, \ldots, c_k \in \mathrm{Lit}$, then $c_1 \otimes \cdots \otimes c_k$ is an $\otimes$-expression;

(iii) nothing else is an $\otimes$-expression.

In addition we stipulate that $\otimes$ obeys the following property (duplication and contraction on the right):

$$\bigotimes_{i=1}^{n} a_i = \Big(\bigotimes_{i=1}^{k-1} a_i\Big) \otimes \Big(\bigotimes_{i=k+1}^{n} a_i\Big)$$

where there exists j such that $a_j = a_k$ and $j < k$.

Given an $\otimes$-expression A, the *length* of A is the number of literals in it. Given an $\otimes$-expression $A \otimes b \otimes C$ (where A and C can be empty), the *index* of b is the length of $A \otimes b$. We also say that b appears at index n in $A \otimes b$ if the length of $A \otimes b$ is n.

The meaning of a compensation chain

$$c_1 \otimes c_2 \otimes \cdots \otimes c_n$$

is that $\mathsf{O}c_1$ is the primary obligation, and when violated (i.e., $\neg c_1$ holds), then $\mathsf{O}c_2$ is in force and it compensates for the violation of the obligation of c_1. Moreover, when $\mathsf{O}c_2$ is violated, then $\mathsf{O}c_3$ is in force, and so on until we reach the end of the chain when a violation of the last element is a non-compensable violation where the norm corresponding to the rule in which the chain appears is not complied with.

We adopt the standard DL definitions of *strict rules*, *defeasible rules*, and *defeaters* [1]. However, for the sake of simplicity, and to better focus on the non-monotonic aspects that DL offers, in the remainder we use only defeasible rules and defeaters. Also, we have to take the obligation operator into account.

Definition 2.2 Let Lab be a set of arbitrary labels. Every rule is of the type

$$r\colon A(r) \hookrightarrow C(r)$$

where

(i) $r \in$ Lab is the name of the rule;

(ii) $A(r) = \{a_1, \dots, a_n\}$, the *antecedent* (or *body*) of the rule, is the set of the premises of the rule (alternatively, it can be understood as the conjunction of all the elements in it). Each a_i is either a literal, a deontic literal or a conjunctive obligation;

(iii) $\hookrightarrow \in \{\Rightarrow, \Rightarrow_{\mathsf{O}}, \leadsto, \leadsto_{\mathsf{O}}\}$ denotes the type of the rule. If $\hookrightarrow$ is $\Rightarrow$, the rule is a *defeasible rule*, while if $\hookrightarrow$ is $\leadsto$, the rule is a *defeater*. Rules without the subscript O are constitutive rules, while rules with such a subscript are prescriptive rules.

(iv) $C(r)$ is the *consequent* (or *head*) of the rule. It is a single literal for defeaters and constitutive rules, and an $\otimes$-expressions for prescriptive defeasible rules.

As we will see, prescriptive rules are used to derive obligations.

Given a set of rules R, we use the following abbreviations for specific subsets of rules:

- R_d denotes the set of defeasible rules in the set R;
- $R[q, n]$ is the set of rules where q appears at index n in the consequent. The set of rules where q appears at any index n is denoted by $R[q]$;
- R^{O} denotes the set of prescriptive rules in R, i.e., the set of rules with O as their subscript;
- R^C denotes the set of constitutive rules in R, i.e., $R \setminus R^{\mathsf{O}}$.

The above notations can be combined. Thus, for example, $R^{\mathsf{O}}_d[q, n]$ stands for the set of defeasible prescriptive rules such that q appears at index n in the consequent of the rule.

Definition 2.3 A *Defeasible Theory* is a structure $D = (F, R, >)$, where F, the set of facts, is a set of literals and deontic literals, R is a set of rules and $>$, the superiority relation, is a binary relation over R.

A theory corresponds to a normative system, i.e., a set of norms, where every norm is modelled by some rules. The superiority relation is used for conflicting rules, i.e., rules whose conclusions are complementary literals, in case both rules fire. We do not impose any restriction on the superiority relation: it just determines the relative strength between two rules.

Definition 2.4 A *proof* (or derivation) P in a defeasible theory D is a linear sequence $P(1) \dots P(z)$ of *tagged literals* in the form of $+\partial q$, $-\partial q$, $+\partial_{\mathsf{O}} q$, $-\partial_{\mathsf{O}} q$,

$+\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$ and $-\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$ where $P(1) \ldots P(z)$ satisfy the proof conditions given in Definitions 2.8–2.13.

