Alessandro Giordani
Matteo Pascucci

Generalizing Deontic Action Logic

Abstract. We introduce a multimodal framework of deontic action logic which encodes the interaction between two fundamental procedures in normative reasoning: conceptual classification and deontic classification. The expressive power of the framework is noteworthy, since it combines insights from agency logic and dynamic logic, allowing for a representation of many kinds of normative conflicts. We provide a semantic characterization for three axiomatic systems of increasing strength, showing how our approach can be modularly extended in order to get different levels of analysis of normative reasoning. Finally, we discuss ways in which the framework can be used to capture other formalisms proposed in the literature, as well as to model searching problems in Artificial Intelligence.

Keywords: Action logic, Action types, Deontic reasoning, Normative conflicts, Searching problems.

1. Introduction

In the present article we provide a general multimodal framework for normative reasoning that combines insights from deontic logic, action logic, hyperintensional semantics and the study of non-deterministic systems.

The basic idea underlying the construction of the system is that we can decompose the analysis of the normative conduct of an agent into three levels. At the first level we have *action tokens* (like Alice's pushing a button in particular circumstances) which witness that the agent is interacting with the rest of the environment in a certain way. An action token is typically associated with an agent's conduct at a specific state of a system. At the second level we have *action types* (like pushing a button), which are used to classify the action tokens performed by the agent. Action tokens may be classified under more than one type, depending on the point of view from which the action is considered. At the third level we have *deontic values*, which allow us to determine whether an action is obligatory, permitted, etc. The deontic value of an action type depends on the normative source we

Presented by **Jacek Malinowski**; *Received* March 1, 2021

Studia Logica (2022) 110: 989–1033
https://doi.org/10.1007/s11225-022-09989-y

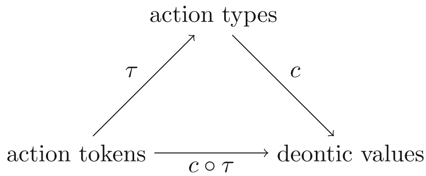


Figure 1. Normative assessment triangle

consider: different sources may assign different values to the same action type. Here, by a normative source we simply mean a set of norms —which may be, for instance, a portion of a legal code. However, we work under the assumption, specified also later in the formal part, that two different normative sources do not share any norm.

The interactions between these three levels can be described in terms of two procedures that are rooted in everyday normative reasoning: *conceptual classification* and *deontic classification*. The former leads from action tokens to action types, answering the question 'what did the agent do?'; the latter from action types to deontic values, answering the question 'how is the agent's conduct to be assessed?'. In the case where only one point of view and one normative source are at work, action tokens, action types, and deontic values are connected as per the *normative assessment triangle* (Figure 1).

In the triangle τ and c are classification functions; τ provides a conceptual classification of action tokens under a set of action types; c provides a deontic classification of action types under a set of deontic values. Conceptual classification is to be viewed as relative to a certain interpretation of a scenario. For instance, the same movements can be viewed as disconnected gestures in some scenario and as a dance in another one. Deontic classification is to be viewed as relative to a set of norms. For instance, the same kind of dance can be permitted in a certain society and prohibited in a more conservative one. The triangle suggests that the hard problem of assessing the deontic value of a particular action can be better addressed once it is decomposed into *two less difficult tasks*, which are typically solved when a normative source —e.g., a legal code— is designed (deontic classification) and when a trial is held (conceptual classification):

1. determine the action type under which the action falls (this is accomplished by a jury);
2. determine the deontic value of that action type (this is accomplished by a legislator).

When τ and c are available, we can conclude that an agent is acting in a wrong way if and only if she is performing

an action instantiating a type which is classified as wrong.

Though very intuitive, this idea is not fully implemented in current deontic systems. In fact, in the tradition of 'action deontic logic' (or 'deontic action logic') stemming from von Wright's foundational work (see, e.g., [3, 13, 22–25, 27]), the algebra of actions and the way in which the deontic value of composite actions depends on the values of their component is the main topic, but action tokens are not explicitly considered and no explicit reference to the structure of the set of deontic values is made. Similarly, in the tradition of dynamic deontic logic (see, e.g., [1, 16, 17]), neither the relation between action tokens and action types nor the structure of the set of deontic values is represented. Finally, in the $STIT$ tradition (see, e.g., [4, 8]) action tokens are primarily studied, while their relation to action types and the structure of the set of deontic values is not represented (but for interesting development in this direction see [9, 14]).

The primary aim of this paper is to provide systems of modal logic which capture the basic framework just sketched. We will start with a basic system where only one agent, one interpretation of a scenario and one normative source are available. Then, we introduce further complications by enriching the formal framework so as to account for multiple normative sources. Finally, we provide two applications related to the study of conflicts in a setting where different normative sources are at work and the implementation of searching problems.

2. Initial System

2.1. Formal Language

Our first system is a version of deontic action logic whose language $\mathcal{L}_0$ is built over a set $\mathtt{Var}$ of *elementary propositions*, denoted by p, q, r, a set $\mathtt{Agt}$ of agents, denoted by i, j, k, etc., and a set $\mathtt{Act}$ of *elementary action types*, denoted by $\mathfrak{a}$, $\mathfrak{b}$, $\mathfrak{c}$, etc. Elementary action types are descriptions of actions without reference to any temporal parameters or specific circumstances, such as 'paying taxes', 'playing chess', 'opening this window' etc. These action types can be combined via algebraic operations, so as to get the set $\mathtt{Act}^*$ of all action types, denoted by α, β, γ, etc., in accordance with the following grammar, where $\mathfrak{a} \in \mathtt{Act}$:

$$\alpha ::= \mathfrak{a} \mid 1 \mid \overline{\alpha} \mid \alpha \sqcup \alpha \mid \alpha \sqcap \alpha$$

We stipulate the following definition: $0 =_{\text{def}} \overline{1}$. The term $\overline{\alpha}$ denotes the complement of action type α, that is, the type of any action not instantiating α; $\alpha \sqcap \beta$ denotes the conjunction of action types α and β, that is, the type of any action instantiating both α and β; $\alpha \sqcup \beta$ the disjunction of action types α and β, that is, the type of any action instantiating either α or β; 1 denotes the type of every possible action token and 0 the type of no possible action token.

The set $\texttt{Wff}$ of well-formed formulas of the language $\mathcal{L}_0$, denoted by ϕ, ψ, χ, etc., is defined by the grammar below, where $i \in \texttt{Agt}$, $\alpha \in \texttt{Act}^*$ and $p \in \texttt{Var}$:

$$\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \mathbf{A}\phi \mid [i]\phi \mid \Box_i\phi \mid done_i(\alpha) \mid \mathbf{O}\alpha$$

The boolean operators $\neg$ and $\wedge$ represent negation and conjunction, respectively. The modal operators $\mathbf{A}$, $[i]$, $\Box_i$, $done_i$ and $\mathbf{O}$ can be divided into two groups. The first three are functions of the type $(\texttt{Wff}, \texttt{Wff})$: they take a well-formed formula as an input and produce a well-formed formula as an output. The last two are functions of the type $(\texttt{Act}^*, \texttt{Wff})$: they take an action type as an input and produce a well-formed formula as an output. The dual operators $\mathbf{E}$, $\langle i\rangle$, $\Diamond_i$ and $\mathbf{P}$ are defined as follows: $\mathbf{E}\phi =_{\text{def}} \neg\mathbf{A}\neg\phi$, $\langle i\rangle\phi =_{\text{def}} \neg[i]\neg\phi$, $\Diamond_i\phi =_{\text{def}} \neg\Box_i\neg\phi$ and $\mathbf{P}\alpha =_{\text{def}} \neg\mathbf{O}\overline{\alpha}$.

Language $\mathcal{L}_0$ will be interpreted in terms of relational models equipped with a deontic algebra in Section 2.4. For the time being, we only anticipate the basic intuitions behind the interpretation of formulas whose main operator is a modal one, on the standard assumption that the truth-value of a formula depends on the state in which it is evaluated.

1. A formula like $\mathbf{A}\phi$ says that ϕ is true at every state, so that $\mathbf{A}$ is the universal modality, whereas $\mathbf{E}\phi$ says that ϕ is true at some state.
2. A formula like $[i]\phi$ says that ϕ is a consequence of the conduct of agent i at the state of evaluation, i.e., that the action i is performing can only result in states where ϕ is the case, while $\langle i\rangle\phi$ says that ϕ is compatible with the current conduct of i. These operators convey the idea of a temporal transition from the state of evaluation to states in its immediate future. Accordingly, since the action of an agent is typically unable to determine a unique successive state, we assume that the same conduct can result in different states. As a consequence, the truth of $[i]\phi$ at a state depends on the truth of ϕ at all states possibly resulting from the agent's conduct (Figure 2).
   Similarly, $\langle i\rangle\phi$ is true at a state precisely when ϕ is true at some states possibly resulting from the agent's conduct.

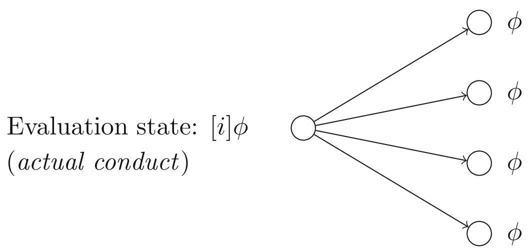


Figure 2. States that are possible given the agent's conduct

3. A formula like $\Box_i\phi$ says that ϕ is, at the state of evaluation, an unavoidable state of affairs for agent i, i.e., that all the actions i is able to perform are performed in states where ϕ is the case, while $\Diamond_i\phi$ says that it is currently possible for i to act in a state where ϕ holds true. The intuition behind this is that, at a given state, an agent is able to give rise to various conducts resulting in different sets of states —e.g. in front of a crossroad, she can go in different directions. Thus, the truth of $\Box_i\phi$ at a state depends on the truth of ϕ at all states where the agent gives rise to a possibly alternative conduct. This allows us to capture the notion of an unavoidable consequence of the agent's conduct. Indeed, we can see that ϕ is an unavoidable consequence of the conduct of i if and only if ϕ is true at all states resulting from all possible agent's conducts, that is, if and only if $\Box_i[i]\phi$ is true —e.g. if ϕ is true no matter what the agent can do at the crossroad (Figure 3).
   Similarly, $\Diamond_i\phi$ is true at a state precisely when ϕ is true at some states where the agent adopts a possibly different conduct and $\Diamond_i[i]\phi$ is true at a state precisely when ϕ is a consequence of a possible conduct of i.
4. A formula like $done_i(\alpha)$ says that an action of type α has just been performed by the agent. It is worth noting that in the present system action

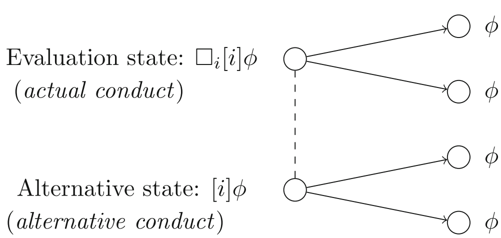


Figure 3. States that are possible given the possible conducts

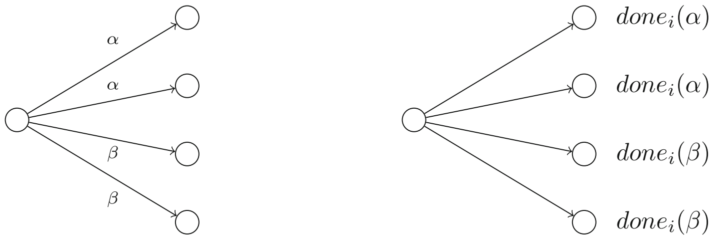


Figure 4. Transitions

types are represented by sets of states, whereas in systems based on dynamic logic, like the one proposed in [17], action types are represented by labeled transitions, according to the idea that a set of α-transitions represents all the ways in which action type α can be performed at a given state. The difference between the two approaches lies in the fact that in labeled transition systems states are assumed to be memoryless, while in the present system states store the information about the most recent actions from which they result. A consequence of this is that states resulting from different actions are always different, precisely because they have memory of their origin. The present approach is slightly more general, since α-transitions can be defined without difficulty in terms of transitions landing to α-states.

Figure 4 depicts α- and β-transitions both as represented in labeled transition systems and as represented in terms of generic transitions ending in states where actions of type α and β have just been accomplished.

