An ASPIC-based legal argumentation framework for deontic reasoning

Leendert VAN DER TORRE [a], Serena VILLATA [b,1]

[a] *University of Luxembourg, Luxembourg*

[b] *INRIA Sophia Antipolis, France*

Abstract. In the last years, argumentation theory has been exploited to reason about norms, argue about enforced obligations and permissions, and establish the validity of norms seen as argumentative claims. In this paper, we start from the dynamic legal argumentation framework recently proposed by Prakken and Sartor, and we extend their ASPIC-based system by introducing deontic modalities, to include also normative concepts like factual and deontic detachment, and normative dynamics. Properties of the original and proposed legal argumentation system are presented and discussed, and related to deontic logic and logics of normative systems.

Keywords. deontic modalities, dynamic legal argumentation, input/output logic

1. Introduction

Norms regulate our everyday life, and are used to assess the conformance of our behavior with respect to the regulations holding in specific contexts. Given the profound importance of norms in our lives, it is fundamental to understand which norms are valid in certain environments, how to interpret them, the legal conclusions of such norms, which norms can be derived from the existing ones, etc. In order to understand norms, people discuss about them to assess the validity or applicability of a certain norm subject to particular conditions, to derive the obligations and permissions to be enforced, or claim that a certain normative conclusion cannot be derived from the existing regulations. Several frameworks have been proposed for legal argumentation [3], but no comprehensive formal model of legal reasoning from arguments has been proposed yet. In this paper, we answer the research question: *how to enrich legal argumentation with a formal account of deontic modalities?*

Prakken and Sartor [11] recently introduced a new instance of ASPIC$^+$ [8] to capture the inference schemes of arguments about norms like *legislative and interpretative* arguments. More precisely, they define a so-called *dynamic legal argumentation system* as a specification of ASPIC$^+$ argumentation system, and a knowledge base K is a normative system consisting of an input, which is composed of a set of literals, and a set of norms, where each norm has the form $L_1 \wedge \ldots \wedge L_n \rightsquigarrow L$ where $L_1, \ldots, L_n, L$ are literals. However, the introduced dynamic legal argumentation system has the following drawback: norms

[1]Corresponding Author: Serena Villata, INRIA Sophia Antipolis, 2004 Route des Lucioles, BP 93, Sophia Antipolis Cedex, France; E-mail: serena.villata@inria.fr.

are expressed without using deontic operators for specifying obligations, permissions and prohibitions, as remarked by the authors themselves in their future perspectives. As a consequence, normative concepts like deontic detachment and the equivalence of normative systems are not considered, the system is not related to deontic logic or logics of normative systems, and no properties of the framework are provided. In this paper, we address these issues by extending the dynamic legal argumentation system of Prakken and Sartor [11] with deontic modalities, adopting the input/output logic methodology [6] for the analysis. We study how this extended dynamic legal argumentation system is related to deontic logic, and we formalize how to change the system to obtain alternative ways to reason with norms.

The remainder of the paper is organized as follows. In Section 2, we study the logical properties of the static core legal argumentation system [11] with respect to the deontic modalities we introduce, and we analyze it by reformulating it in a normative perspective. Section 3 introduces the notion of attack and shows how to verify norms' equivalence in our system. Section 4 introduces deontic modalities for obligation and permission in the system, and Section 5 extends Prakken and Sartor dynamic legal argumentation framework with modalities. Finally, further normative concepts are introduced and discussed, and a comparison with Prakken and Sartor system is addressed. A running example illustrates all new concepts.

2. The Prakken-Sartor (PS) system: unconstrained

We first adapt the definition of dynamic legal argumentation system of Prakken and Sartor [11]. To facilitate our formal analysis, we give only a core static propositional fragment in this section. Moreover, for our analysis we define all the notions like inference rules and arguments as expressions or sentences, that is, as sequences of symbols.

Definition 1 (LAS-PS) *Given a set of propositional atoms. The literals, norms and legal language are given by the following BNF.*

$L ::= P \mid \neg P$ *with* P *in propositional atoms*

$N ::= L \wedge \ldots \wedge L \rightsquigarrow L$

$\alpha ::= L \mid N$

A Legal Argumentation System (LAS) is a tuple $\langle \mathscr{L}, -, \mathscr{R} \rangle$ *where* $\mathscr{L}$ *is the legal language of all sentences* α*,* $- : \mathscr{L} \to 2^{\mathscr{L}}$ *is a function given by* $-(P) = \{\neg P\}$*,* $-(\neg P) = \{P\}$ *and* $-(N) = \emptyset$*, and* $\mathscr{R}$ *contains the following expressions for inference rules Defeasible Modus Ponens (DMP), one rule for each possible norm* $\phi_1 \wedge \ldots \wedge \phi_n \rightsquigarrow \psi$ *that variable* N *may assume.*

DMP: $\phi_1, \ldots, \phi_n, \phi_1 \wedge \ldots \wedge \phi_n \rightsquigarrow \psi \Rightarrow \psi$*;*

In order to illustrate our legal argumentation framework, we adapt the running example discussed by Prakken and Sartor [11].