The tagged literal $+\partial q$ means that q is *defeasibly provable* as an institutional statement, or in other terms, that q holds in the normative system encoded by the theory. The tagged literal $-\partial q$ means that q is *defeasibly refuted* by the normative system. Similarly, the tagged literal $+\partial_{\mathsf{O}} q$ means that q is *defeasibly provable* in D as an obligation, while $-\partial_{\mathsf{O}} q$ means that q is *defeasibly refuted* as an obligation. For $+\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$ the meaning is that the conjunctive obligation $\mathsf{O}(c_1 \wedge \cdots \wedge c_m)$ is defeasibly derivable; and that a conjunctive obligation $\mathsf{O}(c_1 \wedge \cdots \wedge c_m)$ is defeasibly refuted corresponds to $-\partial_{\mathsf{O}}(c_1 \wedge \cdots \wedge c_m)$. The initial part of length i of a proof P is denoted by $P(1..i)$.

The first thing to do is to define when a rule is applicable or discarded. A rule is *applicable* for a literal q if q occurs in the head of the rule, all elements in the antecedent have been defeasibly proved (eventually with the appropriate modalities). On the other hand, a rule is *discarded* if at least one of the modal literals in the antecedent has not been proved. However, as literal q might not appear as the first element in an $\otimes$-expression in the head of the rule, some additional conditions on the consequent of rules must be satisfied. Defining when a rule is applicable or discarded is essential to characterise the notion of provability for constitutive rules and then for obligations ($\pm\partial_{\mathsf{O}}$).

Definition 2.5 Given a proof P, a rule $r \in R$ is *body-applicable* at step $P(n+1)$ iff for all $a_i \in A(r)$:

(i) if $a_i = \mathsf{O}l$ then $+\partial_{\mathsf{O}} l \in P(1..n)$;

(ii) if $a_i = \neg\mathsf{O}l$ then $-\partial_{\mathsf{O}} l \in P(1..n)$;

(iii) if $a_i = \mathsf{O}(c_1 \wedge \cdots \wedge c_m)$ then $+\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m \in P(1..n)$;

(iv) if $a_i = l \in \mathrm{Lit}$ then $+\partial l \in P(1..n)$.

A rule $r \in R[q, j]$ is *body-discarded* at step $P(n+1)$ iff $\exists a_i \in A(r)$ such that

(i) if $a_i = \mathsf{O}l$ then $-\partial_{\mathsf{O}} l \in P(1..n)$;

(ii) if $a_i = \neg\mathsf{O}l$ then $+\partial_{\mathsf{O}} l \in P(1..n)$;

(iii) if $a_i = \mathsf{O}(c_1 \wedge \cdots \wedge c_m)$ then $-\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m \in P(1..n)$;

(iv) if $a_i = l \in \mathrm{Lit}$ then $-\partial l \in P(1..n)$.

Definition 2.6 Given a proof P, a rule $r \in R^{\mathsf{O}}[q, j]$ such that $C(r) = c_1 \otimes \cdots \otimes c_m$ is *applicable* for literal q at index j at step $P(n+1)$ (or, simply, applicable for q), with $1 \leq j < m$, in the condition for $\pm\partial_{\mathsf{O}}$ iff

(i) r is body-applicable at step $P(n+1)$; and

(ii) for all $c_k \in C(r)$, $1 \leq k < j$, $+\partial_{\mathsf{O}} c_k \in P(1..n)$ and $+\partial{\sim} c_k \in P(1..n)$.

Conditions (i) represents the requirements on the antecedent stated in Definition 2.5; condition (ii) on the head of the rule states that each element c_k prior to q must be derived as an obligation, and a violation of such obligation has occurred.

Definition 2.7 Given a proof P, a rule $r \in R^{\mathsf{O}}[q,j]$ such that $C(r) = c_1 \otimes \cdots \otimes c_m$ is *discarded* for literal q at index j at step $P(n+1)$ (or, simply, discarded for q), with $1 \leq j \leq m$, in the condition for $\pm\partial_{\mathsf{O}}$ iff
(i) r is body-discarded at step $P(n+1)$; or
(ii) there exists $c_k \in C(r)$, $1 \leq k < l$, such that either $-\partial_{\mathsf{O}} c_k \in P(1..n)$ or $+\partial c_k \in P(1..n)$.

In this case, condition (institutional) ensures that an obligation prior to q in the chain is not in force or has already been fulfilled (thus, no reparation is required).