What is particularly interesting is that in the present system the *ability* of an agent can be represented in terms of the actions that the agent performs in alternative states. The idea is that an agent is able to perform a certain action precisely when there is an alternative state, consistent with the ability of the agent, where the agent is actually performing that action [5]. First, note that the fact that i is doing α is described by a formula like $[i]done_i(\alpha)$, stating that the actual conduct of i has $done_i(\alpha)$ as a consequence, that is, will result in states where $done_i(\alpha)$ holds (Figure 5).

Then, note that the fact that i is able to do β in a state in which she is actually doing α can be described by a formula like $\Diamond_i[i]done_i(\beta)$.

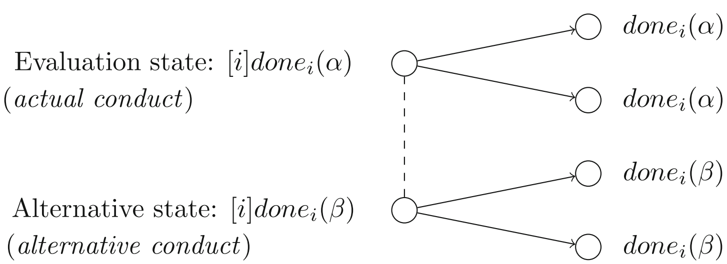


Figure 5. States that are consequences of possible conducts

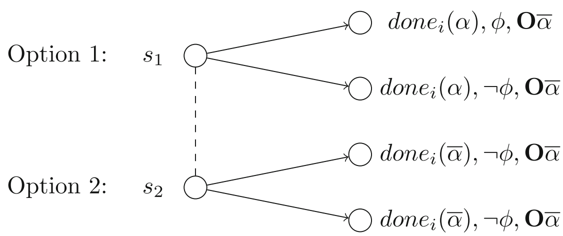


Figure 6. States that are possible given the possible conducts

5. Lastly, $\mathbf{O}\alpha$ says that α is obligatory given the set of norms that are in force at the current state, while $\mathbf{P}\alpha$ says that none of such norms precludes the agent from doing α.

Illustration. Let us consider the crucial point in Antigone's story. Antigone has two options: burying her brother Polynices or not. She has no other possibility and, whatever she does, her conduct will be an instance of burying her brother (action type α) or an instance of not burying her brother (action type $\overline{\alpha}$). Furthermore, some soldiers are watching over the body of Antigone's brother. If she decides to bury him, it is possible that they catch her (ϕ). Finally, in burying her brother she is also violating the civil law ($\mathbf{O}\overline{\alpha}$). She opts for the first option.

As shown in Figure 6, at s_1 the following propositions are true:

1. $[i]done_i(\alpha)$: Antigone is burying her brother.
2. $\Diamond_i[i]done_i(\overline{\alpha})$: Antigone could have acted differently.
3. $\langle i\rangle\phi$: it is possible, given what she is doing, that she gets caught.
4. $\langle i\rangle\neg\phi$: it is possible, given what she is doing, that she does not get caught.

5. $\Diamond_i[i]\neg\phi$: she could have acted so as to avoid getting caught.
6. $[i]done_i(\alpha) \wedge \mathbf{O}\overline{\alpha}$: she is breaking the law.
7. $\Diamond_i[i]done_i(\overline{\alpha}) \wedge \mathbf{O}\overline{\alpha}$: she could have avoided breaking the law.
8. $\Box_i[i](done_i(\overline{\alpha}) \rightarrow \neg\phi)$: not burying excludes getting caught.
9. $\Diamond_i\langle i\rangle(done_i(\alpha) \wedge \phi)$: burying does not exclude getting caught.

As a final remark note that, while, for the sake of a simpler exposition, we will be mostly working with one agent only, whence assuming that $\texttt{Agt} = \{i\}$, the language we are using allows for representing problems of normative reasoning in a multi-agent context.

2.2. Expressive Power

Language $\mathcal{L}_0$ is powerful enough to express the fact that an agent is acting in a wrong way. As we saw, an agent is acting in a wrong way precisely when *she is performing an action instantiating a type which is classified as wrong*, i.e., precisely when the following formula holds:

$$[i]done_i(\alpha) \wedge \mathbf{O}\overline{\alpha}$$

Indeed, a formula like $[i]done_i(\alpha)$ states that having performed α is a result of i's current conduct, i.e., that one of the actions i is performing is of type α, while $\mathbf{O}\overline{\alpha}$ states that it is obligatory to avoid doing α, and so that α is wrong according to a certain deontic standard (in $\mathcal{L}_0$ a single normative source is taken into account). In more detail, we are in a position to distinguish different kinds of wrong action, by enriching and modifying this preliminary definition. For instance, let us consider the differences between schema 1, our preliminary definition, and schemata 2 and 3 below:

1. $[i]done_i(\alpha) \wedge \mathbf{O}\overline{\alpha}$
2. $\Box_i[i]done_i(\alpha) \wedge \mathbf{O}\overline{\alpha}$
3. $\Box_i[i]done_i(\alpha_1 \sqcup \alpha_2) \wedge [i]done_i(\alpha_1) \wedge \mathbf{O}(\overline{\alpha_1}) \wedge \mathbf{O}(\overline{\alpha_2})$

All three schemata concern cases in which a wrong action is performed. Schema 1 leaves open the possibility of whether agent i is able, in the circumstances she is acting, to avoid performing an action of type α, even if she ends up performing such an action. In the case of schema 2, the agent is unable to avoid doing α, and so she performs a wrong action in a condition where she is forced to do that. In the case of schema 3, the agent performs α_1 and it is left open whether she could avoid performing α_2, but she could not avoid performing a wrong action, being forced to perform an action either of type α_1 or of type α_2, and so being in a condition where she is unable to avoid doing something wrong. To the best of our knowledge, no other extant system of action deontic logic is rich enough to provide such an account of wrong doing.

An important expressive gain with respect to other formal frameworks for deontic reasoning is the fact that, due to the possible interactions between the various operators available, $\mathcal{L}_0$ can also keep track of the *distinction between wrong actions whose performance will lead from the state of evaluation to a new state (hence, whose result will obtain in the new state) and wrong actions whose performance led from a previous state to the moment of evaluation*. In the first case the normative classification of an action has *prospective value* (i.e., it is future-oriented), whereas in the second case it has *retrospective value* (i.e., it is past-oriented). Schemata 1–3 described above concern the first case, whereas schema 4 below concerns the second case:

4. $done_i(\alpha) \wedge \mathbf{O}(\overline{\alpha})$.

Furthermore, we can mutually exchange α and $\overline{\alpha}$ in schemata 1–4 in order to represent an agent's negligence with respect to types of actions considered obligatory. This offers a complementary perspective on what can count as a normatively wrong conduct.[1]

Finally, it is worth noting that the multimodal nature of $\mathcal{L}_0$ allows one to express more complex deontic notions via combinations of two or more operators. As a simple example, consider the following definitions of operators of type $((\texttt{Act}^*, \texttt{Wff}), \texttt{Wff})$ to express *conditional obligations*:

1. $\mathbf{O}(\alpha \mid \phi) =_{\text{def}} \mathbf{E}\phi \wedge \mathbf{A}(\phi \rightarrow \mathbf{O}\alpha)$.
2. $\mathbf{O}_i(\alpha \mid \phi) =_{\text{def}} \Diamond_i \phi \wedge \Box_i(\phi \rightarrow \mathbf{O}\alpha)$.

Here ϕ is the antecedent of the conditional and α its consequent. While $\mathbf{O}(\alpha \mid \phi)$ prescribes that action type α be realized *in general* under the condition that ϕ holds true, $\mathbf{O}_i(\alpha \mid \phi)$ prescribes that α be realized *in the circumstances* specified by the state of evaluation, under the same condition.[2]

[1] When the language includes many agents, it is also possible to express conflicts that result from their interaction, such as: $\Box[i,j](done_i(\alpha) \vee done_j(\beta)) \wedge \mathbf{O}(\overline{\alpha}) \wedge \mathbf{O}(\overline{\beta})$. In the case at hand, it is unavoidable that either agent i or agent j ends up performing a prohibited action.

[2] The logic related to such conditionals is very rich. In system $\mathbf{L}_0$ the previous notions have specific properties due to the interaction with the axioms and rules available. For instance, axioms **O1**–**O5** determine corresponding deductive properties of the consequent of a conditional obligation: $\neg\mathbf{O}(0 \mid \phi)$, $\mathbf{O}(\alpha \mid \phi) \wedge \mathbf{O}(\beta \mid \phi) \rightarrow \mathbf{O}(\alpha \sqcap \beta \mid \phi)$. A thorough investigation of these conditional notions is left for future work.

Summing up, the combination of alethic, agentive and deontic modal operators provides us with a quite flexible framework, whose expressive power goes beyond that of languages of deontic logic based on an algebra of actions, since the latter usually include only operators of the type $(\texttt{Act}^*, \texttt{Wff})$ ([22–24]). In addition, $\mathcal{L}_0$ is also more expressive than those including only operators of the type $(\texttt{Wff}, \texttt{Wff})$, such as the language of standard deontic logic (SDL). This approach can then be located within the tradition of proposals which tend to unify insights from agency logic, such as the explicit reference to agents, and from action logic, such as the explicit reference to action types (ADL).

2.3. System $\mathbf{ADeL_0}$

We here introduce a minimal logical system over the formal framework adopted. It is called $\mathbf{ADeL}_0$ (the acronym standing for 'Action Deontic Logic') and axiomatized via the following list of deductive principles:

0. Axioms and rules of the Classical Propositional Calculus;
1. Modal axioms and rules of system $KT5$ for the operator $\mathbf{A}$;
2. Modal axioms of system $KT5$ for the operator $\Box_i$;
3. Modal axioms of system KD for the operator $[i]$;
4. The following axioms for the operator $done_i$:

   D1 $done_i(1)$
   D2 $done_i(\alpha) \leftrightarrow \neg done_i(\overline{\alpha})$
   D3 $done_i(\alpha \sqcup \beta) \leftrightarrow done_i(\alpha) \vee done_i(\beta)$
   D4 $done_i(\alpha \sqcap \beta) \leftrightarrow done_i(\alpha) \wedge done_i(\beta)$

5. The following axioms for the operator $\mathbf{O}$:

   O1 $\mathbf{O}(\alpha \sqcap \alpha) \leftrightarrow \mathbf{O}\alpha$
   O2 $\mathbf{O}(\alpha \sqcap \beta) \leftrightarrow \mathbf{O}(\beta \sqcap \alpha)$
   O3 $\mathbf{O}(\alpha \sqcap (\beta \sqcap \gamma)) \leftrightarrow \mathbf{O}((\alpha \sqcap \beta) \sqcap \gamma)$
   O4 $\neg\mathbf{O}0$
   O5 $(\mathbf{O}\alpha \wedge \mathbf{O}\beta) \rightarrow \mathbf{O}(\alpha \sqcap \beta)$

6. The following bridge axioms:

   B1 $\mathbf{A}\phi \rightarrow \Box_i\phi$
   B2 $\mathbf{A}\phi \rightarrow [i]\phi$
   B3 $[i]\phi \rightarrow [i]\Box_i\phi$

Axioms **O1**–**O5** play the same role as algebraic equations employed in related approaches, such as [13]. Here we opt for a logical —rather than algebraic—

representation of the properties of $\sqcap$, and for this reason we do not make use of the notion of identity between algebraic terms.[3] The deductive properties of the operator $\mathbf{O}$ in this minimal system are very weak and allow one to avoid issues with many traditional paradoxes of deontic reasoning. Indeed, in view of the semantics introduced below, it is not difficult to see that it is not possible to infer $\mathbf{O}(\alpha \sqcup \beta)$ from $\mathbf{O}\alpha$, (thus avoiding Ross's paradox) and it is not possible to infer $\mathbf{O}\alpha$ from $\mathbf{O}(\alpha \sqcap \beta)$ (thus avoiding the Good Samaritan Paradox). Axioms **B1** and **B2**, together with the rule of necessitation for $\mathbf{A}$ (if ϕ is provable, then $\mathbf{A}\phi$ is provable as well) allow us to obtain the rule of necessitation for $\Box_i$ and for $[i]$.[4]

The choice of a $KT5$ logic for $\Box_i$ and $\mathbf{A}$ is due to their intended meaning, as explained in Section 2.1. In fact, both behave as universal quantifiers over states that constitute an equivalence class (the class of all states representing alternative conducts of an agent and the class of all states in a model, respectively). By contrast, the logic of the operator $[i]$ is KD, since such operator behaves as a universal quantifier over the states that are in the immediate future of the state of evaluation; the relation of being 'in the immediate future of' is here simply characterized as serial.

