Detecting Deontic Conflicts in Dynamic Settings

Silvano Colombo Tosatto[1,3], Guido Governatori[2], and Pierre Kelsen[1]

[1] University of Luxembourg
silvano.colombotosatto@uni.lu
pierre.kelsen@uni.lu
[2] NICTA and Queensland University of Technology
guido.governatori@nicta.com.au
[3] Universitá di Torino

Abstract. Regulations, through the use of obligations and permissions, are widely used in modern society to define acceptable behaviours. Thus it is indeed important that these regulations do not conflict with each other and contain contradicting obligations. In the present paper we focus on identifying conflicts between obligations in dynamic settings. We first show the need of an alternative semantics rather than the more classic modelled by standard deontic logic. Second we introduce a new semantics for the obligations capable of representing and reasoning about them in these dynamic settings, and lastly we use it to identify the necessary and sufficient conditions to identify conflicting obligations.

Keywords: Normative Reasoning, Conflicts, Time

1 Introduction

Nowadays, wherever we may go, there are always regulations influencing what we can and cannot do. The modern society makes a heavy use of regulations to define which are the desirable behaviours in almost any foreseeable scenario.

We argue that a clear understanding about how obligations interact is imperative to avoid situations where the obligations contradict each other, turning into dilemmas [16], where desirable behaviours are not discernible anymore.

Example 1. The "working week" defines that workers have to work from monday to friday. Islam defines that friday is an holy day and it is forbidden to work.

The example describes a conflicting situation resulting from merging different regulations, religious and business. The issue of conflicting regulations has been already studied in *normative reasoning*, like by Elhag et al. [6], Beirlaen and Straßer [2], and Sartor [21]. In particular, since regulations define what is obligatory, prohibited and permitted, *deontic logic* [14] and its variants have been extensively used to reason about them. For instance Hansen [12] studies the conflicts between obligations using *dyadic deontic logic*.

The deficiency of standard deontic logic to deal with conflicts has been already studied by Beirlaen et al. [3]. Whereas Beirlaen et al. focus on identifying conflicts between both permissions and obligations in single time instants, in this paper we study conflicts in a dynamic setting, consisting of scenarios evolving through time and we refer to them as *traces*. We show that standard deontic logic appears to be too restrictive while reasoning about normative conflicts in dynamic settings. Therefore we propose an alternative formalisation capable of reasoning about the obligations and detecting conflicts in these settings.

The paper is structured as follows: Section 2 introduces standard deontic logic. Section 3 introduces the traces. Section 4 introduces an alternative semantics to reason about obligations using the traces. Section 5 redefines the concept of deontic conflicts according to the alternative semantics. Section 6 describes how conflicts can be detected between the obligations using the alternative semantics. Sections 7 and 8 extend the alternative semantics introducing respectively preemptive and compensable obligations, and study how conflicts are detected given these additional semantics. Section 9 concludes the paper.

2 Standard Deontic Logic

Firstly introduced in 1951 by von Wright [26] as a system for reasoning about what is necessary or allowed, *Standard Deontic Logic* is one of the successors of this system.

The syntax of this logic is composed of an infinite set of propositional variables, the classical logical operators $(\neg, \wedge, \vee, \rightarrow)$ and two modal operators $\mathcal{O}$ and $\mathcal{P}$ used respectively to identify what is obligatory and what is permitted.

2.1 Consistency

Standard deontic logic is a normal *KD* logic where the axioms: $\mathcal{P}\top$ and $\mathcal{O}\alpha \rightarrow \neg\mathcal{O}\neg\alpha$, and the equivalence $\mathcal{O}\alpha \equiv \neg\mathcal{P}\neg\alpha$ hold. The equivalence expresses a relation between obligations and permissions, in other words it states that if something is obligatory, then the opposite is not permitted. The first axiom: $\mathcal{P}\top$, states that tautologies are always permitted and the second axiom: $\mathcal{O}\alpha \rightarrow \neg\mathcal{O}\neg\alpha$, states that if something is obligatory then its complement must not be obligatory.

We define internal consistency and external consistency using the two axioms and the equivalence. Internal consistency expresses the fact that something contradictory, like a proposition and its negation, cannot be obligatory.

Definition 1 (Internal Consistency). *A set of norms is internally consistent iff there is no formula α such that $\mathcal{O}(\alpha \wedge \neg\alpha)$ is entailed by the set of norms.*

Accordingly internal consistency corresponds to axiom $\mathcal{P}\top$:

$$\neg\mathcal{O}(\alpha \wedge \neg\alpha) \equiv \neg\mathcal{O}\bot \equiv \mathcal{P}\top$$

External consistency expresses that two contradictory obligations cannot coexist, like for instance the obligation of performing an action along with the prohibition of performing it.

Definition 2 (External Consistency). *A set of norms is externally consistent iff there are no formulae* α *such that* $\mathcal{O}\alpha \wedge \mathcal{O}\neg\alpha$ *is entailed by the set of norms.*

Accordingly internal consistency corresponds to axiom $\mathcal{O}\alpha \rightarrow \neg\mathcal{O}\neg\alpha$:

$$\neg(\mathcal{O}\alpha \wedge \mathcal{O}\neg\alpha) \equiv \mathcal{O}\alpha \rightarrow \neg\mathcal{O}\neg\alpha \equiv \mathcal{O}\alpha \rightarrow \mathcal{P}\alpha$$

In standard deontic logic the two axioms $\mathcal{P}\top$ and $\mathcal{O}\alpha \rightarrow \neg\mathcal{O}\neg\alpha$ are equivalent. The two consistency measures defined in the present section are used in standard deontic logic to identify inconsistencies. Although inconsistencies involving permissions are also possible, in this paper we focus on inconsistencies between obligations.

