Towards More Conflict-Tolerant Deontic Logics By Relaxing the Interdefinability Between Obligations And Permissions*

Christian Straßer and Mathieu Beirlaen

October 4, 2010

Abstract

While conflict-tolerant logics (CTDLs) usually allow for obligation-obligation conflicts, they fall short of tolerating obligation-permission conflicts (OP-conflicts) of the type $\mathsf{O}A \wedge \mathsf{P}\neg A$. Moreover, for the sake of conflict-tolerance these logics usually do not validate the very intuitive principle (D), $\mathsf{O}A \supset \mathsf{P}A$.

We demonstrate in this paper that by relaxing the interdefinability between obligations and permission $\mathsf{P}A =_{\mathrm{df}} \neg\mathsf{O}\neg A$ that is characteristic for most deontic logics, the logics get more conflict-tolerant since they allow for OP-conflicts. Moreover, this way they can be equipped with (D) without the need to sacrifice conflict-tolerance. In this paper we offer a generic procedure that transforms a given CTDL into a logic that tolerates OP-conflicts and validates (D).

Key Words: deontic logic, interdefinability, conflict-tolerance, deontic conflicts, deontic dilemmas

1 Introduction

Deontic logics intend to formalize reasoning with and about norms. In order to do so they employ logical operators such as O where $\mathsf{O}A$ expresses the obligation to bring about A, or P where $\mathsf{P}A$ expresses the permission to bring about A. Similarly conditional deontic logics employ dyadic operators $\mathsf{O}(A \mid B)$ expressing that in the context B the obligation to bring about A is in force (and analogously for permissions). We will focus in this paper on unary normative operators since this simplifies the technical level of the discussion. However, our arguments and results apply in a straightforward way also to the dyadic case.

The last decades of research in deontic logics have been characterized by a strongly increasing awareness of the existence of deontic conflicts ([9, 18, 11])

*Research for this paper was supported by the Research Fund of Ghent University by means of Research Projects 01G01907 and G011409.

and the importance of developing formal frameworks that are able to deal with them ([8, 15, 4, 10, 14]). Deontic conflicts occur whenever norms offer conflicting directives. For instance, one norm may oblige you to bring about A while, (i), another one may oblige you to bring about not-A, or, (ii), another one may permit you to bring about not-A.

The challenge is to develop conflict-tolerant deontic logics (henceforth, CTDLs), i.e., logics that do not validate forms of deontic explosion when being confronted with premise sets that feature deontic conflicts. The most commonly considered form of deontic explosion in the literature occurs if from a given set of premises it is derivable that anything is obliged. There are basically two approaches to deal formally with deontic conflicts, one that stays within the classical logic framework and another one employing paraconsistent logics. This paper is mainly situated in the former, classical enterprise. We will discuss the paraconsistent case shortly in Section 9.

It is remarkable that scholars have been focusing mainly on only one type of deontic conflicts, namely conflicts between obligations such as for instance between $\mathsf{O}A$ and $\mathsf{O}\neg A$. However, another type of conflict has been entirely neglected, namely conflicts between obligations and permissions such as for instance between $\mathsf{O}A$ and $\mathsf{P}\neg A$. In this respect the numerous proposed CTDLs are not "fully" conflict tolerant. This creates an unnecessary and moreover unmotivated asymmetry in the modeling offered by these logics.

This is a severe defect of CTDLs. However, it is not a fatal one. We will propose in this paper a transformation procedure that turns a given CDTL $\mathbf{L}$ into a CTDL $\mathbf{L}\star$ that is also conflict-tolerant concerning obligation-permission conflicts. The mainspring of our transformation is to give up the interdefinability between obligations and permission that is characteristic for most deontic logics (henceforth, IP for *interdefinability principle*). Usually, either $\mathsf{P}A$ is defined by $\neg\mathsf{O}\neg A$ or there is an axiom which enforces the equivalence. In the transformed logic $\mathbf{L}\star$ only one direction of the equivalence holds.

The logics gained by our transformation procedure offer many advantages. First and foremost, acknowledging the so far neglected conflict type does not necessitate the development of entirely new deontic systems, but the already proposed systems can be transformed in a way that preserves their strengths and modeling features while making them at the same time more conflict-tolerant.

Giving up IP weakens the transformed logic compared to $\mathbf{L}$. In the first instance this sounds like a shortcoming. However, this makes it possible to add a very intuitive principle that is usually not validated by CTDLs, namely axiom (D) that allows to derive from the obligation to bring about A, the permission to bring about A. Given the intuitive appeal of (D) we will argue that this is indeed a very good compensation for losing one direction of IP.

The paper is structured as follows. In Section 2 we will introduce two types of deontic conflicts, namely conflicts between obligations (OO-conflicts) on the one hand and conflicts between obligations and permissions (OP-conflicts) on the other hand. In Standard Deontic Logic, both types of conflicts lead to the problem of deontic explosion which is discussed in Section 3. In Section 4 we will show that there are certain shortcomings to classical CTDLs due to the fact that they validate IP. In Section 5 we will present a generic procedure to transform a given CTDL into a system that is conflict tolerant with respect to both OO- and OP-conflicts. Interesting meta-theoretic properties of this translation are presented in Section 6. In Sections 7 and 8 we will illustrate the procedure for some existing CTDLs.

In order to improve the readability of the paper we collected the sometimes technical proofs of the meta-theoretic results in the Appendix [19].

2 Deontic conflicts

Deontic conflicts occur whenever various norms offer incompatible directives. Often there are resolutions to such situations. For instance, if in a certain situation we have the obligation to bring about A as well as the obligation to bring about not-A, but the former obligation has higher preference (e.g. it may have been issued by a higher authority than the latter one), then the conflict can be resolved by obeying $\mathsf{O}A$ while neglecting $\mathsf{O}\neg A$. All-things-considered the former obligation is our proper, actual obligation which guides our actions. In cases in which there is (at present) no normative procedure that offers a directive in order to resolve the conflicting norms, authors often speak of deontic dilemmas. It is a common fact of life that such situations occur. Whether or not all of them are in the end resolvable, is a question that we believe to be independent of the practical need for being able to reason in their presence.

Deontic conflicts are usually taken to be constituted by obligations that cannot be mutually realized. Where $\mathsf{O}A$ denotes the obligation to bring about A, a deontic conflict is for instance given by $\mathsf{O}A$ and $\mathsf{O}\neg A$, or by $\mathsf{O}A, \mathsf{O}B$ and $\mathsf{O}\neg(A \wedge B)$. In the remainder, we refer to this type of conflicts as *obligation-obligation conflicts*, or simply *OO-conflicts*.

As an example of what we mean by an OO-conflict, consider people who think that unpreventable pain ought not to be tolerated and that human life ought not to be deliberately shortened. These people will face a deontic conflict in deciding whether or not to withdraw life support from a dying patient ([12]).

Deontic conflicts are also widely present in legal contexts. Consider, for example, the case of SWIFT, a Belgium-based company with offices in the United States which operates a worldwide messaging system used to transmit, inter alia, bank transaction information. According to the U.S. Treasury, information derived from the use of SWIFT data has enhanced the United States' and third countries' ability to identify financiers of terrorism, to map terrorist networks and to disrupt the activities of terrorists and their supporters. However, in September 2006 the Belgian Data Protection Authority stated that SWIFT processing activities for the execution of interbank payments are in breach of Belgian data protection law. American diplomats and politicians claim that SWIFT ought to continue passing information to the U.S. Treasury, whereas according to Belgian law SWIFT ought not to pass this information, since this activity is in breach of Belgian data protection law.

Next to OO-conflicts, we can distinguish a second, much neglected type of conflicts: conflicts between one or more obligation(s) and one or more permission(s). We dub these conflicts *obligation-permission conflicts*, or shortly *OP-conflicts*. The most simple forms of an OP-conflict are given by $\mathsf{O}A \wedge \mathsf{P}\neg A$ (where $\mathsf{P}A$ denotes the permission to bring about A) and $\mathsf{O}\neg A \wedge \mathsf{P}A$. Suppose, for instance, that you are walking along a wild river, and that you see someone drowning in it. Since you are a good swimmer, you feel morally obliged to save this person ($\mathsf{O}S$). But since the river is wild, there is a chance that you will also drown in trying to rescue this person. This risk factor permits you to not save the drowning person ($\mathsf{P}\neg S$).

As another example, consider the case of Yilmaz, who ought not to drink alcohol according to his religious beliefs ($\mathsf{O}\neg A$). However, according to the laws of his country, he is permitted to drink alcohol ($\mathsf{P}A$).

Both OO-conflicts and OP-conflicts can be seen as *morally over-determined* situations in the sense that they impose contradicting normative directives on the moral agent. For instance in an OO-conflict $\mathsf{O}A \wedge \mathsf{O}\neg A$ we are determined to break either the obligation to bring about A or to break the obligation to bring about $\neg A$. It is exactly this over-determination that causes the conflict. The main difference between OO- and OP-conflicts is that the latter are weaker in the sense that one does not *need* to break an obligation in order to fulfill all moral considerations at hand. Obviously deciding to bring about A facing the OP-conflict $\mathsf{O}A \wedge \mathsf{P}\neg A$ is a choice in which no normative directive is violated (presupposing it conforms with other given norms). Hence, OP-conflicts generally have a less dilemmatic character than OO-conflicts.