We now introduce the proof conditions for $\pm\partial$ and $\pm\partial_{\mathsf{O}}$:

Definition 2.8 The proof condition of *defeasible provability for an institutional statement* is
$+\partial$: If $P(n+1) = +\partial q$ then
(1) $q \in F$ or
(2.1) $\sim q \notin F$ and
(2.2) $\exists r \in R_d[q]$ such that r is applicable for q, and
(2.3) $\forall s \in R[\sim q]$, either
(2.3.1) s is discarded for $\sim q$, or
(2.3.2) $\exists t \in R[q]$ such that t is applicable for q and $t > s$.

As usual, we use the strong negation to define the proof condition for $-\partial$

Definition 2.9 The proof condition of *defeasible refutability for an institutional statement* is
$-\partial$: If $P(n+1) = -\partial q$ then
(1) $q \notin F$ and
(2.1) $\sim q \in F$ or
(2.2) $\forall r \in R_d[q]$: either r is discarded for q, or
(2.3) $\exists s \in R[\sim q]$, such that
(2.3.1) s is applicable for $\sim q$, and
(2.3.2) $\forall t \in R[q]$ either t is discarded for q or not $t > s$.

The proof conditions for $\pm\partial$ are the standard conditions in defeasible logic, see [1] for the full explanations.

Definition 2.10 The proof condition of *defeasible provability for obligation* is
$+\partial_{\mathsf{O}}$: If $P(n+1) = +\partial_{\mathsf{O}} q$ then
(1) $\mathsf{O}q \in F$ or
(2.1) $\mathsf{O}{\sim}q \notin F$ and $\neg\mathsf{O}q \notin F$ and
(2.2) $\exists r \in R^{\mathsf{O}}_d[q,i]$ such that r is applicable for q, and
(2.3) $\forall s \in R^{\mathsf{O}}[\sim q, j]$, either
(2.3.1) s is discarded for $\sim q$, or
(2.3.2) $\exists t \in R^{\mathsf{O}}[q,k]$ such that t is applicable for q and $t > s$.

To show that q is defeasibly provable as an obligation, one must show either that: (1) the obligation of q is a fact, or (2) q must be derived by the rules of the theory. In the second case, three conditions must hold: (2.1) q does not appear as not obligatory as a fact, and $\sim q$ is not provable as an obligation using the set of deontic facts at hand; (2.2) there must be a rule introducing the obligation for q which can apply; (2.3) every rule s for $\sim q$ is either discarded or defeated by a stronger rule for q.

The strong negation of Definition 2.10 gives the negative proof condition for obligation.

Definition 2.11 The proof condition of *defeasible refutability for obligation* is

$-\partial_{\mathsf{O}}$: If $P(n+1) = -\partial_{\mathsf{O}} q$ then
(1) $\mathsf{O}q \notin F$ and either
  (2.1) $\mathsf{O}{\sim}q \in F$ or $\neg\mathsf{O}q \in F$ or
  (2.2) $\forall r \in R_d^{\mathsf{O}}[q,i]$ either r is discarded for q, or
  (2.3) $\exists s \in R^{\mathsf{O}}[{\sim}q,j]$ such that
    (2.3.1) s is applicable for $\sim q$, and
    (2.3.2) $\forall t \in R^{\mathsf{O}}[q,k]$, either t is discarded for q or $t \not> s$.

Notice that, given the intended correspondence between $\mathsf{O}l$ and $+\partial_{\mathsf{O}} l$, see Definition 2.5, we will refer to "the derivation of $\mathsf{O}l$" when, strictly speaking, we should use "the derivation of $+\partial_{\mathsf{O}} l$.

We are now ready to provide the proof condition under which a conjunctive obligation can be derived. The condition essentially combines two requirements: the first that a conjunction holds only when all the conjuncts hold (individually). The second requirement is that the derivation of one of the individual obligations does not depend on the violation of the other conjunct. To achieve this, we determine the line of the proof when the obligation appears, and then we check that the negation of the other elements of the conjunction does not occur in the previous derivation steps.

Definition 2.12 The proof condition of *defeasible provability for a conjunctive obligation* is

If $P(n+1) = +\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$, then
$\forall c_i$, $1 \leq i \leq m$,
(1) $+\partial_{\mathsf{O}} c_i \in P(1..n)$ and
(2) if $P(k) = +\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$, $k \leq n$, then
  $\forall c_j$, $1 \leq j \leq m$ and $c_j \neq c_i$, $+\partial{\sim}c_j \notin P(1..k)$.

Again, the proof condition to refute a conjunctive obligation is obtained by strong negation from the condition to defeasibly derive a conjunctive obligation.