3 Traces: a Dynamic Setting

The information contained in single time instants is often not sufficient to decide whether a real obligation has been fulfilled or violated. This problem has been previously approached by Segerberg [22] using dynamic deontic logic.

Example 2. The authors of this paper must submit it to Δeon before the deadline, which is set on Sunday. This also means that the paper has to be finished before the submission deadline.

The scenario contained in Example 2 illustrates a situation comprising an obligation for which considering unique time instants to decide whether it is violated or not is often not sufficient. Because even when considering a time instant where the submission is executed, in order to evaluate the obligation we also need the information regarding whether the submission has been executed before or after the deadline.

To evaluate the obligation in Example 2 we need to consider a time interval. More precisely, we consider the time instants occurring between the event triggering the obligation (being an author of this paper) and the deadline terminating it (Sunday). Considering these time instants between the trigger and the deadline allows to evaluate the obligation by verifying if the paper is submitted in one of them.

3.1 Traces

Each time instant is associated to a state describing the world at that precise point in time. We use finite sets of literals to describe the situation holding in a point in time.

Definition 3 (Universe $\mathcal{L}$). *Given a finite set of atomic elements* E*, the* universe $\mathcal{L}$ *is* $E \cup \{\neg e \mid e \in E\}$*. For* $e \in E$*, let* $\overline{a} = \neg e$ *iff* $a = e$ *and* $\overline{a} = e$ *iff* $a = \neg e$.

Definition 4 (Consistent Set). *A set of literals* L *is* consistent *if and only if* $\forall l \in L, \neg\exists\neg l : \neg l \in L$.

Definition 5 (State). *Let $\mathcal{I} = (t_1, t_2, \dots)$ be a discrete linear order of instants of time and L a consistent finite set of literals. A* state *is a tuple $\sigma = (t_i, L)$.*

The sequence of states contained in a *trace* describes the evolution of the world during that time interval.

Definition 6 (Trace). *Given a potentially infinite discrete linear order $\mathcal{I} = (t_1, t_2, \dots)$, a trace θ is a sequence of states: $(\sigma_1, \dots, \sigma_n, \dots)$, such that for each $\sigma_i = (t_i, L_i)$ and $\sigma_j = (t_j, L_j)$, $\sigma_i \prec \sigma_j$ if and only if $t_i \prec t_j$ in $\mathcal{I}$.*

Example 3 (Trace). Considering again Example 2, a hypothetical trace on which is enforced the obligation of submitting the paper can be the following:

$$\theta = ((t_1, \{a\}), \dots, (t_i, \{s\}), \dots, (t_k, \{d\}), \dots)$$

The state $(t_1, \{a\})$ represents the trigger of the obligation, where the authors are acknowledged, The state $(t_i, \{s\})$ represents where submission of the paper is executed and finally $(t_k, \{d\}))$ when the deadline is reached.

The trace illustrated in Example 3 does not violate the obligation of submitting the paper before the deadline. A violating trace can be constructed from it by exchanging the state at time t_k with the one at time t_i.

4 Obligations' Semantics

In this paper we adopt a simpler semantics than Segerberg's [22] to describe and reason about the obligations. We use linear time models, avoiding branching time, and our semantics focuses on obligations leaving permissions out of the picture.

From Example 2 and as previously pointed out by Governatori et al. [7], an obligation requires a lifeline (a trigger), a deadline (determining the obligation termination) and a condition determining what is required from the obligation. In this paper we disregard the first two elements and adopt a more abstract approach by using a function that given a state of a trace, returns the set of obligations holding in that state.

Definition 7 (Obligation in force). *Given a state σ, we define a function*

$$Force : 2^{\mathcal{I}} \times 2^{\mathcal{L}} \mapsto 2^{\circledcirc}$$

where $\circledcirc$ is a set of obligations.

Definition 8 (Obligation). *An obligation is a structure $\langle t, c \rangle$, where $t \in \{s, a, m\}$ represents the type of the obligation. The element c is a propositional formula composed by elements in $\mathcal{L}$ and represents the content of the obligation.*

We use $\mathcal{O}^t\langle c \rangle$ to represent an obligation.

The content c of an obligation $\mathcal{O}^t\langle c\rangle$ is obligatory in the deontic sense. The formal semantics of how the content is obligatory depends on the type of the obligation t considered. We distinguish three types of obligations: standard ($\mathcal{O}^s$) which replicates the semantics of the obligations of standard deontic logic; achievement ($\mathcal{O}^a$), which captures the semantics of the obligations like the one in Example 2; and maintenance ($\mathcal{O}^m$), which captures the semantics of obligations similar to the one in the following example.

Example 4. To access secure data, the proper credentials must be retained for the whole access period.

Considering the trace illustrated in Example 3 and the obligation in Example 2, we can use s to represent the condition. The obligation, according to Definition 8, is represented as follows: $\mathcal{O}^a\langle s\rangle$.

4.1 Evaluating the Formulae

The states are not necessarily complete, meaning that given a proposition α, a state can either contain such proposition, its negation ($\neg\alpha$) or neither of them.

Definition 9 (Formula Entailment). *Let $\models$ be the standard propositional entailment. Given a state $\sigma = (t, L)$ and a formula α, $\sigma \models \alpha$ if and only if $\bigwedge x \models \alpha$, where each $x \in L$.*

4.2 Standard Obligations

Obligations in standard deontic logic are evaluated in a single state. We can mirror these obligations by forcing the instances of these obligations to be in force for exactly one state. We refer to the mirrored obligations as standard obligations. A standard obligation is represented as follows: $\mathcal{O}^s\langle c\rangle$.