Considering the fact that a normative authority that issues a norm expressing an obligation $\mathsf{O}A$ typically is in support of the permission $\mathsf{P}A$, OO-conflicts typically give rise to OP-conflicts.[1] Indeed, given the OO-conflict, $\mathsf{O}A \wedge \mathsf{O}\neg A$, and the fact that $\mathsf{O}A$ entails $\mathsf{P}A$ (resp. $\mathsf{O}\neg A$ entails $\mathsf{P}\neg A$), we end up with the OP-conflicts $\mathsf{O}\neg A \wedge \mathsf{P}A$ and $\mathsf{O}A \wedge \mathsf{P}\neg A$.

OO- and OP-conflicts as defined above are not the only types of deontic conflicts. Conflicts of the type $\mathsf{O}A \wedge \neg\mathsf{O}A$ or $\mathsf{P}A \wedge \neg\mathsf{P}A$ have been neglected in our discussion. In order to handle them, paraconsistent logics are necessary, since logics that employ the classical negation validate the ex contradictione quodlibet' principle, i.e. $A \wedge \neg A \vdash B$. Since the goal of this paper is to stay as conflict-tolerant as possible within a classical framework, we do not deal with such conflicts. We will, however, shortly discuss the paraconsistent case in Section 9.

3 The problem of deontic explosion

Both OO- and OP-conflicts pose a problem for the deontic logician due to the problem of deontic explosion. We can demonstrate this by taking a look at the very well-known system Standard Deontic Logic (henceforth **SDL**). We formulate **SDL** in the language $\mathcal{L}$ that features two unary deontic operators, O for obligations and P for permissions, as well as all the classical connectives. We

[1] We argue more in favor of this in Section 4.1.

write $\mathcal{W}$ for the set of all well-formed formulas in $\mathcal{L}$. **SDL** is for instance defined by enriching classical propositional logic (henceforth **CL**) by the following axioms and rules:

$$\mathsf{O}A \supset \mathsf{P}A \tag{D}$$

$$\neg\mathsf{O}\bot \tag{N}$$

$$\mathsf{O}\top \tag{P}$$

$$\big(\mathsf{O}A \wedge \mathsf{O}B\big) \supset \mathsf{O}(A \wedge B) \tag{AND}$$

$$\text{If } \vdash A \supset B, \text{ then } \vdash \mathsf{O}A \supset \mathsf{O}B. \tag{RM}$$

$$\mathsf{P}A \equiv \neg\mathsf{O}\neg A \tag{DfP}$$

Instead of using the richer language $\mathcal{L}$ that also features the permission operator P, **SDL** may also be formulated in the weaker language $\mathcal{L}'$ that only features the obligation operator O in addition to the classical symbols. Instead of the axiom (DfP), the permission operator is then defined by $\mathsf{P}A =_{\mathrm{df}} \neg\mathsf{O}\neg A$. The presented axiomatization is only one of many alternative equivalent axiomatizations of **SDL**. We chose it since it allows us to introduce certain rules and axioms that will play a rule in the further discussion.

Note that in **CL** and hence in **SDL** also the following 'ex contradictione quodlibet' principle holds:

$$(A \wedge \neg A) \supset B \tag{ECQ}$$

(AND) is often called the aggregation principle, (RM) the inheritance principle. In the presence of a conflict $\mathsf{O}A \wedge \mathsf{O}\neg A$, we can derive $\mathsf{O}(A \wedge \neg A)$ by (AND), and by (ECQ) and (RM) it follows that $\mathsf{O}B$. Hence, in **SDL**, the presence of a deontic conflict $\mathsf{O}A \wedge \mathsf{O}\neg A$ allows us to derive that, for any proposition B, it holds that $\mathsf{O}B$:

$$\big(\mathsf{O}A \wedge \mathsf{O}\neg A\big) \supset \mathsf{O}B \tag{OO-EX}$$

(OO-EX) states that triviality ensues whenever a premise set contains an OO-conflict. This phenomenon is usually called *deontic explosion*.

Scholars have developed various strategies in order to invalidate deontic explosion. Systems of deontic logic that do not validate (OO-EX) will be called *conflict-tolerant deontic logics* (CTDLs). Since it is clear from our discussion above that each CTDL must either reject or restrict at least one of (AND), (RM) and (ECQ), we can distinguish between three main approaches. The first approach consists of restricting or rejecting (AND), the second approach consists of restricting or rejecting (RM), and the third approach consists of restricting or rejecting (ECQ).[2] The first and second approach are *classical* in the sense that they validate all theorems of **CL**. The third approach is non-classical in this sense, because it rejects some theorems of **CL**. In this paper, we will mainly be

[2]Some systems that were presented fall in various categories. For instance some of Lou Goble's **DPM** logics in [8] restrict inheritance and aggregation, and Van Der Torre and Tan's "2-phase logic" in [21] is a sequential system that first abandons inheritance and in the second phase abandons aggregation.

concerned with the classical approaches and shortly discuss the paraconsistent case in Section 9.

Bearing in mind the fact that there are various kinds of deontic conflicts, it would be a too narrow view to consider (OO-EX) as the only principle of deontic explosion. In analogy to (OO-EX), we can also state an explosion principle for OP-conflicts:[3]

$$\big(\mathsf{O}A \wedge \mathsf{P}\neg A\big) \supset \mathsf{O}B \tag{OP-EX}$$

Note that (OP-EX) is valid in **SDL** since $\mathsf{P}\neg A$ entails by (DfP), $\neg\mathsf{O}\neg\neg A$ and hence by (RM), $\neg\mathsf{O}A$. However, in view of (ECQ), the latter causes logical explosion together with $\mathsf{O}A$.

We will elaborate more on deontic explosion types in Section 6. However, the two mentioned types, (OO-EX) and (OP-EX), are sufficient for our discussion at this point.

4 Two shortcomings of classical CTDLs

To our present knowledge, all of the hitherto proposed classical CTDLs validate the following principle:[4]

$$\mathsf{P}A \supset \neg\mathsf{O}\neg A \tag{DfP1}$$

We will argue in this section that this gives rise to two serious problems. On the one hand CTDLs do not validate the very intuitive principle (D). On the other hand, they are explosive with respect to OP-conflicts. That is to say, they validate the explosion principle (OP-EX).

4.1 Principle (D)

There are two formulations of principle (D) that are frequent in the literature:

$$\mathsf{O}A \supset \mathsf{P}A \tag{D}$$

$$\mathsf{O}A \supset \neg\mathsf{O}\neg A \tag{D'}$$

In logics that validate (DfP1) both formulations are equivalent. Classical CTDLs that validate (DfP1), do not validate (D). The reason is simple: suppose that there is an OO-conflict between two premises $\mathsf{O}A$ and $\mathsf{O}\neg A$. By (D) we would be able to derive $\mathsf{P}A$ from $\mathsf{O}A$. However, by (DfP1), $\mathsf{P}A$ implies $\neg\mathsf{O}\neg A$. Together with $\mathsf{O}\neg A$ this trivializes the premises. Thus, conflict-tolerant logics that are formulated within classical (modal) logic do not validate both, (D) and (DfP1). Neither do classical CTDLs validate (D'). Evidently in case of a deontic conflict, $\mathsf{O}A \wedge \mathsf{O}\neg A$, by (D') e.g. $\neg\mathsf{O}\neg A$ is derivable. However this leads to triviality.

[3]Strictly speaking we should also feature the analogous principle $\big(\mathsf{O}\neg A \wedge \mathsf{P}A\big) \supset \mathsf{O}B$. However, we will only discuss systems that validate $\mathsf{P}A \equiv \mathsf{P}\neg\neg A$. Evidently, in these logics both principles are equivalent.

[4]In most CTDLs (DfP1) is trivially the case due to the fact that the permission operator P is defined by $\mathsf{P}A =_{\mathrm{df}} \neg\mathsf{O}\neg A$.

These observations have led some authors to reject (D) and (D') without further argument. Routley & Plumwood for instance, claim that (D) resp. (D') "is deontically incorrect because it rules out consistent inclusion of moral dilemmas" ([14], p.667). This argument holds evidently for (D'). However, it is dubious for (D), since (D) only rules out consistent inclusion of conflicts if it is assumed that (DfP1) is already valid.

Given the intuitive appeal of (D), it is unfortunate that authors tend to reject this principle without further ado. In command theories for example, we do not see how one could possibly command someone to do A without also permitting this person to do A. In legal contexts too, (D) seems perfectly intuitive: if someone is legally obliged to bring about A, he or she is also legally permitted to bring about A. Nevertheless, both in command theories and in legal contexts, deontic conflicts may occur.

It can also be argued more generally that, given the obligation to bring about A, there is or has to be assumed a source (such as a legal authority, an institution, a deity, etc.) that issued this very obligation. Moreover, it is reasonable to assume further that this source, implicitly or explicitly, supports also the permission to bring about A. If that were not so, then our normative source would be severely incoherent: it would oblige us to do something without at the same time permitting us to do so.