2.4. Semantics

Language $\mathcal{L}_0$ will be interpreted on relational models equipped with a deontic algebra. Models for the logic $\mathbf{ADeL}_0$ constitute a proper subclass of all models for the language $\mathcal{L}_0$. For the sake of brevity, we will refer to the latter as $\mathbf{ADeL}_0$-models and their peculiar properties will be specified at various points within the general definition of models for language $\mathcal{L}_0$ provided below.[5]

---

[3] While the two approaches ultimately yield similar results, the present approach is consistent with the idea that action types are conceptual constructions, and so that they are to be identified in terms of their components and the way of their composition. In this sense, for instance, $\alpha \sqcap \beta$ and $\beta \sqcap \alpha$ turn out to be different types, even if this difference has no significant impact on the resulting logic.

[4] In a multi-agent language we can take the axiomatic basis of $\mathbf{ADeL}_0$ as a minimal system that does not impose any restriction on the possible interactions among agents. However, a more sophisticated approach can be obtained by extending the axiomatic basis of $\mathbf{ADeL}_0$ with an axiom-schema aggregating the possibility of action-types performed by groups of agents. In such way, we can also incorporate the basic principles of $STIT$-logic: for instance, the principle known as *independence of agents* becomes $(\langle A\rangle done_i(\alpha) \wedge \langle A\rangle done_j(\beta)) \rightarrow \langle A\rangle(done_i(\alpha) \wedge done_j(\beta))$, where $A \subseteq \mathsf{Agt}$ is the relevant group of agents.

[5] For the sake of a concise exposition, the following definitions will make reference to a single agent i. However, all these definitions can be straightforwardly generalized in order to deal with many agents.

Definition 1. Model for $\mathcal{L}_0$.

A model for $\mathcal{L}_0$ is a tuple $M = \langle W, R_i, A_i, D_i, \mathbb{D}, V \rangle$, whose elements are to be thought of as follows:

- W is a non-empty set of possible states denoted by w_1, w_2, w_3, etc. (or w, v, u, etc.).
- $R_i : W \to \wp(W)$ is the function returning the set of states that are accessible given the agent's conduct, so that $R_i(w)$ is the set of states that can result from her conduct at w. In the case of $\mathbf{ADeL}_0$-models, R_i is such that $R_i(w) \neq \varnothing$, for every $w \in W$.

Comments. When an agent acts a certain way in a state w different states can be produced depending on the way other agents are acting and the occurrence of non-deterministic events.

- $A_i : W \to \wp(W)$ is the function returning the set of states that are accessible given the ability of the agent, so that $A_i(w)$ is the set of states where her conduct is one of the possible conducts she is able to select at w. In the case of $\mathbf{ADeL}_0$-models A_i is such that

  $A1$. $v \in A_i(w) \Leftrightarrow A_i(v) = A_i(w)$ for every $w, v \in W$.
  $A2$. $v \in R_i(w) \Rightarrow A_i(v) \subseteq R_i(w)$ for every $w, v \in W$.

Comments. We want to be able to say that an agent which is acting in a certain way could have acted differently. Thus, the state w in which the agent is acting in a certain way is associated to states in a set $A_i(w)$ where the agent is acting in a possibly alternative way. The idea is that, if many courses of action are open to the agent at w, then $A_i(w)$ contains the course that the agent is actually following plus all the alternative courses. Condition $A1$ formalizes the idea that, if v is one of the states where the agent is acting in a possibly alternative way, then the alternatives of v coincide with the the alternatives of w. E.g., if the agent is able to push a button, then the A_i-accessible states are the one where she pushes the button and the one where she does not push it, no matter what she does. Hence, if w is the state where she pushes the button and v is the state where she avoids pushing it, $A_i(w) = A_i(v)$. Condition $A2$ formalizes the idea that the agent is not able to change the result of an action: past actions are settled. Therefore, if v is one of the states that can result from i's conduct at w, then all the states where i is acting in a possibly alternative way relative to v are states that can result from i's conduct at w. E.g., if the agent has pushed a button, then the states where i is acting in a possibly alternative way after having pushed the button are all states where the button has been pushed.

- $D_i : \texttt{Act}^* \to \wp(W)$ is the function that determines which action types are performed at a state according to the following conditions:

  $D1$. $D_i(1) = W$
  $D2$. $D_i(\overline{\alpha}) = W - D_i(\alpha)$
  $D3$. $D_i(\alpha \sqcap \beta) = D_i(\alpha) \cap D_i(\beta)$
  $D4$. $D_i(\alpha \sqcup \beta) = D_i(\alpha) \cup D_i(\beta)$

Comments. Action types are associated to possible states in such a way that the action algebra is completely mirrored in the algebra of states. Hence, since $D_i(\alpha)$ is the set of states where α has just been realized and $D_i(\beta)$ is the set of states where β has just been realized, $D_i(\alpha) \cap D_i(\beta)$ is the set of states where $\alpha \sqcap \beta$ has just been realized, given that the realization of $\alpha \sqcap \beta$ coincides with the realization of both α and β, and $D_i(\alpha) \cup D_i(\beta)$ is the set of states where $\alpha \sqcup \beta$ has just been realized, given that the realization of $\alpha \sqcup \beta$ coincides with the realization of one of α and β. Similarly, $\overline{\alpha}$ is realized at all the states where α is not realized. Finally, 1 is realized at all the states altogether, and so 0, as defined above, is realized at no possible state.

- $\mathbb{D}$, called a *deontic system*, is a tuple $(DV, SV, \wedge_{\mathbb{D}}, c)$, which is based on a deontic algebra $(DV, \wedge_{\mathbb{D}})$.

  $\mathbb{D}1$. $DV \neq \emptyset$ is a set of *deontic values*;
  $\mathbb{D}2$. $\wedge_{\mathbb{D}}$ is a binary operation on $\mathbb{D}$ called *meet*;
  $\mathbb{D}3$. $(DV, \wedge_{\mathbb{D}})$ is a *meet semilattice*;
  $\mathbb{D}4$. $SV \subseteq DV$ is a set of *selected deontic values*;
  $\mathbb{D}5$. $c : W \times \texttt{Act}^* \to DV$ is a deontic assignment function.

In the light of $\mathbb{D}3$, $\wedge_{\mathbb{D}}$ is an idempotent, commutative, associative operation on the set DV of deontic values. In respect of $\mathbf{ADeL}_0$-models, SV and c are required to satisfy the additional properties below.

Conditions on SV:

$s1$: $SV \neq \varnothing$

$s2$: if $d \in SV$ and $d' \in SV$, then $d \wedge_{\mathbb{D}} d' \in SV$

Conditions on function c:

$c1$: $c(w, \alpha \sqcap \alpha) = c(w, \alpha)$

$c2$: $c(w, \alpha \sqcap \beta) = c(w, \beta \sqcap \alpha)$

$c3$: $c(w, \alpha \sqcap (\beta \sqcap \gamma)) = c(w, (\alpha \sqcap \beta) \sqcap \gamma)$

$c4$: $c(w, 0) \notin SV$;

$c5$: if $c(w, \alpha), c(w, \beta) \in SV$, then $c(w, \alpha \sqcap \beta) \in SV$

Comments. The *deontic system* $\mathbb{D} = (DV, SV, \wedge_{\mathbb{D}}, c)$ is one of the novel elements of the present framework. DV is a set of deontic values and $SV \subseteq DV$ is a set of selected values. Here we are opting for a *positive* use of SV: the selected values are the ones that allow us to define which action types are obligatory; we could also opt for a *negative* use of SV and select the values that allow us to define which action types are prohibited. Since it is often the case that obligations, or prohibitions, have a different strength, SV will include, in general, more than one value. Indeed, an important advantage of having many values in the set DV and in its subset SV is the possibility of representing alternative solutions to normative conflicts, as will be extensively discussed below in Section 4. In addition, even if we will not pursue this line here, using different selected values is convenient for distinguishing between actions that are both obligatory and optimal and actions that are obligatory but sub-optimal, a distinction that occurs very often in normative reasoning and that can be used to address some paradoxes of deontic reasoning, as explained in [12, 15].[6]

In the light of $s1$–$s2$ values in SV are such that $d \in SV$ and $d' \in SV$ imply $d \wedge_{\mathbb{D}} d' \in SV$, since we want to be able to say that, if a combination $d \wedge_{\mathbb{D}} d'$ of two values is not in SV, this is because one of the combined values is not in SV. In contrast, we do not require that SV is a filter on DV: specifically, we do not require that, if $d \in SV$ and d' is more valuable than d, then $d' \in SV$, so that, when selecting a value, we select all the more valuable values as well. This choice is justified by the fact that we want to allow for the existence of elements of DV that are highly valuable but do not induce obligations. This may be the case, for instance, when the system is used to represent supererogatory statements.

Finally, c is the function that classifies action types in terms of their values. Since, in general, it is possible for the same action type to be classified in different ways at different states, given the conditions in which the actions are performed, c is assumed to be parameterized relative to the states in W. The first three conditions on c ensure that this classification is independent

[6] As a typical case, consider the following scenario: students ought to return all borrowed books to the University Library by the end of the current academic year (this can be thought of as an obligation expressing what is normatively optimal, and so it receives a maximal value d in SV); however, if any student fails to do so, she ought to return the borrowed books during the first week of the next academic year (this can be thought of as an obligation expressing what is normatively sub-optimal and, as such, it should receive a value d' in SV which is ranked as lower than d).

from the way the types are presented, which reflects the natural assumption that, at the algebraic level, the following would hold:

- α coincides with $\alpha \sqcap \alpha$;
- $\alpha \sqcap \beta$ coincides with $\beta \sqcap \alpha$;
- $\alpha \sqcap (\beta \sqcap \gamma)$ coincides with $(\alpha \sqcap \beta) \sqcap \gamma$.

The motivation of the last two conditions on c (i.e., $c4$ and $c5$) is the following. First, $c(w, 0) \notin SV$ excludes that impossible actions possess a positive deontic value. In fact, there is no sense in admitting that: why should we spend our time to assign a positive value to an action that is impossible to perform? Next, $c5$ wants to capture the idea that the realization of $\alpha \sqcap \beta$ is at least as demanding as the realization of α and β separately, and so it is at least as valuable from a deontic perspective. In doing that we prevent situations where the joint performance of two actions with a positive deontic value does not receive, as a whole, a positive value.

- $V : \texttt{Var} \to \wp(W)$ is the function that determines which elementary propositions are true at a state. Accordingly, $V(p)$ is the set of states where p is true.

Comments. This is the usual modal valuation function.

Dropping the function V from the ordered tuple describing a model M, we get an ordered tuple describing the *frame* on which M is based.

Definition 2. A frame for $\mathcal{L}_0$ is a tuple $F = \langle W, R_i, A_i, D_i, \mathbb{D}\rangle$ whose elements can be described as in Definition 1. An $\mathbf{ADeL}_0$-frame is a frame for $\mathcal{L}_0$ whose elements satisfy the additional specific conditions mentioned in Definition 1.

The truth-conditions of formulas in $\mathcal{L}_0$ are now defined as follows.

Definition 3. Truth in a model for $\mathcal{L}_0$.