Definition 10 (Comply with Standard). *Given a standard obligation $\mathcal{O}^s\langle c\rangle$ and a trace θ, θ is compliant with $\mathcal{O}^s$ if and only if: $\forall\sigma_i \in \theta$ such that $\mathcal{O}^s\langle c\rangle \in Force(\sigma_i), \sigma_i \models c$*

Consistency of Standard Obligations We expect that both internal and external consistency measures (Definitions 1 and 2) still apply to standard obligations.

Proposition 1. *$\neg\exists\theta|\theta$ complies with $\mathcal{O}^s\langle\alpha \wedge \neg\alpha\rangle$.*

Proof (Sketch). If we assume an obligation $\mathcal{O}(\alpha \wedge \neg\alpha)$ to be possible, then the translated standard obligation would be the following: $\mathcal{O}^s\langle\alpha \wedge \neg\alpha\rangle$.

From Definition 10 it follows that a trace must contain a state σ_i such that $\sigma_i \models \alpha \wedge \neg\alpha$ in order to comply with the standard obligation unless *Force* of each state of θ is empty (eg. $\forall\sigma \in \theta, Force(\sigma) = \emptyset$). However such state could not exist according to Definition 5 since each state must be consistent. Thus a standard obligation whose condition is a contradiction would never be complied by any trace.

Therefore internal consistency applies to standard obligations. □

Proposition 2. *$\neg\exists\theta|\theta$ complies with $\mathcal{O}^s\langle\alpha\rangle$ and θ complies with $\mathcal{O}^s\langle\neg\alpha\rangle$.*

Proof (Sketch). Assume a trace containing the state σ_i, where $\{\mathcal{O}^s\langle\alpha\rangle, \mathcal{O}^s\langle\neg\alpha\rangle\} \in$ *Force*(σ_i).

From Definition 2, $\mathcal{O}\alpha \wedge \mathcal{O}\neg\alpha$ is translated in standard obligations as follows: $\mathcal{O}^s\langle\alpha\rangle$ and $\mathcal{O}^s\langle\neg\alpha\rangle$ both belonging to the same set returned by applying *Force* the a given state σ_i of a given trace. According to Definition 10, since both standard obligations are in force in σ_i, then both conditions have to be verified in the same state.

A state σ_i, in order to fulfil both obligations, needs to contain in its state both α and $\neg\alpha$, however this is in contradiction with Definition 5, stating that a state has to be consistent. Thus it follows that a state σ_i satisfying both α and $\neg\alpha$ cannot exist.

Therefore a trace compliant with both standard obligations $\mathcal{O}^s\langle\alpha\rangle$ and $\mathcal{O}^s\langle\neg\alpha\rangle$ cannot exist. Thus no solution can exist when such pair of obligations is considered. □

4.3 Non-Standard Obligations

We define the semantics of the additional types of obligations in a similar way as already defined by Governatori et al. [7].

Achievement obligations require that at least a state included in their in force interval satisfies the condition.

Definition 11 (Comply with Achievement). *Given an achievement obligation $\mathcal{O}^a\langle c\rangle$ and a trace θ, θ is compliant with $\mathcal{O}^a\langle c\rangle$ if and only if:*

$\forall$ maximal subsequences $\theta_s \in \theta$ such that $\forall\sigma_i \in \theta_s, \mathcal{O}^a\langle c\rangle \in$ Force$(\sigma_i), \exists\sigma_h \in \theta_s$ such that $\sigma_h \models c$.

An operator with a similar semantics to the one just presented has been defined and analysed by Broersen et al. [5], the operator designed combines the semantics of computation tree logic and standard deontic logic.

Similarly, a maintenance obligation also requires to verify the condition when they are in force. However as we can see from Example 4, for each state where a maintenance obligation is in force, the state needs to satisfy the obligation's condition.

Definition 12 (Comply with Maintenance). *Given a maintenance obligation $\mathcal{O}^m\langle c\rangle$ and a trace θ, θ is compliant with $\mathcal{O}^m\langle c\rangle$ if and only if:*

$\forall\sigma_i \in \theta$ such that $\mathcal{O}^m\langle c\rangle \in$ Force$(\sigma_i), \sigma_i \models c$.

Relations with Standard Obligations Standard obligations are a particular case of both achievement and maintenance obligations. If we constrain the activation period of an achievement obligation to a single state, then such state must satisfy the condition. The same applies to maintenance obligations, if the activation period is limited to only one state, then such state has to fulfil the condition. Therefore if the activation is limited to a single state, then the semantics of both achievement and maintenance obligations collapse in the semantics of standard obligations.

5 Deontic Conflicts

We show here that the external consistency measure of standard deontic logic is too restrictive when used in a dynamic setting. The following example extends Example 2.

Example 5. The authors of this paper must submit it to Δeon before the deadline, which is set on Sunday. This also means that the paper has to be finished before the submission deadline. However, as usual on the weekends, the authors must go to the pub to meet their friends on Saturday or Sunday.

Example 5 contains two obligations: submitting the paper and going to the pub. We assume that the authors cannot finish and submit the paper while at the pub, hence we consider these obligations to be complementary. Thus if the proposition α represents "finishing and submitting the paper", then we can use $\neg\alpha$ to represent "going to the pub".

To formalise the example we discretise time in days. We use the propositions *sat* and *sun* to represent Saturday and Sunday respectively. Lastly we use the proposition *aut* to represent being an author of the present paper. We formalise the obligation of going to the pub: $\mathcal{O}^a\langle\neg\alpha\rangle$ and the obligation of submitting the paper as: $\mathcal{O}^a\langle\alpha\rangle$.