4.2 Tolerance concerning OP-conflicts

Classical CTDLs have yet another shortcoming. Although they allow for OO-conflicts, they do not allow for OP-conflicts (without causing explosion). Note that in classical CTDLs that validate (DfP1), $\mathsf{O}A \wedge \mathsf{P}\neg A$ is equivalent to $\mathsf{O}A \wedge \neg\mathsf{O}A$. By (ECQ) this does not just cause deontic explosion, i.e., the derivability of $\mathsf{O}B$ for any formula B, but even triviality, i.e., the derivability of any formula B. This is obviously undesired. First, it creates an asymmetry concerning the deontic conflict type that is tolerated, namely OO-conflicts, and the one that is neglected, namely OP-conflicts, which is hard to justify. To put it more bluntly: Why do CTDLs tolerate OO-conflicts but not OP-conflicts? Let us consider two possible replies:

1) One answer may be that OO-conflicts are prioritized by deontic logicians due to the fact that they are more frequent. However, as we have pointed out in Section 2, OO-conflicts typically give rise to OP-conflicts. This on the one hand undercuts the argument and on the other hand sheds even more bad light on CTDLs. Since CTDLs do not validate (D), in these logics OO-conflicts do not give rise to OP-conflicts.

2) Another answer may be that OO-conflicts possibly represent deontic dilemmas while OP-conflicts do not or have at least a clearly less dilemmatic character. Indeed, as argued in Section 2, given the conflict $\mathsf{O}A \wedge \mathsf{P}\neg A$, there is a way to act such that no norm is violated, namely by bringing about A. Given an OO-conflict $\mathsf{O}B \wedge \mathsf{O}\neg B$ there is evidently no way to avoid the violation of one of the two obligations. However, we turn the tables also on this argument. If a deontic logic is able to deal with deontic conflicts that have a more or less

severe dilemmatic character then, a fortiori, the logic should also be able to deal with lighter types of conflicts.

In conclusion we state that, considering the fact that OP-conflicts are no less frequent than OO-conflicts as pointed out in 1), as well as the less problematic character of OP-conflicts highlighted in 2), it is highly dubious that CTDLs are tolerant with respect to the former but not with respect to the latter type of conflict. This motivates the following requirement: CTDLs should be conflict-tolerant with respect to both types of conflicts, OO-, and OP-conflicts.

A last "excuse" for CTDLs not to be tolerant with respect to OP-conflicts would be that it is either technically very difficult or even impossible to realize or that realizing OP-conflict tolerance comes with prizes (in terms of derivative strength, complexity, etc.) that are too high to pay. However, we will demonstrate in this paper that neither is the case.

4.3 Solving the problems

As elaborated above, (DfP1) is a common root to both problems. This suggests that by giving up on (DfP1) both problems can be solved.

On the one hand, by abandoning (DfP1), CTDLs gain a higher degree of conflict-tolerance. Not just are they able to deal with OO-conflicts, but furthermore are they able to deal with OP-conflicts.

On the other hand, giving up (DfP1) offers the possibility to add (D) to the axiomatization of a CTDL without causing the problems pointed out above. That is to say, we gain conflict tolerant logics with respect to both conflict types that, on top, also validate the very intuitive (D).

Not only would (DfP1) cause triviality in CTDLs that allow for OP-conflicts. Depending on our interpretation of the deontic operators, (DfP1) may also be intuitively incorrect. Under a descriptive reading of O and P for instance, (D) is intuitively acceptable and OP-conflicts may very well occur, yet (DfP1) is unwanted. Suppose that we take a formula $\mathsf{O}A$ (resp. $\mathsf{P}A$) to express that there is a norm which obliges (resp. permits) us to do A. Similarly, we take a formula $\neg\mathsf{O}A$ (resp. $\neg\mathsf{P}A$) to express that there is no such norm. Now assume we have $\mathsf{P}A$. Then it might very well be that there is also an obligation $\mathsf{O}\neg A$, possibly issued by another normative source. Hence, to derive $\neg\mathsf{O}\neg A$, expressing that there is no norm offering the obligation to bring about $\neg A$, is unwanted here.

Next to the descriptive reading of the deontic operators, there is also a prescriptive reading. In the latter, $\mathsf{O}A$ (resp. $\mathsf{P}A$) expresses that we or the agent(s) in question are obliged (resp. permitted) to bring about A. $\neg\mathsf{O}A$ (resp. $\neg\mathsf{P}A$) expresses that we are not obliged (resp. permitted) to bring about A. In a prescriptive context, (DfP1) seems more intuitive. But here too (D) is intuitively correct and OP-conflicts may occur. Hence overall it is better to give up (DfP1) than to invalidate (D) and to verify (OP-EX)[5].

[5] An additional argument against (DfP1) is that, although it seems more appealing in case O and P are interpreted prescriptively, some authors have argued quite convincingly against such a reading of O and P (see, for instance, [22], Ch. 8, Sec. 2; [23], p. 11; [1]).

Now that we explained why we want adequate CTDLs to invalidate (DfP1), we still need to show that this will not cause any technical drawbacks. We will do so by presenting an algorithm for turning existing CTDLs into CTDLs that validate (D), that allow for the consistent inclusion of OP-conflicts, and that invalidate (DfP1) (Section 5). In Section 6 we show that logics obtained by this algorithm are sufficiently conflict-tolerant and offer a deductive strength that is comparable to the one of the original logic.

5 The generic transformation

In this section we will present a generic procedure for transforming a given CTDL into a system without the interdefinability between obligations and permissions. More precisely the transformed logic does not validate (DfP1) and is hence able to tolerate OP-conflicts. Moreover, this enables us to add the intuitive axiom (D).

We will first present the syntactic and then the semantic approach in terms of an algorithm. In Sections 7 and 8 we will give paradigmatic examples for our transformation based on concrete CTDLs that have been proposed in the literature.

In the following we presuppose (a) that a given CTDL **L** is formulated in the language $\mathcal{L}$ that features the unary operators O and P, and the classical connectives; (b) that **L** is axiomatized by means of all rules and axioms of classical logic and a some of the following rules and axioms: (N), (P), (rAND), (rRM), (DfP), and

$$\text{If } \vdash A \equiv B, \text{ then } \mathsf{O}A \equiv \mathsf{O}B. \tag{RE}$$

(c) that (RE) and (DfP) are part of the axiomatization of **L**; (d) that **L** is a rank-1 modal logic.[6]

Many CTDLs do indeed satisfy these points. Deontic logics that do not feature (DfP) as an axiom are usually formulated in the weaker language $\mathcal{L}$ that does not feature a permission operator and hence define $\mathsf{P}A$ by $\neg\mathsf{O}\neg A$. Evidently, a logic of the latter type can be straightforwardly transformed into a logic of the former kind that has the same consequence relation by simply adding the axiom (DfP) (see also [23], p. 17).

For the sake of simplicity, we do not allow for nested occurrences of O's and P's. We moreover presuppose that with the exception of (DfP) all the rules and axioms of a given CTDL are stated without occurrences of P-operators. Due to the fact that **L** features the axiom (DfP), every other rule and axiom has a canonical **L**-equivalent representation like that. That is to say, we only have to replace every occurrence of $\mathsf{P}A$ by $\neg\mathsf{O}\neg A$.[7]

[6] Rank-1 modal logics are axiomatized in such a way that there are no nested occurrences of modal operators in the rules and axioms. Note that the language of these logics may very well allow for nested occurrences of modal operators, the restriction only concerns the formulation of the axioms and rules.

[7] Due to aesthetic reasons we also remove double negations '$\neg\neg$' in- and outside of the scope of O. By (RE) and classical logic this yields equivalent rules and axioms.

5.1 Three types of CTDLs

As discussed in Section 3, there are basically two classical strategies to gain OO-conflict tolerance: (1) by restricting or abandoning the aggregation principle (AND), or (2) by restricting or abandoning the inheritance principle (RM). Of course, there are also (3) hybrid methods combining (1) and (2).

We will henceforth refer to logics following strategy (1) as *A-CTDLs*. Either these logics abandon the aggregation principle entirely (e.g. [3, 5, 6]), or restrict it along the following lines:

$$\big(\mathsf{O}A \wedge \mathsf{O}B \wedge R^{\wedge}_{A,B}\big) \supset \mathsf{O}(A \wedge B) \tag{rAND}$$

where $R^{\wedge}_{A,B}$ is a wff expressing a restriction on the aggregation principle. We presuppose that $R^{\wedge}_{A,B}$ is a formula in the closure of $\{A, B\}$ under the classical logical connectives and O.[8] Let us give some examples:

$$\big(\mathsf{O}A \wedge \mathsf{O}B \wedge \neg\mathsf{O}\neg(A \wedge B)\big) \supset \mathsf{O}(A \wedge B) \tag{PAND}$$

$$\big(\mathsf{O}A \wedge \mathsf{O}B \wedge \neg\mathsf{O}\neg A \wedge \neg\mathsf{O}\neg B\big) \supset \mathsf{O}(A \wedge B) \tag{PAND'}$$

(PAND) was used by Goble in [8], (PAND') was introduced in [20]. In the presence of (ECQ) a logic that abandons aggregation is equivalent to the same logic enriched with the axiom (rAND) where $R^{\wedge}_{A,B} = A \wedge \neg A$. Thus, we will henceforth presuppose that all A-CTDLs validate a form of the axiom (rAND).

