Abstract Normative Systems: Semantics and Proof Theory

Silvano Colombo Tosatto
SnT, University of Luxembourg
Luxembourg

Guido Boella
University of Turin
Italy

Leendert van der Torre
University of Luxembourg
Luxembourg

Serena Villata
INRIA
France

Abstract

In this paper we introduce an abstract theory of normative reasoning, whose central notion is the generation of obligations, permissions and institutional facts from conditional norms. We present various semantics and their proof systems. The theory can be used to classify and compare new candidates for standards of normative reasoning, and to explore more elaborate forms of normative reasoning than studied thus far.

Introduction

The modal logic framework has been the standard for normative reasoning since Von Wright (1951) wrote his influential paper "deontic logic" sixty years ago. It has been plagued by many counterintuitive or "paradoxical" theorems, it has been extended, for example with action and time (von Wright 1963) or conditionals (Hansson 1969), and adapted, for example to minimal deontic logic (Chellas 1980). Makinson (1999) criticizes the hegemony of modal logic and proposes an alternative iterative approach. In the handbook of deontic logic (Gabbay et al. To appear a), which is currently in preparation, the classical modal logic framework is mainly confined to the historical chapter. A chapter presents the alternatives to the modal framework, and three chapters discuss concrete approaches, namely input/output logic (Makinson and van der Torre 2000; 2001; 2003a), the imperativist approach (Hansen 2006), and the algebraic conceptual implication structures or cis approach (Lindahl and Odelstad 2003). There are also other candidates for a new standard, such as nonmonotonic logic (Horty 1993) or deontic update semantics (van der Torre and Tan 1999).

With this rise of candidates for new standards for normative reasoning, the need emerges to have a common framework to compare and analyze these new proposals. Lindahl and Odelstad's cis approach seems to be the most abstract, but it does not seem to cover the wide range of deontic logics covered by the input/output logic framework or the imperativist approach, which are extensions of propositional logic and provide a range of proof systems. We therefore study the following question:

Research question. How to build an abstract framework for normative reasoning, of which the cis, input/output, and imperativist approaches are instances?

The success criterion of our research question is that it covers a wide range of topics discussed in the first volume of the handbook of deontic logic (Gabbay et al. To appear a), namely:

Conditionals and rules, we do not restrict ourselves to monadic approaches, but also capture dyadic ones.

Contrary to duties, well known from deontic paradoxes (Chisholm 1963; Forrester 1984), must be represented without generation of counterintuitive conclusions.

Dilemmas and defeasibility, if two norms give contradictory advice, then we must to be able to reason with these two alternatives and their consequences.

Permissive norms must be represented explicitly and used to generate permissions.

Constitutitive norms, such as count-as conditionals, must be represented explicitly, and used to generate institutional facts (Boella and van der Torre 2004).

Makinson and van der Torre (2000) do not to introduce a single system of normative reasoning, but a framework for many systems. We follow this tradition. Our methodology is both semantic and proof theoretic. The starting point is a combination of Lindahl and Odelstad's cis approach for the high level of abstraction, and Makinson and van der Torre's input/output logic for the proof systems. Whereas Makinson and van der Torre start from propositional logic, in this paper we stay at the abstract level. The proof systems are based on acceptance and redundancy (van der Torre and Tan 1999; van der Torre 2010). We focus on the conceptual framework and the results, and leave the proofs to a technical report.

We do not address the topics discussed in the second volume of the handbook of deontic logic (Gabbay et al. To appear b), such as time, action, norm change, epistemic norms, games, *etc*.

The layout of this paper is as follows. We address the five issues to be incorporated, where we deal with contrary-to-duty and dilemmas in the same section. We introduce semantics, proof systems, and completeness results, illustrated by examples.

Example

Abstract norms can refer to propositions, like "there should not be a fence," modal propositions like "you should know whether there is a fence," or actions like "you should not build a fence." An advantage of abstract norms is that we can reason even when we do not detail the content of norms, as we discuss below. We illustrate the representation, semantics and proof systems of abstract normative systems by an example of Cottage Regulations visualized in Figure 1.

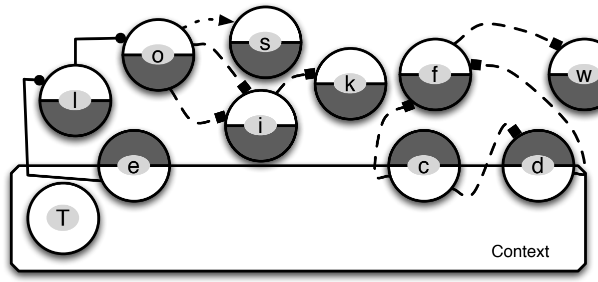


Figure 1: Cottage Regulations.

Representation

This figure must be read as follows. A circle visualizes two atomic elements, for example the light part of f represents that there is a fence, and the dark part represents that there is no fence. An arrow is a *conditional* norm, with straight arrows representing constitutive norms like "a legal cottage contract (l) counts as owning the cottage (o)," dashed lines representing regulative norms like "if you have an id card (i), then you must keep it with you (k)," and dotted lines are permissive norms like "if you own a cottage (o), then you may sell it (s)." Finally, the example contains a context of elements that are assumed to hold, namely the contract to buy the cottage is signed electronically (e), there is a cottage (c) and there is dog (d). Note that if there is a cottage (c), then there should *not* be a fence (f), since the arrow hits the dark side of the fence element.

Constitutive norms define legal concepts like ownership, which are also called institutional facts to distinguish them from brute facts like the existence of a fence, or the physical act of signing a contract. The example contains two constitutive norms: "an electronically signed cottage contract (e) counts as a legal cottage contract (l)" and "a legal cottage contract counts as owning the cottage (o)." The regulations prescribe that a person living in a cottage (c) must follow these norms: "dogs (d) are not allowed," "the cottage should not have a fence (f)," "if there is a dog, then there must be a fence," "if there is a fence, then it must be white (w)," "if someone owns a cottage (o), then he (or she) must have an id card (i)," "if someone does not own a cottage, then he (or she) must have an id card" and "if someone has an id card, then he (or she) must keep it with him (or her) (k)." The example presents only one permissive norm: "the owner of a cottage is allowed to sell it (s)."

Reasoning: semantics

The meaning of a normative system is represented by the institutional facts, obligations and permissions which can be generated from the constitutive, regulative and permissive norms respectively. A semantics generates the institutional fact that there is a legal contract (l), and the obligation there should be no fence (f), directly from the context using a single norm. This is known as factual detachment.

Transitivity Semantics may disagree on the amount of transitivity they allow. For example, can the institutional fact that there is a legal contract (l) itself be used to generate also the institutional fact that we own the cottage (o), and can we use the institutional fact that we own the cottage (o) generate the strong permission to sell the house (s)? Likewise, if there is a dog, then there is the obligation to have a fence (f), and can we generate from this obligation that there must be a white fence (w)? The latter generation is known as deontic detachment.

Reasoning by cases A second distinction is whether a semantics supports reasoning by cases. Consider the id card (i) in Figure 1. It becomes obligatory to have an id card if we own the cottage (o) as well as if we do not own the cottage. Using reasoning by cases we can generate an obligation to have an id card in every context.

Dilemmas and defeasibility The cottage (c) and the dog (d) are part of the context and make the existence of a fence (f) both obliged and forbidden. In some systems this would lead to explosion in the sense that everything becomes obligatory. With multiple obligation sets, we can use subsets of the norms to derive sets of obligations without complementary pairs. In this case we have on the one hand that there must be a fence (because there is a dog), and it must be a white fence, and on the other hand that it is forbidden to have a fence (since there is a cottage).