Both obligations are of type achievement. Despite the conditions of the obligations being complementary, it is still possible to provide a trace complying with both obligations.

$$\theta = (\ldots, (t_i, \{aut\}), \ldots, (t_j, \{sat, \neg\alpha\}), (t_k, \{sun, \alpha\}))$$

Assuming that $\mathcal{O}^a\langle\neg\alpha\rangle$ is in force in both $(t_j, \{sat, \neg\alpha\})$ and $(t_k, \{sun, \alpha\})$ and $\mathcal{O}^a\langle\alpha\rangle$ is in force from $(t_i, \{aut\})$ till the end of θ.

Example 5 describes a situation where two complementary obligations coexist during the weekend, but can be both fulfilled. According to the consistency measures provided by standard deontic logic, this situation would result in a conflict since it violates the external consistency measure. From the present analysis it follows that standard deontic logic is ill suited to reason about dynamic settings, more precisely the external consistency measure is too restrictive.

5.1 Redefining Conflicts

We now propose a new definition of inconsistent obligations, suited to be used in dynamic settings.

Definition 13 (Dynamic Conflict). *A set of obligations, written* $\circledcirc$*, is conflicting if and only if it is not possible to construct a trace in such a way that it is compliant with each obligation belonging to the set,* $\neg\exists\theta|\theta$ *compliant with* $\mathcal{O}, \forall\mathcal{O} \in \circledcirc$

The necessary conditions for two obligations to be conflicting is that their fulfilment conditions are complementary and their activation periods need to overlap. Depending on the type of obligations considered, the necessary condition may not be sufficient to determine whether they are conflicting. To focus on this aspect of the problem we introduce a function *Interval*, which when applied to an obligation and a trace returns the sub-intervals of the trace in which the obligation is activated.

Definition 14 (Interval). *Given a trace* θ*, let* θ_p *be the complete set of the sub-intervals of* θ*. Given an obligation* $\mathcal{O}$*, the partial function* Interval *is defined as follows:*

$$\mathit{Interval} : 2^{\mathcal{O}} \times 2^{\theta} \mapsto 2^{\theta_p} \text{ such that } \forall\varphi \in \mathit{Interval}(\mathcal{O}, \theta), \forall\sigma \in \varphi, \mathcal{O} \in \mathit{Force}(\sigma)$$

The function $Interval$ returns all the intervals of a trace in which an obligation is active. $Interval$ is defined as a partial function since it can be the case that an obligation is never activated in a trace, hence the set of intervals determining the activation period would be represented by the empty set.

5.2 Pair-wise Conflicts

In Definition 13, conflicts are defined for sets of obligations. The following example illustrates a case where a conflict arises from a set of obligations and, when any proper subset of the obligations is considered a conflict does not arise.

Example 6 (Conflicting Set). Assume a trace θ and a set of obligations composed of a single achievement obligation $\mathcal{O}^a\langle\alpha\rangle$ and k standard obligations $\mathcal{O}^s\langle\neg\alpha\rangle$ such that $\mathit{Interval}(\mathcal{O}^a\langle\alpha\rangle, \theta) \equiv \bigcup I \in \mathit{Interval}(\mathcal{O}^s\langle\neg\alpha\rangle, \theta)$ and $\bigcap I \in \mathit{Interval}(\mathcal{O}^s\langle\neg\alpha\rangle, \theta) = \emptyset$. In other words the activation periods of the standard obligations are all distinct and entirely cover the activation period of the achievement obligation.

From Example 6, we can see that a trace compliant with all the obligations belonging to the set proposed cannot exist because it would require a state containing both α and $\neg\alpha$.

The behaviour of the standard obligations in Example 6 can be simulated using a single maintenance obligation. The behaviour required from a trace to be compliant with the set of standard obligations ($\mathcal{O}^s\langle\neg\alpha\rangle$) is that in such trace $\neg\alpha$ holds for the interval determined by the obligations. The same result can be obtained by using a single maintenance obligation requiring $\neg\alpha$ to hold for the same interval. Thus the set of standard obligations can be substituted with a single maintenance obligation satisfying the following condition on the activation period:

$$\bigcup \varphi \in \mathit{Interval}(\mathcal{O}^s\langle\neg\alpha\rangle, \theta) \equiv \mathit{Interval}(\mathcal{O}^m\langle\neg\alpha\rangle, \theta)$$

Therefore we focus on analysing pair-wise conflicts between obligations.

6 Conflict Detection

The two necessary conditions to detect whether two obligations conflict are the following:

1. Their fulfilment conditions have to be complementary: $\mathcal{O}_1\langle\alpha\rangle$ and $\mathcal{O}_2\langle\beta\rangle$, such that $\alpha \wedge \beta \rightarrow \bot$.
2. The intersection of their activation periods must be not empty: $\exists x, y | x \in \mathit{Interval}(\mathcal{O}_1\langle\alpha\rangle, \theta), y \in \mathit{Interval}(\mathcal{O}_2\langle\beta\langle, \theta)$ and $x \cap y \neq \emptyset$.

We identify here the sufficient conditions to decide whether two obligations are conflicting. Being standard obligations a special case of both achievement and maintenance, it is sufficient to analyse the three combinations involving these types ($\mathcal{O}^m - \mathcal{O}^m$, $\mathcal{O}^m - \mathcal{O}^a$ and $\mathcal{O}^a - \mathcal{O}^a$). To do so we introduce two auxiliary functions, which applied to an interval or a trace, returns the first state belonging to them: *min*, or the last state: *max*.

6.1 Maintenance - Maintenance

We consider here two maintenance obligations.