Logics following strategy (2) will be denoted from now on *I-CTDLs*. Either these logics abandon the inheritance principle (RM) or they restrict it along the following lines:

$$\text{If } A \vdash B \text{ and } \nvdash \neg A, \text{ then } \big(\mathsf{O}A \wedge R^{\supset}_{A,B}\big) \supset \mathsf{O}B. \tag{rRM}$$

where $R^{\supset}_{A,B}$ is a wff expressing a restriction on the inheritance principle. Again we presuppose that $R^{\supset}_{A,B}$ is a formula in the closure of $\{A, B\}$ under the classical logical connectives and O. Examples for restricted inheritance principles are:

$$\text{If } A \vdash B \text{ and } \nvdash \neg A, \text{ then } \big(\mathsf{O}A \wedge \neg\mathsf{O}\neg A\big) \supset \mathsf{O}B. \tag{RPM}$$

$$\text{If } A \vdash B \text{ and } \nvdash \neg A, \text{ then } \big(\mathsf{O}A \wedge \neg\mathsf{O}\neg B\big) \supset \mathsf{O}B. \tag{RPM'}$$

The first one was used for Goble's **DPM** systems [8]. We are not aware of any logics employing the second proposal. Again, a logic that abandons inheritance is in the presence of (ECQ) equivalent to the same logic enhanced by (rRM) where $R^{\supset}_{A,B} = A \wedge \neg A$. Thus, we presuppose that all I-CTDLs validate a form of (rRM).

Hybrid logics employ a restricted version of aggregation (rAND) as well as

[8]The closure of $\{A, B\}$ under the classical connectives and O is the smallest set of formulas Ψ such that $\{A, B\} \subseteq \Psi$, if $C, D \in \Psi$ then $C \wedge D, C \vee D, C \supset D, C \equiv D \in \Psi$, and if $C \in \Psi$ then $\neg C, \mathsf{O}C \in \Psi$.

a restricted version of inheritance (rRM).[9] Some hybrid approaches that we are aware of are the logics **DPM.2** (see [8]) and **DPM.2′** (see [20]). We refer to hybrid CTDLs by *H-CTDLs*.[10]

5.2 The Axiomatization

Given a CTDL **L**, let us call it the *base logic*, we transform it into an OO- and OP-conflict tolerant logic $\mathbf{L}\star$ by means of the following steps. Our starting point is the axiomatization of **L**. The four steps indicate how this axiomatization is altered in order to arrive at the axiomatization of $\mathbf{L}\star$.

Step 1: Removing (DfP) As discussed before, the main feature that will turn the transformed logics OP-conflict tolerant is that they do not validate (DfP1). Hence we remove (DfP) from the axiomatization of $\mathbf{L}\star$.

Step 2: Adding (DfP2) Since our axiomatization does not feature anymore (DfP), (DfP1) and

$$\neg \mathsf{P}A \supset \mathsf{O}\neg A \tag{DfP2}$$

are not anymore validated per default. Thus, we add (DfP2) to the axiomatization of $\mathbf{L}\star$.[11] Since, as argued above, (DfP1) is causing deontic explosion for OP-conflicts, we obviously keep it out of our axiomatization.

Step 3: Adding (D) As discussed in Section 4, the lack of (D) is a serious shortcoming of classical CTDLs. One of the merits of removing (DfP1) is that it allows for the addition of (D) to the axioms without the usual disadvantages. That is to say, adding (D) does not make the logic less conflict-tolerant (see our discussion in Section 4). Hence, we add axiom (D) to the axiomatization of $\mathbf{L}\star$.

[9] The reader may wonder why hybrid cases are at all needed since, as we pointed out in our discussion, it is enough to either restrict aggregation or to restrict inheritance in order to achieve conflict-tolerance concerning OO-conflicts. However, besides the technical point of achieving conflict-tolerance there are other reasons that motivate the restriction of the aggregation and inheritance principles. It is not clear why aggregation (resp. inheritance) should hold unrestrictedly. For instance: Should aggregation (resp. inheritance) be applied to conflicting obligations? Should aggregation (resp. inheritance) be applied in cases it leads to deontic conflicts? A negative answer to these questions motivates the restriction of aggregation in addition to a restriction of inheritance (resp. vice versa).

[10] There are also approaches offered in the literature that sequentially apply CTDLs of different types. One example is the Two-Phase-Logic in [21]. Here first an A-CTDL and then an I-CTDL is applied. The generic transformation procedure that is presented in Section 5.2 and 5.3 can also be applied to such proposals by treating each logic in the sequence separately.

[11] By adding (DfP2) to our axiomatization we remain conservative with respect to the base logic **L** where (DfP2) is valid. If the reader finds (DfP2) counterintuitive or not fitting for the intended application, then step 2 of our transformation may be skipped. Nothing in our construction essentially relies on (DfP2). In general, invalidating (DfP2) does not make a logic any more conflict-tolerant than it already is. It does however make the logic *gap-tolerant* to some extent, i.e. invalidating (DfP2) is necessary if one wants to allow for situations where a wff A is neither permitted nor forbidden. This discussion, however, is beyond the scope of this paper.

Step 4: Further strengthenings Due to the fact that we give up on (DfP1), some intuitive theorems of **L** do not hold anymore. Since our aim is, on the one hand, to weaken **L** in order to gain OP-conflict-tolerance, and, on the other hand, to stay close to the deductive power of **L**, we are going to add certain further axioms and rules that are not in the way of the former goal.

(a) In case **L** is an A-CTDL, we add

$$\text{If } \vdash A \supset B, \text{ then } \vdash \mathsf{P}A \supset \mathsf{P}B. \tag{P-RM}$$

Note that, while the contra-position to (RM) is in A-CTDLs equivalent to (P-RM), this is not the case after giving up on (DfP1). Only the following weakened version of (P-RM) is valid in the transformed A-CTDLs:

$$\text{If } \vdash B \supset A, \text{ then } \vdash \big(\mathsf{P}B \wedge \neg\mathsf{O}\neg B\big) \supset \mathsf{P}A. \tag{RM-OP}$$

Since full inheritance is valid for obligations, it is intuitive to allow also for full inheritance for permissions. Hence, we strengthen our translation by the rule (P-RM).

(b) In case **L** is an I-CTDL or an H-CTDL, note that without (DfP1) the congruence principle for permission operators is not anymore entailed by the other axioms and rules. Thus, we add

$$\text{If } \vdash A \equiv B, \text{ then } \vdash \mathsf{P}A \equiv \mathsf{P}B. \tag{P-RE}$$

5.3 The semantics

In order to give a generic account concerning the semantics of CTDLs we need to consider a semantic frame powerful enough to represent most CTDLs, such as the two systems that we are going to present in Sections 7 and 8. We settle for neighborhood semantics[12] since they offer a well-known and very generic semantic framework in terms of which many CTDLs can be represented in a technically straightforward way. Moreover, strong completeness for the canonical formulations presented in this paper has been generically proven in [16].[13]

One of the basic ideas of the neighborhood semantics is that propositions are interpreted in terms of sets of worlds. Moreover, each world has associated with it propositions, i.e. sets of worlds. The idea is that an obligation $\mathsf{O}A$ is true at a world w, in case A is one of its associated propositions. Let us first look at the semantic framework for logics that validate (DfP1) and (DfP2).

[12] See Segerberg [17] and Chellas [3].

[13] The semantics that we introduce in this section are very similar to the way Goble defined neighborhood semantics for his **DPM** logics in [7, 8]. However, they vary from the former in two ways: (1) We make use of an actual world in our semantics. This makes the semantics philosophically more intuitive for the type of applications we have in mind. We are interested in deriving obligations and permissions from given premises and not only in modeling theoremhood. Thus, we will define a semantic consequence relation. (2) We alter the semantics in order to avoid the interdefinability between obligations and permissions.

5.3.1 Neighborhood semantics for CTDLs

Let $\wp(X)$ denote the power set of some set X. An O-frame F is a tuple $\langle W, \mathcal{O}\rangle$ where W is a set of points and $\mathcal{O} : W \to \wp(\wp(W))$. We call elements of W worlds. Thus, $\mathcal{O}$ assigns to each world $w \in W$ a set of propositions, i.e., $\mathcal{O}(w) \subseteq \wp(W)$. We write from now on $\mathcal{O}_w$ instead of $\mathcal{O}(w)$. An F-model M on an O-frame F is a triple $\langle F, v, @\rangle$ where $@ \in W$ is called the actual world and $v : \mathcal{S} \to \wp(W)$ with $\mathcal{S}$ being the set of propositional atoms. A propositional atom is mapped by v into the set of worlds in which it is supposed to hold. Validity at a world in a model is defined as usual for the classical connectives:

$$M, w \models A \text{ iff } w \in v(A), \text{ where } A \in \mathcal{S} \tag{M-$\mathcal{S}$}$$

$$M, w \models \neg A \text{ iff } M, w \not\models A \tag{M-$\neg$}$$

$$M, w \models A \vee B \text{ iff } (M, w \models A \text{ or } M, w \models B) \tag{M-$\vee$}$$

$$M, w \models A \wedge B \text{ iff } (M, w \models A \text{ and } M, w \models B) \tag{M-$\wedge$}$$

$$M, w \models A \supset B \text{ iff } M, w \models \neg A \vee B \tag{M-$\supset$}$$

Moreover, where $w \in W$, $|A|_M =_{\text{df}} \{w \in W \mid M, w \models A\}$, we define:

$$M, w \models \mathsf{O}A \text{ iff } |A|_M \in \mathcal{O}_w \tag{M-$\mathcal{O}$}$$

$$M, w \models \mathsf{P}A \text{ iff } M, w \models \neg\mathsf{O}\neg A \tag{M-DfP}$$

Furthermore, $M \models A$ iff $M, @ \models A$. Where $\Gamma \subseteq \mathcal{W}$, we say that M is an F-model of Γ iff M is an F-model and $M \models A$ for all $A \in \Gamma$. This can be generalized for O-frames F and classes of O-frames $\mathcal{F}$ in the following way: $\Gamma \Vdash_F A$ iff for all F-models M of Γ, $M \models A$, and $\Gamma \Vdash_{\mathcal{F}} A$ iff $\Gamma \Vdash_F A$ for all $F \in \mathcal{F}$.