Example 1 (Running example concerning smoking regulations) *Let the set of propositional atoms be* $\{a, b, c, d, e, f\}$*. LAS is a legal argumentation system* $\langle \mathscr{L}, -, \mathscr{R} \rangle$*, as given in Definition 1 for this set of propositional atoms.*

- *Propositional atoms* $P ::= a|b|c|d|e|f$ *where*
  - $*$ *a: "people want to smoke in a closed space,"*

- ∗ *b: “the public place has special designated areas for smoking,”*
- ∗ *c: “people need to smoke cannabis on medical grounds,”*
- ∗ *d: “people are forbidden from smoking cannabis and tobacco in public places,”*
- ∗ *e: “cannabis is allowed for medical treatment,”*
- ∗ *f: “people are permitted to smoke cannabis in recreational cannabis establishments.”*

- *Literals* $L ::= a|\neg a|b|\neg b|c|\neg c|d|\neg d|e|\neg e|f|\neg f$;
- *Norms* $N ::= a \rightsquigarrow a|a \rightsquigarrow b|a \rightsquigarrow c|a \rightsquigarrow d|a \rightsquigarrow e|a \rightsquigarrow f|a \wedge \neg c \rightsquigarrow d|a \wedge c \rightsquigarrow d|\neg a \wedge c \rightsquigarrow d|\neg a \wedge \neg c \rightsquigarrow d|\dots$ *is the set of all possible norms built from L;*
- *Legal language* $\mathscr{L}$ *contains all sentences* α *that variables L and N may assume;*
- $-$: $-(a) = \{\neg a\}$, $-(\neg a) = \{a\}$, $-(b) = \{\neg b\}$, $-(\neg b) = \{b\}$, $-(c) = \{\neg c\}$, $-(\neg c) = \{c\}$, $-(d) = \{\neg d\}$, $-(\neg d) = \{d\}$, $-(e) = \{\neg e\}$, $-(\neg e) = \{e\}$, $-(f) = \{\neg f\}$, $-(\neg f) = \{f\}$;
- $\mathscr{R}$ *contains expressions for inference rules DMP, like for instance:*
  - ∗ $a, a \rightsquigarrow b \Rightarrow b$; $a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow d$; $\dots$

Prakken and Sartor [11] follow Modgil and Prakken [8] and do not consider a model theoretic semantics for this language. Instead, they define a set of arguments.

Definition 2 (LAS PS arguments) *A knowledge base K is a set of sentences of* $\mathscr{L}$*. The set of arguments A on the basis of a knowledge base K in a legal argumentation system LAS is called* $Arg(LAS, K)$ *and is the smallest set of expressions containing K and closed under the following rule:*

if $A_1, \dots, A_n \subseteq Arg(LAS, K)$ *and* $concl(A_1), \dots, concl(A_n) \Rightarrow L \in \mathscr{R}$ *then* $(A_1, \dots, A_n \Rightarrow L) \in Arg(LAS, K)$,
where $concl(A)$ *is defined by* $concl(\alpha) = \alpha$ *and* $concl(A_1, \dots, A_n \Rightarrow L) = L$*. We may leave out the brackets if there is no risk of confusion.*

To study this notion of norm-based argument, we define consequence by considering only the conclusions of the arguments, in other words, by abstracting away the explicit arguments. Following input/output logic conventions, we call the consequences the unconstrained output, or *Out*. Note that at this stage we do not introduce an attack relation on arguments yet: this choice is part of the formal machinery here (like in other formal approaches, e.g., adaptive logic [2], or explanatory non-monotonic reasoning [4]) where the unconstrained is distinguished from the constrained output.

Definition 3 (Output PS) $Out(LAS, K) = \{concl(A) \mid A \in Arg(LAS, K)\}$.

Example 2 (Continued) *Consider now* $K_1 = \{a, b, c, e, a \wedge b \rightsquigarrow \neg d, c \wedge \neg d \wedge e \rightsquigarrow f\}$ *where the norms state that*

- *if people want to smoke in a closed space and the public place has smoking special secluded areas, then people are not forbidden from smoking cannabis and tobacco in public places;*
- *if people need to smoke cannabis on medical grounds and it is not forbidden from smoking cannabis and tobacco in public places and cannabis is allowed for medical treatment, then people are permitted to smoke cannabis in recreational cannabis establishments;*

Arguments (i.e., including the elements of K as the literals and the norms) can be constructed combining DMP inference rules as follows:

- $A_1 : a, b, a \wedge b \rightsquigarrow \neg d \Rightarrow \neg d$;
- $A_2 : c, (a, b, a \wedge b \rightsquigarrow \neg d \Rightarrow \neg d), e, c \wedge \neg d \wedge e \rightsquigarrow f \Rightarrow f$.

The latter is often presented in ASPIC+ as A_2' below. However, in our presentation, arguments are expressions and A_1 is not part of our language, and thus not part of the argument.

- $A_2' : c, A_1, e, c \wedge \neg d \wedge e \rightsquigarrow f \Rightarrow f$.

From arguments A_1 and A_2, we have that $concl(A_1) = \neg d$ and $concl(A_2) = f$. We conclude that $Out(LAS, K_1) = \{a, b, c, \neg d, e, f, a \wedge b \rightsquigarrow \neg d, c \wedge \neg d \wedge e \rightsquigarrow f\}$.

We now introduce the logical properties of the framework, not given by Prakken and Sartor. We use a proof system with expressions $K \therefore L$ to be read as: L can be derived from K. The proof system contains four rules, called Identity (ID), Strengthening of the input (SI), Factual Detachment (FD), and Deontic Detachment (DD). The former is sometimes called Monotony (Mon), and the latter two are sometimes called Modus Ponens (MP) Cumulative Transitivity (CT). The notion of consequence is called simple-minded reusable throughput or out_3^+ by Makinson and van der Torre [6]. Strengthening of the input reflects that we consider unconstrained output only.