Definition 2.13 The proof condition of *defeasible refutability for a conjunctive obligation* is

If $P(n+1) = -\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$, then
$\exists c_i$, $1 \leq i \leq m$, such that either
(1) $-\partial_{\mathsf{O}} c_i \in P(1..n)$ or
(2) if $P(k) = +\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_m$, $k \leq n$, then
  $\exists c_j$, $1 \leq j \leq m$ such that $c_j \neq c_i$ and $+\partial{\sim}c_j \in P(1..k)$.

In case of a binary conjunctive obligation the positive proof condition boils down to

$+\partial_{\mathsf{O}\wedge}$: If $P(n+1) = +\partial_{\mathsf{O}} p \wedge q$ then
(1) $+\partial_{\mathsf{O}} p \in P(1..n)$ and
(2) $+\partial_{\mathsf{O}} q \in P(1..n)$ and
(3) if $P(k) = +\partial_{\mathsf{O}} p$ $(k \leq n)$,then $+\partial{\sim}q \notin P(1..k)$ and
(4) if $P(k) = +\partial_{\mathsf{O}} q$ $(k \leq n)$, then $+\partial{\sim}p \notin P(1..k)$.
Similarly, for the condition for $-\partial_{\mathsf{O}\wedge}$.

Before moving on proving some theoretical results about the logic defined we give some examples that illustrate the behaviour of the logic. In what follows we use $\dots \Rightarrow c$ to refer to an applicable rule for c where we assume that the elements are not related (directly or indirectly) to the other literals used in the examples.

Compensatory Obligations The first case we want to discuss is when the conjunctive obligation corresponding to the Pragmatic Oddity has as conjuncts an obligation and its compensation. This scenario is illustrated by the rule:

$$\dots \Rightarrow_{\mathsf{O}} a \otimes b$$

In this case, it is clear that we cannot derive the conjunctive obligation of a and b, since the proof condition that allows us to derive $+\partial_{\mathsf{O}} b$ explicitly requires that $+\partial{\sim}a$ has been already derived (condition 2 of Definition 2.6). In this case, it is impossible to have the obligation of b without the violation of the obligation of a.

Contrary-to-duty The second case is when we have a CTD. The classical representation of a CTD is given by the following two rules:

$$\dots \Rightarrow_{\mathsf{O}} a \qquad\qquad \neg a \Rightarrow_{\mathsf{O}} b$$

In this case, it is possible to have situations when the obligation of b is in force without having a violation of the obligation of a, namely, when a is not obligatory. However, as soon as we have $\mathsf{O}a$, we need to derive $\neg a$ to trigger the derivation of $\mathsf{O}b$ (Definition 2.5).

Pragmatic Oddity via Intermediate Concepts The situations in the previous two cases can be easily detected by a simple inspection of the rules involved; there could be more complicated cases. Specifically, when the second conjunct does not immediately depends on the first conjunct, but it depends through a reasoning chain. The simplest structure for this case is illustrated by the following three rules:

$$\dots \Rightarrow_{\mathsf{O}} a$$
$$\neg a \Rightarrow b$$
$$b \Rightarrow_{\mathsf{O}} c$$

Here to derive $\mathsf{O}c$, we need first to prove b. To prove b we require that $\neg a$ has already been proved.

Negative Support In the previous case the support was through an intermediate concepts. However, given the non-monotonic nature of Defeasible Deontic Logic, we can have cases where the support is not to directly derive the other obligation from the violation, but the violation prevents the derivation of the prohibition (or the permission of the opposite) of the other conjunct. This situation is illustrated by the following set of rules: [2]

$$\cdots \Rightarrow_{\mathsf{O}} a$$
$$\cdots \Rightarrow_{\mathsf{O}} b$$
$$c \Rightarrow_{\mathsf{O}} \neg b$$
$$\cdots \Rightarrow c$$
$$\neg a \leadsto \neg c$$

To derive $\mathsf{O}b$, we have to ensure that the rule for $\mathsf{O}\neg b$ is discarded. This means that c should be rejected (i.e., $-\partial c$). We have two options, either the rule for c is discarded, or the rule for $\neg c$ is applicable. This implies that to prove $+\partial_{\mathsf{O}} b$ we have to prove first $+\partial\neg a$. Thus, one of the two elements of the conjunctive obligation $\mathsf{O}(a \wedge b)$ depends on the violation of the other.