Violations and contrary-to-duty If we consider a case where there is a fence, and thus (f) would be part of the context, then we would have the obligation to not have a fence just outside the context which represents a violation. The obligation to have a white fence (w) must be fulfilled in case the obligation not to have a fence is violated.

Reasoning: proof system

At first sight, it may seem that the generation of obligations from norms itself is a proof system. However, how to distinguish different ways to generate them, and how to prove soundness and completeness with such a proof theory?

The proof systems in this paper are based on equivalence of abstract normative systems. If adding the norm that always there should be an id card (i), by adding a dashed arrow from ($\top$) in the context to (i), does not change the semantics, in other words if we can still generate the same obligations, permissions and institutional facts, then we derive the norm. Likewise if the addition of the constitutive norm that "if we sign the contract (e), then we own the cottage (o)" does not change the semantics, then we derive this norm.

Rules and Conditionals

Abstract normative system

Makinson and van der Torre (2000) represent rules by ordered pairs (a, x), where the body a is thought of as an input, representing some condition or situation, and the head x is thought of as an output, representing what the rule tells us to be desirable, obligatory or whatever in that situation. They quickly move to what they call the logical level, where the universe contains all sentences of a logical language, and in particular they study propositional logic.

We study their abstract level, to which we add two ideas. First, each element in the universe comes with its "anti-element." This is the minimal extension to represent violations, namely elements in the input whose anti-element is in the output. Second, there is an element in the universe called $\top$, contained in every context. We consider only a finite universe in this paper.

Definition 1 (Universe L) *Given a* finite *set of atomic elements E, the universe L is $E \cup \{\sim e \mid e \in E\} \cup \{\top\}$. For $e \in E$, let $\overline{a} = \sim e$ iff $a = e$, $\overline{a} = e$ iff $a = \sim e$, and undefined iff $a = \top$.*

An abstract normative system is a directed graph, and a context is a set of nodes of the graph containing $\top$.

Definition 2 (ANS $\langle L, N\rangle$) *An abstract normative system ANS is a pair $\langle L, N\rangle$ with $N \subseteq L \times L$ a set of conditional norms, and a context $A \subseteq L$ is a subset of the universe such that $\top \in A$.*

In a context, an abstract normative system generates or produces an obligation set, a subset of the universe, reflecting the obligatory elements of the universe.

Definition 3 (Deontic operation $\bigcirc$) *The deontic operation $\bigcirc$ is a function from an abstract normative system $\langle L, N\rangle$ and a context A to a subset of the universe $\bigcirc(\langle L, N\rangle, A) \subseteq L$. Since L is always clear from context, we write $\bigcirc(N, A)$ for $\bigcirc(\langle L, N\rangle, A)$.*

Semantics

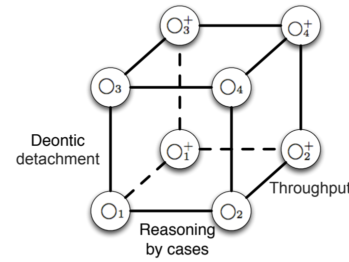
Simple-minded output or $\bigcirc_1$ is Makinson and van der Torre's minimal system. Basic output or $\bigcirc_2$ allows for reasoning by cases, which now means that if something is obligatory in the context of a and its complement $\overline{a}$, then it is obligatory also in the minimal context. Reusable output or $\bigcirc_3$ allows for deontic detachment, which now corresponds to iteration of the rules. Throughput or $\bigcirc_i^+$ allows for identity. All possible combinations lead to eight input/output operations.

Definition 4 (Eight deontic operations) *A context $A \subseteq L$ is complete if for all $e \in E$, it contains either e or $\overline{e}$ (or both).*

$\bigcirc_1(N, A) = N(A) = \{x \mid (a, x) \in N \text{ for some } a \in A\}$
$\bigcirc_2(N, A) = \cap\{N(V) \mid A \subseteq V, V \text{ complete}\}$
$\bigcirc_3(N, A) = \cap\{N(B) \mid A \subseteq B \supseteq N(B)\}$
$\bigcirc_4(N, A) = \cap\{N(V) \mid A \subseteq V \supseteq N(V), V \text{ complete}\}$
$\bigcirc_i^+(N, A) = \bigcirc_i(N \cup \{(a, a) \mid a \in L\}, A)$

Equivalently, we can define $\bigcirc_3(N, A)$ as $N(B)$ where B is the *smallest* set containing A and closed under N, i.e. $A \subseteq B \supseteq N(B)$. Moreover, to emphasize symmetry, we can define $\bigcirc_1(N, A)$ equivalently as $\cap\{N(B)|A \subseteq B\}$.

Proposition 1 (Inclusion among deontic operations) *We have $\bigcirc_1(N, A) \subseteq \bigcirc_2(N, A) \subseteq \bigcirc_4(N, A)$ (reasoning by cases), $\bigcirc_1(N, A) \subseteq \bigcirc_3(N, A) \subseteq \bigcirc_4(N, A)$ (deontic detachment), and $\bigcirc_i(N, A) \subseteq \bigcirc_i^+(N, A)$ (throughput).*



The following example illustrates that the eight deontic operations are all distinct.

Example 1 *Let $N = \{(o, i), (\overline{o}, i), (i, k)\}$, as visualized in Figure 2. Notice that we are considering a fragment of the example represented in Figure 1.*

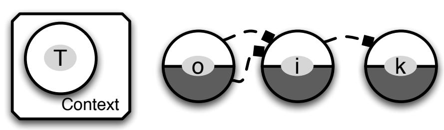


Figure 2: Both o and $\overline{o}$ make i obligatory (Example 1).

The tables compare the obligation sets for the two contexts $A = \{\top\}$ and $A = \{o, \top\}$ for all eight deontic operations.

A	$\bigcirc_1(N, A)$	$\bigcirc_2(N, A)$	$\bigcirc_3(N, A)$	$\bigcirc_4(N, A)$
$\{\top\}$	$\{\}$	$\{i\}$	$\{\}$	$\{i, k\}$
$\{o, \top\}$	$\{i\}$	$\{i\}$	$\{i, k\}$	$\{i, k\}$

A	$\bigcirc_1^+(N, A)$	$\bigcirc_2^+(N, A)$	$\bigcirc_3^+(N, A)$	$\bigcirc_4^+(N, A)$
$\{\top\}$	$\{\}$	$\{i\}$	$\{\}$	$\{i, k\}$
$\{o, \top\}$	$\{o, i\}$	$\{o, i\}$	$\{o, i, k\}$	$\{o, i, k\}$

E.g., $\bigcirc_2(N, \{\top\}) = \{i\}$ can be calculated by taking all 64 complete contexts V, of which the ones without complementary pairs are given in the table below, and calculate the associated $N(V)$, which gives $\{i\}$ as their intersection.

V	$N[V]$	V	$N[V]$
$\{o, i, k, \top\}$	$\{i, k\}$	$\{\overline{o}, i, k, \top\}$	$\{i, k\}$
$\{o, i, \overline{k}, \top\}$	$\{i, k\}$	$\{\overline{o}, i, \overline{k}, \top\}$	$\{i, k\}$
$\{o, \overline{i}, k, \top\}$	$\{i\}$	$\{\overline{o}, \overline{i}, k, \top\}$	$\{i\}$
$\{o, \overline{i}, \overline{k}, \top\}$	$\{i\}$	$\{\overline{o}, \overline{i}, \overline{k}, \top\}$	$\{i\}$

$\bigcirc_3(N, \{o, \top\}) = \cap\{N(B)|B \in \{\{o, i, k, \top\}, \ldots L\}\} = \{i, k\}$, *since the only B containing a and satisfying the constraint that $N(B) \subseteq B$ are $B = \{o, i, k, \top\}$ and its supersets.*

Proof systems

A pair (a, x) of formulae is derivable using rules from a set N of such pairs iff (a, x) is in the least set that includes N, and is closed under the rules. In the particular case of simple-minded output, we use the core rule *Strengthening of the Input* (SI) determining an operation d_1. In the case of basic output, we add *Disjunction* (OR) to it, obtaining an operation d_2. In the case of reusable output, we add *Transitivity* (T) instead of (OR), obtaining an operation d_3. In the case of reusable basic output, we add both (OR) and (T) to the duo, obtaining an operation d_4. Finally, for each d_i we add *Identity* (ID) to obtain the throughput variant d_i^+.

