An integrated framework for ought–to–be and ought–to–do constraints (extended abstract)

P. d'Altan
J.– J.Ch. Meyer*
R.J. Wieringa
Department of Mathematics and Computer Science – Vrije Universiteit
De Boelelaan 1081a – 1081 HV Amsterdam – The Netherlands
Email: daltan@cs.vu.nl, jules@cs.vu.nl, roelw@cs.vu.nl

1 Introduction

Deontic logic is the logic to reason about ideal and actual behaviour. Besides the traditional role as an underlying logic for law and ethics (for a survey see [MW]), in the realm of computer science, deontic logic has been proposed as a logic for the specification of legal expert systems [BMT87],[Sta80], authorization mechanisms [ML85], decision support systems [KL88], [Lee88b],[Lee88a], database security rules [GMP89], fault–tolerant software [KM87], [Coe], and database integrity constraints[WMW89], [WWMD91]. A survey of applications can be found in [WM]. In all these areas, we must be able to reason about the difference between ideal and actual behaviour. In many cases, it is important to distinguish *ought–to–do* statements (which express imperatives of the form "an actor ought to perform an action") from *ought–to–be* statements (which express a desired state of affairs without necessarily mentioning actors and actions that have a relation with that state of affairs). There are situations where we would like to relate the two oughts with each other. For example, suppose we want to specify deontic integrity constraints for a bank data base. From the ought–to–be constraint

> (1.) The balance of a bank account must be non–negative

we would like to conclude the ought–to–do statement

> (2.) If the balance of a bank account is n and $n - m < 0$, then it is forbidden to withdraw m from the account.

In addition, we would like to be able to express

> (3.) If the balance of a bank account is n and $n < 0$, then an action $deposit(-n)$ ought to be performed.

*also at the Katholieke Universiteit Nijmegen

In most systems of deontic logic, the derivation (1.) $\vdash$ (2.) cannot be made and norm (3.) cannot be expressed.

In standard deontic logic (SDL) [Han71], norms are expressed by applying a sentential operator O to sentence letters p. Now, Op cannot be read as "p ought to be done", for then p would not be a sentential letter. In

> (4.) $deposit(m)$ ought to be done

$deposit(m)$ is not a sentence. So Op must be read as an ought–to–be statement, as in

> (5.) that $(balance \geq 0)$ ought to be.

So the reason why in SDL we cannot get from the ought–to–be statement (1.) to the ought–to–do statement (2.) or to express the ought–to–do statement (3.) is that in SDL we cannot talk about actions.

The distinction between ought–to–be and ought–to–do is not only a matter of syntax, as the opinions of several philosophers clearly show. Castañeda, for instance, asserts that

> Deontic statements divide neatly into: (i) those that involve agents and actions and support imperatives, and (ii) those that involve states of affairs and are agentless and have by themselves nothing to do with imperatives. The former belong to what used to be called the *Ought–to–do* and the latter to the *Ought–to–be*. ([Cas70], 452)

Castañeda's distinction may be easily interpreted as a rejection of SDL as a logic of normative concepts. A radical position in this sense has been assumed by P. Geach:

> obligation essentially relates to an agent, it is *somebody*'s obligation; if instead we try to think of the ought–to–be–ness, Sein–sollen, of a situation involving the agent, then our thinking is going to be *confused*; our mental vision, so to say, is prevented from coming to a proper focus. ([Gea81], 2–3).

Geach tries accordingly to show that ought–to–be statements are to be understood as ought–to–do ones[1]:

> 'There ought to be a law against smoking in buses' (if the speaker is really thinking) will mean that the predicable '— ought to make a law against smoking in buses' is true of some person or persons. ([Gea81], 4)

Similar ideas seem to have been already defended by W. D. Ross and before him by A. Prichard[2]

Other authors, e.g., von Wright, defend a more moderate position. Instead of eliminating one of the two terms of the question, as Geach does, they assign to the ought–to–be an autonomous sphere of meaning: we quote from [vW63]

> There is, however, a group of norms which are immediately concerned, not with action, but with things that ought to or may or must not *be*. German writers sometimes make a distinction between *Tunsollen* and *Seinsollen*. [...] Following G.E. Moore, I shall call norms which are concerned with being rather than with doing, *ideal rules*. Ideal rules are referred to, for example, when we say that a man ought to be generous, truthful, just, temperate, etc., and also when we say that a soldier in the army should be brave, hardy, and disciplined; a schoolmaster patient with children, firm and understanding; a watchman alert, observant, and resolute; and so forth. [...] Ideal rules are closely connected with the concept of *goodness*. [...] The features which ideal rules require to be present in good members of a class or kind of human beings can be termed the *virtues* characteristic of men of that class or kind. ([vW63], 13–14).