Definition 15 ($\mathcal{O}^m - \mathcal{O}^m$ Conflict). *Let $\mathcal{O}^m\langle\alpha\rangle$ and $\mathcal{O}^m\langle\beta\rangle$ be two complementary maintenance obligations. $\mathcal{O}^m\langle\alpha\rangle$ and $\mathcal{O}^m\langle\beta\rangle$ are conflicting if and only if:*

$$\exists I \in \mathit{Interval}(\mathcal{O}^m\langle\alpha\rangle, \theta) \text{ and } \exists I' \in \mathit{Interval}(\mathcal{O}^m\langle\beta\rangle, \theta) : I \cap I' \neq \emptyset$$

Proposition 3 ($\mathcal{O}^m - \mathcal{O}^m$ Conflict). *Let $\mathcal{O}^m = \langle\alpha\rangle$ and $\mathcal{O}'^m = \langle\beta\rangle$ be conflicting maintenance obligations, then there does not exist a trace complying with both obligations.*

Proof ($\mathcal{O}^m - \mathcal{O}^m$ Conflict). We prove that the condition provided in Definition 15 is sufficient to identify whether two maintenance obligations are conflicting.

1. Let $\mathcal{O}^m = \langle\alpha\rangle$ and $\mathcal{O}'^m = \langle\beta\rangle$ be two complementary maintenance obligations, meaning that $\alpha \wedge \beta \rightarrow \bot$.
2. From the hypothesis we know that $\exists I, I'$ such that $I \in \mathit{Interval}(\mathcal{O}^m, \theta), I' \in \mathit{Interval}(\mathcal{O}'^m, \theta)$ and $\exists\sigma$ such that $\sigma \in I$ and $\sigma \in I'$.
3. From Definition 12 and 2. it follows that $\forall\sigma \in I, \sigma \models \alpha$ and $\forall\sigma' \in I'\sigma' \models \beta$.
4. Assume that there exists a trace θ such that θ is compliant with $\mathcal{O}^m$ and $\mathcal{O}'^m$.

5. From 4. it follows that $\forall I, I'$ such that $I \in \mathit{Interval}(\mathcal{O}^m, \theta)$ and $I' \in \mathit{Interval}(\mathcal{O}'^m, \theta)$, $I \subseteq \theta$ and $I' \subseteq \theta$ and $\forall \sigma \in I, \sigma \models \alpha$ and $\forall \sigma' \in I', \sigma' \models \beta$.
6. From 2. and 5. it follows that $\exists \sigma \theta$ such that $\sigma \models \alpha$ and $\sigma \models \beta$.
7. From Definition 9 and 6. it follows that $\exists \sigma \in \theta$ such that $\{\alpha, \beta\} \in \sigma$.
8. From 1. we know that $\alpha \wedge \beta \rightarrow \bot$, hence from 7. and Definition 5 it follows that a state σ is inconsistent and a trace containing such state cannot exists.

Therefore we have proven that the condition provided in Proposition 3 is sufficient to identify to conflicting complementary maintenance obligations. □

Two maintenance obligations are conflicting as soon as they are complementary and their activation periods overlap. In this case the sufficient condition is also the necessary condition previously introduced.

We do not provide propositions and formal proofs for the following definitions since they are analogous of the one provided for Definition 15.

6.2 Maintenance - Achievement

We consider here a maintenance and an achievement obligation.

Definition 16 ($\mathcal{O}^m - \mathcal{O}^a$ Conflict). *Let $\mathcal{O}^m\langle\alpha\rangle$ be a maintenance obligation and $\mathcal{O}^a\langle\beta\rangle$ be a complementary achievement obligation. $\mathcal{O}^m\langle\alpha\rangle$ and $\mathcal{O}^a\langle\beta\rangle$ are conflicting if and only if:*

$$\exists I \in \mathit{Interval}(\mathcal{O}^a\langle\beta\rangle, \theta) \text{ and } \exists I' \in \mathit{Interval}(\mathcal{O}^m\langle\alpha\rangle, \theta) : I \subseteq I'$$

The sufficient condition captures the fact that an achievement obligation requires be fulfilled in a single state, hence a conflict arise only if the activation period of the maintenance obligation is a superset of the activation period of the achievement obligation.

6.3 Achievement - Achievement

We consider here two achievement obligations.

Definition 17 ($\mathcal{O}^a - \mathcal{O}^a$ Conflict). *Let $\mathcal{O}^a\langle\alpha\rangle$ and $\mathcal{O}^a\langle\beta\rangle$ be two conflicting achievement obligations. $\mathcal{O}^a\langle\alpha\rangle$ and $\mathcal{O}^a\langle\beta\rangle$ are conflicting if and only if:*

$$\exists I \in \mathit{Interval}(\mathcal{O}^a\langle\alpha\rangle, \theta) : I \in \mathit{Interval}(\mathcal{O}^a\langle\beta\rangle, \theta) \text{ and } ||I|| = 1$$

The sufficient condition requires that there exists an activation period common to the two complementary achievement obligations and that such activation period is of length one. These restrictive conditions are necessary due to the flexibility allowed to comply with achievement obligations. Two achievement obligations are actually conflicting if and only if both behave as standard obligations in at least a shared activation period.

The sufficient condition required to identify conflicting standard obligations (Definition 10) is the following:

$$\exists I \in \mathit{Interval}(\mathcal{O}^s\langle\alpha\rangle, \theta) : I \in \mathit{Interval}(\mathcal{O}^s\langle\beta\rangle, \theta)$$

As it is expected, this sufficient condition is a particular case of all the other conditions identified in the present section.

7 Preemptive Obligations

Achievement and maintenance are capable of representing a good deal of obligations used in real world scenarios. However, there are still some which could be not translated using their semantics.