A formula ϕ is true at a state w in the domain of a model M (in symbols, $M, w \models \phi$) precisely under the following conditions:

$M, w \models p_i \;\Leftrightarrow\; w \in V(p_i)$, for every $p_i \in \texttt{Var}$

$M, w \models \neg\phi \;\Leftrightarrow\; M, w \not\models \phi$

$M, w \models \phi \wedge \psi \;\Leftrightarrow\; M, w \models \phi$ and $M, w \models \psi$

$M, w \models \mathbf{A}\phi \Leftrightarrow \forall v(M, v \models \phi)$

$M, w \models [i]\phi \;\Leftrightarrow\; \forall v(v \in R_i(w) \Rightarrow M, v \models \phi)$

$M, w \models \Box_i\phi \;\Leftrightarrow\; \forall v(v \in A_i(w) \Rightarrow M, v \models \phi)$

$M, w \models done_i(\alpha) \Leftrightarrow\ w \in D_i(\alpha)$

$M, w \models \mathbf{O}\alpha \Leftrightarrow\ c(w, \alpha) \in SV$

In system $\mathbf{ADeL}_0$ we do not exploit the full power furnished by $\mathbb{D}$, since we do not make any assumption on the set of non-selected deontic values, that is, $DV - SV$: given the positive use of SV, this choice corresponds to focusing on the simple opposition between obligatory action types and non-obligatory action types. Furthermore, given that the descriptive power of the language of $\mathbf{ADeL}_0$ relative to deontic conditions is limited to formulas like $\mathbf{O}\alpha$, we are allowed, as we will see below, to define a canonical model for $\mathbf{ADeL}_0$ where DV contains only two values, one of which in SV. However, in a more general setting, the semantic framework just proposed can be used to interpret more powerful languages and capture more deontic distinctions, as shown in Sections 4 and 5. For instance, a standard instantiation of $\mathbb{D}$ is $DV = \{\mathbf{o}, \mathbf{f}, \mathbf{i}\}$, where $\mathbf{o}$ = obligatory, $\mathbf{f}$ = forbidden and $\mathbf{i}$ = indifferent, and $SV = \{\mathbf{o}\}$. Having in mind this instantiation, one can formulate extensions of $\mathbf{ADeL}_0$ that include deductive principles able to capture formal relations between permitted and prohibited actions.

Definition 4. Satisfiability and validity.

A formula ϕ is *satisfiable* in a model M iff there is some state w in the domain of M s.t. $M, w \models \phi$. A formula ϕ is *valid* in a model M iff for all states w in the domain of M we have $M, w \models \phi$. A formula ϕ is satisfiable (resp. valid) in a class of frames C iff it is satisfiable (resp. valid) in some (resp. every) model base on some (resp. every) frame in C.

Comparisons with Similar Approaches. Before closing this section, let us briefly consider the connection between our way of interpreting the accomplishment of an action and some related ways in the literature, focusing on approaches that are based on relational semantics and that employ the ideas developed in the tradition of dynamic logic [17] and $STIT$-logic [7].[7] Then, in Section 5.2, when studying the representation of search problems, we will prove a general theorem showing how transition systems can be represented in the present framework.

In [17] the effect of the accomplishment of an action α at a given state w is captured by a relation $R_\alpha(w)$, whose instances are thought of as labeled transitions. Intuitively, we have $v \in R_\alpha(w)$ when v is accessible from w

[7]These are the approaches that are more similar to the one presented here. For further comparison, see [18].

via action α, i.e., when v is one of the states that may obtain after action α is performed. Notice that in the framework of dynamic logic there is no explicit reference to agents; thus, while it is possible to say that something is the consequence of an action, it is not possible to say that something is the consequence of an agent's behaviour. A refinement of this approach, making explicit reference to agents, is exploited in [7]. In that context, following the intuitions underlying the construction of $STIT$-logic, and rephrasing these in a semantics for multimodal logic, different constraints are introduced to model the relations between the actions of different agents, which are represented as $R_{i:\alpha}$, $R_{j:\beta}$, etc. Relations of the latter kind can be defined in our setting as follows.

Definition 5. (*Relation* $R_{i:\alpha}$) Given a model $M = \langle W, R_i, A_i, D_i, \mathbb{D}, V\rangle$, the relation $R_{i:\alpha}$, for $\alpha \in \texttt{Act}^*$ and $i \in \texttt{Agt}$, is such that:

$v \in R_{i:\alpha}(w)$ iff $v \in R_i(x) \cap D_i(\alpha)$ for some x such that $x \in A_i(w)$

Hence, v is accessible from w via action α when it is possible for agent i to act in such a way that her action token is both of type α and leading to v. This definition shows that our approach is general enough to recover the ideas at the basis of previous systems and open a line of research aimed at studying the connections between these systems and the present one. Here we just notice that every relation $R_{i:\alpha}$ can be syntactically associated with an operator saying that agent i can perform an action of type α. Such an operator can be defined in two steps as follows:

Step 1: $do_i(\alpha) := [i]done_i(\alpha)$. A formula like $do_i(\alpha)$ says that the agent is successfully doing an action of type α, since her conduct necessarily leads to a state where such an action is performed.

Step 2: $can_i(\alpha) := \Diamond_i do_i(\alpha)$. A formula like $can_i(\alpha)$ says that the agent has the ability of doing an action of type α, since she is able to entertain a conduct that necessarily leads to a state where such an action is performed.

We can finally see that the following truth-conditions hold:

$M, w \models can_i(\alpha)$ iff there is some $v \in R_{i:\alpha}(w)$

highlighting the idea underlying the introduction of $R_{i:\alpha}$.

2.5. Soundness

Theorem 1. (Soundness) *Every theorem of* $\mathbf{ADeL}_0$ *is valid in the class of* $\mathbf{ADeL}_0$*-models.*

Proof. The proof is a standard induction on the length of derivations. We focus on axioms on $done_i$ and **O**.

Axioms **D1**–**D4** are valid due to conditions $D1$–$D4$ on D_i. We just illustrate the case of **D4**. Assume that there is a state w of an $\mathbf{ADeL}_0$-model M s.t. $M, w \models done_i(\alpha \sqcap \beta)$, while $M, w \not\models done_i(\alpha) \wedge done_i(\beta)$. Then, either $M, w \not\models done_i(\alpha)$ or $M, w \not\models done_i(\beta)$. From this it follows that either $w \notin D_i(\alpha)$ or $w \notin D_i(\beta)$, both of which entails $w \notin D_i(\alpha \sqcap \beta)$, whence $M, w \not\models done_i(\alpha \sqcap \beta)$: contradiction. Conversely, assume that there is a state w of an $\mathbf{ADeL}_0$-model M s.t. $M, w \models done_i(\alpha) \wedge done_i(\beta)$ whereas $M, w \not\models done_i(\alpha \sqcap \beta)$. Then, $M, w \models done_i(\alpha)$ and $M, w \models done_i(\beta)$. From this it follows that $w \in D_i(\alpha)$ and $w \in D_i(\beta)$; whence, $w \in D_i(\alpha \sqcap \beta)$ and $M, w \models done_i(\alpha \sqcap \beta)$: contradiction.

Axioms **O1**–**O5** are valid due to the conditions on c. We just illustrate the case of **O4** and **O5**. Since $c(w, 0) \notin SV$, $M, w \models \mathbf{O}0$ for no $w \in W$, and so $M, w \models \neg\mathbf{O}0$. Suppose now $M, w \models \mathbf{O}\alpha$ and $M, w \models \mathbf{O}\beta$. Then $c(w, \alpha) \in SV$ and $c(w, \beta) \in SV$. Thus $c(w, \alpha \sqcap \beta) \in SV$, by $c5$. ■

Corollary 1. *Every theorem of* $\mathbf{ADeL}_0$ *is valid in the class of* $\mathbf{ADeL}_0$-*frames.*

Proof. The result follows directly from the fact that in the proof of Theorem 1 no reference to a particular valuation function V was made. ■

Having proved soundness, we are able to show that $\mathbf{ADeL}_0$ is a safe system with respect to the traditional puzzles of deontic reasoning mentioned above. Indeed, in order to see that principles like $\mathbf{O}1$, $\mathbf{O}(\alpha \sqcap \beta) \to \mathbf{O}(\alpha)$, and $\mathbf{O}\alpha \to \mathbf{O}(\alpha \sqcup \beta)$ are not derivable in it, it is sufficient to set $DV = \{\mathbf{1}, \mathbf{0}\}$, $SV = \{\mathbf{1}\}$, $\wedge_{\mathbb{D}}$ such that $d \wedge_{\mathbb{D}} d' = \mathbf{1}$ iff $d = \mathbf{1}$ and $d' = \mathbf{1}$, and define c so that, for every $w \in W$, $c(w, 1) = \mathbf{0}$, $c(w, \mathfrak{a}) = \mathbf{1}$, $c(w, 1 \sqcup \mathfrak{a}) = \mathbf{0}$, and $c(w, 1 \sqcap \mathfrak{a}) = \mathbf{1}$, where $\mathfrak{a} \in \mathtt{Act}$. It is plain that, in a model endowed which this deontic system, we have $M, w \not\models \mathbf{O}1$, $M, w \models \mathbf{O}\mathfrak{a}$, $M, w \not\models \mathbf{O}(1 \sqcup \mathfrak{a})$, and $M, w \models \mathbf{O}(1 \sqcap \mathfrak{a})$, which invalidates all the principles we pointed out.

2.6. Completeness

Completeness is proved through the construction of a canonical model equipped with an algebra of deontic values. So far, we have presented the semantics for language $\mathcal{L}_0$ and for system $\mathbf{ADeL}_0$ in its full generality. In particular, we allowed the set DV of deontic values in $\mathbf{ADeL}_0$-models to be an arbitrary set with at least two elements. In fact, due to condition $\mathbb{D}1$ in Definition 1, there is at least one element in DV; moreover, due to conditions $\mathbb{D}4$ and $c4$, there are at least two distinct elements in DV, and at least one of these is also an element of SV. This choice reflects the intuitions discussed in Section 2.4: having more than one selected deontic value allows one to represent alternative approaches to normative conflicts (as illustrated below in Section 4), as well as obligations with varying strength (e.g., those associated with optimal or sub-optimal normative scenarios). We know, by Theorem 1, that $\mathbf{ADeL}_0$ is sound with respect to this semantics. Yet, we will now show that the completeness of $\mathbf{ADeL}_0$ can be proven by building a canonical model in which DV includes exactly two values, namely, **1** (selected) and **0** (non-selected); let us say that models of this kind are *binary* $\mathbf{ADeL}_0$-models. Since the class of binary $\mathbf{ADeL}_0$-models is a subclass of the class of all $\mathbf{ADeL}_0$-models, we will be able to infer: (i) that $\mathbf{ADeL}_0$ is sound with respect to the class of binary $\mathbf{ADeL}_0$-models, and (ii) that $\mathbf{ADeL}_0$ is complete with respect to the class of all $\mathbf{ADeL}_0$-models. Additionally, in Section 2.7, we will show that any $\mathbf{ADeL}_0$-model M can be converted into a binary $\mathbf{ADeL}_0$-model M' s.t. M and M' are invariant with respect to the truth of $\mathcal{L}_0$ formulas at a state (Theorem 3). This implies that the cardinalities of DV and SV are not describable in $\mathcal{L}_0$.

We now proceed with the construction of the canonical model for $\mathbf{ADeL}_0$. The notion of a maximal $\mathbf{ADeL}_0$-consistent set of formulas is defined in the usual way and satisfies the usual properties.

Definition 6. Canonical model for $\mathbf{ADeL}_0$.

Let X be an $\mathbf{ADeL}_0$-consistent set of formulas and x be a maximal $\mathbf{ADeL}_0$-consistent set of formulas extending X. Let $x/\mathbf{A}$ be the set $\{\phi : \mathbf{A}\phi \in x\}$, that is, the set of formulas that are within the scope of $\mathbf{A}$ in x. We take an analogous definition for the sets $x/[i]$ and $x/\Box_i$. The canonical model for $\mathbf{ADeL}_0$ based on x is the tuple $M = \langle W, R_i, A_i, D_i, \mathbb{D}, V\rangle$, where:

1. W is the set of all maximal $\mathbf{ADeL}_0$-consistent sets of formulas w such that $x/\mathbf{A} \subseteq w$;
2. R_i is such that $v \in R_i(w)$ iff $w/[i] \subseteq v$;
3. A_i is such that $v \in A_i(w)$ iff $w/\Box_i \subseteq v$;
4. D_i is such that $v \in D_i(\alpha)$ iff $done_i(\alpha) \in v$;
5. $\mathbb{D} = (DV, SV, \wedge_{\mathbb{D}}, c)$ is such that:
   (a) $DV = \{\mathbf{1}, \mathbf{0}\}$
   (b) $SV = \{\mathbf{1}\}$
   (c) $\wedge_{\mathbb{D}}$ is specified via the matrix

$\wedge_{\mathbb{D}}$	**1**	**0**
1	**1**	**0**
0	**0**	**0**

(d) c is such that $c(w, \alpha) = \mathbf{1}$ iff $\mathbf{O}\alpha \in w$

6. V is such that $v \in V(p)$ iff $p \in v$.

We remark that, by definition, $(DV, \wedge_{\mathbb{D}})$ is a commutative, associative and idempotent algebraic structure in this model.