Definition 5 (Derivability d**)** *$d_1(N)$ ($d_2(N)$ / $d_3(N)$ / $d_4(N)$) is the smallest set containing N and closed under SI (SI+OR / SI+T / SI+OR+T). Moreover, $d_i^+(N)$ is the smallest set containing N and closed under the rules of d_i together with ID:*

$$\frac{(\top, x)}{(a, x)}SI \qquad \frac{(a, x)\ (\overline{a}, x)}{(\top, x)}OR \qquad \frac{(a, x)\ (x, y)}{(a, y)}T \qquad \frac{}{(a, a)}ID$$

Example 2 *We refine Example 1 by replacing $(\overline{o}, i)$ by $(\overline{o}, z)$ and (z, i) where z means: "to have a legal mail address." Let $N = \{(o, i)(\overline{o}, z)(z, i)(i, k)\}$ and $A = \{\top\}$ as visualized in Figure 3.*

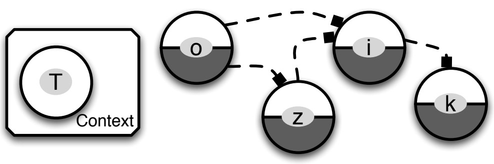


Figure 3: Using $\bigcirc_4$, y is obligatory (Examples 2 and 3).

The following two derivations show $y \in d_4(N, A)$. The non-repetition derivation (i) *uses each premise at most once, and the universal order derivation* (ii) *uses the transitivity rule T only before the disjunction rule OR. Non-repetition and universal order properties are used by Makinson and van der Torre in their completeness proofs.*

(i) *Non-repetition derivation: premises used at most once*

$$\dfrac{\dfrac{(o, x) \qquad \dfrac{(\overline{o}, z) \qquad (z, i)}{(\overline{o}, i)}\,T}{(\top, i)}\,OR \qquad (i, k)}{(\top, k)}\,T$$

(ii) *Universal order derivation: T only before OR*

$$\dfrac{\dfrac{(o, i) \qquad (i, k)}{(o, k)}\,T \qquad \dfrac{\dfrac{(\overline{o}, z) \qquad (z, i)}{(\overline{o}, i)}\,T \qquad (i, k)}{(\overline{o}, k)}\,T}{(\top, k)}\,OR$$

Theorem 1 shows the soundness and completeness of the eight proof systems with respect to the corresponding deontic operations. This completeness result is phrased in an unusual way to deal with the abstract nature of the normative systems, but the deeper structure is similar to more standard completeness results. More precisely, there are two challenges to define a proof system and completeness theorem for an abstract normative system, here as well as in the following sections:

1. Analogously to the completeness result of input/output logic, repeated as Theorem 3 later in this paper, deontic operations based on obligation sets must be related to proof systems based on conditionals. As in input/output logic, the intuition is that $x \in \bigcirc(N, \{a\})$ if and only if $(a, x) \in d(N)$.
2. The deontic operations are defined for a context being a set, whereas the antecedent of a conditional is a single formula. Since we are in an abstract setting, in contrast to the input/output logic framework we cannot use conjunction to join elements of the context.

For the second challenge, we adopt the following idea. In classical logic, it is well known that if a set of formulas S implies a formula ϕ, we may equivalently that S is equivalent to $S \cup \{\phi\}$. For example, this is how a Boolean algebra is defined. This idea has been further explored in dynamic semantics such as deontic update semantics (van der Torre and Tan 1999), where – using the terminology of this paper – a normative system N *accepts* the norm (a, x) if and only if adding the norm (a, x) to N does not change the obligation set, i.e. $\forall A : \bigcirc(N, A) = \bigcirc(N \cup \{(a, x)\}, A)$. In other words, the idea is that *equivalence* of normative systems replaces implication as the basic notion. Likewise, van der Torre (2010) studies when a norm $(a, x) \in N$ is redundant, in the sense that it can be removed from N without changing the obligation set. The same idea is used here to establish the relation between the deontic operations and the proof systems.

Theorem 1 (Completeness) *For all deontic operations $\bigcirc \in \{\bigcirc_i, \bigcirc_i^+ \mid i = 1, \ldots, 4\}$ and corresponding proof systems $d \in \{d_i, d_i^+ \mid i = 1, \ldots, 4\}$ we have:*

$$(a, x) \in d(N) \textit{ iff } \forall A : \bigcirc(N, A) = \bigcirc(N \cup \{(a, x)\}, A)$$

Example 3 (continued from Example 2) *We have again $N = \{(o, i)(\overline{o}, z)(z, i)(i, k)\}$ and $A = \{\top\}$. From $k \in d_4(N, A)$ and the completeness theorem, we can deduce $\forall B : \bigcirc(N, B) = \bigcirc(N \cup \{(\top, k)\}, B)$, and consequently $k \in \bigcirc_4(N, A)$. This can also be verified as follows. From the complete contexts built from $\{o, i, z, k, \overline{o}, \overline{i}, \overline{z}, \overline{k}, \top\}$, only $\{o, i, z, k, \top\}$, $\{o, i, \overline{z}, k, \top\}$, $\{\overline{o}, i, z, k, \top\}$ and their extensions satisfy $N(V) \subseteq V$. $N(V)$ is $\{i, k\}$, $\{i, k\}$ and $\{i, z, k\}$ respectively, and $\bigcirc_4(N, A)$ is their intersection $\{i, k\}$.*

Dilemmas and contrary-to-duties

At least since the work of Horty (1993), nonmonotonic techniques have been used to deal with reasoning in the context of dilemmas, contrary-to-duty reasoning, and defeasible norms.

Dilemmas are two (or more) obligations with contradictory content, like the obligation for a and the obligation for $\overline{a}$.

Contrary-to-duty or secondary obligations (a, x) are in force only in case of *violation* of a primary obligation, e.g., generated using $(\top, \overline{a})$.

Defeasible deontic logic is concerned with violations *and exceptions* (van der Torre 1997; Nute 1997).

A typical consequence of Horty's approach is that there is no longer necessarily a single obligation set, but there may be several of them. We do not have to change the definition of abstract normative system ANS, but we have to change the definition of deontic operation. We write $\odot$ for deontic operations that may have multiple obligation sets.

Definition 6 (Deontic operation $\odot$) *The deontic operation $\odot$ is a function from an abstract normative system $\langle L, N\rangle$ and a context A to a set of subsets of the universe $\odot(\langle L, N\rangle, A) \subseteq 2^L$. We write $\odot(N, A)$ for $\odot(\langle L, N\rangle, A)$.*

The following example illustrates a use of multiple obligation sets.

Example 4 *Let $N = \{(\top, x), (\top, \overline{x}), (x, y), (\overline{x}, \overline{y})\}$ and context $A = \{\top\}$ visualized in Figure 4. The two obligation sets $\{x, y\}$ and $\{\overline{x}, \overline{y}\}$ represent two deontic alternatives.*

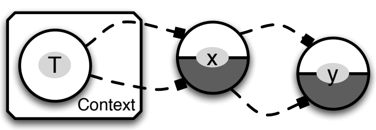


Figure 4: Dilemma with two alternatives (Example 4 and 7).