Pragmatic Un-pragmatic Oddity What about when there are multiple norms both prescribing the contrary-to-duty obligation, and at least one of the norms is not related to the violation of the primary norm?

$$r_1 : \cdots \Rightarrow_{\mathsf{O}} a \otimes b$$
$$r_2 : \cdots \Rightarrow_{\mathsf{O}} b$$
$$\neg a$$

In this situation you can have a derivation:

(1)	$+\partial\neg a$	fact
(2)	$+\partial_{\mathsf{O}} a$	from r_1
(2)	$+\partial_{\mathsf{O}} b$	from r_1 and (1) and (2)

where the derivation of $\mathsf{O}b$ ($+\partial_{\mathsf{O}} b$) depends on the violation of the primary obligation of r_1. In this case, we cannot derive the conjunctive obligation of a and b. However, there is an alternative derivation, namely:

(1)	$+\partial_{\mathsf{O}} a$	from r_1
(2)	$+\partial_{\mathsf{O}} b$	from r_2
(3)	$+\partial\neg a$	fact
(4)	$+\partial_{\mathsf{O}} a \wedge b$	from (1) and (2)

that demonstrates the independence of $\mathsf{O}b$ from $\neg a$, given that the derivation of $\neg a$ occurs in a line after the line where $+\partial_{\mathsf{O}} b$ is derived.

[2] It is worth noting that, in the theory below, the rules for $\neg b$ and $\neg c$ can be either defeasible rules or defeaters producing the same result as far as the derivation of $\mathsf{O}(a \wedge b)$ is concerned.

3 Independence

As we have discussed the idea of the proof conditions above is to ensure that the individual obligations do not depend on the violations of the others. Accordingly, the question now is what does it mean that a formula is independent from another formula. In classical logic, given a theory T, a formula A depends on the formula B if $T \cup B \vdash A$, but $T \setminus B \nvdash A$. In Defeasible Deontic Logic, we have to remove all possible reasons to conclude the literal; this means we have to remove it from the facts and we have to remove the rules where it appears in the head of the rule. Since we are interested in removing only non deontic literals we can restrict the removal to the constitutive rules whose head is the literal to be removed. Accordingly, we can define the following transformation.

Definition 3.1 Given a defeasible theory $D = (F, R, >)$ and a literal l, the *Pragmatic Oddity Transformation* of D based on l, noted as $pot(D, l)$ is the defeasible theory $D' = (F', R', >')$ satisfying the following conditions:

(i) $F' = F \setminus \{l\}$;

(ii) $R' = R \setminus R[l]$;

(iii) $>' = > \setminus \{(r, s)\colon r \notin R' \vee s \notin R'\}$.

The transformation is to create a theory similar to the original theory but, as we said, without l. The condition on F is obvious. The second condition ensures that the rules that can derive the literal are removed. Then the literal is no longer derivable, since the resulting theory does not contain rules for the literal anymore. Given that $R'[l] = \emptyset$, the following result is immediate.

Observation 1 *Given a Defeasible Theory D and a literal l, $-\partial l$ is not derivable in $pot(D, l)$.*

It worth noting that we do not have to remove rules where the literal appears in the antecedent of the rule. Such rules are immediately discarded. Similarly, for prescriptive rules where the complement of the removed literal appears in the head of the rules. Such rules are no longer applicable for any elements appearing after the complement of the removed literal. Thus if you have a rule with the $\otimes$-chain $c_1 \otimes \cdots \otimes c_n \otimes \neg l \otimes c_{n+1} \cdots$, the rules in $R^{\mathsf{O}}[c, m]$ for any $m \geq n + 1$ are not applicable. Remember, that to derive $+\partial_{\mathsf{O}} c_{n+1}$ we have to prove both $+\partial_{\mathsf{O}} \neg l$ and $+\partial l$. The transformation *pot* is then extended to the case of a (finite) set of literals $L = \{l_1, \ldots, l_n\}$ by applying the transformation to all the literals in L; thus $pot(D, L) = pot(\cdots (pot(D, l_1), \cdots l_n)$ for an arbitrary sequence of all the elements in L.

We can now specify when a (deontic) literal is independent from a set of plain literals in Defeasible Deontic Logic

Definition 3.2 Given a defeasible theory D, a set L of plain literals and a literal m, m is *independent from* L iff m is defeasibly provable in D and in $pot(D, L)$.

We can now show that the condition (2) in the proof conditions for a conjunctive obligation ensures the independence of the obligations from the violations. However, before proving this result we have to recall a general property about Defeasible (Deontic) Logic: First of all a defeasible theory is consistent if F does not contain a literal l and its complement $\neg l$. Second, given a logical formula expressing a proof condition the strong negation of the formula/conditions is obtained by replacing every occurrence of a positive proof tag with the corresponding negative proof tag, replacing conjunctions with disjunctions, disjunctions with conjunctions, existential with universal and universal with existential. It is immediate to observe that all negative proof conditions given in this section are the strong negation of the corresponding positive one (and the other way around). If corresponding proof conditions are defined using the principle of strong negation outlined above, then, given a derivation, it is not possible to have that the literal (conjunctive obligation) is both derivable and refutable in the same derivation.