Definition 4 (Derivations PS) *$der(LAS)$ is the smallest set of expressions $K \therefore L$ closed under the following four rules.*
ID: *$\{L\} \therefore L$ for a literal L*
SI: *from $K \therefore L$ derive $K \cup K' \therefore L$*
FD: *$\{L_1, \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L\} \therefore L$ for a norm $L_1 \wedge \ldots \wedge L_n \rightsquigarrow L$*
DD: *from $K \therefore L_i$ for $1 \leq i \leq n$ and $K \cup \{L_1 \ldots, L_n\} \therefore L$ derive $K \therefore L$*

Theorem 1 shows the close relation between arguments and derivations in a deontic logic or a logic of normative systems. This is not surprising, as the similarity is quite clear from the structure of arguments. However, making the relation precise by framing the legal argument system into an input/output logic highlights two properties, which we give as corollaries. These two properties lead to the definition of variants later in Section 4 of this paper. Moreover, Theorem 1 will be the basis for Theorem 2 of this paper, relating argumentation semantics and constrained output operations.

Theorem 1 (Characterization PS) *$K \therefore L \in der(LAS)$ iff $L \in Out(LAS, K)$.*
Proof. *(sketch) The proof is not complicated, but it may be unusual for readers not familiar with input/output logic methodology. We therefore give some details.*

Soundness: If $K \therefore L \in der(LAS)$, then $L \in Out(LAS, K)$. First we show that each rule is sound, which is immediate.
ID: *$L \in Out(LAS, L)$ due to inclusion of L in $Arg(LAS, \{L\})$.*
SI: *Assume $K \therefore L$. So there is an argument A using some of the premises of K, which has conclusion L. This argument A can also be built from $K \cup K'$. Thus $K \cup K' \therefore L$*
FD: *Assume a knowledge base $K = \{L_1, \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L\}$. Then we have that $A = L_1, \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L \Rightarrow L \in Arg(LAS, K)$, and thus we also have that $L = concl(A) \in Out(LAS, K)$.*

DD: *Assume $L_i \in Out(LAS, K)$ and $L \in Out(LAS, K \cup \{L_1, \ldots, L_n\})$. In the argument for L, replace all occurrence of L_i by the argument for L_i. This gives an argument for L, i.e., $L \in Out(LAS, K)$.*

Soundness follows by induction on the structure of the derivation.

Completeness: If $L \in Out(LAS, K)$, then $K \therefore L \in der(LAS)$. The proof is by construction. Assume $L \in Out(LAS, K)$. In case the argument does not use a norm, we can directly derive it using ID. Otherwise, the argument makes use of a set of norms. Now consider the following derivation. Use FD and SI to strengthen these norms $L_1, \ldots, L_n \rightsquigarrow L$ to $K \cup \{L_1, \ldots, L_n\} \therefore L$. Then combine them using DD, using the structure of the argument.

The theorem highlights a drawback of the legal argumentation system of Prakken and Sartor: simple-minded reusable throughput is often adopted for default logics and logic programs, but rarely for normative reasoning. Note that though it is unconstrained output, we can not derive p from $\{q \rightsquigarrow p, r, \neg r\}$, i.e., we do not have explosion. The following two corollaries highlight the drawbacks.

Corollary 1 (Reasoning by cases) *The system does not satisfy reasoning by cases, e.g. from $K = \{a \rightsquigarrow x, \neg a \rightsquigarrow x\}$ we cannot derive x. This is reflected in the proof system by the lack of the disjunction rule: OR: from $K \cup \{P\} \therefore L$ and $K \cup \{\neg P\} \therefore L$ derive $K \therefore L$.*

Corollary 1 highlights an open problem: How to introduce reasoning by cases in the PS system? At first sight, it may seem that we can close the norms under disjunction. This, however, does not work, as we cannot derive x from the knowledge base $K = \{a \rightsquigarrow p, p \rightsquigarrow x, \neg a \rightsquigarrow x\}$. We leave this problem for future research.

Corollary 2 (Throughput) *The system satisfies throughput, that is, if $L \in K$ then we have $L \in Out(AS, K)$.*

Corollary 2 highlights a second open problem: how to prevent that facts are obligatory? This is sometimes called the Is-Ought problem in deontic logic and normative reasoning. In the running example, this issue is dealt with by introducing special propositions read like an obligation. Prakken and Sartor explicitly mention this problem as future work to be addressed, and they suggest that a modal operator may be introduced. We consider this alternative in Section 4 in this paper.

3. The PS system: constrained

To establish our results with constrained input/output logic, we consider only rebut, not undercut (which needs names of norms and rules). Moreover, we do not consider defeasible knowledge and undermining. So the only attack is the attack of an argument with an opposite literal. This is obviously a very simple notion of attack which is of little use in most applications, but it useful to establish the relation with logical approaches.

Definition 5 (Attack PS) *The set of sub-arguments of argument B is the smallest set containing B that is closed under the rule: if $A_1, \ldots, A_n \Rightarrow L$ is a sub-argument of B, then also A_1, ..., A_n are sub-arguments of B.*

A attacks B iff $B \notin K$ and there is a sub-argument B' of B such that $concl(A) \in -(concl(B'))$. We write $attack(LAS, K)$ for the set of all attacks among $Arg(LAS, K)$.