Horty uses Reiter's default logic, which has the drawback that it is a throughput operation, i.e., it satisfies identity. Makinson and van der Torre define a deontic operation of constrained output via the definition of maximal sets of norms, under a consistency constraint. In our abstract setting, for each deontic operation $\bigcirc$, we define an operation $\odot^*$ with the output constraint that there is no complementary pair in the output (for dilemmas) and an operation $\odot$ with the input/output constraint that there is no complementary pair if we combine the input with the output (for dilemmas and contrary-to-duty reasoning). $\odot^*$ is defined using the $m_i^*(N, A)$ operation, selecting the maximal subsets of N that do not return complementary pairs in the output, and $\odot$ is defined using the $m_i(N, A)$ operation considering both the input and the output. For throughput operations the output and the input/output constraint coincide, but in general this is not the case, leading to twelve distinct operations.

Definition 7 (Twelve deontic operations) *Let $coh(A)$ be true if and only if A does not contain a complementary pair $\{a, \overline{a}\}$, and let $\bigcirc \in \{\bigcirc_i, \bigcirc_i^+ \mid i = 1, \ldots, 4\}$. Given an abstract normative system $\langle L, N\rangle$ and a context A, maximal rules sets $m_i^*(N, A)$ and $m_i(N, A)$ are as follows:*

$$m^*(N, A) = max_{\subseteq}\{N' \subseteq N | coh(\bigcirc(N', A))\}$$

$$m(N, A) = max_{\subseteq}\{N' \subseteq N | coh(A \cup \bigcirc(N', A))\}$$

The deontic operations $\odot^*$ and $\odot$ are as follows:

$$\odot^*(N, A) = \{\bigcirc(N', A) | N' \in m^*(N, A)\}$$

$$\odot(N, A) = \{\bigcirc(N', A) | N' \in m(N, A)\}$$

We illustrate the new operations by several examples. The following example of Möbius strip (Makinson 1999) illustrates the difference between the output and input/output constraint, and that obligation sets do not have to be maximal.

Example 5 (Möbius strip) $N = \{(a, x), (x, y), (y, \overline{a})\}$ *and context $A = \{a, \top\}$,*

- $m^*(N, A) = \{N\}$ *and* $\odot_3^*(N, A) = \{\{x, y, \overline{a}\}\}$,
- $m(N, A) = \{\{(a, x), (x, y)\}, \{(a, x), (y, \overline{a})\}, \{(x, y), (y, \overline{a})\}\}$ *and* $\odot_3(N, A) = \{\{x, y\}, \{x\}, \{\}\}$.

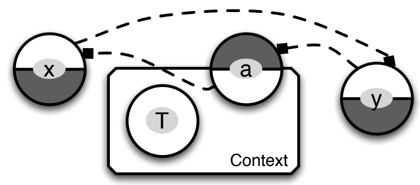


Figure 5: The Möbius strip (Example 5).

(Chisholm 1963)'s paradox illustrates that adding a transitive closure may change the output for $\odot_3$.

Example 6 (Chisholm set) $N = \{(\top, a), (a, t), (\overline{a}, \overline{t})\}$, *and $A = \{\overline{a}, \top\}$ as visualized in Figure 6, where a stands for a man going to the assistance of his neighbors and t for telling them that he will come. We have $\odot_3(N, A) = \{\{\overline{t}\}\}$ and $\odot_3(N \cup \{(\top, t)\}, A) = \{\{\overline{t}\}, \{t\}\}$, thus adding the transitive closure leads to a new obligation set. We do have $\odot_3^*(N, A) = \{\{a, \overline{t}\}, \{a, t\}\}$ and $\odot_3^*(N \cup \{(\top, t)\}, A) = \{\{a, \overline{t}\}, \{a, t\}\}$.*

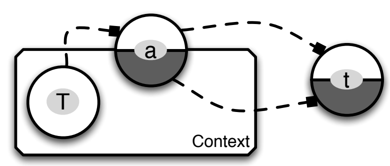


Figure 6: Chisholm's paradox (Example 6).

Moreover, the following example illustrates that also for $\odot_3^*$, adding the transitive closure may change the output.

Example 7 *Consider $N = \{(\top, x), (\top, \overline{x}), (x, y), (\overline{x}, \overline{y})\}$ and $A = \{\top\}$ from Example 4, visualized in Figure 4. We have $\odot_3^*(N, A) = \{\{x, y\}, \{\overline{x}, \overline{y}\}\}$ and $\odot_3^*(N \cup \{(\top, y)\}, A) = \{\{x, y\}, \{\overline{x}, \overline{y}\}, \{\overline{x}, y\}\}$, thus adding the transitive closure adds the third obligation set.*

The following example illustrates operation $\odot_3$.

Example 8 $N = \{(a, x), (x, y), (\overline{x}, y)\}$. *We have for $A = \{a, \top\}$ that $m(N, A) = \{\{(a, x), (x, y), (\overline{x}, y)\}\}$ and $\odot_3 = \{\{x, y\}\}$, for $A = \{a, \overline{x}, \top\}$ that $m(N, A) = \{\{(x, y), (\overline{x}, y)\}\}$ and $\odot_3 = \{\{y\}\}$, and in general it can be shown that we have $\forall A : \odot_3(N, A) = \odot_3(N \cup \{(a, y)\}, A)$.*

Assume $N = \{(a, x), (x, y), (\overline{x}, y), (y, \overline{x})\}$ and $A = \{a, \top\}$. We have $m(N, A) = \{\{(a, x), (x, y), (\overline{x}, y)\},$

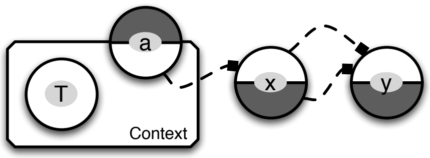


Figure 7: Reasoning by cases (Example 8).

$\{(a,x),(\overline{x},y),(y,\overline{x})\},\{(x,y),(\overline{x},y),(y,\overline{x})\}\}$, *and therefore* $\bigodot_3(N,A)=\{\{x,y\},\{x\},\{\}\}$.

Assume now $N'=\{(a,x),(x,y),(\overline{x},y),(y,\overline{x}),(a,y)\}$ *and* $A=\{a,\top\}$. *We have* $m(N',A)=\{\{(a,x),(x,y),(\overline{x},y),(a,y)\},\{(a,x),(\overline{x},y),(y,\overline{x})\},\{(x,y),(\overline{x},y),(y,\overline{x})\}\}$, $\bigodot_2^*(N',a)=\{\{x,y\},\{x\},\{y,\overline{x}\}\}$, *and thus* $\bigodot_3(N)\neq\bigodot_3(N\cup\{(a,y)\})$.

Many more semantics could be studied, for example, based on maximization of the obligation sets rather than maximizing norms in the function m, as in (Horty 1993), introducing priorities among the norms, or introducing a causal reasoning principle (Bochman 2005). Since we focus on completeness theorems in this paper, we leave these variants for further research, and we turn to proof systems.

Proof system

For $\bigodot^*$, compared to $\bigcirc$, the rules SI and ID still hold, but T no longer holds. Consequently, the proof systems for $\bigodot_1^*$ and $\bigodot_3^*$ may be the same, and likewise for $\bigodot_2^*$ and $\bigodot_4^*$.

For $\bigodot$ we have the same properties as for $\bigodot^*$, with in addition the new rule *Violation* ($\sim$) for all deontic operators.

$$\frac{}{(a,\overline{a})}\sim$$

If we define the completeness theorem analogously to Theorem 1, then the proof system would have to be non-monotonic. We do not study this issue here. We just observe that we may enforce monotonicity in N by considering equivalence under addition of norms, in the following way. For all deontic operations $\bigodot\in\{\bigodot_i,\bigodot_i^+\mid i=1,\ldots,4\}$ and corresponding proof systems $\delta\in\{\delta_i,\delta_i^+\mid i=1,\ldots,4\}$ we have $\forall A\subseteq L\forall M\subseteq L\times L$:

$$(a,x)\in\delta(N)\text{ iff }\bigodot(N\cup M,A)=\bigodot(N\cup M\cup\{(a,x)\},A)$$

Permissions

We consider two of the three kinds of permissions studied by Makinson and van der Torre (2003a), because these are the only deontic logics for strong permissions we know of, which come with a proof system.