Completeness and soundness is achieved with respect to a class of O-frames that is characterized by conditions that correspond to the given axioms of the logic in question. Let us for instance demonstrate this by giving the frame conditions for **SDL**. We define the following requirements on O-frames $F = \langle W, \mathcal{O}\rangle$. For all $w \in W$ we require the following:

$$\text{For all } X \in \mathcal{O}_w, W \setminus X \notin \mathcal{O}_w \tag{F-D}$$

$$\emptyset \notin \mathcal{O}_w \tag{F-P}$$

$$W \in \mathcal{O}_w \tag{F-N}$$

$$\text{For all } X, Y \subseteq W, \text{ if } X, Y \in \mathcal{O}_w, \text{ then } X \cap Y \in \mathcal{O}_w \tag{F-AND}$$

$$\text{For all } X, Y \subseteq W, \text{ if } Y \subseteq X \text{ and } Y \in \mathcal{O}_w, \text{ then } X \in \mathcal{O}_w \tag{F-RM}$$

Note that conditions "(F-X)" correspond to rules resp. axioms "(X)".

All we have to do for a concrete CTDL **L** is to give the appropriate frame conditions corresponding to the axiomatization of **L**. The way the axioms and rules of **L** are translated into frame-conditions is straightforward, as demonstrated above.[14] Where $\mathcal{F}$ is the class of O-frames that satisfies these conditions,

[14] We will explicate this for two more examples in Sections 7 and 8.

a semantic consequence relation is defined by $\Gamma \Vdash_{\mathbf{L}} A$ iff $\Gamma \Vdash_{\mathcal{F}} A$. Schröder and Pattinson have presented a generic soundness and (strong) completeness result for all such logics $\mathbf{L}$ with respect to the class of frames that fulfill the respective frame conditions (see [16]).

5.3.2 Neighborhood semantics for the translation

Now we will present the semantic part of our algorithm that turns a given CTDL $\mathbf{L}$ into an OO- and OP-conflict tolerant logic $\mathbf{L}\star$. The four steps of the algorithm are semantic counterparts to the four steps presented in Section 5.2.

Step 1: Add the relation $\mathcal{P}$ for permissions In order to model an autonomous permission operator, we introduce another assignment besides $\mathcal{O}$, namely $\mathcal{P} : W \to \wp(\wp(W))$. The idea is analogous to the one for obligations: $\mathsf{P}A$ is true at a world w, in case A is one of its associated propositions with respect to $\mathcal{P}$. We hence generalize O-frames: an OP-frame is a tuple $\langle W, \mathcal{O}, \mathcal{P}\rangle$ where $\langle W, \mathcal{O}\rangle$ is an O-frame. For an OP-frame $F = \langle W, \mathcal{O}, \mathcal{P}\rangle$ we define an F-model again by the triple $\langle F, v, @\rangle$. The validity relation $\models$ is defined by (M-$\mathcal{S}$)–(M-$\supset$) and (M-$\mathcal{O}$) as above, only without (M-DfP) and the following requirement is added for permissions:

$$M, w \models \mathsf{P}A \text{ iff } |A|_M \in \mathcal{P}_w \tag{M-$\mathcal{P}$}$$

In order to define the class of OP-frames that correspond to $\mathbf{L}\star$ we need to introduce some additional frame conditions corresponding to steps 2–4 in the axiomatic construction of $\mathbf{L}\star$.

Step 2: Add the frame condition for (DfP2) The frame condition for (DfP2) is given by[15]

$$\text{For all } X \subseteq W,\ \text{if } W \setminus X \notin \mathcal{O}_w,\ \text{then } X \in \mathcal{P}_w. \tag{FP-DfP2}$$

Step 3: Add the frame condition for (D) The frame condition for (D) is given by:

$$\text{For all } X \subseteq W,\ \text{if } X \in \mathcal{O}_w,\ \text{then } X \in \mathcal{P}_w. \tag{FP-D}$$

Step 4: Add the frame conditions for the further strengthenings Analogous to the axiomatization presented in Section 5.2, in case $\mathbf{L}$ is a A-CTDL we add:

$$\text{For all } X, Y \subseteq W,\ \text{if } X \subseteq Y \text{ and } X \in \mathcal{P}_w,\ \text{then } Y \in \mathcal{P}_w. \tag{FP-P-RM}$$

Note that in case $\mathbf{L}$ is an I-CTDL or an H-CTDL, there is no need to add a frame-condition that corresponds to (P-RE) (cp. point 4 (b) in the syntactical part of

[15] In case the reader is interested in a system that does not validate (DfP2), step 2 can be skipped. Compare the discussion on giving up on (DfP2) in Footnote 11.

the algorithm) since the congruence principles for the two deontic operators are by definition valid in the neighborhood semantics.[16]

The class of OP-frames corresponding to $\mathbf{L}\star$ consists thus of all frames $F_\star = \langle W, \mathcal{O}, \mathcal{P}\rangle$ where $\langle W, \mathcal{O}\rangle$ is an $\mathbf{L}$-frame and where $F_\star$ satisfies the additional frame conditions given in steps 2–4.

The following results are direct consequences of the way OP-frames where defined. It is easy to see that OP-frames are a generalization of O-frames in view of the following fact:

Fact 1. *Given a set of frame conditions $\mathcal{C}$ for O-frames, let $\mathcal{F}_O$ be the class of O-frames satisfying $\mathcal{C}$. Moreover let $\mathcal{F}_{OP}$ be the class of OP-frames satisfying $\mathcal{C}$,*

$$\text{For all } X \subseteq W, \text{ if } X \in \mathcal{O}_w \text{ then } W \setminus X \notin \mathcal{P}_w. \tag{FP-DfP1}$$

and (FP-DfP2)*. We have, $\Gamma \Vdash_{\mathcal{F}_O} A$ iff $\Gamma \Vdash_{\mathcal{F}_{OP}} A$.*

Let $\mathcal{W}_{\mathsf{O}}$ be all formulas in the language $\mathcal{L}'$, i.e., all formulas without occurrences of the P-operator.

Fact 2. *Where $F = \langle W, \mathcal{O}\rangle$ is an O-frame, $F_\star = \langle W, \mathcal{O}, \mathcal{P}\rangle$ is an OP-frame that satisfies* (FP-DfP2)*, $M = \langle F, v, @\rangle$ and $M_\star = \langle F_\star, v, @\rangle$ we have: (i) For all $A \in \mathcal{W}_{\mathsf{O}}$, $M \models A$ iff $M_\star \models A$, (ii) if $M \models \mathsf{P}A$ then $M_\star \models \mathsf{P}A$, and (iii) $|A|_M = |A|_{M_\star}$ for all $A \in \mathcal{W}_{\mathsf{O}}$.*

6 Some meta-theory

Let $\mathbf{L}\star$ be the translation of a given CTDL $\mathbf{L}$ gained by the algorithm presented in Section 5. In this section we will present some meta-theoretic insights for $\mathbf{L}\star$. In the Appendix [19] the reader can find the meta-theory for various strengthenings of $\mathbf{L}\star$.

Let us first compare the consequence relations that are characterized by $\mathbf{L}$ and $\mathbf{L}\star$. Evidently, they are not identical. In case a premise set features OP-conflicts this is desired, since $\mathbf{L}$ is explosive in these cases and we expect our $\mathbf{L}\star$ not to have explosive behavior. Moreover, they are also not equivalent for premise sets that do not give rise to OP-conflicts. For instance $\mathsf{P}A \vdash_{\mathbf{L}} \neg\mathsf{O}\neg A$, though $\mathsf{P}A \nvdash_{\mathbf{L}\star} \neg\mathsf{O}\neg A$. This is due to the fact that $\mathbf{L}$ validates (DfP1), while $\mathbf{L}\star$ doesn't. Similarly, $\mathsf{O}A \vdash_{\mathbf{L}\star} \mathsf{P}A$ while $\mathsf{O}A \nvdash_{\mathbf{L}} \mathsf{P}A$ due to that fact that only $\mathbf{L}\star$ validates (D). However, the two logics are equivalent with respect to premise sets that only feature formulas without occurrences of P-operators.

Theorem 1. *Where $\Gamma \subseteq \mathcal{W}_{\mathsf{O}}$ and $A \in \mathcal{W}_{\mathsf{O}}$, $\Gamma \vdash_{\mathbf{L}} A$ iff $\Gamma \vdash_{\mathbf{L}\star} A$.*

Given a set of premises Γ we define Γ_{O} by replacing every $A \in \Gamma$ by $\pi(A)$ where $\pi(A)$ is the result of replacing every occurrence of a formula $\mathsf{P}B$ in A by $\neg\mathsf{O}\neg B$. Hence $\Gamma_{\mathsf{O}} = \{\pi(A) \mid A \in \Gamma\} \subseteq \mathcal{W}_{\mathsf{O}}$. The following fact is an immediate consequence of $\mathbf{L}$ validating (DfP).