No argument can attack a norm, so in this static system, all norms of the knowledge base are accepted. A semantics associates sets of extensions with an argumentation framework, where each extension consists of a set of arguments. For each extension, the output consists of the set of conclusions of the arguments, as for *Out* before. A semantics thus gives us a set of sets of conclusions, which we call an *Outfamily*.

Definition 6 (Outfamily PS) *An extension is a set of arguments, and an argumentation semantics $sem(arg, attack)$ is a function that takes as input a set of arguments and a binary attack relation among the arguments, and as output a set of extensions.*
$Outfamily(K, sem) = \{\{concl(A) \mid A \in S\} \mid S \in sem(arg(AS, K), attack(AS, K))\}$.

Constrained output can be defined as an iterative procedure. Def. 7 presents a common construction, where each extension corresponds to an order in which norms are applied.

Definition 7 (Outf) *Let $\geq$ be a total order (i.e., transitive, irreflexive and connected) on the norms K^N. For a set of formulas $S \subseteq \mathscr{L}$, let $AN(S, K^N) \subseteq K^N$ be the set of (applicable) norms such that $L_1, \ldots, L_n \in S$, $L \notin S$ and $-(L) \notin S$, and let $o(S, K^N, \geq) = \{L\}$ be the consequent of the maximal (applicable) norm $L_1 \wedge L_n \rightarrow L$ of $N(S, K^N)$ in the total order $\geq$ if $AN(S, K^N)$ is nonempty, $o(S, K^N, \geq) = \emptyset$ otherwise.*
$E(K, \geq) = \cup_{i=0\ldots\infty} E_i$, *where* $E_0 = K^L$ *and* $E_{i+1} = E_i \cup o(S, K^N, \geq)$.
$Outf(K) = \{E(K, \geq) \mid \geq$ *is a total preorder on* $K^N\}$

Due to space limitations, we do not repeat the well known definitions of argumentation semantics, and the proof of Theorem 2 is omitted.

Theorem 2 (Characterization PS) $Outfamily(K, sem) = Outf(K)$ *for sem is stable.*

Finally, we observe that also other definitions of outfamily can be given. Definition 8 below defines constrained output in input/output logic framework, which is inspired by maximal consistent set constructions in belief revision and non-monotonic reasoning. *Maxf* takes the maximal sets of norms of K such that the output of K is consistent, and *Outf* takes the output of these maximal norm sets. In Definition 8, by *consistent* we mean that it does not contain any complementary pair of literals. Defining an attack relation to correspond to this *Outfamily* is left as further work.

Definition 8 (Outf IOL) *Let $K = K^L \cup K^N$ consist of literals K^L and norms K^N.*
$Conf(K) = \{N \subseteq K^N \mid Out(K^L \cup N) consistent\}$
$Maxf(K) = \{N \subseteq K^N \mid N \text{ maximal w.r.t. } \subseteq \text{ in } Conf(K)\}$
$Outf_{IOL}(K) = \{Out(K^L \cup N) \mid N \in Maxf(K)\}$

4. Introducing obligations

We add an additional modal operator O to the language. All norms are of the form $L_1 \wedge \ldots \wedge L_n \rightsquigarrow L$, as before, or $L_1 \wedge \ldots \wedge L_n \rightsquigarrow OL$. The body contains simple literals and the head contains either a literal or an obligation.

4.1. Without identity

In this section we redefine the concepts or *LAS*, *Out*, *der*, etc. As there is no risk for confusion, we refer to them with the same names as in the previous sections.

Definition 9 (LAS O) *Given a set of propositional atoms. The literals, norms and legal language $\mathscr{L}$ are given by the following BNF.*

$L ::= P \mid \neg P$ *with P in propositional atoms*

$M ::= L \mid OL$

$N ::= L \wedge \ldots \wedge L \rightsquigarrow OL$

$\alpha ::= L \mid N$

A Legal Argumentation System with Obligations (LAS-O) is as in Definition 1, where the $-$ function is extended to obligations: $-(OL) = O - (L)$.

The definition of arguments is adapted in the obvious way. In the output, we now consider only the obligatory propositions. Note that, in this paper, we adopt input/output logic as methodology to analyze deontic logics. To classify a deontic logic, we relate what is the case to what is obliged to be the case. This relation is not based on a modal operator, but on purely factual sentences. If a factual sentence is in the input we interpret it as a fact, and when it is in the output it is interpreted as an obligation. Thus, it is the position which determines whether a sentence is factual or obligatory.

Definition 10 (Output O) $Out(LAS\text{-}O, K) = \{L \mid A \in Arg(LAS\text{-}O, K), concl(A) = OL\}$.