Negative permission

A negative permission operation is defined analogously to a deontic operation.

Definition 8 (Negative permission operation) *The negative permission operation* P^- *is a function from an abstract normative system and a context to a subset of the universe* $P^-(\langle L,N\rangle,A)\subseteq L$. *We write* $P^-(L,A)$ *for* $P^-(\langle L,N\rangle,A)$.

Something is negatively permitted if it is not explicitly forbidden.

Definition 9 (Negative permission) *Let* $\bigcirc$ *be a deontic operation and* P^- *a negative permission operation.*

- $x\in P^-(N,A)$ *iff* $\overline{x}\notin\bigcirc(N,A)$

We use Makinson and van der Torre (2003a)'s running example to illustrate our operations.

Example 9 $N=\{(work,tax)\}$.

A	$P_1^-(N,A)$
$\{\top\}$	$\{tax,\overline{tax},work,\overline{work}\}$
$\{work,\top\}$	$\{tax,work,\overline{work}\}$

In context $\{\top\}$, *we have* $\bigcirc(N,\{\top\})=\{\}$, *so everything is negatively permitted.*

Properties

Makinson and van der Torre show that so-called inverse rules hold for negative permission in input/output logic, and the following proposition illustrates similar properties in our abstract setting, obtained by inverting the proof rules of $\bigcirc$.

Proposition 2 *The following property holds for all eight negative permission operations.*

$$\frac{x\in P^-(N,A)}{x\in P^-(N,\top)}SI^{-1}$$

However, as Makinson and van der Torre observe, it is not clear how these properties could lead to a characterization of P^- as the closure of some basis under the rules, for it is not clear what the basis could be.

Static positive permission

Positive permissions are explicitly permitted by the rules. To formalize positive permissions, we need a set P of explicit permissive norms, along with the set N of explicit obligations.

Definition 10 (ANS^P) *An abstract normative system with permissive norms ANS^P is a triple* $\langle L,N,P\rangle$ *with* $N,P\subseteq L\times L$ *two sets of conditional norms. To distinguish permissive norms from obligations, we write them as* $<a,x>$.

Definition 11 (Positive permission operation) *The positive permission operation* P^0 *is a function from an abstract normative system with permissive norms* $\langle L,N,P\rangle$ *and a context* A *to a subset of the universe* $P^0(\langle L,N,P\rangle,A)\subseteq L$.

In the line of Von Wright's later approach, Makinson and van der Torre say that there is a static positive permission to realize x in context A if x is generated under these conditions either by the norms in N alone, or the norms in N together with a single explicit permission in P.

Definition 12 (Static positive permission) *Let $\bigcirc$ be a deontic operation and P^0 a positive permission operation.*

- $x \in P^0(N, P, A)$ *iff* $x \in \bigcirc(N \cup Q, A)$ *for some singleton or empty* $Q \subseteq P$

Example 10 (continued from Example 9) *Besides norm* $N = \{(work, tax)\}$, *we have two explicit permissive norms* $P = \{\lhd 18y, vote\rhd, \lhd 18y, work\rhd\}$ *visualized in Figure 8.*

A	$P_1^0(N, P, A)$	$P_3^0(N, P, A)$
$\{\top\}$	$-$	$-$
$\{work, \top\}$	$\{tax\}$	$\{tax\}$
$\{18y, \top\}$	$\{vote, work\}$	$\{vote, work, tax\}$

In context $\{\top\}$ *and using* P_1^0, *no positive permissions are derived. In context* $\{work, \top\}$ *we have* $\{tax\} \in \bigcirc_1(N \cup Q, A)$ *with* $Q = \emptyset$. *In context* $\{18y, \top\}$ *we derive* $\{vote, work\}$ *using* $Q = \{\lhd 18y, vote\rhd\}$ *and* $Q = \{\lhd 18y, work\rhd\}$ *respectively. Using* P_3^0, *also* tax *is derived.*

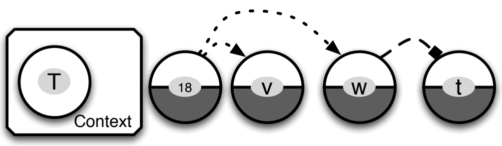


Figure 8: Combining N and P norms (Examples 10 - 11).

Proof system

Proof systems for permissions presuppose a proof system for obligations. The so-called subverse proof rules (Makinson and van der Torre 2003a) are derived from proof rules of $\bigcirc$, where the conclusion and one of the premises is replaced by a corresponding permission.

Definition 13 *Given* $d(N)$, $p_1(N, P)$ / $p_2(N, P)$ / $p_3(N, P)$ *is the smallest set containing* P *and closed under* OP *and* $SI^{\downarrow}$ (OP+$SI^{\downarrow}$+$OR^{\downarrow}$ / OP+$SI^{\downarrow}$+$T^{\downarrow 1}$+$T^{\downarrow 2}$):

$$\frac{(a,x)}{\lhd a,x\rhd}OP \qquad \frac{\lhd\top,x\rhd}{\lhd a,x\rhd}SI^{\downarrow} \qquad \frac{(a,x)\lhd\overline{a},x\rhd}{\lhd\top,x\rhd}OR^{\downarrow}$$

$$\frac{(a,x)\lhd x,y\rhd}{\lhd a,y\rhd}T^{\downarrow 1} \qquad \frac{\lhd a,x\rhd(x,y)}{\lhd a,y\rhd}T^{\downarrow 2}$$

For throughput, the same holds, when $d^+(N)$ *is used.*

Example 11 (continued) $\lhd 18y, tax\rhd \in p_3(N, P)$ *as a consequence of rule* $T^{\downarrow 2}$.

The completeness theorem is defined in the same spirit as Theorem 1 for obligations, and proven using a non-repetition lemma in the same way as the completeness theorems of (Makinson and van der Torre 2003a).

Theorem 2 (Completeness) *For all permission operations* $P^0 \in \{P_i^0, P_i^{0+} \mid i = 1 \ldots 3\}$ *and corresponding proof systems* $p \in \{p_i, p_i^+ \mid i = 1 \ldots 3\}$ *we have:*

$\forall A : P^0(N, P, A) = P^0(N, P \cup \{(a, x)\}, A)$ *iff* $(a, x) \in p(N, P)$

We conjecture that for some derivations there is no proof that allows non repetition usage of norms as shown in the following example.

Example 12 *Assume* $N = \{(\overline{b}, \overline{x}), (b, y), (y, \overline{b})\}$ *and from the rules we want to derive* $(\top, \overline{x})$.

$$\dfrac{\dfrac{(b,y) \qquad \dfrac{(y,\overline{b}) \qquad (\overline{b},\overline{x})}{(y,\overline{x})}\,T}{(b,\overline{x})}\,T \qquad\qquad (\overline{b},\overline{x})}{(\top,\overline{x})}\,OR$$

The rule $(\overline{b}, \overline{x})$ *is used twice; first in the starting T derivation and second in the last OR.*

Though the failure of the completeness proof of P_4 is on the one hand disappointing, on the other hand it is reassuring to find that the same behavior of permission under input/output logic is shown here again at the abstract level.