LEMMA 1. (Truth Lemma) *If M is the canonical model for* $\mathbf{ADeL}_0$*, then for every $\mathcal{L}_0$-formula ϕ and every maximal* $\mathbf{ADeL}_0$*-consistent set w, we have $M, w \models \phi$ iff $\phi \in w$.*

PROOF. The procedure is standard. We consider only the cases of formulas of the type $done_i(\alpha)$ and of the type $\mathbf{O}\alpha$.

1. $M, w \models done_i(\alpha)$ iff $w \in D_i(\alpha)$, by the definition of truth, iff $done_i(\alpha) \in w$, by the canonical definition of D_i;
2. $M, w \models \mathbf{O}\alpha$ iff $c(w, \alpha) \in SV$, by the definition of truth, iff $c(w, \alpha) = \mathbf{1}$, by the canonical definition of SV, iff $\mathbf{O}\alpha \in w$, by the definition of c.

This concludes the proof. ■

THEOREM 2. (Completeness) *Every formula that is valid in the class of* $\mathbf{ADeL}_0$*-models is a theorem of* $\mathbf{ADeL}_0$*.*

PROOF. We prove this by showing that the canonical model for $\mathbf{ADeL}_0$ belongs to the class of $\mathbf{ADeL}_0$-models. Properties of accessibility relations associated to modal operators can be checked following standard procedures for systems of multimodal logic. Properties of the function D_i easily follow from the fact that every maximal $\mathbf{ADeL}_0$-consistent set of formulas includes all instances of axioms **D1-D4**. So, let us check the remaining properties, that is, the conditions on SV and c.

Conditions on SV: immediate, from the definition of SV and $\wedge_{\mathbb{D}}$.

Conditions on c:

$c1$. for every $w \in W$, $c(w, \alpha \sqcap \alpha) = \mathbf{1}$ iff $c(w, \alpha) = \mathbf{1}$. This follows from the fact that w is closed under axiom **O1**.

$c2$. for every $w \in W$, $c(w, \alpha \sqcap \beta) = \mathbf{1}$ iff $c(w, \beta \sqcap \alpha) = \mathbf{1}$. This follows from the fact that w is closed under axiom **O2**.

$c3$. for every $w \in W$, $c(w, \alpha \sqcap (\beta \sqcap \gamma)) = \mathbf{1}$ iff $c(w, (\alpha \sqcap \beta) \sqcap \gamma) = \mathbf{1}$. Again, this follows from the fact that w is closed under axiom **O3**.

$c4$. $c(w, 0) \neq \mathbf{1}$, since, by axiom **O4**, $\mathbf{O}0 \notin w$.

$c5$. If $c(w, \alpha), c(w, \beta) \in SV$, then $\mathbf{O}\alpha \in w$ and $\mathbf{O}\beta \in w$, by the definition of c, and so $\mathbf{O}(\alpha \sqcap \beta) \in w$, by axiom **O5**. Therefore $c(w, \alpha \sqcap \beta) = \mathbf{1}$, which is sufficient for concluding that $c(w, \alpha \sqcap \beta) \in SV$.

Thus, the canonical model for $\mathbf{ADeL}_0$ based on x satisfies all properties of $\mathbf{ADeL}_0$-models. Together with Lemma 1, this entails the desired result by contraposition: if a formula ϕ is not a theorem of $\mathbf{ADeL}_0$, then $\{\neg\phi\}$ is an $\mathbf{ADeL}_0$-consistent set, and ϕ is falsifiable in the canonical model for $\mathbf{ADeL}_0$ based on any maximal $\mathbf{ADeL}_0$-consistent extension of $\{\neg\phi\}$. ■

Corollary 2. *Every formula that is valid in the class of* $\mathbf{ADeL}_0$*-frames is a theorem of* $\mathbf{ADeL}_0$*.*

Proof. The result follows from the fact that the frame of the canonical model used in the proof of Theorem 2 satisfies all properties of an $\mathbf{ADeL}_0$-frame. ■

2.7. Finite Model Property

In this section we show that system $\mathbf{ADeL}_0$ has the finite model property. We will prove this by adapting a method illustrated in [10]. From the finite model property and the fact that $\mathbf{ADeL}_0$ is finitely axiomatized, it follows that $\mathbf{ADeL}_0$ is decidable. For the sake of convenience, in the description of an $\mathbf{ADeL}_0$-model we will here explicitly mention a universal accessibility relation R associated with the operator $\mathbf{A}$; hence, a model will have the form:

$$M = \langle W, R, R_i, A_i, D_i, \mathbb{D}, V\rangle$$

In accordance with this, for all $w \in W$, we will have:

- $R_i(w) \subseteq R(w)$
- $A_i(w) \subseteq R(w)$

where $R(w) = \{v : R(w, v)\}$.

First of all, we observe the following:

Theorem 3. (Binary version of a model) *Every* $\mathbf{ADeL}_0$*-model* M *can be transformed into an* $\mathbf{ADeL}_0$*-model* M' *s.t.:*

- *M' is exactly as M, possibly except for $\mathbb{D}'$, where $DV' = \{\boldsymbol{1}, \boldsymbol{0}\}$, $SV' = \{\boldsymbol{1}\}$, $\wedge_{\mathbb{D}'}$ is specified as in the canonical model for* $\mathbf{ADeL}_0$ *and c' is such that, for every $w \in W$, $c'(w, \alpha) = \boldsymbol{1}$ iff $c(w, \alpha) \in SV$;*
- *for every $\mathcal{L}_0$-formula ϕ and every $w \in W$ it holds that $M, w \models \phi$ iff $M', w \models \phi$.*

We will call M' the binary version *of M.*

Proof. The result follows from the fact that, due to its definition, $\mathbb{D}'$ satisfies all properties of deontic systems in $\mathbf{ADeL}_0$-models (as shown in Theorem 2). In particular, this ensures that for every formula of the form $\mathbf{O}\alpha$, we have $M, w \models \mathbf{O}\alpha$ iff $M', w \models \mathbf{O}\alpha$. ∎

Let us assume standard definitions of (proper) sub-term of an algebraic term in $\mathtt{Act}^*$ and of (proper) sub-formula of a formula in $\mathcal{L}_0$. Given a set Δ_0 of $\mathcal{L}_0$-formulas, we will denote by Δ the smallest set obtained via the following three steps:

1. we add $\neg done_i(0)$ to Δ_0, obtaining a set Δ_1;
2. we close Δ_1 under sub-formulas, obtaining a set Δ_2;
3. we close Δ_2 under atomic formulas including proper sub-terms and their complements, obtaining the final set Δ, namely:
    - if $done_i(\alpha) \in \Delta_2$ and β is a proper sub-term of α, then $done_i(\beta), done_i(\overline{\beta}) \in \Delta$;
    - if $\mathbf{O}\alpha \in \Delta_2$ and β is a proper sub-term of α, then $\mathbf{O}(\beta), \mathbf{O}(\overline{\beta}) \in \Delta$.

According to this procedure, $\Delta_0 \subseteq \Delta_1 \subseteq \Delta_2 \subseteq \Delta$.

For any world w in a model M, let $Th(w) = \{\phi \in \mathcal{L}_0 : M, w \models \phi\}$. We say that two worlds w and v are *equivalent modulo* a set of $\mathcal{L}_0$-formulas Δ, in symbols $E_\Delta(w, v)$, iff, for every $\phi \in \Delta$, $M, w \models \phi$ iff $M, v \models \phi$. This is the same as saying that $Th(w) \cap \Delta = Th(v) \cap \Delta$. The relation E_Δ gives rise to a partition of W into equivalence classes that we can denote as $|w|_\Delta$, $|v|_\Delta$, $|u|_\Delta$, etc. We will say that such a partition is *induced by* Δ.

Definition 7. Filtration of an $\mathcal{L}_0$-model.

A filtration of a model $M = \langle W, R, R_i, A_i, D_i, \mathbb{D}, V\rangle$ through a set of $\mathcal{L}_0$-formulas Δ is any model $M^* = \langle W^*, R^*, R_i^*, A_i^*, D_i, \mathbb{D}^*, V^*\rangle$ such that:

1. W^* consists of one world from each equivalence class of the partition induced by Δ.
2. R^*, A_i^* and R_i^* satisfy the following conditions:
    (a) for every $w, v \in W^*$, if $[v]_\Delta \cap R(w) \neq \varnothing$, then $v \in R^*(w)$;
    for every $w, v \in W^*$, if $[v]_\Delta \cap A_i(w) \neq \varnothing$, then $v \in A_i^*(w)$;
    for every $w, v \in W^*$, if $[v]_\Delta \cap R_i(w) \neq \varnothing$, then $v \in R_i^*(w)$;
    (b) for every $w, v \in W^*$, if $v \in R^*(w)$, then $Th(w)/\mathbf{A} \cap \Delta \subseteq Th(v)$;
    for every $w, v \in W^*$, if $v \in A_i^*(w)$, then $Th(w)/\Box_i \cap \Delta \subseteq Th(v)$;
    for every $w, v \in W^*$, if $v \in R_i^*(w)$, then $Th(w)/[i] \cap \Delta \subseteq Th(v)$.

3. for every $w \in W^*$ and every $\alpha \in \mathtt{Act}^*$ s.t. $done_i(\alpha) \in \Delta$, $w \in D_i^*(\alpha)$ iff $w \in D_i(\alpha)$.
4. $\mathbb{D}^* = (DV^*, SV^*, \wedge_{\mathbb{D}^*}, c^*)$ satisfies the usual conditions of $\mathcal{L}_0$-models; furthermore, DV^* is a finite set and, for every $w \in W^*$ and every $\alpha \in \mathtt{Act}^*$ s.t. $\mathbf{O}\alpha \in \Delta$, $c^*(w, \alpha) \in SV^*$ iff $c(w, \alpha) \in SV$.
5. V^* consists of the original V restricted to the elements of W^*.

We prove the following

THEOREM 4. (Intended filtration) *Given an* $\mathbf{ADeL}_0$*-model* M*, there is a filtration of* M *which is an* $\mathbf{ADeL}_0$*-model.*

PROOF. Take an $\mathbf{ADeL}_0$-model $M = \langle W, R, R_i, A_i, D_i, \mathbb{D}, V\rangle$ and let model $M^* = \langle W^*, R^*, R_i^*, A_i^*, D_i, \mathbb{D}^*, V^*\rangle$ be defined in such a way that (for every $w, v \in W^*$):

- W^* consists of one world from each equivalence class of the partition induced by Δ.
- $v \in R^*(w)$ iff, for every formula $\mathbf{A}\phi \in \Delta$, $M, w \models \mathbf{A}\phi \Leftrightarrow M, v \models \mathbf{A}\phi$.
- $v \in A_i^*(w)$ iff, for every formula $\Box_i\phi \in \Delta$, $M, w \models \Box_i\phi \Leftrightarrow M, v \models \Box_i\phi$.
- $v \in R_i^*(w)$ iff, for every formula $[i]\phi \in \Delta$, $M, w \models [i]\phi \Rightarrow M, v \models \Box_i\phi \wedge \phi$.
- D_i^* is such that for every $\mathfrak{a} \in \mathtt{Act}$, $w \in D_i^*(\mathfrak{a})$ iff $M, w \models done_i(\mathfrak{a})$ and

  $D_i^*(1) = W^*$
  $D_i^*(\overline{\alpha}) = W^* - D_i^*(\alpha)$
  $D_i^*(\alpha \sqcap \beta) = D_i^*(\alpha) \cap D_i^*(\beta)$
  $D_i^*(\alpha \sqcup \beta) = D_i^*(\alpha) \cup D_i^*(\beta)$
- $\mathbb{D}^* = \mathbb{D}$;
- V^* consists of the original V restricted to the elements of W^*.

We have to prove that M^* is a filtration of M and an $\mathbf{ADeL}_0$-model. To prove the first point, we have to show that properties 1-5 of Definition 7 are satisfied by M^*, so that M^* is a filtration of M. Since most of the arguments are standard, we focus on the novel ones and show that properties 2(a-b) are satisfied by R_i^*, leaving the other cases to the reader.