Example 3 *Let us consider that LAS-O is defined as follows:*

- *Propositional atoms $P ::= a|b|c|d|e$ where*
  - *a: "the person wants to smoke in a closed space,"*
  - *b: "the person is in a private space,"*
  - *c: "the person needs to smoke on medical grounds,"*
  - *d: "the person is forbidden from smoking,"*
  - *e: "use electronic cigarettes."*
- *Literals $L ::= a|\neg a|b|\neg b|c|\neg c|d|\neg d|e|\neg e$;*
- $M ::= a|\neg a|b|\neg b|c|\neg c|d|\neg d|e|\neg e|Oa|O\neg a|Ob|O\neg b|Oc|O\neg c|Od|O\neg d|Oe|O\neg e$
- $N ::= a \rightsquigarrow a|a \rightsquigarrow b|a \rightsquigarrow c|a \rightsquigarrow d|a \rightsquigarrow e|a \wedge \neg c \rightsquigarrow d|a \wedge c \rightsquigarrow d|a \wedge \neg c \rightsquigarrow Od|a \wedge c \rightsquigarrow Od|\ldots$ *is the set of all possible norms built from L;*
- *$\mathscr{L}$ contains all sentences α that variables L and N may assume;*
- *Function $-$: $-(a) = \{\neg a\}$, $-(\neg a) = \{a\}$, $-(Oa) = \{O\neg a\}$, $-(b) = \{\neg b\}$, $-(\neg b) = \{b\}$, $-(Ob) = \{O\neg b)\}$, $-(c) = \{\neg c\}$, $-(\neg c) = \{c\}$, $-(Oc) = \{O\neg c\}$, $-(d) = \{\neg d\}$, $-(\neg d) = \{d\}$, $-(Od) = \{O\neg d)\}$, $-(e) = \{\neg e\}$, $-(\neg e) = \{e\}$, $-(Oe) = \{O\neg e\}$;*
- *$\mathscr{R}$contains expressions for inference rules like for instance:*
  - $a, a \rightsquigarrow b \Rightarrow b$;
  - $a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow Od$;

Consider now the knowledge base $K_2 = \{a, \neg b, \neg c, a \wedge \neg c \rightsquigarrow d, a \wedge b \rightsquigarrow \neg d, c \rightsquigarrow \neg d, a \wedge d \rightsquigarrow Oe\}$ where the norms state that

- *if the person is in a closed space and she does not need to smoke on medical grounds, then the person is forbidden from smoking;*

- *if the person wants to smoke in a closed space and she is in a private space, then the person is not forbidden from smoking;*
- *if the person needs to smoke on medical grounds, then she is not forbidden from smoking;*
- *if the person wants to smoke in a closed space and she is forbidden from smoking, then it is obligatory to use electronic cigarettes;*

We can construct the following arguments (in addition to arguments containing a single element of K):

- $A_1 : a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow d;$
- $A_2 : a, (a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow d), a \wedge d \rightsquigarrow Oe \Rightarrow Oe;$

We have that $concl(A_1) = d$ and $concl(A_2) = Oe$, and we conclude $Out(LAS\text{-}O, K_2) = \{e\}$ i.e., it is obligatory to use electronic cigarettes.

The constrained version can be defined analogously. The proof system contains three rules, Strengthening of the Input (SI), Factual Detachment (FD) and a kind of contraction (C). The notion of consequence is called simple-minded output or out_1 in [6].

Definition 11 (Derivations O) *$der(LAS\text{-}O)$ is the smallest set of expressions $K \therefore L$ closed under the following three rules.*
SI: *from $K \therefore L$ derive $K \cup K' \therefore L$*
FD: $\{L_1, \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow OL\} \therefore L$ *for a norm* $L_1 \wedge \ldots \wedge L_n \rightsquigarrow OL$
C: *from $K \cup \{L_1 \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L, L\} \therefore L'$ derive $K \cup \{L_1 \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L\} \therefore L'$*

Theorem 3 (Characterization O) *$K \therefore L \in der(LAS\text{-}O)$ iff $L \in Out(LAS\text{-}O, K)$.*
Proof. *Analogous to the proof of Theorem 1. Soundness is straightforward to check, and completeness is again by construction.*

Corollary 3 (Deontic detachment) *The system does not satisfy deontic detachment, e.g. from $K = \{a, a \rightsquigarrow Ob, b \rightsquigarrow Oc\}$ we cannot derive Oc. This is reflected in the proof system by the lack of the DD rule.*

We have thus succeeded in blocking the ID rule, but as a consequence we have also lost the DD rule. It is a topic of ongoing debate whether the DD rule can be used in deontic logic. Some people argued that we cannot accept it, not even in a defeasible deontic logic, whereas others have argued that we cannot do without it. To use the legal argumentation system to contribute to this debate, we need to represent it first. The following definition does so, by adapting the argumentation system.

4.2. Re-introducing deontic detachment

In this section, we redefine again the concepts or *LAS*, *Out*, *der*, etc., referring to them with the same names as in the previous sections.

Definition 12 (LAS O+DD) *Given a set of propositional atoms. The literals, norms and legal language are given by the following BNF.*

$L ::= P \mid \neg P$ *with P in propositional atoms*
$M ::= L \mid OL$

$N ::= L \wedge \ldots \wedge L \rightsquigarrow M$
$\alpha ::= L \mid N$
A Legal Argumentation System (LAS) is a tuple $\langle \mathscr{L}, -, \mathscr{R} \rangle$ *where* $\mathscr{L}$ *is the legal language of all sentences* α, $- : \mathscr{L} \to 2^{\mathscr{L}}$ *is a function given by* $-(P) = \{\neg P\}$, $-(\neg P) = \{P\}$ *and* $-(N) = \emptyset$, *and* $\mathscr{R}$ *contains the single inference rule Defeasible Modus Ponens (DMP), where we have either* $\phi_i = \phi'_i$ *or* $\phi_i = O\phi'_i$.
DMP: $\phi_1, \ldots, \phi_n, (\phi'_1 \wedge \ldots \wedge \phi'_n \rightsquigarrow \psi) \Rightarrow \psi$;