Constitutive Norms

Constitutive norms are usually represented as so-called count-as conditionals "X counts as Y in the context Z" (Searle 1969). However, a drawback of this representation is that there is no consensus on the representation of the context. Lindahl and Odelstad (2003) abstract from the context and represent constitutive norms as rules "X counts as Y," in the same way as regulative norms are represented. Boella and van der Torre (2006) use the same framework for constitutive norms deriving institutional facts, as they use for regulative norms deriving obligations. Here we also use the abstract normative system in Definition 1 for constitutive norms generating institutional facts, we call the corresponding operation I for *institutional fact operation.*

In particular, we are interested in the interaction among regulative and constitutive norms, just like in the previous section on positive permissions we were interested in the interaction among regulative and permissive norms.

Definition 14 (ANS^C) *An abstract normative system with constitutive norms ANS^C is a triple* $\langle L, N, C\rangle$ *with* $N, C \subseteq L \times L$ *two sets of conditional norms.*

Definition 15 (*IO* operation) *The combined IO operation is a function from an abstract normative system with constitutive norms* $\langle L, N, C\rangle$ *and a context* A *to a subset of the universe* $IO(\langle L, N, C\rangle, A) \subseteq 2^L$. *To distinguish the kinds of norms, we write* $[a, p]$ *for constitutive norms,* (p, x) *for regulative norms as before, and* $[a, x)$ *for the combined operation.*

Semantics

Lindahl and Odelstad use constitutive norms to generate a particular kind of institutional facts, called *intermediate concepts*. Their combination is visualized in Figure 9 considering Example 13. The context of the combined system consists of so called brute facts A, which are used to generate the input I of the regulative norms N. This input consists of institutional facts intermediate between the two black boxes of constitutive C and regulative norms N. The output of the regulative norms are again obligations and permissions O. In Figure 1 we have brute facts like “the contract is signed (e)” that leads to the intermediate concepts: “there is a legal contract (l)” and “there is a cottage owner (o)”.

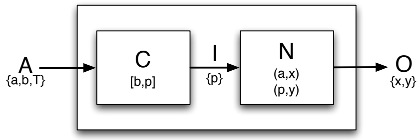


Figure 9: Logical Architecture

We consider two options of the combined operation.

Definition 16 *Let I be an institutional fact operation and $\bigcirc$ a deontic operation.*

- $\ominus^*(N, C, A) = \bigcirc(N, \{\top\} \cup I(C, A))$
- $\ominus(N, C, A) = \bigcirc(N, A \cup I(C, A))$

The following example illustrates the difference between the two operations $\ominus^*$ and $\ominus$.

Example 13 *Given $N = \{(a, x)(p, y)\}$, $C = \{[b, p]\}$ and $A = \{a, b, \top\}$ as visualized in Figure 9 and 10. Consider the following two choices for I and $\bigcirc$:*

- *I_3 and $\bigcirc_3$. We have $\ominus^*(N, C, A) = \{y\}$ because $I_3(C, A) = \{p\}$ and $O_3(N, \{p, \top\}) = \{y\}$, and $\ominus(N, C, A) = \{x, y\}$ because $I_3(C, A) = \{p\}$ and $O_3(N, \{a, b, p, \top\}) = \{x, y\}$.*
- *I_3^+ and $\bigcirc_3$. We have $\ominus^*(N, C, A) = \{x, y\}$ because $I_3^+(C, A) = \{a, b, p\}$ and $O_3(N, \{a, b, p, \top\}) = \{x, y\}$, and $\ominus(N, C, A) = \{x, y\}$ because $I_3^+(C, A) = \{a, b, p\}$ and $O_3(N, \{a, b, p, \top\}) = \{x, y\}$.*

Consequently, if the desired output is $\{x, y\}$, then we can use either $\ominus^*$ with throughput, or $\ominus$.

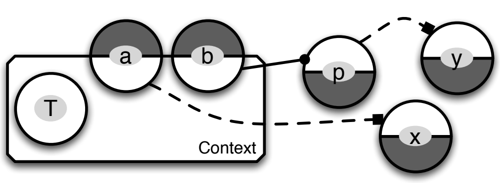


Figure 10: Combining N and C (Examples 13 and 14).

Proof systems

As far as we know, no proof systems have been developed thus far for logical architectures. There seem to be various questions one could pursue. First, one could describe the behavior of $\ominus^*$ and $\ominus$ in terms of the behavior of I and O, together with bridge rules.

Definition 17 *Given $d^I(C)$ for constitutive norms and $d^O(N)$ for regulative norms, $d^{IO*}(C, N)$ ($d^{IO}(C, N)$) is the smallest set containing $d^I(C)$, $d^O(N)$ and closed under IO^1 (IO^1 and IO^2):*

$$\frac{[a, p]\ (p, x)}{[a, x]} IO^1 \qquad \frac{(a, x)}{[a, x]} IO^2$$

Example 14 (Continued) *$N = \{(a, x)(p, y)\}$ and $C = \{[b, p]\}$. With $\ominus^*$ and throughput for I, we derive $[a, x)$ from (a, x) using ID and IO^1 as follows:*

$$\dfrac{\dfrac{}{[a, a]}\,ID \qquad (a, x)}{[a, x]}\,IO^1$$

With $\ominus$, we can derive it directly using IO^2.

However, it is less clear how we can define a completeness theorem for a semantics using this proof system.

Another kind of proof system for the logical architecture in Figure 9 could be based on the idea that one is only interested in the input/output behavior of the whole system, and therefore not in the intermediate concepts. In other words, the underlying notion of equivalence of normative systems could abstract away the institutional facts.

The system can be further developed by introducing both permissive and constitutive norms in the same abstract normative system, and consider constraints for the constitutive and permissive rules too. For example, Artosi *et al.* (2004) argue that constitutive norms are defeasible. We do not pursue this issue in this paper.

Instantiation

Explosion

To instantiate the ANS with propositional logic to obtain a fragment of input/output logic, we have to deal with the property that if an input or output is inconsistent, then it is the whole language L. In this section, we therefore make two assumptions. First, if the context contains a complementary pair, then it is equivalent to the universe. Moreover, if the output contains a complementary pair, then it is the whole universe.

Definition 18 (Eight deontic operations with explosion) *We assume that the context satisfies the property that if $a, \overline{a} \in A$, then $A = L$. Given an ANS $\langle L, N \rangle$ and context A, $N[A] = N(A)$ if there is no $e \in E$ with $e, \overline{e} \in N(A)$, $N[A] = L$ otherwise.*

$O_1(N, A) = N[A]$
$O_2(N, A) = \cap\{N[V] | A \subseteq V, V \textit{ complete}\}$
$O_3(N, A) = \cap\{N[B] | A \subseteq B \supseteq N[B]\}$
$O_4(N, A) = \cap\{N[V] | A \subseteq V \supseteq N[V], V \textit{ complete}\}$
$O_i^+(N, A) = O_i(N \cup I, A)$

We have that the following two inference rules hold:

$$\frac{(a, \overline{a})\ (b, x)}{(a, x)} V \qquad \frac{(a, x)\ (a, \overline{x})}{(a, y)} \perp$$

Example 15 $N = \{(\top, x), (a, \overline{x}), (\overline{a}, y), (b, \overline{y})\}$. *The following derivation shows that $z \in O_2(N, \{b, \top\})$.*

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{(\top,x)}{(a,x)}SI \quad (a,\overline{x})}{(a,y)}\bot \quad (\overline{a},y)}{(\top,y)}OR}{(b,y)}SI \quad (b,\overline{y})}{(b,z)}\bot$$

For negative permission we have that now $\bot^{-1}$ holds (see Proposition 2) and for positive permission that $\bot^{\downarrow}$ holds (see Theorem 2).

$$\dfrac{x \in \bigcirc(N,A) \quad y \in P^-(N,A)}{x \in P^-(N,A)}\bot^{-1} \quad \dfrac{(a,x)\ \langle a,\overline{x}\rangle}{\langle a,y\rangle}\bot^{\downarrow}$$

Instantiation: input/output logic

To compare our abstract operations to the unconstrained input/output operations, we repeat the relevant definitions and theorems from the latter. Makinson and van der Torre (2000) emphasize that this is a relatively simple setting, abstracting from important aspects of deontic reasoning, such as contrary-to-duty reasoning or permissions. These aspects are discussed in (Makinson and van der Torre 2001; 2003a), leading to much more involved input/output operations, which we discuss in following sections. The construction of the semantics is analogous to our abstract normative systems, adding the closure of input and output under propositional consequence.