Example 7. Anti-Money Laundering and Counter-Terrorism Financing Act 2006. Clause 54 (Timing of reports about physical currency movements).

1. A report under Section 53 must be given:
   (a) if the movement of the physical currency is to be effected by a person bringing the physical currency into Australia with the person-at the time worked out under subsection (2); or
   ...
   (d) in any other case-at any time before the movement of the physical currency takes place.

Example 7 illustrates an Australian regulation aimed at monitoring physical currency movements. The obligation states that a report must be provided when transaction occurs, however clause **(d)** states that this report can be provided before the transaction takes place. This obligation is still an achievement obligation, however due to clause **(d)**, this obligation can be preemptively achieved as it has been defined by Governatori and Rotolo [9].

We introduce a sub-type of achievement obligations called *preemptive achievement obligation*, denoted $\mathcal{O}^{-a}$, which allow to be fulfilled in states preceding their triggering state.

Definition 18 (Comply with Preemptive Achievement). *Given a preemptive achievement obligation $\mathcal{O}^{-a}\langle c\rangle$ and a trace θ, θ is compliant with $\mathcal{O}^{-a}$ if and only if:*

$\forall$ maximal subsequences $\theta_s \in \theta$ such that $\forall \sigma_i \in \theta_s, \mathcal{O}^{-a}\langle c\rangle \in Force(\sigma_i), \exists \sigma_h \in \theta_s$ and $\exists \sigma_j \in \theta$ such that $\sigma_j \models c$ and $\sigma_j \preceq \sigma_h$.

7.1 Conflicts for Preemptive Achievement Obligations

As it has been done previously for the two main types of obligations, we define the sufficient conditions to identify conflicts involving a preemptive achievement obligation.

Maintenance - Preemptive Achievement We now consider a maintenance and a preemptive achievement obligation.

Definition 19 ($\mathcal{O}^m - \mathcal{O}^{-a}$ Conflict). *Let $\mathcal{O}^m\langle\alpha\rangle$ be a maintenance obligation and $\mathcal{O}^{-a}\langle\beta\rangle$ be a complementary preemptive achievement obligation. $\mathcal{O}^m\langle\alpha\rangle$ and $\mathcal{O}^{-a}\langle\beta\rangle$ are conflicting if and only if:*

$$\exists I \in Interval(\mathcal{O}^{-a}\langle\beta\rangle, \theta), \exists I' \in Interval(\mathcal{O}^m\langle\alpha\rangle, \theta) \colon I \subseteq I' \; and \, min(I') = min(\theta)$$

The sufficient condition is an extension of Definition 16, to which has been added the additional condition, requiring that the activation period of the maintenance obligation contains the first state of the trace. The stricter sufficient condition follows from the less strict fulfilment condition for preemptive achievement obligations (Definition 18) with respect to achievement obligations (Definition 11).

Preemptive Achievement - Achievement We now consider a preemptive achievement and an achievement obligation.

Definition 20 ($\mathcal{O}^a - \mathcal{O}^{-a}$ Conflict). *Let $\mathcal{O}^a\langle\alpha\rangle$ be an achievement obligation and $\mathcal{O}^{-a}\langle\beta\rangle$ be a preemptive achievement obligation. $\mathcal{O}^a\langle\alpha\rangle$ and $\mathcal{O}^{-a}\langle\beta\rangle$ are conflicting if and only if:*

$$\exists I \in Interval(\mathcal{O}^{-a}\langle\beta\rangle, \theta) \colon I \in Interval(\mathcal{O}^a\langle\alpha\rangle, \theta), ||I|| = 1 \text{ and } min(I) = min(\theta)$$

The sufficient condition is an extension of Definition 17. The additional constraint: "and $min(I) = min(\theta)$" requires that an activation period for both obligation to include the first sate of the trace and be of length 1.

Preemptive Achievement - Preemptive Achievement The sufficient condition to identify whether two preemptive achievement obligations are conflicting is the same as the one identified between an achievement and a preemptive achievement obligation in Definition 20.

8 Compensable Obligations

In complex systems, the possibility that regulations may not be followed has to be taken into account. Lomuscio and Sergot [17] studied this in the context of multi-agent systems. Compensable obligations define in addition to their obligation, which we call primary from now on, also what needs to be done when they are violated through secondary obligations as defined by Governatori and Rotolo [11]. Secondary obligations are a particular type of obligation whose activation depends on the violations of the primary obligation they try to compensate.

Definition 21 (Activation). *An activation of an obligation $\mathcal{O}^t\langle c\rangle$ in a trace θ consists of a maximal subsequence θ_s of θ where $\forall \sigma_i \in \theta_s, \mathcal{O}^t\langle c\rangle \in Force(\sigma_i)$.*

A violation can raise for each activation in which a primary obligation is not complied with. This means that if there is no state satisfying the condition, then an achievement obligation (for both types of obligations, Definitions 11 and 18) is raised in the last state belonging to the activation. For maintenance obligations (Definition 12), if exists a state in the activation which does not satisfy the condition, then a violation is raised in the earliest[4] state of the activation which does not satisfy the condition.

[4] We consider the earliest to be the one not satisfying the condition and not preceded by any other which does not satisfy the condition.

Definition 22 (Violations). *Given an activation θ_s of an obligation $\mathcal{O}^t\langle c\rangle$, a violation v of $\mathcal{O}^t\langle c\rangle$ is identified a function $V(\theta, \mathcal{O}^t\langle c\rangle)$ which identifies the earliest state of θ_s where $\mathcal{O}^t\langle c\rangle$ is not complied with.*

Compensable obligations are composed of two separate components: a primary obligation which describe the obligation which has to be complied with for each activation of the compensable obligation, and a secondary obligation that needs to be complied with for each violation of the primary obligation.