Definition 13 (Derivations O+DD) $der(LAS)$ *is the smallest set of expressions* $K \therefore L$ *closed under the following four rules.*
SI: *from* $K \therefore L$ *derive* $K \cup K' \therefore L$
FD: $\{L_1, \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L\} \therefore L$ *for a norm* $L_1 \wedge \ldots \wedge L_n \rightsquigarrow L$
C: *from* $K \cup \{L_1 \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L, L\} \therefore L'$ *derive* $K \cup \{L_1 \ldots, L_n, L_1 \wedge \ldots \wedge L_n \rightsquigarrow L\} \therefore L'$
DD: *from* $K \therefore L_i$ *for* $1 \leq i \leq n$ *and* $K \cup \{L_1 \ldots, L_n\} \therefore L$ *derive* $K \therefore L$

Theorem 4 (Characterization O+DD) $K \therefore L \in der(LAS)$ *iff* $L \in Out(LAS, K)$.
Proof. *Analogous to the proof of Theorem 1. Soundness is straightforward to check, and completeness is again by construction.*

4.3. Permissions

For permissive norms, we add norms of the form $L_1 \wedge \ldots \wedge L_n \rightsquigarrow \neg OL$.

Definition 14 (LAS OP) *Given a set of propositional atoms. The literals, norms and legal language are given by the following BNF.*
$L ::= P \mid \neg P$ *with* P *in propositional atoms*
$M ::= L \mid OL \mid \neg OL$
$N ::= L \wedge \ldots \wedge L \rightsquigarrow M$
$\alpha ::= L \mid N$
A Legal Argumentation System (LAS) is as defined before, extending the function $-$ *also to permissions,* $-(\neg OL) = \{OL\}, -(OL) = \{O - (L), \neg OL\}$.

Definition 15 (Permission) $Perm(LAS, K) = \{L \mid A \in Arg(LAS, K), concl(A) = \neg OL\}$

Example 4 *Let us consider LAS such that:*
- *Propositional atoms* $P ::= a|b|c|d|e|f$ *where* a, b, c, e *are as in Example 1, and*
  - $*$ *d: "smoke cannabis and tobacco in public places,"*
  - $*$ *f: "smoke cannabis in cannabis establishments."*
- *Literals* $L ::= a|\neg a|b|\neg b|c|\neg c|d|\neg d|e|\neg e|f|\neg f$;
- $M ::= a|\neg a|b|\neg b|c|\neg c|d|\neg d|e|\neg e|Oa|O\neg a|\neg Oa|Ob|O\neg b|\neg Ob|Oc|O\neg c|\neg Oc|Od |O\neg d|\neg Od|Oe|O\neg e|\neg Oe|Of|O\neg f|\neg Of$
- $N ::= a \rightsquigarrow a|a \rightsquigarrow b|a \rightsquigarrow c|a \rightsquigarrow d|a \rightsquigarrow e|a \rightsquigarrow f|a \wedge \neg c \rightsquigarrow d|a \wedge c \rightsquigarrow d|a \wedge \neg c \rightsquigarrow Od|a \wedge c \rightsquigarrow Od|a \wedge c \rightsquigarrow \neg Od|\ldots$ *is the set of all possible norms built from L;*
- $\mathscr{L}$ *contains all sentences* α *that variables L and N may assume;*
- *Function* $-$: $-(a) = \{\neg a\}$, $-(\neg a) = \{a\}$, $-(Oa) = \{O\neg a\}$, $-(b) = \{\neg b\}$, $-(\neg b) = \{b\}$, $-(Ob) = \{O\neg b\}$, $-(c) = \{\neg c\}$, $-(\neg c) = \{c\}$, $-(Oc) = \{O\neg c\}$, $-(d) = \{\neg d\}$, $-(\neg d) = \{d\}$, $-(Od) = \{O\neg d\}$, $-(e) = \{\neg e\}$, $-(\neg e) = \{e\}$, $-(Oe) = \{O\neg e\}$, $-(f) = \{\neg f\}$, $-(\neg f) = \{f\}$, $-(Of) = \{O\neg f\}$;

- *$\mathscr{R}$contains inference rules like for instance:*
  - $*$ $a, a \rightsquigarrow b \Rightarrow b$;
  - $*$ $a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow Od$;
  - $*$ $a, \neg c, a \wedge \neg c \rightsquigarrow d \Rightarrow \neg Od$

Consider now the knowledge base $K_3 = \{a, \neg b, c, e, a \wedge \neg b \rightsquigarrow O\neg d, c \wedge e \rightsquigarrow \neg Of\}$ where the norms state that

- *if people want to smoke in a closed space (a) and the public place has not smoking special secluded areas ($\neg b$), then smoking cannabis and tobacco in public places is forbidden (i.e., it is obligatory not to smoke cannabis and tobacco in public places ($O\neg d$));*
- *if people need to smoke cannabis on medical grounds (c) and cannabis is allowed for medical treatment (e), then smoking cannabis in cannabis establishments is permitted ($\neg Of$);*

We can construct the following arguments:

- $A_1 : a, \neg b, a \wedge \neg b \rightsquigarrow O\neg d \Rightarrow O\neg d$;
- $A_2 : c, e, c \wedge e \rightsquigarrow \neg Of \Rightarrow \neg Of$;

We have that $concl(A_1) = O\neg d$ and $concl(A_2) = \neg Of$. We conclude $Perm(LAS, K_3) = \{f\}$, i.e., smoking cannabis in cannabis establishments is permitted.

They behave like positive strong permissions in the classification of Makinson and van der Torre [7]. We can also include the obligations among the strong permissions. We leave the formal analysis for future research.

5. Dynamic legal systems

In this section, we illustrate how our definitions can be extended to the full ASPIC system, directly building on the work of Prakken and Sartor, where the set of norms is dynamic. We extend their system with the modal operator as before, and we illustrate it with an example. A formal analysis is left for future research.