- R_i^* satisfies property 2(a). In fact, suppose $[v]_\Delta \cap R_i(w) \neq \emptyset$. This entails that for every formula $[i]\phi$, if $M, w \models [i]\phi$, then for every $u \in [v]_\Delta \cap R_i(w)$, we have $M, u \models \phi$.
  Furthermore, due to the fact that **B3** is valid in M, we have that $M, w \models [i]\Box_i\phi$ and that, for every $u \in [v]_\Delta \cap R_i(w)$, $M, u \models \Box_i\phi$. Thus, for every

such u we have $M, u \models \Box_i\phi \wedge \phi$. Thus, we can infer that $M, v \models \Box_i\phi \wedge \phi$, whence, by the definition of M^*, $v \in R_i^*(w)$.

- R_i^* satisfies property 2(b). Suppose that $v \in R_i^*(w)$. Then, by the definition of M^*, for every formula $[i]\phi \in \Delta$, $M, w \models [i]\phi$ entails $M, v \models \Box_i\phi \wedge \phi$. Thus, it cannot be the case that there is some formula $[i]\phi \in \Delta$ s.t. $M, w \models [i]\phi$ and $M, v \not\models \phi$.

Now, we prove that M^* is an $\mathbf{ADeL}_0$-model. We show that A_i^* and R_i^* are related in the right way, leaving again the other cases to the reader.

- A_i^* and R_i^* are such that $v \in R_i^*(w) \Rightarrow A_i^*(v) \subseteq R_i^*(w)$ for every $w, v \in W$. In fact, suppose that $v \in R_i^*(w)$. Then, for every formula $[i]\phi \in \Delta$, $M, w \models [i]\phi \Rightarrow M, v \models \Box_i\phi \wedge \phi$. Let $x \in A_i^*(v)$: then, by the definition of M^*, for every formula $\Box_i\phi \in \Delta$, $M, v \models \Box_i\phi$ iff $M, x \models \Box_i\phi$. Thus, by transitivity and the fact that $\Box_i\phi \to \phi$ is valid in M, for every formula $[i]\phi$ in Δ, $M, w \models [i]\phi \Rightarrow M, x \models \Box_i\phi \wedge \phi$, so that $u \in R_i^*(w)$.

This concludes the proof. ■

We will refer to model M^* as the *intended filtration* of model M. We can now prove the following

THEOREM 5. (Invariance under filtrations) *Let M be an $\mathbf{ADeL}_0$-model and Δ a set of $\mathcal{L}_0$-formulas. If M^* is a filtration of M through Δ, then, for every $\psi \in \Delta$ and $w \in M^*$, $M^*, w \models \psi$ iff $M, w \models \psi$.*

PROOF. By induction on the construction of ϕ. The proof with respect to the modal formulas is standard —see Theorem 1— and the proof for formulas of the form $\mathbf{O}\alpha$ is straightforward. Thus, we only focus on the proof in relation to formulas of the form $done_i(\alpha)$.
In the case of ϕ being $done_i(\alpha)$, we proceed by induction on the construction of α.

Suppose $M, w \models done_i(\alpha)$ and α is atomic. Then $w \in D_i^*(\alpha)$, and so $M^*, w \models done_i(\alpha)$, by the definition of M^*.

Suppose $M, w \models done_i(\alpha)$ and $\alpha = \alpha_1 \sqcap \alpha_2$. Then $M, w \models done_i(\alpha_1 \sqcap \alpha_2)$, and so $M, w \models done_i(\alpha_1)$ and $M, w \models done_i(\alpha_2)$. Thus, by the induction hypothesis, $M^*, w \models done_i(\alpha_1)$ and $M^*, w \models done_i(\alpha_2)$, since both $done_i(\alpha_1) \in \Delta$ and $done_i(\alpha_2) \in \Delta$, and so $M^*, w \models done_i(\alpha_1 \sqcap \alpha_2)$.

The other cases are similar. ■

Combining the results obtained so far, we get the finite model property for system $\mathbf{ADeL}_0$, which can be formulated as below:

COROLLARY 3. (Finite model property) *Since, for every* $\mathbf{ADeL}_0$*-model* M*, we can first define a model* M' *which is the binary version of* M *and then a model* M'' *which is the intended filtration of* M'*, the following claims are pairwise equivalent (for every* $\mathcal{L}_0$*-formula* ϕ *and world* w *in the domain of* M*):*

- $M, w \models \phi$;
- $M', w \models \phi$;
- $M'', w \models \phi$.

PROOF. A direct consequence of Theorems 3–7. ∎

3. Extended Framework

3.1. Formal Language

Let us now see how to exploit the ideas proposed before in order to deal with conflicts. To do that, we have to enrich our language.

Conflicts are situations where an agent is required to perform actions that are mutually exclusive: thus, in a certain situation, an agent could be required to perform an action of type α and an action of type β, even if it is impossible for α and β to be performed at once. Conflicts are typically generated by the fact that different norms prescribe different things. Therefore, in order to track the origin of a conflict, we have to use a language where the fact that an obligation stems from a certain norm is expressible.

While we keep the set of action types $\texttt{Act}^*$ as in Section 2.1, we define the new language $\mathcal{L}_1$ as follows:

$$\phi ::= p \mid \neg\phi \mid \phi \wedge \phi \mid \mathbf{A}\phi \mid [i]\phi \mid \Box_i \phi \mid done_i(\alpha) \mid \mathbf{O}\alpha \mid \mathbf{O}_s\alpha$$

where $s \in \mathsf{NS}$ and $\mathsf{NS} \neq \varnothing$ is a set of normative sources.

3.2. Conflicts

We are now in a position to distinguish different kinds of conflict. A first distinction we can introduce is between *abstract conflicts* (characterized in terms of action types) and *concrete conflicts* (characterized in terms of action tokens).

Abstract conflicts. In some cases abstract conflicts stem from different sources (s_1 and s_2), because what the sources require is not realizable in general.

- $\mathbf{O}_{s_1}\alpha \wedge \mathbf{O}_{s_2}\beta \wedge \mathbf{A}\neg done_i(\alpha \sqcap \beta)$.

In other cases abstract conflicts stem from a unique source (say, s_1):

- $\mathbf{O}_{s_1}\alpha \wedge \mathbf{O}_{s_1}\beta \wedge \mathbf{A}\neg done_i(\alpha \sqcap \beta)$.

Note the following.

1. In the first case, what the agent does is not necessarily wrong according to s_1, since she may perform α, nor it is wrong according to s_2, since she may perform β, but she could be wrong according to $\{s_1, s_2\}$, and indeed she is, provided we have no way to prioritize one of the sources over the other. Still, we are not entitled to conclude that i's action is wrong according to $\{s_1, s_2\}$ without further justification, precisely because it is possible for one of the sources to be more important than the other.
2. In the second case, what the agent can do is necessarily wrong according to s_1, since $\mathbf{O}_{s_1}\alpha \wedge \mathbf{O}_{s_1}\beta$ implies $\mathbf{O}_{s_1}(\alpha \sqcap \beta)$ and $\mathbf{A}\neg done_i(\alpha \sqcap \beta)$ is equivalent to $\mathbf{A}done_i(\overline{\alpha \sqcap \beta})$. Thus, we get $\mathbf{O}_{s_1}(\alpha \sqcap \beta) \wedge \mathbf{A}done_i(\overline{\alpha \sqcap \beta})$, which implies wrong doing: $\mathbf{O}_{s_1}(\alpha \sqcap \beta) \wedge [i]done_i(\overline{\alpha \sqcap \beta})$.
3. In the second case, the fact that what the agent does is wrong according to s_1 does not entail that what the agent does is absolutely wrong, since s_1 could include norms that do not apply to the circumstances of evaluation or that are superseded by others in the circumstances of evaluation.

Concrete conflicts. In some cases concrete conflicts stem from a different classification of the same conduct in different sources:

- $\mathbf{O}_{s_1}\alpha \wedge \mathbf{O}_{s_2}\beta \wedge \Box_i[i]\neg done_i(\alpha \sqcap \beta)$.

In other cases concrete conflicts stem from a different classification of the same conduct within one source:

- $\mathbf{O}_{s_1}\alpha \wedge \mathbf{O}_{s_1}\beta \wedge \Box_i[i]\neg done_i(\alpha \sqcap \beta)$.

As before, in the second case, what the agent does is wrong according to s_1, while, in the first case, it is not wrong according to s_1, nor it is wrong according to s_2, even if it could be wrong according to the more complex source $\{s_1, s_2\}$, but this depends on the way in which s_1 and s_2 are aggregated.

Solvable conflicts. In some cases abstract deontic conflicts can be solved by ordering the sources. Similarly, concrete deontic conflicts can be solved by accepting one source only. Finally, some cases are such that deontic conflicts cannot be solved.

We further analyze conflicts in Section 4, where we aim at (i) refining the characterization of the structure of a conflict; (ii) identifying the source of the conflict given its characterization; (iii) understanding whether a conflict is avoidable or unavoidable, solvable or unsolvable, given its source, from the point of view of an agent.

3.3. Systems $\mathbf{ADeL_{1.1}}$ and $\mathbf{ADeL_{1.2}}$

We here introduce two logical systems over the new language $\mathcal{L}_1$, that will be called $\mathbf{ADeL}_{1.1}$ and $\mathbf{ADeL}_{1.2}$. The axiomatic basis for the first system, $\mathbf{ADeL}_{1.1}$, is obtained from the axiomatic basis for $\mathbf{ADeL}_0$ by removing **O1** and adding the following schemata, for $s \in \mathsf{NS}$:

O1s $\mathbf{O}_s(\alpha \sqcap \alpha) \leftrightarrow \mathbf{O}_s\alpha$

O2s $\mathbf{O}_s(\alpha \sqcap \beta) \leftrightarrow \mathbf{O}_s(\beta \sqcap \alpha)$

O3s $\mathbf{O}_s(\alpha \sqcap (\beta \sqcap \gamma)) \leftrightarrow \mathbf{O}_s((\alpha \sqcap \beta) \sqcap \gamma)$

O4s $\neg\mathbf{O}_s 0$

O5s $\mathbf{O}_s\alpha \wedge \mathbf{O}_s\beta \rightarrow \mathbf{O}_s(\alpha \sqcap \beta)$

B4 $\mathbf{O}\alpha \rightarrow \mathbf{O}_s\alpha$, for all $s \in \mathsf{NS}$.

Axiom **O1** is not needed, since it becomes derivable from the new ones. Note that here we do not assume that the set of normative sources NS is finite. Under that assumption, one could replace **B4** with:

$\mathbf{B4}_{fin}$ $\mathbf{O}\alpha \leftrightarrow \bigwedge_{s\in\mathsf{NS}} \mathbf{O}_s\alpha$.

The axiomatic basis for the second system, $\mathbf{ADeL}_{1.2}$, is obtained by extending the one for $\mathbf{ADeL}_{1.1}$ with the following schema:

B5 $(\mathbf{O}_s\alpha \wedge \mathbf{O}_s\beta) \rightarrow \Diamond_i done_i(\alpha \sqcap \beta)$.

Thus, $\mathbf{ADeL}_{1.2}$ is stronger than $\mathbf{ADeL}_{1.1}$ and encodes the additional intuition that it is always possible to jointly perform a list of obligatory action types, at least when the types are prescribed by the same source.

3.4. Semantics

Language $\mathcal{L}_1$ is interpreted in the following semantics, which is a generalization of the one for language $\mathcal{L}_0$. In case of a finite set NS of normative sources, there is no problem in adapting the semantics for the basic system;

the main result obtained here is a generalization to any set NS. Properties of $\mathbf{ADeL}_{1.1}$-models and of $\mathbf{ADeL}_{1.2}$-models will be specified within the general definition. In accordance with the comparable deductive power of the two systems, we will define an $\mathbf{ADeL}_{1.2}$-model so as to be also an $\mathbf{ADeL}_{1.1}$-model.

DEFINITION 8. Model for $\mathcal{L}_1$.