For certain aspects, Von Wright's conception has already been maintained since the beginning of this century by the phenomenological school (Brentano, Husserl, Meinong and Mally) and reached its highpoint with Scheler and Hartmann, who grounded ought–to–do upon ought–to–be (a conclusion opposite to Geach's one). According to these authors, not only SDL would not be a logic of the ought–to–do but also it could not express ought–to–be statements adequately for reasoning about what is good and bad. The main reason is that speaking of bad and good does not seem to require any notion of necessity at all differently from what is implicit assumption in SDL. This might better be used to reason about possible (moral or not moral) worlds. Of course, this means also that SDL is inadequate for representing moral norms.

However, this is no real drawback for our purposes. For us, SDL is useful for describing desiderable states of affairs devoid of any moral content. We are indeed going to use it for expressing more prosaic things like static deontic constraints such as (1.), where the deontic operator expresses an "ideal" necessity but no moral attitude at all (i.e., we will indeed consider a particular kind of alethic modalities, those that satisfy axiom D).

But we also need to extend SDL to deal with dynamic deontic contraints like (2.) and (3.), this will be done by using propositional dynamic logic.

We first briefly review Anderson's reduction of ought to be statements to alethic ones and successively Meyer's reduction of ought to do statements to dynamic ones. In the last sections we will show how by making use of both reductions it is possible to solve the expressivity problems sketched in the first part of this introduction.

2 Anderson reduction of ought–to–be sentences to alethic modal ones

The deontic system Anderson considers in [And67] is von Wright's system M without the T–axiom (i.e., $Op \rightarrow p$), in the classification of Lemmon [Seg77] and Chellas [Che80] it is called D and KD, respectively.

Anderson's reduction may be seen as a kind of Dawson's modelling [Daw59] for SDL in which O–sentences are interpreted by means of T–formulas of the form $\Box(\neg p \rightarrow V)$, where V is a propositional constant interpreted by Anderson rather freely as either expressing some form of sanction or a bad thing or also as stating that "things go wrong". Thus, Anderson reduction of Op has to be read as "necessarily, $\neg p$ implies a sanction (that things go wrong)". V is subject to three conditions:

1) $\Diamond\neg V$ is assumed (or proved) valid

2) $\neg V$ cannot be proved valid

3) V cannot be proved valid

Viewed model–theoretically, models for the deontic system are just a special kind of models for modal sentences. The nature of these models can be shown in a rather straightforward way. Let $\perp$ represent a logical contradiction. In propositional calculus (PC) we can prove that for every p it holds true that

$$p \equiv (\neg p \rightarrow \perp).$$

In every normal modal system this involves also the truth of

$$\Box p \equiv \Box(\neg p \rightarrow \perp).$$

As it should be apparent, we have here an analogon of Anderson's reduction where V is replaced by a contradiction[3].

[1] For a critical appraisal of Geach's paper see [Gar86]

[2] Cfr. [Gar86], 276 n22. The paper of Prichard can also be found in [SH70], 86–96.

[3] This also represents the classical view about necessity according to which what is necessary is by definition contrary to what implies contradiction.

Intuitively, contradiction can be seen as a sort of logical "things go wrong" in the sense that if p is necessarily true then assuming it false involves a sort of logical breakdown. The same piece of reasoning may however be applied to the deontic case: if the state of affairs p ought to be, then if it were not the case, something would go deontically wrong. In other words, "things go wrong" (deontically) is a sort of deontic contradiction. Of course, it is much less strong qua logical nature than an alethic one, since we cannot prove it to be a contradiction (condition 2 above). A logical contradiction is moreover false in every accessible possible world, so that

$$\Box\neg\bot$$

is derivable in every normal system. It is now evident that necessity operators may be deontically interpreted only where V is false. That is, if we define Op as $\Box(\neg p \rightarrow V)$, then a sufficient condition for $w \models \Box p \equiv Op$ is

$$\forall w'(wRw' \Rightarrow w' \models \neg V).$$

This explains why we cannot in general consider $\Box$ and O as representing the same notion of necessity.

We will deviate from Anderson's formulation at least in one important aspect. We have seen that he interprets V in terms of sanction or in terms of an unspecified bad thing. We propose a more specific reading in terms of "violation". We read Op as "necessarily, $\neg p$ implies violation of the normative system to which Op belongs", where with normative system we simply intend a set of deontic constraints, thus not necessarily (or preferably not) moral norms. There are principally two reasons that justify our reading of V:

1. It avoids the counterintuitive readings of a few derivations of SDL. These readings are counterintuitive when V is interpretated as sanction (cfr. [Cas60]). Consider, for example, $\Box(V \rightarrow V)$ and $Pp \equiv \Diamond(p \wedge \neg V)$. The former formula means in Anderson's system "sanctions are forbidden" and this is evidently counterintuitive. For the latter, consider the following example taken from [Cas60], 46:

   > Let V, i.e. the sanction mentioned in the Penal Code, be 'you will be put in jail for 10 years'; and p be 'you will be put in jail for 9 years'. Clearly, it is logically possible to put you in jail for 9, but not 10 years. Thus, it follows logically that it is permitted to put you in jail for 9 years — without ado!