Definition 16 (Terms, literals and norms) *Given four disjoint sets for variables, constants, functions including monadic function DMP, and predicate symbols, including binary predicate $=$ for equality, and monadic Valid and Applicable, and natural number for each element of functions and predicates representing their arity.*

$\tau ::= x \mid c \mid f(\tau, \ldots, \tau) \mid N(\phi)$ *where x in variables, c in constants, and f in functions and number of arguments corresponding to the arity of f.*

$L ::= P(\tau, \ldots, \tau) \mid \neg P(\tau, \ldots, \tau)$ *with P in predicates and number of arguments corresponding to the arity of P*

$M = L \mid OL \mid \neg OL$

$\phi ::= \forall(M \wedge \ldots \wedge M \rightsquigarrow M)$

$\alpha ::= M \mid \phi$

We add an inference rule for equality (it is not completely clear to us how this is dealt with in the Prakken and Sartor system, maybe it is supposed to work as a silent rule in the background). For the remainder of the definitions, see Prakken and Sartor [11].

Definition 17 *$\mathscr{R} = \mathscr{R}_s \cup \mathscr{R}_d$ is the set of inference rules, both strict ($\mathscr{R}_s$) and defeasible ($\mathscr{R}_s$), of the form $\phi_1, \ldots, \phi_n \rightarrow \phi$ and $\phi_1, \ldots, \phi_n \Rightarrow \phi$ respectively. $\mathscr{R}$ consists of:*

- *Equality: From* $\phi \wedge \tau = \tau' \rightarrow \phi[\tau/\tau']$, *where* $[\tau/\tau']$ *is a uniform substitution of expressions* τ *with expressions* τ'
- *Validity:* $Valid(N(\phi)) \rightarrow \phi$ *where* $N(\phi)$ *is the name of norm* ϕ*;*
- *Instantiation:* $\forall(\phi \rightsquigarrow \psi) \rightarrow (\phi \rightsquigarrow \psi)[x_1/t_1,\ldots,t_n]$, *where* $[x_1/t_1,\ldots,t_n]$ *is a substitution of variables* $x_1,\ldots,x_n$ *with ground terms* $t_1,\ldots,t_n$*;*
- *Defeasible modus ponens (DMP):* $\phi_1,\ldots,\phi_n,(\phi_1 \wedge \ldots \wedge \phi_n \rightsquigarrow \psi) \Rightarrow \psi$*;*
- *Undercutting:* $\neg Applicable(w) \rightarrow \neg DMP(w)$ *where w is the name of norm* $\phi_1 \wedge \ldots \wedge \phi_n \rightsquigarrow \psi$ *and* $DMP(w)$ *is the name of the DMP inference rule* $\phi_1,\ldots\phi_n,\phi_1 \wedge \ldots \wedge \phi_n \rightsquigarrow \psi \Rightarrow \psi$*;*

Example 5 *We have the following predicates:* $ICS(x)$*: "x is in a closed space",* $IMPS(x)$*: "x is in a private space",* $NSMG(x)$*: "x needs to smoke on medical grounds",* $EP(z)$*: "Parliament enacts norm z",* $BS(x)$*: "x is a burning substance",* $FIE(x,y)$*: "x inhales or exhales y". Prakken and Sartor include also predicate* $FFS(s)$*: "x is forbidden from smoking". As we introduce operator O, we model prohibitions by adding the following predicate:* $O\neg SM(x)$ *"it is obligatory for x to not smoke", i.e., x is forbidden from smoking. K, built according to Definition 16, is as follows (the first four premises give names for the norms):* $N(EP(w) \rightsquigarrow Valid(w)) = C_0$*;* $N(ICS(x) \rightsquigarrow O\neg SM(x)) = C_1$*;* $N(IMPS(x) \rightsquigarrow \neg(Applicable(C_1(x))) = C_2$*;* $N(NSMG(x) \rightsquigarrow \neg O\neg SM(x)) = C_3$*;* $Valid(C_0)$*;* $EP(C_1)$*;* $EP(C_2)$*;* $EP(C_3)$*;* $ICS(John)$*;* $ICS(Mary)$*;* $ICS(Tom)$*;* $NSMG(Mary)$*;* $IMPS(Tom)$*. As in [11], norm* C_0 *is valid (factual premise), norms* C_1*,* C_2 *and* C_3 *have been enacted by the Parliament, and agents John, Mary and Tom are in a closed space, while Mary needs to smoke on medical grounds and Tom is in a private space.*