A model for $\mathcal{L}_1$ is a tuple $M = \langle W, R_i, A_i, D_i, \mathbb{D}, V\rangle$, where all the elements are as in Section 2.4, except for the *deontic system*, which becomes $\mathbb{D} = (DV, SV, \wedge_{\mathbb{D}}, \{c_s\}_{s\in\mathsf{NS}}, \mathfrak{C})$, where:

- DV, SV and $\wedge_{\mathbb{D}}$ are defined in the usual way and have the usual properties;
- $\{c_s\}_{s\in\mathsf{NS}}$ is a set of deontic classification functions, each being associated with one normative source, namely $c_s : W \times \mathtt{Act}^* \to DV$;
- $\mathfrak{C} : W \times \mathtt{Act}^* \to DV$ is an *all-things-considered* deontic classification function.

Hence, $c_s(w, \alpha)$ is the deontic value associated to α at w by source s, whereas $\mathfrak{C}(w, \alpha)$ is the deontic value associated to α at w by aggregating normative sources. In the case of $\mathbf{ADeL}_{1.1}$-models (and, *a fortiori*, of $\mathbf{ADeL}_{1.2}$-models) we have, for each deontic function c_s and for $\mathfrak{C}$, all the properties that we had in $\mathbf{ADeL}_0$-models for the function c. Furthermore, we have the following property, which captures the idea of aggregation:

- $\mathfrak{C}(w, \alpha) \in SV$ iff $c_s(w, \alpha) \in SV$ for every $s \in \mathsf{NS}$.

This condition entails axiom **B4** and, in the case of a finite set NS of normative sources, it corresponds to axiom $\mathbf{B4}_{fin}$.

In $\mathbf{ADeL}_{1.2}$-models we also have the following property:

- if $c_s(w, \alpha) \in SV$ and $c_s(w, \beta) \in SV$, then there is a state $v \in R_i(w) \cap D_i(\alpha \sqcap \beta)$.

The idea behind the last condition is that norms are in themselves consistent, so that no possible composite action types is such that its component types receive contrary deontic values. This condition correspond to axiom **B5**.

The following truth conditions are added/changed accordingly in models for the language $\mathcal{L}_1$:

- $M, w \models \mathbf{O}_s\alpha \;\Leftrightarrow\; c_s(w, \alpha) \in SV$
- $M, w \models \mathbf{O}\alpha \;\Leftrightarrow\; \mathfrak{C}(w, \alpha) \in SV$.

In $\mathbf{ADeL}_{1.1}$-models (and, *a fortiori*, $\mathbf{ADeL}_{1.2}$-models), the latter truth-condition is equivalent to:

- $M, w \models \mathbf{O}\alpha \Leftrightarrow$ for all $s \in \mathsf{NS}$, $c_s(w, \alpha) \in SV$.

Finally, frames are defined as per the following

Definition 9. A frame for $\mathcal{L}_1$ is a tuple $F = \langle W, R_i, A_i, D_i, \mathbb{D}\rangle$ whose elements can be described as in Definition 8. An $\mathbf{ADeL}_{1.x}$-frame, for $x \in \{1, 2\}$, is a frame for $\mathcal{L}_1$ whose elements satisfy the conditions mentioned in Definition 8.

3.5. Soundness

Theorem 6. (Soundness) *Every theorem of* $\mathbf{ADeL}_{1.x}$*, where* $x \in \{1, 2\}$ *is valid in the class of* $\mathbf{ADeL}_{1.x}$*-models.*

Proof. Analogous to the one of Theorem 1. We just illustrate the case of axioms **B4** and **B5** as examples.

Suppose that there is a state w of an $\mathbf{ADeL}_{1.1}$-model M s.t. $M, w \models \mathbf{O}\alpha$ and, for some $s \in \mathsf{NS}$, $M, w \not\models \mathbf{O}_s\alpha$. Then, by the truth-conditions, we have that $\mathfrak{C}(w, \alpha) \in SV$; but the latter claim in $\mathbf{ADeL}_{1.1}$-models is the same as the claim that, for every $s' \in \mathsf{NS}$, $c_{s'}(w, \alpha) \in SV$. From this, by the truth-conditions, it follows that $\mathfrak{M}, w \models \mathbf{O}_s$: contradiction.

Suppose that there is a state w of an $\mathbf{ADeL}_{1.2}$-model M and a source $s \in \mathsf{NS}$ s.t. $M, w \models \mathbf{O}_s\alpha \wedge \mathbf{O}_s\beta$ whereas $M, w \not\models \Diamond_i done_i(\alpha \sqcap \beta)$. Then, $M, w \models \mathbf{O}_s\alpha$ and $M, w \models \mathbf{O}_s\beta$; by the truth-conditions, $c_s(w, \alpha), c_s(w, \beta) \in SV$. From this, due to the properties of $\mathbf{ADeL}_{1.2}$-models, we can infer that there is a state $v \in R_i(w) \cap D_i(\alpha \sqcap \beta)$. Thus, $M, v \models done_i(\alpha \sqcap \beta)$ and $M, w \models \Diamond_i done_i(\alpha \sqcap \beta)$: contradiction. ■

Corollary 4. *Every formula that is a theorem of* $\mathbf{ADeL}_{1.x}$ *is valid in the class of* $\mathbf{ADeL}_{1.x}$*-frames, for* $x \in \{1, 2\}$.

Proof. The result follows from the fact that the proof of Theorem 6 does not make reference to any particular valuation function V. ■

3.6. Completeness

Definition 10. Canonical models for $\mathbf{ADeL}_{1.1}$ and $\mathbf{ADeL}_{1.2}$.

The canonical model for $\mathbf{ADeL}_{1.1}$ can be defined by replacing the deontic system used in the canonical model for $\mathbf{ADeL}_0$ with the deontic system $\mathbb{D} = (DV, SV, \wedge_{\mathbb{D}}, \{c_s : s \in \mathsf{NS}\}, \mathfrak{C})$. In this new system $(DV, \wedge_{\mathbb{D}})$ and SV are defined as in the previous case. The set NS, the functions of type c_s and the function $\mathfrak{C}$ are defined as below:

- $\mathsf{NS} = \mathsf{NS}^+ \cup \{s^*\}$, where $\mathsf{NS}^+ = \{s : \mathbf{O}_s\alpha \in \mathcal{L}_1\}$ and $\mathsf{NS}^+ \cap \{s^*\} = \emptyset$;
- for every $s \in \mathsf{NS}^+$, $c_s(w, \alpha) = \mathbf{1}$ iff $\mathbf{O}_s\alpha \in w$;
- $c_{s^*}(w, \alpha) = \mathfrak{C}(w, \alpha) = \mathbf{1}$ iff $\mathbf{O}\alpha \in w$.

Lemma 2. (Truth Lemma) *If M is the canonical model for* $\mathbf{ADeL}_{1.x}$*, where $x \in \{1, 2\}$, then for every $\mathcal{L}_1$-formula ϕ and every maximal* $\mathbf{ADeL}_{1.x}$*-consistent set w, we have $M, w \models \phi$ iff $\phi \in w$.*

Proof. The procedure is standard. We consider only the cases of formulas of the type $\mathbf{O}_s\alpha$, for $s \in \mathsf{NS}^+$, and of the type $\mathbf{O}\alpha$.

1. $M, w \models \mathbf{O}_s\alpha$ iff $c_s(w, \alpha) \in SV$, by the definition of truth, iff $c_s(w, \alpha) = \mathbf{1}$, by the canonical definition of SV, iff $\mathbf{O}_s\alpha \in w$, by the definition of c_s;
2. $M, w \models \mathbf{O}\alpha$ iff $\mathfrak{C}(w, \alpha) \in SV$, by the definition of truth, iff $\mathfrak{C}(w, \alpha) = \mathbf{1}$, by the canonical definition of SV, iff $\mathbf{O}\alpha \in w$, by the definition of $\mathfrak{C}$.

This concludes the proof. ■

Theorem 7. (Completeness) *Every formula that is valid in the class of* $\mathbf{ADeL}_{1.x}$*-models is a theorem of* $\mathbf{ADeL}_{1.x}$*, for $x \in \{1, 2\}$.*

Proof. Conditions on functions of the type c_s, for $s \in \mathsf{NS}^+$ as well as on the function $\mathfrak{C}$, can be checked with the same procedure used in the case of $\mathbf{ADeL}_0$, possibly relying on the new axioms **O1s-O5s** and **B4**. Notice that, since the function s^* is defined in the same way as the function $\mathfrak{C}$, it satisfies those conditions as well. Furthermore, the addition of s^*, which does not correspond to any deontic operator in $\mathcal{L}_1$, allows us to get the desired property on the relation between $\mathfrak{C}$ and the set of normative sources: $\mathfrak{C}(w, \alpha) = \mathbf{1}$ iff, for all $s \in \mathsf{NS}$, $c_s(w, \alpha) = \mathbf{1}$. Indeed, for the left-to-right direction, if $\mathfrak{C}(w, \alpha) = \mathbf{1}$, then $\mathbf{O}\alpha \in w$ and $c_{s^*}(w, \alpha) = \mathbf{1}$. Furthermore, due to **B4**, we have $\mathbf{O}_{s'}\alpha \in w$ for all $s' \in \mathsf{NS}^+$, whence $c_{s'}(w, \alpha) = \mathbf{1}$. Thus, for all $s \in \mathsf{NS}$, we get $c_s(w, \alpha) = \mathbf{1}$. For the right-to-left direction, suppose that $\mathfrak{C}(w, \alpha) = \mathbf{0}$. Then, $\mathbf{O}\alpha \notin w$ and $c_{s^*}(w, \alpha) = \mathbf{0}$. Thus, it is not the case that for every $s \in \mathsf{NS}$ we have $c_s(w, \alpha) = \mathbf{1}$.

Conditions on the relations R_i and A_i, as well as conditions on the function D_i, can be checked as usual.

Moving from $\mathbf{ADeL}_{1.1}$ to $\mathbf{ADeL}_{1.2}$, we can build the canonical model in the same way, but we also have to check that the following additional property is satisfied, for every $s \in \mathsf{NS}$:

- if $c_s(w, \alpha) = \mathbf{1}$ and $c_s(w, \beta) = \mathbf{1}$, then there is $v \in R_i(w) \cap D_i(\alpha \sqcap \beta)$.

In the case of $s \in \mathsf{NS}^+$ the property holds due to axiom **B5**. In the case of s^*, assume that $c_{s^*}(w, \alpha) = \mathbf{1}$ and $c_{s^*}(w, \beta) = \mathbf{1}$; then $\mathbf{O}\alpha, \mathbf{O}\beta \in w$, so, by axiom **B4**, we can infer that $\mathbf{O}_s\alpha, \mathbf{O}_s\beta \in w$, that is, $\mathbf{O}_s\alpha \wedge \mathbf{O}_s\beta \in w$, for all $s \in \mathsf{NS}^+$. Thus, since NS^+ is non-empty, we can further infer by axiom **B5** that $\Diamond_i done_i(\alpha \sqcap \beta) \in w$. Suppose that there is no state $v \in R_i(w) \cap D_i(\alpha \sqcap \beta)$. Then, for all states u, we have that either $u \notin R_i(w)$ or $u \notin D_i(\alpha \sqcap \beta)$; yet, R_i is serial and the latter claim is the same as the claim that, for every $u \in R_i(w)$, we have $done_i(\alpha \sqcap \beta) \notin u$. Then, by the canonical model construction, we have that $\Diamond_i done_i(\alpha \sqcap \beta) \notin w$: contradiction. ■

Corollary 5. *Every formula that is valid in the class of* $\mathbf{ADeL}_{1.x}$*-frames is a theorem of* $\mathbf{ADeL}_{1.x}$*, for* $x \in \{1, 2\}$*.*

Proof. The result follows also in this case from the fact that the frame of the canonical model used in the proof of Theorem 7 satisfies all properties of an $\mathbf{ADeL}_{1.x}$-frame, for $x \in \{1, 2\}$. ■

4. Perspectives on Conflicts

In the present section we apply our framework to recent proposals to handle normative conflicts in systems of action deontic logic. We start by discussing a preliminary issue, that is, the distinction between aggregation of action types and aggregation of normative sources.

4.1. Aggregating Actions and Sources

In the framework provided here aggregation of action types ($\sqcap$) and aggregation of sources ($\wedge_{\mathbb{D}}$) are kept sharply distinct. We take this to be a crucial feature of action deontic logic. For a comparison, let us consider the view in [13]:

> Thus, if $\alpha \sqcap \beta$ appears in a formula, then α and β have to be different descriptions that can be attached to the same particular action. Usually in this context α and β represent types of action coming from different normative systems and $\alpha \sqcap \beta$ refers to the same action when we express its final deontic status after merging the normative systems.