2. Our reading offers an almost tautological and therefore scarcely objectionable reading of deontic formulas: if I do not fulfil a norm, then I violate it, or if p is not the case, then the relevant constraint has been violated.

3 Reduction of ought–to–do sentences to dynamic ones

Inspired by Anderson's not so satisfactory (wrt ought–to–do) reduction to alethic modal logic, and after analyzing the reason why it failed, Meyer [Mey88] proposed another reduction, in this case to propositional dynamic logic. A consequence of the use of dynamic logic is the distinction between propositions (assertions) and actions (practitions, cf. [Cas81]). Meyer's reduction uses Anderson's violation atom V to indicate that an action has occurred that violates one of the deontic constraints.

(Propositional) Dynamic Logic (PDL, cf. [Har84]) consists of the normal propositional language, extended with modal operator $[\alpha]$ for every action α in the language. These actions are either atomic (primitive) or composed by means of operators. An expression $[\alpha]\phi$ is read as "the performance (execution) of the action α leads necessarily to a state (possible world) in which ϕ holds". In this approach, α is forbidden ($\hat{F}\alpha$), permitted ($\hat{P}\alpha$), and obligated ($\hat{O}\alpha$) are reduced to dynamic expressions as follows:

Definition 3.1

$$\hat{F}\alpha \equiv [\alpha]V \qquad (1)$$

$$\hat{P}\alpha \equiv \neg\hat{F}\alpha\,(\equiv \langle\alpha\rangle\neg V) \qquad (2)$$

$$\hat{O}\alpha \equiv \hat{F}\overline{\alpha}\,(\equiv [\overline{\alpha}]V) \qquad (3)$$

Here for the reduction of the obligation operator $\hat{O}$, we employed the *negation* of an action α, denoted $\overline{\alpha}$, expressing *refraining* from the perfomance of α. The concept of action negation is discussed in [Mey89], [DM90] and [WM91].

The formal semantics is given by means of a Kripke structure where there are accessibility relations R_α associated with each action α. In particular, PDL is characterized by irreflexive frames of possible worlds.

Definition 3.2 *A model M for PDL is given by $M = \langle \wp^+(A), W, [\![\alpha]\!]_R, \models \rangle$ where $\wp^+(A)$ represents the power set of sets of actions, W a set of possible worlds, $[\![\alpha]\!]_R$ a function that associates to action α and world w, the set of possible worlds to which the performance of α leads, and $\models$ the usual truth–relation between worlds and sentences.*

The truth–definitions for dynamic formulas are as follows:

Definition 3.3

$$w \models [\alpha]\phi \quad \text{iff} \quad \forall w'(w' \in [\![\alpha]\!]_R(w) \Rightarrow w' \models \phi)$$

i.e., sentence $[\alpha]\phi$ is true in w iff ϕ holds true in every world accessible from w by performing α.

The approach of reducing standard deontic logic to dynamic logic has the advantage of allowing integration of static deontic constraints (Seinsollen or ought–to–be sentences) with dynamic deontic constraints (Tunsollen or ought–to–do sentences). This will be the subject of the following section.

4 Axiomatization

The system we are going to assume as basic is characterized as a mixed modal–dynamic logic with the following axioms

Axiom 4.1

$$\Box(p \rightarrow q) \rightarrow (\Box p \rightarrow \Box q) \qquad K$$

$$\Box p \rightarrow p \qquad T$$

$$\neg\Box p \rightarrow \Box\neg\Box p \qquad 5$$

$$\Diamond\neg V \qquad D$$

$$[\alpha](p \rightarrow q) \rightarrow ([\alpha]p \rightarrow [\alpha]q) \qquad AK$$

The rules are, as usual, necessitation for both operators (N and [N], respectively) and modus ponens (MP).

5 From ought–to–be to ought–to–do

The integration requires no particular technical step besides the definition of modal and dynamic ought in terms of Anderson and Meyer reduction, respectively. For modal fomulas we will adopt a S5–semantics adapted in order to cope with axiom $\Diamond\neg V$[4] and retain the standard semantics of PDL for dynamic formulas.

We consider the formal counterpart of the example presented in the introduction.

Let b means "balance", $\mathrm{w}(m)$ "withdraw at least m from the account", and $\mathrm{d}(m)$ "deposit at least m on the account", where b is a variable, n and m constants.

We make use of the following

Definition 5.1

$$Fp = \Box(p \rightarrow V) \qquad \text{(a)}$$

$$Op = \Box(\neg p \rightarrow V) \qquad \text{(b)}$$

and of axioms of arithmetic.