Definition 23 (Compensable Obligation). *A compensable obligation, written $\mathbf{O} = \mathcal{O} \otimes \mathcal{O}_c$, is composed of a primary obligation $\mathcal{O}$ and a compensation $\mathcal{O}_c$.*

The relations between the activation periods of $\mathcal{O}$ and $\mathcal{O}_c$ are the following: $\forall I \in Interval(\mathcal{O}_c, \theta), \exists v \in V(\mathcal{O}, \theta) : min(I) = v$. Moreover $\forall v \in V(\mathcal{O}, \theta), \exists I \in Interval(\mathcal{O}, \theta) : max(I) = v$.

The compensation $\mathcal{O}_c$ can be as well a compensable obligation.

Compensable obligations can be seen as sequences of obligations connected by the operator $\otimes$.

Definition 24 (Comply with Compensable Obligations). *Given a trace θ and a compensable obligation $\mathbf{O} = \mathcal{O} \otimes \mathcal{O}_c$. θ is compliant with $\mathbf{O}$ if and only if θ is compliant with $\mathcal{O}_c$.*

A trace is compliant with a compensable obligation if it is compliant with its secondary obligation. This follows from Definitions 11, 12 and 18, where a trace θ is always considered to be compliant with an obligation $\mathcal{O}$ if $Interval(\mathcal{O}, \theta) = \emptyset$. This means in this case that either the primary obligation is not violated or if it is violated, then each violation has been compensated.

Example 8. An example of compensable obligations is the following: When you dine at a restaurant you have to pay for your meal. If you don't, then you have to wash the dishes.

This compensable obligation can be formalised as follows: $\mathcal{O}^a\langle\alpha\rangle \otimes \mathcal{O}^a\langle\beta\rangle$, where α represents "paying the bill" and β represents "washing the dishes".

8.1 Conflicts for Compensable Obligations

We define now the sufficient conditions to identify pair-wise conflicts involving compensable obligations. A compensable obligation is not a new type of obligation, but rather a way of structuring the existing types of obligations. A non compensable obligation is a special case of compensable which compensable obligation cannot be fulfilled if triggered. Therefore we analyse the more general case of deciding which are the sufficient conditions to determine whether two compensable obligations are conflicting.

Definition 25 ($\boldsymbol{\odot}$ - $\boldsymbol{\odot}$ Conflict). *Let* $\boldsymbol{\odot} = \mathcal{O} \otimes \mathcal{O}_c$ *and* $\boldsymbol{\odot}' = \mathcal{O}' \otimes \mathcal{O}'_c$ *be two compensable obligations.* $\boldsymbol{\odot}$ *and* $\boldsymbol{\odot}'$ *are conflicting if and only if:*

$$\mathcal{O}_c \textit{ is conflicting with } \mathcal{O}'_c$$

To determine whether two obligation "conflict" we reuse the sufficient conditions from Definitions 15, 16, 17, 19 and 20. The sufficient condition expressed in Definition 25 requires that the compensations of the two compensable obligations are conflicting. A compensation $\mathcal{O}_c$ is triggered by a violation of the primary obligation $\mathcal{O}$, hence $||Interval(\mathcal{O}_c, \theta)|| = ||V(\mathcal{O}, \theta)||$. If the two secondary obligation are conflicting, it means that both $V(\mathcal{O}, \theta)$ and $V(\mathcal{O}', \theta)$ are not empty due existing conflicts between $\mathcal{O}$ and an obligation in $\boldsymbol{\odot}'$ and vice versa.

9 Conclusion

In the present paper we show that standard deontic logic is not well suited to reason about conflicting obligations in a dynamic setting. Therefore we first provide an alternative semantics more suited to reason about obligations in such a setting and second we show how the newly defined semantics can be used to detect conflicting obligations in this type of dynamic setting. The sufficient and necessary condition for identifying conflicting obligations can be used as constraints while designing regulations for normative systems, in order to avoid systems where any behaviour would violate the regulations proposed.

The conditions identified in the present paper can be also used to detect conflicts in existing systems. Resolving these conflicts is also an important part as has been shown by Prakken and Sartor [20]. Another work relative to conflict detection and resolution by Vasconcelos et al. [25], proposes a similar approach as the one in this paper by considering overlapping periods of the obligations for conflict detection. The work of Vasconcelos et al. includes also conflict resolution techniques to solve the conflicts detected, however in the present paper we focus on explicitly identifying and highlighting the sufficient and necessary conditions to detect conflicts between different types of obligations, which we are capable of achieving using a simpler semantics for the obligations. Additionally we claim that a further utility of the conditions identified in the present paper is that they can be used as constraints to design *conflict free* normative systems.

The most closely related work is [10] presenting a temporal version of deontic defeasible logic equipped with deontic operators corresponding to all classes of obligations discussed in the paper (excluding preemptive obligations) and supplemented with an operator for compensatory obligations [11]. The conflicts are not explicitly given but are embedded in the various proof conditions.

Another important element in normative reasoning is constituted by permissions, which, as described by Boella and van der Torre [4], and Makinson and van der Torre [18], can be used as a mean to limit the applicability of obligations and prohibitions, as already has been studied by Stolpe [23] where the semantics is defined using AGM belief revision [1], Input/Output logic [19] and Defeasible Logic [8]. Conflict detection involving permission has already been studied by Hansen [13], however we plan to study it in a dynamic setting as future work.

Acknowledgements

NICTA is funded by the Australian Government as represented by the Department of Broadband, Communications and the Digital Economy and the Australian Research Council through the ICT Centre of Excellence program.