Proposition 3.3 *[6] Given a consistent defeasible theory D, a derivation P, a literal l, and proof tag $\# \in \{\partial, \partial_{\mathsf{O}}\}$ it is not possible that $+\#l, -\#l \in P$.*

Armed with this result we can prove the result linking independence and the proof conditions for conjunctive obligations.

Proposition 3.4 *Given a consistent defeasible theory D, a deontic literal m and a set L of plain literals. m is independent from L iff there is a derivation P in D such that*

- $P(n) = +\partial_{\mathsf{O}} m$ *and*
- $\forall l \in L$, $+\partial l \notin P(1..n)$.

4 Complexity

In this section, we are going to study the computational complexity of the problem of computing whether a conjunctive obligation is derivable from a given defeasible theory. To this end, we adapt the algorithm proposed in [5] to compute the extension of a defeasible theory, where the computation of the extension is linear in the size of the theory. The algorithm is based on a series of transformations that reduce the complexity of the theory, by either removing elements from rules when some elements are provable, and removing rules when they become discarded (and so no longer able to produce positive conclusions). Using the idea in [5] the extension of a defeasible theory D is defined as follows:

Definition 4.1 Given a theory D, the *literal extension* of D is the tuple

$$\langle \partial^+(D), \partial^-(D), \partial^+_{\mathsf{O}}(D), \partial^-_{\mathsf{O}}(D) \rangle$$

where

- $\partial^+(D)$ is the set of literals appearing in D that are defeasibly provable as institutional statements;
- $\partial^-(D)$ is the set of literals appearing in D that are defeasibly refutable as institutional statements;

- $\partial_{\mathsf{O}}^{+}(D)$ is the set of literals appearing in D that are defeasibly provable as obligations;
- $\partial_{\mathsf{O}}^{-}(D)$ is the set of literals appearing in D that are defeasibly refutable as obligations;

The aim of the paper is to determine when conjunctive obligations are either provable or discarded. Accordingly, we have to extend the definition to account for conjunctive obligation. However, if we want to maintain a feasible computational complexity we have to limit the conjunctions we are going to consider: given a set of n literals the set of all possible non logically equivalent conjunctions that can be formed by the n literals contains 2^n conjunctions; hence, we cannot compute in polynomial time for such a set if any element is derivable or refuted by the theory. However, we are going to show that for each individual conjunction we can compute in polynomial time whether it is derivable or refuted.

Definition 4.2 Given a defeasible theory D the *conjunctive extension* of the theory is the tuple:

$$\langle \partial^{+}(D), \partial^{-}(D), \partial_{\mathsf{O}}^{+}(D), \partial_{\mathsf{O}}^{-}(D), \partial_{\wedge}^{+}(D), \partial_{\wedge}^{-}(D) \rangle$$

where $\partial^{+}(D)$, $\partial^{-}(D)$, $\partial_{\mathsf{O}}^{+}(D)$ and $\partial_{\mathsf{O}}^{-}(D)$ are as in Definition 4.1 and

- $\partial_{\wedge}^{+}(D)$ is the set of conjunctive obligations appearing in D (i.e., $c = \mathsf{O}(c_1 \wedge \cdots \wedge c_n)$ and $\exists r \in R$ such that $c \in A(r)$) that are defeasibly provable in D (Definition 2.12);
- $\partial_{\wedge}^{-}(D)$ is the set of conjunctive obligations appearing in D that are defeasibly refutable in D (Definition 2.13).

The algorithm to determine the conjunctive extension of a theory is based on the following data structure (for the full details we refer the reader to [5]). We create a list of the atoms appearing in the theory. Every entry in the list of atoms has an array associated to it. The array has ten cells, where every cell contains pointers to rules depending on whether and how the atom appears in the rule. The first cell is where the atom appears in the head of a constitutive rule, the second where the negation of the atom appears in the head of a constitutive rule, the third where the atom appears in the head of a prescriptive rule, the fourth where the negation of atom appears in the head of a prescriptive rule, the fifth where the atom appears in the body of a rule, the sixth where the negation of the atom appears in the body of a rule, the seventh where the atom appears as an obligation in the body of a rule, the eighth where the negation of the atom appears as an obligation in the body of a rule, the ninth where the atom appears as a negative obligation in the body of a rule, and the tenth where the negation of the atom appears as a negative obligation in the body of a rule. In addition, we maintain a list of conjunctive obligations occurring in the theory, and for every conjunction we associate it to the rules where it appears in the body.