We have a specification consisting of the following constraints:

[4]See [Åqv88], 107–113, for details.

Specification 5.2

$$O(b \geq 0) \qquad (1)$$

$$(b = n \wedge m \geq 0) \rightarrow [\mathrm{w}(m)](b \leq n - m) \qquad (2)$$

$$(b = n \wedge m \geq 0) \rightarrow [\overline{\mathrm{w}(m)}](b > n - m) \qquad (3)$$

$$(b = n \wedge m \geq 0) \rightarrow [\mathrm{d}(m)](b \geq n + m) \qquad (4)$$

$$(b = n \wedge m \geq 0) \rightarrow [\overline{\mathrm{d}(m)}](b < n + m) \qquad (5)$$

Note that we interpret not–withdrawal (m) as a withdrawal of less than m (including depositing any amount of money). This is a choice that is straightforward in this example. In general we may consider other choices.

Let $n, m \geq 0$. We have to show that $(1) + (2)$ semantically imply

Proposition 5.3

$$(b = n \wedge n - m < 0) \rightarrow \hat{F}(\mathrm{w}(m))$$

Proof

$$(b = n \wedge n - m < 0) \rightarrow$$
$$\rightarrow [\mathrm{w}(m)](b \leq n - m)$$

from Specif. 5.2.2

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)](b < 0)$$
$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]\neg(b \geq 0)$$
$$\Box(\neg(b \geq 0) \rightarrow V)$$

from Specif. 5.2.1 and Def. 5.1 b

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]V$$

we will refer to this formula as (6)

$$(b = n \wedge n - m < 0) \rightarrow \hat{F}(\mathrm{w}(m))$$

from Def. 3.1.1

■

In addition, we can derive the constraint about undoing the negative account balance

Proposition 5.4

$$(b = n \wedge n < 0) \rightarrow \hat{O}(\mathrm{d}(-n))$$

Proof

$$(b = n \wedge -n \geq 0) \rightarrow [\overline{\mathrm{d}(-n)}](b < n + (-n))$$

from Specif. 5.2.5

$$\Box(\neg(b \geq 0) \rightarrow V)$$

from Specif. 5.2.1 and Def. 5.1 b

$$(b = n \wedge -n \geq 0) \rightarrow [\overline{\mathrm{d}(-n)}]V$$

■

We may now wonder whether with negative balance it is obligatory or permitted to deposit more than not simply $-n$. Let's begin with permission.

Let $m > -n$. Does the following formula hold?

$$b = n \wedge n < 0 \rightarrow \hat{P}(d(m)) \qquad (7)$$

Perhaps somewhat surprisingly it does not without the addition of more information.

We have to make a few assumptions, one of which is not plausible without notions that we will present in the following section about refinement (cfr. As. 6.10 and As. 6.11).

Furthermore, we cannot derive the following

$$b = n \wedge n < 0 \rightarrow [\overline{\mathrm{d}(m)}](b < n + m)$$

This blocks the derivation of $\hat{O}(d(m))$. The reason is, of course, that from $b < n + m$ we now cannot infer that $b \leq 0$ and, hence, we cannot apply Specif. 5.2.1 in order to conclude $\hat{O}(d(m))$. This is as one may expect. But, rather surprisingly, we cannot infer

$$(b = n \wedge n < 0) \rightarrow \neg\hat{O}(d(m))$$

either. After the refinement of the following section, we shall show how this can be repaired.

6 Refinement

The refinement concerns the formalization of the violation atom. The main idea is that of indexing atoms. In particular, we will consider two different form of indexing. The first one relates a norm and the relative violation to a certain piece of legislation where the norm is considered. The second one considers violation with relation to the norm that has been violated.

6.1 Refinement I

We divide the space of norms in classes the typical representant of which is i. We say that two norms are of the same type i if and only if the relative violations have both the form V_i. As already pointed out, index i refers to a piece of legislation, which we do not specify further, for instance, a paragraph of a code of laws.

The reason for adopting such device is more of conceptual than technical character. We can indeed easily show that derivation of Prop. 5.3 has not to be altered in order to cope with the indexed violation. The justification of our choice is hence to be sought elsewhere. In reduced SDL, we speak of violation in a very general way, that is, when we speaking of non fulfilment of a norm we say that the whole normative system to which the norm belongs has been violated. This is so only by way of approximation, since we know exactly where the violation has arisen. So that relating the violation to the relevant legal trespassing seems more adherent to the reality than not making use of an all–purpose violation atom.

Note that under this rewriting of violation, for every norm (obligations and prohibitions) of type i, we have a permission of the same type. This depends upon the validity in reduced SDL of the equivalence $\Box(\neg p \rightarrow V_i) \equiv \neg\Diamond(\neg p \wedge \neg V_i)$. This implies that the piece of legislation to which i refers not only have to state what norms fall under ought–to–be and what under ought–to–do, but also what is permitted (or compatible) with these norms.