Silvano Colombo Tosatto is supported by the National Research Fund, Luxembourg.

References

1. Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *J. Symb. Log.*, 50(2):510–530, 1985.
2. Mathieu Beirlaen and Christian Straßer. A paraconsistent multi-agent framework for dealing with normative conflicts. In Leite et al. [15], pages 312–329.
3. Mathieu Beirlaen, Christian Straßer, and Joke Meheus. An inconsistency-adaptive deontic logic for normative conflicts. *J. Philosophical Logic*, 42(2):285–315, 2013.
4. Guido Boella and Leendert W. N. van der Torre. Permissions and obligations in hierarchical normative systems. In *ICAIL*, pages 109–118, 2003.
5. Jan Broersen, Frank Dignum, Virginia Dignum, and John-JulesCh. Meyer. Designing a deontic logic of deadlines. In Alessio Lomuscio and Donald Nute, editors, *Deontic Logic in Computer Science*, volume 3065 of *Lecture Notes in Computer Science*, pages 43–56. Springer Berlin Heidelberg, 2004.
6. Abdullatif A.O. Elhag, Joost A. Breuker, and Bob W. Brouwer. On the formal analysis of normative conflicts. In H.Jaap van den Herik et al., editor, *JURIX 1999: The Twelfth Annual Conference*, Frontiers in Artificial Intelligence and Applications, pages 35–46, Nijmegen, 1999. GNI.
7. Guido Governatori, Joris Hulstijn, Régis Riveret, and Antonino Rotolo. Characterising deadlines in temporal modal defeasible logic. In Mehmet A. Orgun and John Thornton, editors, *Australian Conference on Artificial Intelligence*, volume 4830 of *Lecture Notes in Computer Science*, pages 486–496. Springer, 2007.
8. 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.
9. Guido Governatori and Antonino Rotolo. A conceptually rich model of business process compliance. In Sebastian Link and Aditya Ghose, editors, *7th Asia-Pacific Conference on Conceptual Modelling (APCCM 2010)*, volume 110 of *CRPIT*, pages 3–12. ACS, January 18-21 2010.
10. Guido Governatori and Antonino Rotolo. Justice delayed is justice denied: Logics for a temporal account of reparations and legal compliance. In Leite et al. [15], pages 364–382.
11. Guido Governatori and Antonio Rotolo. Logic of violations: A gentzen system for reasoning with contrary-to-duty obligations. *The Australasian Journal of Logic*, 4:193–215, 2006.
12. Jörg Hansen. Conflicting imperatives and dyadic deontic logic. In A. Lomuscio and D. Nute, editors, *Proceedings of the 7th International Workshop on Deontic Logic in Computer Science (DEON 2004), Madeira, Portugal, May 26-28, 2004*, number 3065/2004 in LNCS, pages 146–164. Springer, 2004.
13. Jorg Hansen. *Outstanding Contributions to Logic*, chapter Reasoning About Permission and Obligation. In Sven Ove Hansson and Wansing [24], 2014.

14. Risto Hilpinen and Paul McNamara. Deontic logic: a historical survey and introduction. In Dov Gabbay, John Horty, Ron van der Meyden, Xavier Parent, and Leendert van der Torre, editors, *Handbook of Deontic Logic and Normative Systems*. College Publications, 2013.
15. João Leite, Paolo Torroni, Thomas Ågotnes, Guido Boella, and Leon van der Torre, editors. *Computational Logic in Multi-Agent Systems - 12th International Workshop, CLIMA XII, Barcelona, Spain, July 17-18, 2011. Proceedings*, volume 6814 of *Lecture Notes in Computer Science*. Springer, 2011.
16. E. J. Lemmon. Moral dilemmas. *The Philosophical Review*, 71(2):139–158, April 1962.
17. Alessio Lomuscio and Marek Sergot. Violation, error recovery, and enforcement in the bit transmission problem. In John Horty and Andrew J. I. Jones, editors, *Proceedings of the 6th International Workshop on Deontic Logic in Computer Science (DEON 2002), London, England, May 22-24, 2002*, pages 181–202. Imperial College London, Informal Proceedings, 2002.
18. David Makinson and Leendert van der Torre. Permission from an input/output perspective. *J. Philosophical Logic*, 32(4):391–416, 2003.
19. David Makinson and Leendert W. N. van der Torre. What is input/output logic? input/output logic, constraints, permissions. In Guido Boella, Leendert W. N. van der Torre, and Harko Verhagen, editors, *Normative Multi-agent Systems*, volume 07122 of *Dagstuhl Seminar Proceedings*. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, 2007.
20. Henry Prakken and Giovanni Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artif. Intell. Law*, 4(3-4):331–368, 1996.
21. Giovanni Sartor. Normative conflicts in legal reasoning. *Artificial Intelligence and Law*, 1(2-3):209–235, 1992.
22. Krister Segerberg. DΔL : a dynamic deontic logic. *Synthese*, 185:1–17, 2012.
23. Audun Stolpe. *Outstanding Contributions to Logic*, chapter Abstract Interfaces of Input/Output Logic. In Sven Ove Hansson and Wansing [24], 2014.
24. Lawrence Moss Sonja Smets Sven Ove Hansson, Marcus Kracht and Heinrich Wansing, editors. *Outstanding Contributions to Logic*. Springer Netherlands, Dordrecht, 2014.
25. WambertoW. Vasconcelos, MartinJ. Kollingbaum, and TimothyJ. Norman. Normative conflict resolution in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 19(2):124–152, 2009.
26. Georg Henrik Von Wright. Deontic logic. *Mind*, 60(237):1–15, 1951.