[16]The very easy proof is left to the reader.

Fact 3. *Γ and Γ_{O} are* **L**-*equivalent, i.e.* $\Gamma \vdash_{\mathbf{L}} A$ *iff* $\Gamma_{\mathsf{O}} \vdash_{\mathbf{L}} A$.

We say that $\mathbf{L}\star$ is **L**-*conservative* iff for all $A \in \mathcal{W}_{\mathsf{O}}$ and all **L**-consistent $\Gamma \subseteq \mathcal{W}$, $\Gamma \vdash_{\mathbf{L}} A$ iff $\Gamma_{\mathsf{O}} \vdash_{\mathbf{L}\star} A$. By Theorem 1 and Fact 3 we get the following corollary.

Corollary 1. $\mathbf{L}\star$ *is* **L**-*conservative.*

Moreover, the following theorem holds.

Theorem 2. *Where Γ is* **L**-*consistent, if* $\Gamma \vdash_{\mathbf{L}} \mathsf{P}A$ *then* $\Gamma_{\mathsf{O}} \vdash_{\mathbf{L}\star} \mathsf{P}A$.

Of course, this is not valid the other way around, since for instance $\mathsf{O}A \vdash_{\mathbf{L}\star} \mathsf{P}A$ while $\mathsf{O}A \nvdash_{\mathbf{L}} \mathsf{P}A$. Similarly, Theorem 2 cannot be generalized for negated permissions, since for instance $\mathsf{O}A \vdash_{\mathbf{L}} \neg\mathsf{P}\neg A$, while $\mathsf{O}A \nvdash_{\mathbf{L}\star} \neg\mathsf{P}\neg A$. This is due to the fact that, in order to allow for OP-conflicts, we had to abandon (DfP1).

Let us now have a look at issues concerning conflict-tolerance. Goble argued repeatedly in favor of the following requirement on CTDLs (e.g. [8], p. 464):

> (†): A CTDL should be such that adding (D) as an axiom results in a logic that is equivalent to **SDL**.

Criterion (†) seems to presuppose that CTDLs essentially need to abandon (D) in order to gain conflict-tolerance. However, we have argued that an alternative procedure is to give up on (DfP1) and to keep (D). Hence, normative criteria for CTDLs such as (†) have to be formulated relative to certain basic principles that have been given up in order to achieve conflict-tolerance. While for most CTDLs one such principle is indeed (D), in our case it is (DfP1). Thus, we reformulate the criterion accordingly for our transformation $\mathbf{L}\star$:

> (‡): A CTDL that does not validate (DfP1) should be such that adding (DfP1) as an axiom results in a logic that is equivalent to **SDL**.

There is a strong relation between (†), (‡), **L** and $\mathbf{L}\star$, as the following theorem shows:

Theorem 3. *If* **L** *satisfies (†), then* $\mathbf{L}\star$ *satisfies (‡).*

Criterion (†) is very demanding. Many CTDLs do not validate it. Take for instance logic **F** (see Section 7). It is not enough to add the axiom (D) to the axiomatization of **F** in order to gain a consequence relation that is equivalent to **SDL**. However, if additionally (AND) is added, it is enough. The following result shows that in these cases we also have an analogous principle to the one stated in Theorem 3.

Theorem 4. *If* **L** *strengthened by (D) and the set of axioms and rules Θ characterizes the same consequence relation as* **SDL**, *then* $\mathbf{L}\star$ *strengthened by* (DfP1) *and the axioms and rules in Θ characterizes the same consequence relation as* **SDL**.

Let us introduce another requirement concerning OP-conflicts: for an **L**-consistent premise set the only OP-conflicts that are **L**$\star$-derivable should be the ones resulting from the given OO-conflicts. This is expressed by the following theorem.

Theorem 5. *Where* Γ *is* **L***-consistent: For all finite index sets* I *and all wffs* A_i*,* $\Gamma_{\mathsf{O}} \nvdash_{\mathbf{L}\star} \bigvee_{i \in I} (\mathsf{O}A_i \wedge \mathsf{P}\neg A_i \wedge \neg\mathsf{O}\neg A_i)$*.*

Similarly this can be expressed semantically in terms of the following theorem.

Theorem 6. *For every* **L***-model* M *of* Γ *there is an* **L**$\star$*-model* $M_\star$ *of* Γ_{O} *for which* $M_\star \models \mathsf{O}A \wedge \mathsf{P}\neg A$ *iff* $M \models \mathsf{O}A \wedge \mathsf{O}\neg A$ *iff* $M_\star \models \mathsf{O}A \wedge \mathsf{O}\neg A$*.*

In the remainder of this section we will further investigate in the conflict-tolerance of **L**$\star$. Therefore it is useful to specify certain explosion principles that serve as benchmarks for the conflict-tolerance of CTDLs. We have two types of conflicts, OO- and OP-conflicts, and hence two types of explosion. An obligation explosion occurs if from a given conflict any obligation $\mathsf{O}B$ is derivable.

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{O}B \qquad \text{(OO-EX-O)}$$

$$\mathsf{O}A, \mathsf{P}\neg A \vdash \mathsf{O}B \qquad \text{(OP-EX-O)}$$

In semantic terms this means that there is no model that (i) validates the given conflict and that (ii) does not validate all obligations, or in other words, that validates $\neg\mathsf{O}B$ for some B.

These principles may be refined in various ways. In the paragraph above we have presented a very strict reading of obligation explosions, i.e. that all obligations are derivable given a deontic conflict. Another, weaker type of explosion is the case that for each B, $\mathsf{O}B \vee \mathsf{O}\neg B$ is derivable. Obviously it is not desirable that the logic, if it is confronted with a conflict, entails for each B either that it is obliged or that $\neg B$ is obliged. The same holds for the explosion type that corresponds to the case that for every B, $\mathsf{O}B \vee \neg\mathsf{P}B$ resp. $\mathsf{O}B \vee \mathsf{P}B$ resp. $\mathsf{O}B \vee \neg\mathsf{O}\neg B$ resp. $\mathsf{P}B$ is derivable. This can be formally expressed by:

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{O}B \vee \mathsf{O}\neg B \qquad \text{(OO-EX-OO}\neg\text{)}$$

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{O}B \vee \neg\mathsf{P}B \qquad \text{(OO-EX-O}\neg\text{P)}$$

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{O}B \vee \mathsf{P}B \qquad \text{(OO-EX-OP)}$$

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{O}B \vee \neg\mathsf{O}\neg B \qquad \text{(OO-EX-O}\neg\text{O}\neg\text{)}$$

$$\mathsf{O}A, \mathsf{O}\neg A \vdash \mathsf{P}B \qquad \text{(OO-EX-P)}$$

Note that in the names of the principles the prefix (OO and OP) denotes the *conflict type*, while the suffix (e.g. O, P, OP, etc.) denotes the *explosion type*.

A further requirement is to demand not just that there is a non-explosive model that validates the conflicting norms, but to impose certain normality conditions on this model. For instance, non-explosive models should also validate a non-conflicting obligation, e.g. $\mathsf{O}C$ and $\neg\mathsf{O}\neg C$, and/or a non-conflicting permission, e.g. $\mathsf{P}D$ and $\neg\mathsf{O}\neg D$, and/or there should be a proposition E such that neither E nor $\neg E$ is obliged, i.e. $\neg\mathsf{O}E \wedge \neg\mathsf{O}\neg E$, and/or there is a proposition F such that both, F and $\neg F$, are allowed, i.e. $\mathsf{P}F \wedge \mathsf{P}\neg F$. These conditions obviously hold for the real world and hence there should be also interpretations of deontic conflicts that satisfy these criteria. We will denote such refinements by adding the additional requirements in set brackets after the basic principle, for instance, where $\gamma = \{\neg\mathsf{O}E, \neg\mathsf{O}\neg E\}$ and $\gamma' = \{\mathsf{O}C, \neg\mathsf{O}\neg C, \mathsf{P}D, \neg\mathsf{O}\neg D, \neg\mathsf{O}E, \neg\mathsf{O}\neg E, \mathsf{P}F, \mathsf{P}\neg F\}$,

$$\{\mathsf{O}A, \mathsf{O}\neg A\} \cup \gamma \vdash \mathsf{O}B \vee \mathsf{P}B \qquad (\mathsf{OO}\text{-EX-}\mathsf{OP}\text{-}\gamma)$$
$$\{\mathsf{O}A, \mathsf{P}\neg A\} \cup \gamma' \vdash \mathsf{O}B \vee \mathsf{P}B \qquad (\mathsf{OP}\text{-EX-}\mathsf{OP}\text{-}\gamma')$$

Indeed, any "truly" conflict-tolerant logic should be tolerant concerning any of these principles. We call a logic Conflict-Tolerant iff it does not validate δ-EX-τ-β for any $\delta \in \{\mathsf{OO}, \mathsf{OP}\}$, any $\beta \subseteq \gamma'$ and any $\tau \in \{\mathsf{O}, \mathsf{P}, \mathsf{OO}\neg, \mathsf{O}\neg\mathsf{P}, \mathsf{O}\neg\mathsf{O}\neg\}$. Moreover, a logic is OO-Conflict-Tolerant iff it does not validate OO-EX-τ-β for any $\beta \subseteq \gamma'$ and any $\tau \in \{\mathsf{O}, \mathsf{P}, \mathsf{OO}\neg, \mathsf{O}\neg\mathsf{P}, \mathsf{O}\neg\mathsf{O}\neg\}$.