The algorithm works as follows: at every round we scan the list of atoms. For every atom (excluding the entries for the conjunctions) we look if the atom appears in the head of some rules. If it does not appears in any of the cells for the heads, we can set the corresponding literals as refuted; and we can remove rules, from corresponding cells. So, for example, given an atom p, if there are no prescriptive rules for $\neg p$, then, we can conclude that the theory proves $-\partial_{\mathsf{O}}\neg p$; accordingly, all rules where $\neg\mathsf{O}\neg p$ occurs in the body are (body)-discarded, and we can remove them from the data structure. Similarly, if there are no constitutive rules for $\neg p$, then we can prove $-\partial\neg p$, and, then (i) all the rules where it appears in the body are body-discarded, but also, for each rule r in whose head p appears as an obligation, no elements following p in r can any longer be derived using r and such elements are removed from the appropriate cells. If an atom appears in the head of a rule, we determine (i) if the body of the rule is empty, and (ii) for prescriptive rules, if the atom is the first element of the head. If this is the case, then, the rule is applicable, and we check if there are rules for the negation. If there are no rules for the negation, or the rules are weaker than applicable rules, then the atom/literal is provable with the suitable proof tag, and then we remove the atom/literal from the appropriate rules. We repeat the above steps until we are no longer able to obtain new conclusions. When, we are no able to derive new conclusion we turn our attention to the list of the conjunctive obligations, where we invoke the following (sub)algorithm for every conjunction $c = \mathsf{O}(c_1 \wedge \cdots \wedge c_n)$ in the list (where $C = \{\sim c_i, 1 \leq i \leq n\}$)

Algorithm 1 Evaluate Conjunctive Obligation

for $i \in 1..n$ **do**
  if $c_i \in \partial_{\mathsf{O}}^{-}(D)$ **then**
    $c \in \partial_{\wedge}^{-}(D)$ remove all rules r where $c \in A(R)$
    Exit
  end if
  if $c_i \in \partial_{\mathsf{O}}^{+}(D)$ **then**
    if $\forall c_j \neg c_i, \sim c_j \in \partial^{+}(D)$ **then**
      if $c_i \in +\partial_{\mathsf{O}}^{+}(pot(D, C \setminus \{\sim c_i\})$ **then**
        $i := i + 1$
      else $c \in \partial_{\wedge}^{-}(D)$ remove all rules r where $c \in A(R)$
        Exit
      end if
      if $\exists c_j \neq c_i, \sim c_i \in \partial^{-}(D)$ **then**
        $i := i + 1$
      end if
    end if
  end if
  Exit
end for
$c \in \partial_{\wedge}^{+}(D)$, remove c from all rules r where $c \in A(r)$

For every conjunction the algorithm iterates over the conjuncts. If a conjunct is not provable as an obligation the conjunction is not provable (line 2–4). If the conjunct is provable as an obligation, it checks whether the violations of the other obligations are provable; if so, it has to check whether the obligation of the conjunct is independent from the violations. To determine this, we can repeat the whole algorithm with the the sub-theory obtained by the transformation $pot(D, C \setminus \{c_i\})$. If it is independent we continue with the next element of the conjunction; otherwise, the conjunction is not derivable. Similarly, if some of violations are not derivable we continue with the iteration. The conjunction is provable when the iteration is successful for all the elements of the conjunction.

At the end of the sub-routine, we return to the main algorithm, if there are changes in the rules we repeat the process, otherwise the process terminates.

The algorithm outline above is sound and complete; hence, we can state the following proposition. Essentially, the correctness of the algorithm depends on Proposition 3.4.

Proposition 4.3 *Given a defeasible theory D*

- $+\partial l$ *is defeasibly provable in D iff* $l \in \partial^{+}(D)$;
- $-\partial l$ *is defeasibly provable in D iff* $l \in \partial^{-}(D)$;
- $+\partial_{\mathsf{O}} l$ *is defeasibly provable in D iff* $l \in \partial_{\mathsf{O}}^{+}(D)$;
- $-\partial_{\mathsf{O}} l$ *is defeasibly provable in D iff* $l \in \partial_{\mathsf{O}}^{-}(D)$;
- $+\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_n$ *is defeasibly provable in D iff* $c_1 \wedge \cdots \wedge c_n \in \partial_{\wedge}^{+}(D)$;
- $-\partial_{\mathsf{O}} c_1 \wedge \cdots \wedge c_n$ *is defeasibly provable in D iff* $c_1 \wedge \cdots \wedge c_n \in \partial_{\wedge}^{-}(D)$.