This is not the only shortcoming from which indexing suffers. The real problem is exactly that both ought–to–do and ought–to–be have to be contemplated in the same, say, paragraph of the code and possibly to be put in relation. This device does not explain why the same piece of legal text can put "it ought to be that p" and "it ought to be brought about that p" in relation, without maintaining that the text indeed refer to both ought to do and ought to be, and thus that either the lawgiver has issued two different norms under the same paragraph or he has assumed the existence of a criterion for relating them. But whether the latter exists and how it works is exactly what we are interested in. That is in short why we may not be satisfied with this kind of indexing.

6.2 Refinement II

Instead of relating violations to the norms of which they are violation by mentioning the relevant piece of law where the norm is printed, we may choose the straightforward solution of associating the violation to its related norm. That is, we propose to distinguish violation states according to the cause of the violation, i.e., if α is obligatory ($\hat{O}\alpha$), then we say that refraining from performing α leads to a violation of $\hat{O}\alpha$. We represent this violation by the atom $V_{\hat{O}\alpha}$ where $\hat{O}\alpha$ is called a violation flag.

An important reason for adopting this device is that it provides the possibility to undo the violation. Since we would not want to undo a violation caused by withdrawing too much money from an account by, say, returning an overdue book to the library, we refine V so that it indicates the cause of the violation and consequently the way in which the violation is to be undone.

Using flagged violations, applying Meyer reduction to Prop. 5.3 would have yielded

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]V_{\hat{F}(\mathrm{w}(m))} \qquad (6')$$

Now, (6') is different from (6) since in the latter the violation refers to Specification 5.2.1:

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]V_{O(b \geq 0)} \qquad (6)$$

In order to derive Prop. 5.3 from specifications (1) + (2), we have to relate the two violation in a plausible way.

Firstly, we adopt Segerberg's δ–operator (cf. [Seg89]) to express consequences of actions. We simply recall the intuitive meaning of this operator:

δp = "a choice from all actions α that bring it about that p (make p true)".

Secondly, we introduce the notion of involvement between actions.

Definition 6.1 *Let α and β be actions, ϕ a state of affairs, we define "α involves β" ($\alpha \gg \beta$) as follows:*

$$\alpha \gg \beta \quad \text{iff} \quad [\![\alpha]\!]_R \subseteq [\![\beta]\!]_R$$

From definition 6.1, it follows that

Proposition 6.2

$$\alpha \gg \beta \Rightarrow \hat{F}(\beta) \rightarrow \hat{F}(\alpha)$$

Proof From Def. 6.1 it follows that $[\beta]\phi \rightarrow [\alpha]\phi$, which by substitution (V for ϕ) yields Prop. 6.2. ■

and also that

Proposition 6.3

$$[\alpha]p \vdash \alpha \gg \delta p$$

Proof By definition α belongs to $[\![\delta p]\!]_R$, and hence $[\![\alpha]\!]_R \subseteq [\![\beta]\!]_R$, which yields the theorem. ■

If we consider unflagged violation, we can easily show that semantically the following are validities:

Proposition 6.4

$$[\delta p]p$$

Proof Every performance of δp leads to a world where p holds, i.e., $\forall w'(w' \in [\![\delta p]\!]_R(w) \Rightarrow w' \models p)$, hence the thesis. ■

Proposition 6.5

$$Fp \vdash \hat{F}(\delta p)$$

Proof In every world accessible by performing δp both p and $p \supset V$ holds, hence V and $\hat{F}(\delta p)$. ■

If we assume flagged violation Prop. 6.5 may be proved only by defining dependence relations among flags. The reason is that we can easily prove that

$$Fp \vdash [\delta p]V_{\hat{F}\delta p} \qquad (6.5')$$

but not that

$$[\delta p]V_{Fp} \rightarrow [\delta p]V_{\hat{F}\delta p} \qquad (*)$$

Of course, if we have a constraint of the kind

$$V_{Fp} \rightarrow V_{\hat{F}\delta p} \qquad (9)$$

formula (*) would be easily proved by applying necessitation to (9) and consequently modus ponens to axiom AK (i.e., to $[\delta p](V_{Fp} \rightarrow V_{\hat{F}\delta p}) \rightarrow ([\delta p]V_{Fp} \rightarrow [\delta p]V_{\hat{F}\delta p})$). Now, formula (9) has no clear interpretation in our system, since V_{Fp} should represent a violation of the normative system to which Fp belongs (or, considering the simpler formalization proposed by Meyer and Wieringa in, e.g., [WM], the system in which p is normed) and (9) would consequentely mean that whenever the normative system to which Fp belongs has been violated, then so it has also the system to which $\hat{F}\delta p$ belongs. We have of course to explain how the two systems may be one and the same system.