OO-EX-OP-γ-tolerance is indeed a strong criterion since, as the following theorem shows, it is enough that $\mathbf{L}$ is OO-EX-OP-γ-tolerant, in order to guarantee the conflict-tolerance of $\mathbf{L}\star$.

Theorem 7. *If $\mathbf{L}$ is OO-EX-OP-γ-tolerant then $\mathbf{L}\star$ is* conflict-tolerant.

Of course, if $\mathbf{L}$ is OO-Conflict-Tolerant then it is also OO-EX-OP-γ-tolerant and hence $\mathbf{L}\star$ is Conflict-Tolerant.

The next theorem shows that the conflict tolerance concerning OO-conflicts entails conflict-tolerance concerning OP-conflicts for our transformed logics.

Theorem 8. *Where $\tau \in \{\mathsf{O}, \mathsf{P}, \mathsf{OO}\neg, \mathsf{O}\neg\mathsf{P}, \mathsf{OP}, \mathsf{O}\neg\mathsf{O}\neg\}$ and β is a set of wffs, if $\mathbf{L}\star$ is OO-EX-τ-β-tolerant, then it is also OP-EX-τ-β-tolerant.*

The next theorem generalizes on Theorem 7. It shows that if $\mathbf{L}$ enjoys a certain type of conflict tolerance concerning OO-conflicts then $\mathbf{L}\star$ inherits this property.

Theorem 9. *Let β be set of formulas of the form $\mathsf{O}A$, $\neg\mathsf{O}A$ and $\mathsf{P}A$, and $\tau \in \{\mathsf{O}, \mathsf{OO}\neg, \mathsf{O}\neg\mathsf{P}, \mathsf{OP}, \mathsf{O}\neg\mathsf{O}\neg\}$. If $\mathbf{L}$ is OO-EX-τ-β-tolerant, then*

(i) $\mathbf{L}\star$ *is OO-EX-τ-β-tolerant,*

(ii) $\mathbf{L}\star$ *is OP-EX-τ-β-tolerant,*

(iii) if $\tau = \mathsf{O}\neg\mathsf{O}\neg$, then $\mathbf{L}\star$ is OO-EX-P-β-tolerant and OP-EX-P-β-tolerant.

7 Translating A-CTDLs: the logic F

In Section 5 a generic procedure was presented that transforms a given CTDL $\mathbf{L}$ into an OP-conflict-tolerant CTDL $\mathbf{L}\star$. After having investigated some properties of $\mathbf{L}\star$ in Section 6 it is time to apply the transformation to a concrete CTDL. While in this section we will demonstrate our algorithm on the basis of an A-CTDL, we will focus in the next section on an I-CTDL and a H-CTDL.

In [5] Van Fraassen presented the first axiomatization of a deontic logic that invalidates (AND). Chellas ([3]), Goble ([6]) and Schotch & Jennings ([15]) proposed (nearly) identical systems based on Van Fraassen's system, which we call **F**. We have chosen **F** as a first illustration of the application of our procedure defined in Section 5 since it is a well-known and rather simple system.

Definition 1. $\Psi_{\mathbf{F}}$ is the least set of formulas containing all classical tautologies, plus all instances of (P) and (DfP) that is closed under (RM) and Modus Ponens.

We define in a canonical way, $\vdash_{\mathbf{F}} A$ iff A is a member of $\Psi_{\mathbf{F}}$. Furthermore, where $\Gamma \subseteq \mathcal{W}$, $\Gamma \vdash_{\mathbf{F}} A$ iff for some suitable $B_1, \ldots, B_n \in \Gamma$ we have $\vdash_{\mathbf{F}} (B_1 \wedge \cdots \wedge B_n) \supset A$.[17]

The neighborhood semantics for **F** are defined in terms of the frame conditions (F-RM), and (F-N).

Where the **F**-frames are O-frames that satisfy the conditions above, by [16] we have the following soundness and completeness result: $\Gamma \vdash_{\mathbf{F}} A$ iff $\Gamma \Vdash_{\mathbf{F}} A$.

Concerning the conflict-tolerance of **F** the following theorem is proved in the Appendix [19].

Theorem 10. **F** *is* OO-CONFLICT-TOLERANT.

F was devised by Van Fraassen in order to allow for OO-conflicts without causing explosion. However, although **F** is OO-CONFLICT-TOLERANT, it is not CONFLICT-TOLERANT in the sense specified in Section 6. For instance, it validates OP-EX-O. We will now show that, by applying the algorithm defined in Section 5, we can define a variant of **F** that invalidates (DfP1) and that will turn out to be CONFLICT-TOLERANT:

Definition 2. $\mathbf{F}\star$ is defined according to our algorithm in the following way: (1) we remove axiom (DfP), (2) we add axiom (DfP2), (3) we add axiom (D), (4) we add rule (P-RM). The consequence relation $\vdash_{\mathbf{F}\star}$ is defined analogous to $\vdash_{\mathbf{F}}$.

For the neighborhood semantics of $\mathbf{F}\star$ we need to (1) use OP-frames, (2) add the frame condition (FP-DfP2), (3) add the frame condition (FP-D), and (4) add the frame condition (FP-P-RM). The $\mathbf{F}\star$-frames are hence all OP-frames that satisfy the frame conditions (F-RM), (F-N), (F-D), (FP-P-RM), and (FP-DfP2). Due to [16], $\Gamma \vdash_{\mathbf{F}\star} A$ iff $\Gamma \Vdash_{\mathbf{F}\star} A$.

With our meta-theory in Section 6 we immediately get the following result.

Corollary 2. $\mathbf{F}\star$ *is* **F**-*conservative.*

We expect $\mathbf{F}\star$ to be OO-CONFLICT-TOLERANT as is **F**. Moreover, the logic should now also be conflict-tolerant with respect to OP-conflicts. The following corollary shows that indeed $\mathbf{F}\star$ has both properties. It follows directly from Theorem 7 and Theorem 10.

Corollary 3. $\mathbf{F}\star$ *is* CONFLICT-TOLERANT.

[17] See also [16] where the authors define consequence relations for rank-1 modal logics in this way and prove strong completeness. Van Fraassen originally defined **F** in terms of theoremhood. Since we are interested in modeling the consequences of premises we defined a consequence relation for **F**.

8 Translating I-CTDLs and H-CTDLs: the DPM logics

Goble proposed in [7, 8] a way to deal with deontic dilemmas by restricting the inheritance principle. The full inheritance principle (RM) is replaced by (RPM) (see Section 5.1). The idea is to apply inheritance to an obligation $\mathsf{O}A$ if there is no OO-conflict concerning $\mathsf{O}A$, i.e., in case $\neg\mathsf{O}\neg A$. Goble defines his **DPM.1** along the following lines:

Definition 3. $\Psi_{\mathbf{DPM.1}}$ is the least set of formulas containing all classical tautologies of formulas of $\mathcal{W}$, plus all instances of (N), (AND), (DfP), that is closed under Modus Ponens, (RE), and (RPM). We define in a canonical way, $\vdash_{\mathbf{DPM.1}} A$ iff A is a member of $\Psi_{\mathbf{DPM.1}}$. Furthermore, where $\Gamma \subseteq \mathcal{W}$, $\Gamma \vdash_{\mathbf{DPM.1}} A$ iff for some suitable $B_1, \ldots, B_n \in \Gamma$ we have $\vdash_{\mathbf{DPM.1}} (B_1 \wedge \cdots \wedge B_n) \supset A$.[18]

Semantically **DPM.1** can be represented by O-frames that satisfy conditions (F-P), (F-AND), and

$$\text{If } X \subseteq Y; X \in \mathcal{O}_w \text{ and } W \setminus X \notin \mathcal{O}_w, \text{ then } Y \in \mathcal{O}_w. \tag{F-RPM}$$

In Section 5.1 we have already motivated why H-CTDLs are useful. Let us take a look at a concrete H-CTDL: **DPM.2**. This system stems from the same family of logics as **DPM.1**. Note, that in **DPM.1** axiom (N), $\neg\mathsf{O}\bot$, is not valid. Due to the fact that in **DPM.1** full aggregation is valid, given an OO-conflict, $\mathsf{O}A \wedge \mathsf{O}\neg A$, $\mathsf{O}(A \wedge \neg A)$ is derivable. This violates the Kantian 'ought implies can' principle since the obligation to bring about $A \wedge \neg A$ is impossible to realize. In order to gain a system that validates (N) and that is in the spirit of 'ought implies can', Goble introduces a variant of his I-CTDL. Instead of full aggregation, **DPM.2** features a restricted version of aggregation, namely (PAND) (see Section 5.1). The idea is that aggregation is applicable to two obligations $\mathsf{O}A$ and $\mathsf{O}B$ in the case that $\mathsf{O}(A \wedge B)$ does not cause an OO-conflict. A similar system was proposed under the name **DPM.2′** in [20]. It is just as **DPM.2**, only instead of (PAND) it features the axiom (PAND'). The idea behind (PAND') is that aggregation is only applied to non-conflicting obligations.