Definition 19 (out (Makinson and van der Torre 2000)) *Let L be a propositional logic with Cn the consequence operator of L, $\top$ a tautology of L, a complete set one that is either maxiconsistent or equal to L, and let N be a set of ordered pairs of L (called the generators). A generator (a,x) is read as 'if input a then output x'. The following logical systems are defined:*
$out_1(N,A) = Cn(N(Cn(A))$
$out_2(N,A) = \cap\{Cn(N(V)) : A \subseteq V, V \text{ complete}\}$
$out_3(N,A) = \cap\{Cn(N(B)) : A \subseteq B = Cn(B) \supseteq N(B)\}$
$out_4(N,A) = \cap\{Cn(N(V)) : A \subseteq V \supseteq N(V), V \text{ complete}\}$

To emphasize the similarity between the two proof systems, we use the same names for the more general rules of input/output logic as the names we used for the rules of our abstract normative systems, since it is always clear from context which rule we refer to. In addition, all proof systems satisfy *Weakening of the Output* (WO) and *Conjunction* (AND), and d_3 and d_4 satisfy *Cumulative Transitivity* (CT).

Definition 20 (deriv (Makinson and van der Torre 2000)) *Let L be a propositional logic with $\vdash$ the derivibility relation of L. $deriv_1(N)$ ($deriv_2(N)$ / $deriv_3(N)$ / $deriv_4(N)$) is the smallest set containing $N \cup \{(\top,\top)\}$ and closed under SI, WO and AND (SI+WO+AND+OR / SI+WO+AND+CT / SI+WO+AND+OR+CT), and $deriv_i^+(N)$ is the smallest set containining $N \cup \{(\top,\top)\}$ and closed under the rules of $deriv_i$, and in addition ID:*

$$\dfrac{(a,x)\quad b \vdash a}{(b,x)}SI \quad \dfrac{(a,x)\quad x \vdash y}{(a,y)}WO \quad \dfrac{(a,x)\quad (a,y)}{(a,x\wedge y)}AND$$

$$\dfrac{(a,x)\quad (b,x)}{(a\vee b,x)}OR \quad \dfrac{(a,x)\quad (a\wedge x,y)}{(a,y)}CT \quad \dfrac{}{(a,a)}ID$$

Theorem 3 ((Makinson and van der Torre 2000)) *For all $out \in \{out_i, out_i^+ \mid i = 1,\ldots,4\}$ and $deriv \in \{deriv_i, deriv_i^+ \mid i = 1,\ldots,4\}$ we have:*

$x \in out(N,A)$ iff $(\bigwedge A_0, x) \in deriv(N)$ for a finite $A_0 \subseteq A$

The following proposition shows that the derivations of the abstract normative systems are a subset of the derivations of input/output logic, since T is derivable in out_3 as follows:

$$\dfrac{(a,x) \quad \dfrac{(a,y)}{(a\wedge x,y)}SI}{(a,y)}CT$$

Likewise, AND and WO imply $\bot$.

Proposition 3 *For $d \in \{d_i, d_i^+ \mid i = 1,\ldots,4\}$ we have that all derivations of d are derivations of the corresponding $deriv$ too.*

We make some additional observations regarding the relation between the abstract normative systems and the instantiation in unconstrained input/output logic:

- Theorem 1 looks different from Theorem 3, but note that for the eight input/output logics we have $x \in out(N,\{a,\top\})$ iff $\forall A : out(N,A) = out(N \cup \{(a,x)\}$. It seems that we cannot reformulate Theorem 1 in a similar way as Theorem 3, since we do not have a counterpart of conjunction in the theory of abstract normative systems.
- For further symmetry we could have added a rule corresponding to WO to our abstract normative systems: from (a,x) derive $(a,\top)$. Alternatively, we could have included $(\top,\top)$ in the basis of the proof system. In the semantics, this corresponds to assuming that $\top$ is always in the obligation set. The corresponding axiom in the modal logic framework is that a tautology is always obligatory, which has been criticized in the deontic logic literature. We therefore do not adopt the principle that $\top$ should always be in the obligation set.
- The eight deontic operations for abstract normative systems are distinct, whereas in input/output logic, $out_2^+ = out_4^+$ (there is CT elimination in out_4^+, since it is derivable from all the other rules).
- Compactness plays an important role in input/output logic, we have assumed finite abstract normative systems in this paper (e.g. to prevent infinite chains in $\bigcirc_3$).

Note that neither in the input/output logic framework, nor in the present abstract normative systems framework, we say that a normative system "implies" a norm. Norms are used to generate obligation sets, we can axiomatize deontic operations using a proof system based on conditionals, but this does not mean that norms are "implied" or "derived." The most we can say is that a norm is "accepted" by a normative system (van der Torre and Tan 1999), or "redundant" in a normative system (van der Torre 2010).

The latter point may be related to two philosophical considerations of the input/output logic framework. First, the framework is based on the idea that norms do not have truth values, known as Jorgensen's dilemma in the deontic logic literature (Jörgensen 1937). Second, the role of logic is not to create or determine a distinguished set of norms, but rather to prepare information before it goes in as input to such a normative code, to unpack output as it emerges and, if needed, coordinate the two in certain ways. A set of conditional norms is thus seen as a transformation device, and the task of logic is to act as its "secretarial assistant" (Makinson and van der Torre 2000). See their papers for a further discussion.

Finally, Stolpe (2008) introduces input/output operations without WO, leading to the same abstract systems as above.

Dilemmas and CTD

As far as we know, no completeness results are known. In deontic update semantics (van der Torre and Tan 1999), it has been observed that analogues of WO no longer hold, but instead from (a, x) and (a, y) we can derive $(a, x \vee y)$. Moreover, instead of CT or T the following so-called CTA rule holds: from (a, x) and $(a \wedge x, y)$ derive $(a, x \wedge y)$. Likewise, in input/output logic WO, CT and T do not hold, and in out_3 CTA holds. In addition, the following so-called CTD rule holds: from (a, x) and $(\overline{x}, y)$ derive $(a, \overline{x} \rightarrow y)$.

Permissions

Makinson and van der Torre provide completeness proofs only for P_1^0 to P_3^0, where they replace CT by the CTA rule, which derives $(a, x \wedge y)$ from (a, x) and $(a \wedge x, y)$.

Makinson and van der Torre provide also completeness proofs for so-called dynamic permissions. Since this is a relatively complex operation, we leave that for further research.

Constitutive norms

Besides the algebraic cis approach of Lindahl and Odelstad, two instatiations of constitutive norms are Jones and Sergot (1996)'s and Artosi *et al.* (2006)'s modal frameworks for constitutive norms. The first is relatively weak and only satisfies strengthening of the input and transitivity. The latter develops several logics, of which the so-called classificatory interpretation is strong, in the sense that it satisfies the same rules as $\bigcirc_4^+$. The authors have argued that there are some deeper similarities between their models. Indeed, at the abstract level studied in this paper, they roughly satisfy the same rules.

In this paper we are interested in the combination of constitutive and regulative norms. Concerning logical architectures, the combination of input/output logics has been proposed by Makinson and van der Torre (2003b) under the name of lions or logical input/output nets, but they have not been studied formally.