The problem, however, is that we may not assume relations between violations as more primitive than those norms to which the violations refer. That in our first, simpler system we could avoid this difficulty, shows only that we had to be careful in assuming the existence of essential logical relations between objects (in this case between "ought–to–do" and "ought–to–be") but not that these relations are detectable only at an appropriate level of abstraction, with that implying that the flagged system is not abstract enough.

The most valuable solution is according to us that of taking Proposition 6.5 as an additional assumption. That is to say, accepting that philosophical position outlined at the end of section 1, where ought–to–do was seen as a means in order to realize the ideality espressed by ought–to–be.

In a certain sense, the fact that Proposition 6.5 cannot be derived in the refined system is not surprising. Section 1 should have already shown that any decision about the relations between ought–to–do and ought–to–be cannot be simply taken by logic. Of course, the derivability of Proposition 6.5 would have been a refutation of all that.

Propositions 6.4 and 6.3 are still derivable in the flagged system, since there no flagged violation needs to occur in their proof.

The importance of having something like Proposition 6.5 is, of course, that it allows to bring it about the desidered derivation (i.e., from Specif. 5.2.1, to derive the statement (2.) of section 1).

Assumption 6.6

$$Fp \vdash \hat{F}(\delta p)$$

Definitions 5.1 and 3.1.1 take now the form

Definition 6.7

$$Fp = \Box(p \rightarrow V_{Fp}) \qquad (a)$$

$$Op = \Box(\neg p \rightarrow V_{Op}) \quad \text{(b)}$$

$$\hat{F}(\alpha) = [\alpha]V_{\hat{F}(\alpha)} \quad \text{(c)}$$

To them we add the standard definition

Definition 6.8

$$Op = F\neg p$$

We begin now to perform the derivation of Prop. 5.3, though, as we will see, this will show itself still impossible.

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]\neg(b \geq 0)$$

as before

$$F\neg(b \geq 0)$$

from Specif. 5.2.1 and Def. 6.8)

$$\Box(\neg(b \geq 0) \rightarrow V_{F\neg(b\geq 0)})$$

from Def 6.7 a

$$(b = n \wedge n - m < 0) \rightarrow [\mathrm{w}(m)]V_{F\neg(b\geq 0)}$$

$$(b = n \wedge n - m < 0) \rightarrow (\mathrm{w}(m) \gg \delta(V_{F\neg(b\geq 0)}))$$

from Prop. 6.3

$$(b = n \wedge n - m < 0) \rightarrow (\hat{F}(\delta(V_{F\neg(b\geq 0)})) \rightarrow \hat{F}(\mathrm{w}(m)))$$

from Prop. 6.2

To get the derived result we need

$$\hat{F}(\delta(V_{F\neg(b\geq 0)}))$$

however, this step cannot be performed by applying (As.6.6).

A possible solution to this impasse is considering the relation between the violation of a particular norm and what we might call a generic violation of the normative system to which that particular norm belongs. The idea is that atom V_x, where x represents an arbitrary norm, does not only express the violation of a norm belonging to a given normative system S but, more generally, also a violation within that very system. We can express that by saying that violation of a norm of S is a sufficient condition for the violation of S itself. Formally, we may represent this fact as follows

$$\text{(8)} \qquad V_x \rightarrow V_{FV}$$

where we define V_{FV} as the disjunction of all the violations that can take place within a given normative system (here, FV means approximatively that it is forbidden to break norms or that violating norms is forbidden).

Definition 6.9 *Let x_i, $i \in \mathbb{N}$, represents a generic norm (obligation or prohibition) of a given normative system S. We define the following*

$$V_{FV} =_{\mathrm{def}} \bigvee_i V_{x_i}$$

By logic, we may indeed prove that $F(V_{FV})$ (cfr. remark at the end of section 2) but not that $F(V_x)$ is a theorem of the calculus. The latter result is exactly what we need in order to apply As. 6.6 and Def. 6.9 enables us to derive it.

Since in SDL the following equivalence holds

$$F(b \vee q) \equiv Fb \wedge Fq$$

it is straightforward to conclude that

$$F(V_{FV}) \rightarrow F(V_{F\neg(b\geq 0)})$$

and hence, being able to apply As. 6.6 (remember $F(V_{FV})$ is a theorem of the calculus), we obtain

$$\hat{F}(\delta(V_{F\neg(b\geq 0)}))$$

and eventually by MP and PC

$$(b = n \wedge n - m < 0) \rightarrow \hat{F}(\mathrm{w}(m))$$

■

We can now go back to the problem sketched at the end of the previous section, i.e., whether formula (7) is derivable in our system. We observed that without more information it could not be derived.