Definition 4. The hybrid CTDL **DPM.2** resp. **DPM.2′** is defined just as **DPM.1**, with two exceptions: we add (N) and replace (AND) by (PAND) resp. (PAND'). Semantically we add (F-P) and replace (F-AND) by

$$\text{For all } X, Y \subseteq W, \text{ if } X, Y \in \mathcal{O}_w \text{ and } W \setminus X \cap Y \notin \mathcal{O}_w, \text{ then } X \cap Y \in \mathcal{O}_w. \tag{F-PAND}$$

resp. by

$$\text{For all } X, Y \subseteq W, \text{ if } X, Y \in \mathcal{O}_w, W \setminus X \notin \mathcal{O}_w, \text{ and } W \setminus Y \notin \mathcal{O}_w, \text{ then } X \cap Y \in \mathcal{O}_w. \tag{F-PAND'}$$

We call the resulting frames the **DPM.2**-frames resp. the **DPM.2′**-frames.

[18] Goble originally defines his **DPM.1** in terms of theoremhood. Since we are interested in modeling the consequences of premises we defined a consequence relation for **DPM.1**.

By [16] we get, where $\alpha \in \{1, 2, 2'\}$, $\Gamma \vdash_{\mathbf{DPM}.\alpha} A$ iff $\Gamma \Vdash_{\mathbf{DPM}.\alpha} A$.

As suggested in Section 5 we define variants of **DPM.1**, **DPM.2**, and **DPM.2′** that do not validate (DfP1), that validate (D), and that will turn out to be CONFLICT-TOLERANT.

Definition 5. Where $\alpha \in \{1, 2, 2'\}$, we define the logic $\mathbf{DPM}.\alpha\star$ according to our algorithm in the following way: (1) we remove axiom (DfP), (2) we add axiom (DfP2), (3) we add axiom (D), (4) we add rule (P-RE).

Semantically $\mathbf{DPM}.\alpha\star$ is represented by OP-frames that satisfy the $\mathbf{DPM}.\alpha$ frame conditions, (FP-DfP2), and (FP-D).

By [16] we have, where $\alpha \in \{1, 2, 2'\}$, $\Gamma \vdash_{\mathbf{DPM}.\alpha\star} A$ iff $\Gamma \Vdash_{\mathbf{DPM}.\alpha\star} A$.

Let us take a look at some of the properties of the given logics. It follows immediately by our meta-theory of Section 6 that $\mathbf{DPM}.\alpha\star$ is $\mathbf{DPM}.\alpha$-conservative.

Corollary 4. *Where* $\alpha \in \{1, 2, 2'\}$*,* $\mathbf{DPM}.\alpha\star$ *is* $\mathbf{DPM}.\alpha$*-conservative.*

Theorem 11. **DPM.1** *and* **DPM.2′** *satisfy criterion (†).*

Corollary 5. $\mathbf{DPM.1}\star$ *and* $\mathbf{DPM.2'}\star$ *satisfy criterion (‡).*

In order to demonstrate the full conflict-tolerance of our translations $\mathbf{DPM.1}\star$, $\mathbf{DPM.2}\star$, and $\mathbf{DPM.2'}\star$ we make use of Theorem 7 and of the following result.

Theorem 12. *Where* $\alpha \in \{1, 2, 2'\}$*,* $\mathbf{DPM}.\alpha$ *is* OO-CONFLICT-TOLERANT.

Corollary 6. *Where* $\alpha \in \{1, 2, 2'\}$*,* $\mathbf{DPM}.\alpha\star$ *is* CONFLICT-TOLERANT.

9 Some thoughts on paraconsistent CTDLs

In Section 1, we briefly mentioned the paraconsistent approach for making deontic logics more conflict-tolerant. We did not include this approach in the presentation of our generic procedure in Section 5, because, to the best of our knowledge, all paraconsistent deontic systems presented in the literature are already OO- as well as OP-conflict-tolerant, even though they validate (DfP) (see, for instance, [4, 13]). Typically, explosion principles rest on (i) the derivation of an inconsistency from a deontic conflict and (ii), the application of (ECQ) to this inconsistency, which causes the derivation of the obligation and/or permission of every proposition in the domain. Since paraconsistent deontic systems invalidate (ECQ), step (ii) is blocked and explosion is avoided. Hence there is no need for applying our generic procedure to such systems in order to make them more conflict-tolerant. However, as discussed in Section 4.3, under a descriptive reading of the normative operators it is rather dubious to infer from $\mathsf{O}A$ by (DfP1) $\neg\mathsf{P}\neg A$. Hence, there are applications for which it is also interesting to relax the interdefinability between O and P in a paraconsistent setting.[19] For

[19] In [2], a nonmonotonic paraconsistent deontic logic has been presented that invalidates the application of (DfP) to possibly conflicted norms. We are unaware of monotonic paraconsistent systems that invalidate (DfP).

paraconsistent deontic logics validating both aggregation and inheritance, our generic translation reduces to applying steps 1-3 in the procedure presented in Section 5.

10 Conclusion

In this paper we proposed a way to deal with obligation-permission conflicts of the kind $\mathsf{O}A \wedge \mathsf{P}\neg A$ in deontic logics. The idea was to relax the usual interdefinability between obligations and permissions. This enabled us to validate the very intuitive principle (D) that makes it possible to derive the permission to bring about A from the obligation to bring about A. (D) is usually not valid in classical conflict-tolerant deontic logics. We have presented our idea in terms of an algorithm that turns conflict-tolerant logics into logics that are also non-explosive concerning obligation-permission conflicts. We have shown that the transformed systems have a similar derivative strength compared to the old logics and are as conflict-tolerant concerning obligation-obligation conflicts as the old logics. Although we have presented our results in terms of monadic deontic logics, it is a very easy exercise to adjust the transformation procedure for the conditional case. Altogether our procedure makes available a multitude of highly conflict-tolerant systems that have not been considered before in the literature. Moreover, relaxing the interdefinability between obligations and permissions instead of abandoning principle (D) is a novel design principle for conflict-tolerant logics that has not been addressed in the literature so far and may be fruitful for the designing of new deontic logics that aim for a high degree of conflict-tolerance.

References

[1] Carlos E. Alchourrón. Logic of norms and logic of normative propositions. *Logique et Analyse*, 47:242–268, 1969.

[2] Mathieu Beirlaen, Joke Meheus, and Christian Straßer. A paraconsistent adaptive deontic logic for normative conflicts. Forthcoming.

[3] Brian F. Chellas. *Modal Logic: An Introduction*. Cambridge University Press, 1980.

[4] Newton Da Costa and Walter Carnielli. On paraconsistent deontic logic. *Philosophia*, 16(3):293–305, December 1986.

[5] Bas C. Van Fraassen. Values and the heart's command. *Journal of Philosophy*, 70(1):5–19, 1973.

[6] Lou Goble. Multiplex semantics for deontic logic. *Nordic Journal of Philosophical Logic*, 5:113–134, 2000.

[7] Lou Goble. A proposal for dealing with deontic dilemmas. In Alessio Lomuscio and Donald Nute, editors, *DEON*, volume 3065 of *Lecture Notes in Computer Science*, pages 74–113. Springer, 2004.

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

[9] Christopher W. Gowans, editor. *Moral Dilemmas*. Oxford University Press, New York, 1987.

[10] John Holbo. Moral dilemmas and the logic of obligation. *American Philosophical Quarterly*, 339(3):259–274, 2002.

[11] H.E. Mason, editor. *Moral Dilemmas and Moral Theory*. Oxford University Press, New York, Oxford, 1996.

[12] Terrance McConnell. Moral dilemmas and consistency in ethics. *Canadian Journal of Philosophy*, 8:269–287, 1978.

[13] Graham Priest. *In Contradiction: A Study of the Transconsistent*. Oxford University Press, 2006.

[14] R. Routley and V. Plumwood. Moral dilemmas and the logic of deontic notions. In Graham Priest, Richard Routley, and Jean Norman, editors, *Paraconsistent Logic. Essays on the Inconsistent*, pages 653–690. Philosophia Verlag, München, 1989.

[15] P. K. Schotch and R. E. Jennings. Non-kripkean deontic logic. In R. Hilpinen, editor, *New Studies in Deontic Logic: Norms, Actions, and the Foundations of Ethics*, pages 149–162. Reidel Pub Co., 1981.

[16] Lutz Schröder and Dirk Pattinson. Rank-1 modal logics are coalgebraic. *Journal of Logic and Computation*, December 2008. Forthcoming. Online access `http://dx.doi.org/10.1093/logcom/exn096`.

[17] Krister Segerberg. *An essay in classical modal logic.* Uppsala Universitet, Filosofiska Studier, 1971.

[18] Walter Sinnott-Armstrong. *Moral Dilemmas*. Basil Blackwell, Oxford, 1988.

[19] Christian Straßer and Mathieu Beirlaen. Appendix to towards more conflict-tolerant deontic logics by relaxing the interdefinability between obligations and permissions. `http://logica.ugent.be/centrum/preprints/OP-DFP-appendix.pdf`, 2010.

[20] Christian Straßer, Joke Meheus, and Mathieu Beirlaen. Tolerating deontic conflicts by adaptively restricting inheritance. To appear.

[21] Leendert van der Torre and Yao Hua Tan. Two-phase deontic logic. *Logique et Analyse*, (171-172):411–456, 2000.

[22] G.H. Von Wright. *Norm and Action. A Logical Enquiry.* Routledge & Kegan Paul, London, 1963.

[23] G.H. Von Wright. *An Essay In Deontic Logic and The General Theory of Action.* North-Holland Publishing Company, Amsterdam, 1972.