Consider the application of the inference schemes in Definition 17. As norm C_0 *is valid, then we can conclude that all norms enacted by the Parliament are valid, thus norms* C_1*,* C_2 *and* C_3 *are valid. As a consequence, we can conclude that* $\forall(x)(ICS(x) \rightsquigarrow O\neg SM(x))$*, according to inference rule* $Valid(C_1) \rightarrow \forall(ICS(x) \rightsquigarrow O\neg SM(x))$*. Further conclusions could be* $\forall(x)(IMPS(x) \rightsquigarrow \neg(Applicable(C_1(x)))$*, according to inference rule* $Valid(C_2) \rightarrow \forall(MPS(x) \rightsquigarrow \neg(Applicable(C_1(x)))$*, and* $\forall(x)(NSMG(x) \rightsquigarrow \neg O\neg SM(x))$*, according to inference rule* $Valid(C_3) \rightarrow \forall(NSMG(x) \rightsquigarrow \neg O\neg SM(x))$*. The instantiation scheme allows to derive instances of the general norms. We have that norm* C_1 *is used to derive the instantiated norm* $ICS(John) \rightsquigarrow O\neg SM(John)$ *meaning that John is obliged not to smoke as he is in a closed space. in particular, this conclusion is driven by the DMP scheme that allows to apply the instantiated norms we have obtained, e.g., from DMP rule* $ICS(John), ICS(John) \rightsquigarrow O\neg SM(John) \Rightarrow O\neg SM(John)$ *we can conclude that* $O\neg SM(John)$*. Finally, the undercutting scheme links the inapplicability of a norm and the inapplicability of the corresponding DMP rule, e.g., from* $IMPS(Tom)$ *and* $IMPS(Tom) \rightsquigarrow \neg(Applicable(C_1(Tom))$ *we conclude that DPM rule on* C_1 *stating* $ICS(Tom), ICS(Tom) \rightsquigarrow O\neg SM(Tom) \Rightarrow O\neg SM(Tom)$ *is not applicable to Tom.*

6. Conclusions

Prakken and Sartor system [11] is defined as a tuple $S = \langle \mathscr{L}, -, \mathscr{R}, n \rangle$ where $\mathscr{L}$ is a logical language including symbols for predicates, functions, constants and variables, $=$ for equality, $\neg$ for negation and $\rightsquigarrow$ for normative conditionals, and the universal quantifier $\forall$, $\mathscr{R}$ is the set of inference rules, and n is the naming convention. A norm has the form $\forall(L_1 \wedge \ldots \wedge L_n \rightsquigarrow L)$, where $L_1, \ldots, L_n$ are literals. In particular, they define inference schemes for validity ($Valid(N(\phi)) \rightarrow \phi$), and applicability (i.e., undercutting, $\neg Applicable(w) \rightarrow \neg DMP(w)$). As future direction, the authors foster the extension of the framework by enriching the logical language with a formal account of modalities such as obligation. This is the issue we addressed in this paper. We extend their system with deontic modalities. The framework allows to reason over normative concepts like factual and deontic detachment, and to assess norms' equivalence. The properties of our framework are proved, and new concepts are illustrated by a running example.

The main technical contribution is to give a formal analysis, and a bridge to input/output logic, enabling a fruitful bridge. Compared to other input/output logics, we do not have weakening of the output or aggregation of obligations due to the clausal language. Comparison with prioritized input/output logics is a topic for future research. Compared to abstract normative systems [12], we have not only norms $L_1 \rightsquigarrow L_2$, but also rules with multiple literals in the body. To compare with other deontic logics we can define the inference relation in terms of consequence sets as usual ($K \models \phi$ iff $\phi \in Out(K)$).

Other frameworks for legal argumentation have been proposed like, among others, [9,1,5,10], but all of them concentrate on specific problems of reasoning with legal arguments, whilst the aim of our framework and of Prakken and Sartor as well is to integrate various aspects so far addressed separately towards a logic comprehensive model of dynamic legal argumentation.

Several future directions are planned. First, the introduction of deontic operators in the dynamic legal argumentation system allows us to extend it to capture further normative reasoning issues, like violations. Second, we will explore preferences in *LAS* as defined in ASPIC, and we will study how it relates to discussions on prioritized norms in defeasible deontic logic. Third, the challenge is to represent also aggregative deontic detachment, considering the generation of aggregative deontic conclusions.

References

[1] Kevin D. Ashley. *Modeling legal argument - reasoning with cases and hypotheticals*. Artificial Intelligence and Legal Reasoning. MIT Press, 1990.

[2] Diderik Batens. New arguments for adaptive logics as unifying frame for the defeasible handling of inconsistency. In *Paraconsistency: Logic and Applications*, volume 26, pages 101–122. Springer, 2013.

[3] Trevor Bench-Capon, Henry Prakken, and Giovanni Sartor. *Argumentation in Legal Reasoning*. Argumentation in Artificial Intelligence, Iyad Rahwan and Guillermo Simari Eds. Springer, 2010.

[4] Alexander Bochman. *Explanatory Non-monotonic Reasoning*. Advances in Logic: Volume 4. 2005.

[5] Thomas F. Gordon. The pleadings game. *Artif. Intell. Law*, 2(4):239–292, 1993.

[6] David Makinson and Leendert van der Torre. Input/output logics. *J. Phil. Logic*, 29(4):383–408, 2000.

[7] David Makinson and Leendert van der Torre. Permission from an input/output perspective. *J. Phil. Logic*, 32(4):391–416, 2003.

[8] Sanjay Modgil and Henry Prakken. A general account of argumentation with preferences. *Artif. Intell.*, 195:361–397, 2013.

[9] Henry Prakken. A logical framework for modelling legal argument. In *Procs of ICAIL*, pages 1–9, 1993.

[10] Henry Prakken and Giovanni Sartor. A dialectical model of assessing conflicting arguments in legal reasoning. *Artif. Intell. Law*, 4(3-4):331–368, 1996.

[11] Henry Prakken and Giovanni Sartor. Formalising arguments about norms. In *Procs of JURIX*, volume 259 of *Frontiers in Artificial Intelligence and Applications*, pages 121–130. IOS Press, 2013.

[12] Silvano Colombo Tosatto, Guido Boella, Leendert van der Torre, and Serena Villata. Abstract normative systems: Semantics and proof theory. In *Procs of KR*. AAAI Press, 2012.