If we consider a generic norm, say, Op, we see that an intuitive necessary and sufficient condition for V_{Op} to hold is exactly that $\neg p$ is the case.

Assumption 6.10

$$\Box(\neg p \leftrightarrow V_{Op})$$

Furthermore, if we assume that we can always deposit, we have to admit a further assumption.

Assumption 6.11

$$\langle d(m)\rangle\top$$

Before proving formula (7), we need also to give a more exact definition of permission than not that given as Def. 3.1.2.

Definition 6.12

$$\hat{P}\alpha = \neg[\alpha]V_{FV}$$

Now we can prove formula (7).

Proposition 6.13

$$b = n \wedge n < 0 \rightarrow \hat{P}(d(m)) \quad (7)$$

Proof

$$\Box(b \geq 0 \leftrightarrow \neg V_{O(b \geq 0)})$$

from As. 6.10 and Specif. 5.2.1

$$b = n \wedge n < 0 \rightarrow [d(m)](b \geq 0)$$

from Specif. 5.2.4

$$b = n \wedge n < 0 \rightarrow [d(m)]\neg V_{O(b \geq 0)}$$

$$b = n \wedge n < 0 \rightarrow \langle d(m)\rangle \neg V_{O(b \geq 0)}$$

from As. 6.11

■

Remark 6.14 *The fact that we have to assume extra conditions for deriving (7) should not be really surprising, think, e.g., of a situation where $\Box V$ would hold, we would have as a consequence that in that situation not anything would be permitted!*

We conclude with a few annotations about the non derivability of formula

$$(b = n \wedge n \leq 0) \rightarrow \neg\hat{O}(d(m))$$

already considered at the end of section 5.

In order to prove it we should be able to prove the following (simply apply As. 6.10)

$$(b = n \wedge n \leq 0) \rightarrow \langle\overline{d(m)}\rangle(b \geq 0) \quad (10)$$

We know that $[\overline{d(m)}](b \leq n + m)$ (with $n + m$ positive) holds. However, to derive (10), we need to know whether positive values of b can indeed be obtained. For this purpose we have to refine our specification by adding the following

Specification 6.15

$$b = n \wedge m \geq 0 \rightarrow \langle d(m)\rangle(b = k),\ for\ all\ k \geq n \quad (1)$$

$$b = n \wedge m \geq 0 \rightarrow \langle\overline{d(m)}\rangle(b = k),\ for\ all\ k < n \quad (2)$$

7 Conclusion

We have shown that the distinction between ought–to–do and ought–to–be is relevant for at least some kinds of system specification and that, maintaining a certain degree of generality, it is possible to express both kinds of norms in one system without reducing one of them to the other or even assuming the existence of specific relations between them. We have seen that by increasing the expressive power of the language by flagging violation atoms, we have to state relations connecting the violation of static constraints with that of dynamic ones. In the solution we have presented, we have chosen to relate ought–to–be to ought–to–do considered in its broadest sense by an assumption (As. 6.6). We still have to investigate how this assumption can be fit into our semantic framework.

Current research includes also a characterization of negated actions in the perspective of an application of involvement also to obligations and a study of the logical properties of the system sketched above (consistency and completeness).

References

[And67] A.R. Anderson. The formal analysis of normative systems. In N. Rescher, editor, *The Logic of Decision and Action*, pages 147–213. University of Pittsburgh Press, 1967.

[Åqv88] Lennart Åqvist. *Introduction to deontic logic and the theory of normative systems.* Bibliopolis, Napoli, 1987 [1988].

[BMT87] C. Biagioli, P. Mariani, and D. Tiscornia. ESPLEX: A rule and conceptual based model for representing statutes. In *The First International Conference on Artificial Intelligence and Law*, pages 240–251. ACM, May 1987.

[Cas60] Hector-Neri Castañeda. Obligation and modal logic. *Logique et Analyse*, 3:40–48, 1960.

[Cas70] Hector-Neri Castañeda. On the semantics of the ought–to–do. *Synthese*, 21:449–468, 1970.

[Cas81] Hector-Neri Castañeda. The paradoxes of deontic logic: the simplest solution to all of them in one fell swoop. In Risto Hilpinen, editor, *New Studies in Deontic Logic*, pages 37–85. D. Reidel, Dordrecht – Boston – London, 1981.

[Che80] Brian F. Chellas. *Modal Logic. An Introduction*. Cambridge University Press, Cambridge, 1980.

[Coe] J. Coenen. Top-down development of layered fault-tolerant systems and its problems —a deontic perspective. In J.-J.Ch. Meyer and R.J. Wieringa, editors, *Deontic Logic* *in Computer Science: Normative System Specification*. Wiley. to be published.

[Daw59] E. E. Dawson. A model for deontic logic. *Analysis*, 19:73–78, 1959.