Summary

In this paper, we introduce an easy to use graph based reasoning framework to classify and organize theories of normative reasoning. We are able to reproduce many subtle variants of existing logics at the abstract level, including many input/output logics, as summarized in Table 1. In total, we introduce eight concrete deontic operators with their proof systems, two ways to introduce constraints for each of these deontic operators, and two ways to introduce permissions for each deontic operator, one with its proof systems. Moreover, we introduce two ways to combine regulative and constitutive norms.

Table 1: Proof systems for IOL and ANS operations

Output operation	IOL $deriv$	ANS d_i
Simple-minded	{SI, WO, AND}	{SI, $\perp$}
Basic	{SI, WO, AND}+{OR}	{SI, $\perp$}+{OR}
Reusable	{SI, WO, AND}+{CT}	{SI, $\perp$}+{T}
Reusable basic	{SI, WO, AND}+{OR,CT}	{SI, $\perp$}+{OR,T}

In the course of this paper, we have already indicated several open problems. For example, we have to extend our framework to cope with the issues discussed in the second volume of the handbook (Gabbay et al. To appear b), which covers time, action, norm change, epistemic norms, games and more. Another topic for future research is to study the relation between our abstract theory of normative systems, and Dung (1995)'s abstract theory of argumentation. Though we share the motivation of providing a general framework to classify concrete approaches, and we share the methodology of graph reasoning, there are also substantial differences. In particular, the notion of abstraction used in Dung's theory is of a different kind than the notion of abstraction used in this paper. An initial relation between input/output logic and abstract argumentation has been shown by Bochman (2005), using the input/output logic out_4 to represent causal reasoning, and represent argumentation in the causal framework. The relation for our abstract theory is also left for future research.

The work on abstract argumentation offers more inspiration for further research on abstract normative systems. For example, we may extend abstract normative systems with higher order relations reflecting the addition or removal of norms, as done in the reactive approach (Gabbay 2008), or with preference or value patterns (Villata, Boella, and van der Torre 2011a). Moreover, we may extend abstract normative systems with other patterns for collective norms, as in accrual of arguments, where elements together give rise to new obligations, without doing so individually.

Another unexplored topic is the visualization of normative systems. The derivation of obligations, permissions and institutional facts may be visualized by coloring the nodes, or changing their shape. Inspired by attack semantics for abstract argumentation (Villata, Boella, and van der Torre 2011b), we may emphasize the norms rather than the elements by visualizing the elements as points rather than circles, and coloring or changing the shape of the successful norms.

References

Artosi, A.; Rotolo, A.; and Vida, S. 2004. On the logical nature of count-as conditionals. In Cevenini, C., ed., *The Law and Electronic Agents: Proceedings of the LEA 04 workshop*, 9–34. Gedit.

Bochman, A. 2005. *Explanatory Nonmonotonic Reasoning*. New York, London: World Scientific Publishing Company.

Boella, G., and van der Torre, L. 2004. Regulative and constitutive norms in normative multiagent systems. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth International Conference (KR 2004)*, 255–266.

Boella, G., and van der Torre, L. 2006. A logical architecture of a normative system. In *Deontic Logic and Artificial Normative Systems, 8th International Workshop on Deontic Logic in Computer Science (ΔEON 2006)*, volume 4048 of *LNCS*, 24–35. Springer.

Chellas, B. 1980. *Modal Logic: An Introduction*. Cambridge University Press.

Chisholm, R. 1963. Contrary-to-duty imperatives and deontic logic. *Analyse* 24:33–36.

Dung, P. M. 1995. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming, and n–person games. *Artificial Intelligence* 77:321–357.

Forrester, J. W. 1984. Gentle murder, or the adverbial samaritan. *Journal of Philosophy* 81:193–196.

Gabbay; Horty; van der Meyden; and van der Torre., eds. To appear a. *Handbook of Normative systems*, volume 1. London: College Publications.

Gabbay; Horty; van der Meyden; and van der Torre., eds. To appear b. *Handbook of Normative systems*, volume 2. London: College Publications.

Gabbay, D. M. 2008. Reactive kripke models and contrary to duty obligations. In *Deontic Logic in Computer Science, 9th International Conference, DEON 2008*, volume 5076 of *Lecture Notes in Computer Science*, 155–173. Springer.

Grossi, D.; Meyer, J.-J.; and Dignum, F. 2006. Counts-as: Classification or constitution? an answer using modal logic. In *Procs. of Deontic Logic and Artificial Normative Systems, 8th International Workshop on Deontic Logic in Computer Science, (ΔEON 2006)*, volume 4048 of *LNCS*, 115–130. Springer.

Hansen, J. 2006. Deontic logics for prioritized imperatives. *Artificial Intelligence and Law* 14(1-2):1–34.

Hansson, B. 1969. An analysis of some deontic logics. *Noûs* 3:373–398.

Horty, J. F. 1993. Deontic logic as founded on nonmonotonic logic. *Ann. Math. Artif. Intell.* 9(1-2):69–91.

Jones, A., and Sergot, M. 1996. A formal characterisation of institutionalised power. *Journal of IGPL* 3:427–443.

Jörgensen, J. 1937. Imperatives and logic. *Erkenntnis* 7:288–296.

Lindahl, L., and Odelstad, J. 2003. Normative systems and their revision: An algebraic approach. *Artificial Intelligence and Law* 11(2-3):81–104.

Makinson, D., and van der Torre, L. 2000. Input-output logics. *Journal of Philosophical Logic* 29(4):383–408.

Makinson, D., and van der Torre, L. 2001. Constraints for input-output logics. *Journal of Philosophical Logic* 30(2):155–185.

Makinson, D., and van der Torre, L. 2003a. Permissions from an input-output perspective. *Journal of Philosophical Logic* 32(4):391–416.

Makinson, D., and van der Torre, L. 2003b. What is input/output logic? In *Foundations of the Formal Sciences II: Applications of Mathematical Logic in Philosophy and Linguistics*, volume 17 of *Trends in Logic*. Kluwer.

Makinson, D. 1999. On a fundamental problem of deontic logic. In McNamara, P., and Prakken, H., eds., *Norms, Logics and Information Systems. New Studies on Deontic Logic and Computer Science*, 29–54. IOS Press.

Nute, D., ed. 1997. *Defeasible deontic logic*, volume 263 of *Synthese Library*. Kluwer.

Searle, J. 1969. *Speech Acts: an Essay in the Philosophy of Language*. Cambridge (UK): Cambridge University Press.

Stolpe, A. 2008. Normative consequence: The problem of keeping it whilst giving it up. In *Deontic Logic in Computer Science, 9th International Conference (ΔEON 2008)*, volume 5076 of *LNCS*, 174–188. Springer.

van der Torre, L., and Tan, Y. 1999. Deontic update semantics. In McNamara, P., and Prakken, H., eds., *Norms, Logics and Information Systems. New Studies on Deontic Logic and Computer Science*. IOS Press.

van der Torre, L. 1997. *Reasoning about obligations*. Ph.D. Dissertation, Erasmus University Rotterdam.

van der Torre, L. 2010. Deontic redundancy: A fundamental challenge for deontic logic. In *Deontic Logic in Computer Science, 10th International Conference (ΔEON 2010)*, volume 6181 of *LNCS*, 11–32. Springer.

Villata, S.; Boella, G.; and van der Torre, L. 2011a. Argumentation patterns. In *8th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2011)*, 133–150.

Villata, S.; Boella, G.; and van der Torre, L. 2011b. Attack semantics for abstract argumentation. In Walsh, T., ed., *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI 2011)*, 406–413. IJCAI/AAAI.

von Wright, G. H. 1951. Deontic logic. *Mind* 60:1–15.

von Wright, G. H. 1963. *Norm and Action. A logical Inquiry*. London: Routledge and Kegan Paul.