As far as the computational complexity, [5] proves that the complexity of computing the extension of a defeasible theory without conjunctive obligation is linear in the size of the theory, where the size of the theory is determined by the number of symbols in the theory, and hence if n and r stand for, respectively, the number of atoms and the number of rules in the theory, the complexity is in $O(n * r)$. For the complexity of computing the conjunctive extension of a defeasible theory we have to take into account the complexity of the Evaluate Conjunctive Obligation algorithm and the number of times we have to compute it. This can be determined as follows: let m be the number of conjunctive obligations in the theory, and k the number of conjuncts in the longest conjunctive obligation. For each of them we have to compute the extension of $pot(D, C)$, thus we have to perform $O(m * k * O(n * r))$ computations on top of the computation of the extension (i.e., $O((m + n) * r)$).

Proposition 4.4 *The conjunctive extension of a theory can be computed in polynomial time.*

Notice that the algorithm Evaluate Conjunctive Obligation can be use the evaluate any conjunctive obligation not only the conjunctive obligations occurring in a theory. All we have to do is to compute the conjunctive extension of the theory and then evaluate the single conjunctive obligation, and as we have just seen this can be computed in polynomial time.

5 Summary

We have proposed an extension of Defeasible Deontic Logic able to handle the so called Pragmatic Oddity paradox. The mechanism we used to achieve this result was to provide a schema that allows us to give a guard to the derivation of conjunctive obligations ensuring that each individual obligation does not depend on the violation of the other obligation. The mechanism is given by the proof theory of defeasible logic.

While the complexity of the logic is polynomial and hence feasible the algorithm we propose is not optimal. Nonetheless, this is practical for most real life applications, in which it is likely there will be few conjunctive obligations, each with only a small number of conjuncts; however, the next step is to to devise an optimal algorithm to implement the novel proof conditions.

Acknowledgments

A preliminary version of the paper presenting the idea of the logic was presented at Jurix 2019 [8]. We thanks the anonymous reviewers for their valuable comments on an earlier version of the paper.

References

[1] Grigoris Antoniou, David Billington, Guido Governatori, and Michael J. Maher. Representation results for defeasible logic. *ACM Transactions on Computational Logic*, 2(2):255–287, 2001.

[2] José Carmo and Andrew JI Jones. Deontic logic and contrary-to-duties. In *Handbook of philosophical logic*, pages 265–343. Springer, 2002.

[3] Roderick M Chisholm. Contrary-to-duty imperatives and deontic logic. *Analysis*, 24(2):33–36, 1963.

[4] Lou Goble. A logic for deontic dilemmas. *Journal of Applied Logic*, 3(3-4):461–483, 2005.

[5] Guido Governatori, Francesco Olivieri, Antonino Rotolo, and Simone Scannapieco. Computing strong and weak permissions in defeasible logic. *Journal of Philosophical Logic*, 42(6):799–829, 2013.

[6] Guido Governatori, Vineet Padmanabhan, Antonino Rotolo, and Abdul Sattar. A defeasible logic for modelling policy-based intentions and motivational attitudes. *Logic Journal of the IGPL*, 17(3):227–265, 2009.

[7] Guido Governatori and Antonino Rotolo. Logic of violations: A Gentzen system for reasoning with contrary-to-duty obligations. *Australasian Journal of Logic*, 4:193–215, 2006.

[8] Guido Governatori and Antonino Rotolo. A computational model for pragmatic oddity. In Michał Araszkiewicz and Víctor Rodríguez-Doncel, editors, *JURIX 2019: The 32th international conference on Legal Knowledge and Information Systems*, volume 332 of *Frontiers in Artificial Intelligence and Applications*, pages 187–192, Amsterdam, 2019. IOS Press.

[9] Xavier Parent and Leendert van der Torre. "sing and dance!". In Fabrizio Cariani, Davide Grossi, Joke Meheus, and Xavier Parent, editors, *Deontic Logic and Normative Systems*, pages 149–165, Cham, 2014. Springer International Publishing.

[10] Xavier Parent and Leendert van der Torre. The pragmatic oddity in norm-based deontic logics. In Guido Governatori and Jeroen Keppens, editors, *Proceedings of the 16th edition*

of the International Conference on Artificial Intelligence and Law, pages 169–178. ACM, 2017.

[11] Henry Prakken and Marek J. Sergot. Contrary-to-duty obligations. *Studia Logica*, 57(1):91–115, 1996.