[DM90] F.P.M. Dignum and J.-J.Ch. Meyer. Negations of transactions and their use in the specification of dynamic and deontic integrity constraints. In M.Z. Kwiatkowska, M.W. Shields, and R.M. Thomas, editors, *Semantics for Concurrency*, pages 61–80. Springer, 1990.

[Gar86] J. L. A. García. The *tunsollen*, the *seinsollen*, and the *soseinsollen*. *American Philosophical Quarterly*, 23:267–276, 1986.

[Gea81] Peter Geach. Whatever happened to deontic logic? *Philosophia*, 11:1–12, 1981.

[GMP89] J. Glasgow, G. MacEwen, and P. Panangaden. Security by permission in databases. In C.E. Landwehr, editor, *Database Security II: Status and Prospects*, pages 197–205. North-Holland, 1989. Results of the IFIP WG 11.3 Workshop on Database Security (October 1988), Kingston, Ontario, Canada.

[Han71] Bengt Hansson. An analysis of some deontic logics. In Risto Hilpinen, editor, *Deontic Logic: Introductory and Systematic Readings*, pages 121–147. D.Reidel, Dordrecht, 1971.

[Har84] D. Harel. Dynamic logic. In D.M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic II*, pages 497–604. D. Reidel, 1984.

[KL88] S.O. Kimbrough and R.M. Lee. Logic modeling: A tool for management science. *Decision Support Systems*, 4:3–16, 1988.

[KM87] S. Khosla and T.S.E. Maibaum. The prescription and description of state based systems. In B. Banieqbal, H. Barringer, and A. Pnueli, editors, *Temporal Logic in Specification*, pages 243–294. Springer, 1987. Lecture Notes in Computer Science 398.

[Lee88a] R.M. Lee. Bureaucracies as deontic systems. *ACM Transactions on Office Information Systems*, 6:87–108, 1988.

[Lee88b] R.M. Lee. A logic model for electronic contracting. *Decision Support Systems*, 4:27–44, 1988.

[Mey88] J.-J.Ch. Meyer. A different approach to deontic logic: Deontic logic viewed as a variant of dynamic logic. *Notre Dame Journal of Formal Logic*, 29:109–136, 1988.

[Mey89] J.-J.Ch. Meyer. Using programming concepts in deontic reasoning. In R. Bartsch, J.F.A.K. van Benthem, and P. van Emde Boas, editors, *Semantics and Contextual Expression*, pages 117–145. FORIS publications, Dordrecht/Riverton, 1989.

[ML85] N.H. Minsky and A.D. Lockman. Ensuring integrity by adding obligations to priviliges. In *8th IEEE International Conference on Software Engineering*, pages 92–102, 1985.

[MW] J.-J.Ch. Meyer and R.J. Wieringa. Deontic logic: A concise overview. In J.-J.Ch. Meyer and R.J. Wieringa, editors, *Deontic Logic in Computer Science: Normative System Specification*. Wiley. to be published.

[Seg77] Krister Segerberg, editor. *An introduction to modal logic by E.J. Lemmon (in coll. with D. Scott)*, volume 11 of *APQ Monograph Series*. Basil Blackwell, Oxford, 1977.

[Seg89] Krister Segerberg. Bringing it about. *Journal of Philosophical Logic*, 18:327–347, 1989.

[SH70] Wilfrid Sellars and John Hospers, editors. *Readings in Ethical Theory*. Prentice–Hall, Inc., Englewood Cliffs, NJ, 1970. 2nd ed.

[Sta80] R. Stamper. LEGOL: Modelling legal rules by computer. In B. Niblett, editor, *Com[puter Science and Law*, pages 45–71. Cambridge University Press, 1980.

[vW63] Georg Henrik von Wright. *Norm and Action*. Routledge and Kegan Paul, London, 1963.

[WM] R.J. Wieringa and J.-J.Ch. Meyer. Applications of deontic logic in computer science: A concise overview. In J.-J.Ch. Meyer and R.J. Wieringa, editors, *Deontic Logic in Computer Science: Normative System Specification*. Wiley. to be published.

[WM91] R.J. Wieringa and J.-J.Ch. Meyer. Actors, actions, and initiative in normative system specification. Technical Report IR-257, Faculty of Mathematics and Computer Science, Vrije Universiteit, Amsterdam, October 1991. To be published, *Annals of Mathematics and Artificial Intelligence*.

[WMW89] R.J. Wieringa, J.-J. Ch. Meyer, and H. Weigand. Specifying dynamic and deontic integrity constraints. *Data and Knowledge Engineering*, 4:157–189, 1989.

[WWMD91] R.J. Wieringa, H. Weigand, J.-J. Ch. Meyer, and F. Dignum. The inheritance of dynamic and deontic integrity constraints. *Annals of Mathematics and Artificial Intelligence*, 3:393–428, 1991.