In this passage, Kulicki and Trypuz seem to have in mind aggregation of sources; however, they express this kind of aggregation in terms of aggregation of action types, which could be misleading. If generalized, such an analysis would also weaken the potential expressiveness of action deontic logic since, as it is pointed out in [6], it would not make room for the possibility of assigning a deontic value to the combination of two distinct action types within a single normative source. By contrast, in [6] a choice is made to interpret $\sqcap$ as action aggregation, but the possibility of dealing with the aggregation of sources requires additional machinery. Since the ideas in [13] are very interesting to deepen the analysis of normative conflicts, we show below how one could use these in the present framework.

4.2. Representing Alternative Proposals

In [13] three views are introduced:

- The *pessimistic view*: if an action is obligatory from one point of view and forbidden from another, then it is to be regarded as forbidden;
- The *optimistic view*: if an action is obligatory from one point of view and forbidden from another, then it is to be regarded as obligatory;
- The *neutral view*: if an action is obligatory from one point of view and forbidden from another, then it is to be regarded as neither obligatory nor forbidden.

A fourth view, based on a more sophisticated framework that allows for representing a hierarchy among normative sources, is proposed in [6]:

- The *realistic view*: if an action is obligatory from one point of view and forbidden from another, then it gives rise to a normative conflict that remains unsolved unless a priority between the two points of view can be established.

In the present framework these views can be captured by introducing different deontic systems corresponding to different semi-lattices $(DV, \wedge_{\mathbb{D}})$. All these systems are such that DV contains the following deontic values: $\mathbf{o}$ ('being obligatory'), $\mathbf{f}$ ('being forbidden') and $\mathbf{i}$ ('being indifferent'). Furthermore, in all systems we have $SV = \{\mathbf{o}\}$, $\mathbf{o} \wedge_{\mathbb{D}} \mathbf{i} = \mathbf{o}$ and $\mathbf{i} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{f}$. Finally, we have that, for $h \in \{1, 2\}$, $c_{s_h}(w, \overline{\overline{\alpha}}) = c_{s_h}(w, \alpha)$, as well as that the negation of an obligatory action is forbidden and vice versa, i.e., $c_{s_h}(w, \alpha) = \mathbf{o}$ iff $c_{s_h}(w, \overline{\alpha}) = \mathbf{f}$.

Suppose that the agent is required to perform two action types that are mutually exclusive at a state w. So, suppose that there are action types α_1 and α_2 such that:

1. $c_{s_1}(w, \alpha_1) = \mathbf{o}$, $c_{s_1}(w, \alpha_2) = \mathbf{i}$;
2. $c_{s_2}(w, \alpha_2) = \mathbf{o}$, $c_{s_2}(w, \alpha_1) = \mathbf{i}$;
3. $M, w \models \neg \langle i \rangle \, done_i(\alpha_1 \sqcap \alpha_2)$.

The choice is between $\alpha_1 \sqcap \overline{\alpha_2}$ and $\overline{\alpha_1} \sqcap \alpha_2$, and has to be made taking into account all the relevant normative sources. Hence, defining

$$\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = c_{s_1}(w, \alpha_1 \sqcap \overline{\alpha_2}) \wedge_{\mathbb{D}} c_{s_2}(w, \alpha_1 \sqcap \overline{\alpha_2})$$

we obtain

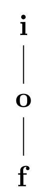
1. $c_{s_1}(w, \alpha_1 \sqcap \overline{\alpha_2}) = c_{s_1}(w, \alpha_1) \wedge_{\mathbb{D}} c_{s_1}(w, \overline{\alpha_2}) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{i} = \mathbf{o}$
2. $c_{s_2}(w, \alpha_1 \sqcap \overline{\alpha_2}) = c_{s_2}(w, \alpha_1) \wedge_{\mathbb{D}} c_{s_2}(w, \overline{\alpha_2}) = \mathbf{i} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{f}$
3. $\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f}$;
4. similarly, $\{c_{s_1}, c_{s_2}\}(w, \overline{\alpha_1} \sqcap \alpha_2) = \mathbf{f} \wedge_{\mathbb{D}} \mathbf{o} = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f}$.
5. and so $\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = \{c_{s_1}, c_{s_2}\}(w, \overline{\alpha_1} \sqcap \alpha_2) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f}$.

Pessimistic system. The pessimistic deontic system is defined so that $DV = \{\mathbf{o}, \mathbf{f}, \mathbf{i}\}$, $SV = \{\mathbf{o}\}$, and $\wedge_{\mathbb{D}}$ is such that

$\wedge_{\mathbb{D}}$	**f**	**i**	**o**
f	**f**	**f**	**f**
i	**f**	**i**	**o**
o	**f**	**o**	**o**

The pessimistic system has it that an action which is prohibited by a source is prohibited *tout court*. Furthermore, we stress that the value **i** can be generated only via an aggregation of two copies of itself.

The corresponding semi-lattice has the following form:



Hence, **f** is the strongest value. The basic intuition is that an action that is good from one point of view and bad from the other point of view is to be eventually assessed as bad. In case of conflict, there is no good solution, since the action an agent performs is bad no matter what the agent does.

1. $\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{f}$;
2. $\{c_{s_1}, c_{s_2}\}(w, \overline{\alpha_1} \sqcap \alpha_2) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{f}$;
3. therefore, whatever we opt for, we get **f**.

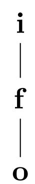
This result makes it evident that the pessimistic system *does not allow for a positive solution* of a dilemma.

Optimistic system. The optimistic deontic system is defined so that $DV = \{\mathbf{o}, \mathbf{f}, \mathbf{i}\}$, $SV = \{\mathbf{o}\}$, and $\wedge_{\mathbb{D}}$ is such that

$\wedge_{\mathbb{D}}$	**f**	**i**	**o**
f	**f**	**f**	**o**
i	**f**	**i**	**o**
o	**o**	**o**	**o**

The optimistic system has it that an action which is required by a source is required *tout court*. Also in the case of the optimistic system it holds that the value **i** can be generated only via an aggregation of two copies of itself.

The corresponding semi-lattice has the following form:



Hence, **o** is the strongest value. The basic intuition is that an action that is good from one point of view and bad from the other point of view is to be eventually assessed as good. In case of conflict, there is always a good solution, since the action an agent performs is good no matter what the agent does.

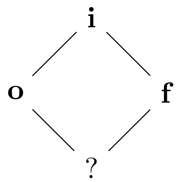
1. $\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{o}$;
2. $\{c_{s_1}, c_{s_2}\}(w, \overline{\alpha_1} \sqcap \alpha_2) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} = \mathbf{o}$;
3. therefore, whatever we opt for, we get **o**.

This result makes it evident that the pessimistic system *does allow for a positive solution* of a dilemma.

Neutral system. The neutral deontic system is defined so that $DV = \{\mathbf{o}, \mathbf{f}, \mathbf{i}, ?\}$, $SV = \{\mathbf{o}\}$, and $\wedge_{\mathbb{D}}$ is such that

$\wedge_{\mathbb{D}}$	**f**	**i**	**o**	?
f	**f**	**f**	?	?
i	**f**	**i**	**o**	?
o	?	**o**	**o**	?
?	?	?	?	?

The neutral system has it that conflicting assessments give rise to a problematic value ?. The corresponding lattice has the following form:



In this framework, the new deontic value ? can be generated via an aggregation of two deontic values that are not necessarily identical to it.

Here, ? is the strongest value. The basic intuition is that an action that is good from one point of view and bad from the other point of view is to be eventually assessed as problematic. In case of conflict, there is no straightforward assessment, since the action an agent performs is both good and bad.

1. $\{c_{s_1}, c_{s_2}\}(w, \alpha_1 \sqcap \overline{\alpha_2}) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} =?$;
2. $\{c_{s_1}, c_{s_2}\}(w, \overline{\alpha_1} \sqcap \alpha_2) = \mathbf{o} \wedge_{\mathbb{D}} \mathbf{f} =?$;
3. therefore, whatever we opt for, we get ?.

It is difficult to see whether the neutral system allows for a solution of a dilemma. Here dilemmas are taken seriously: doing something good implies doing something bad at once.

The realistic view. As long as a single normative source is taken into account, the realistic deontic system can be semantically represented within our framework exactly as the neutral system, since the two agree on the deontic value assigned to each action type within one normative source. In this regard, they differ only conceptually: while the neutral system takes the assignment of value ? to an action type α as saying that α is neither obligatory nor prohibited (whence, its syntactic counterpart is the claim $\neg\mathbf{O}\alpha \wedge \neg\mathbf{O}\overline{\alpha}$), the realistic system takes the assignment of value ? to α as saying that α is both obligatory and prohibited (whence, its syntactic counterpart is the claim $\mathbf{O}\alpha \wedge \mathbf{O}\overline{\alpha}$).[8]

However, the realistic system naturally lends itself to the analysis of normative problems involving more than one normative source: in fact, it is supported by a mechanism to handle normative conflicts by combining normative sources. In our framework a more appropriate representation of the realistic system would require moving from language $\mathcal{L}_0$ to language $\mathcal{L}_1$ and

[8]The notation employed in [6] is slightly different. In particular, in the case of deontic values, the label c replaces the label ?.

adding a priority relation $\prec$ among normative sources which would determine a new way of defining the deontic aggregation function $\mathfrak{C}$. Here we just provide a sketch of the idea, leaving its development to a further work. Consider two normative sources s_1 and s_2, and let $\prec$ be a strict partial order over the set of normative source NS. We say that there is a *winning source* between s_1 and s_2 iff the following holds: either $s_1 \prec s_2$ or $s_2 \prec s_1$. The deontic value assigned by $\mathfrak{C}$ to an action type α in the realistic system is specified by the table below, where values assigned by s_1 are on the vertical axis, values assigned by s_2 are on the horizontal axis, and an expression of the form $\prec /x$ means "if there is a winning source between s_1 and s_2, the deontic value assigned by $\mathfrak{C}$ to α is that of the winning source, otherwise it is x" (see Table 4 in [6]):

$\mathfrak{C}$	$\mathbf{f}$	$\mathbf{i}$	$\mathbf{o}$	?
$\mathbf{f}$	$\mathbf{f}$	$\prec/\mathbf{f}$	$\prec/?$	$\prec/?$
$\mathbf{i}$	$\prec/\mathbf{f}$	$\mathbf{i}$	$\prec/\mathbf{o}$	$\prec/?$
$\mathbf{o}$	$\prec/?$	$\prec/\mathbf{o}$	$\mathbf{o}$	$\prec/?$
?	$\prec/?$	$\prec/?$	$\prec/?$	?

5. Perspectives on Search Problems

The second application of the framework concerns the representation of search problems. As is well-known, a search problem involves agents that search sequences of actions to achieve a specific goal in an optimal way.[9] As such they are problems of normative reasoning, since an agent may solve them by acting in accordance with a certain norm that specifies the best course of events, if any. Therefore, systems of deontic logic constitute natural candidates to formally represent them.

In this section we show how our framework can be suitably used to provide such a representation. Our main aim is to point out that these problems can be encoded in the syntactic and semantic framework of $\mathbf{ADeL}_0$ in a straightforward and intuitive way, differently from other languages of deontic logic in which it is not possible to make reference to action types or to the conduct of an agent. Even the weakest framework we have introduced so far, that is language $\mathcal{L}_0$ of system $\mathbf{ADeL}_0$ and its semantics, is expressive enough to represent interesting aspects of search problems in two steps: first, one transforms a rigorous description of a search problem into a simplified model for $\mathbf{ADeL}_0$; then, one uses the modal operators in $\mathcal{L}_0$ to express

[9]See [11, ch.2] and [21, ch.2] for a general introduction.

actions that are available to an agent at each stage of the search problem and norms that have to be followed.

Let us start by better specifying the sort of problems we are dealing with.

Definition 11. A search problem is a tuple $(S, A, in, end, \mathbf{a}, \mathbf{s}, \mathbf{cost})$ where

$S \neq \varnothing$ is a set of states

$A \neq \varnothing$ is a finite set of action types

$in, end \in S$ are the initial state and the goal state

$\mathbf{a} : S \to \wp(A)$ gives the actions $\mathbf{a}(s)$ available at $s \in S$

$\mathbf{s} : S \times A \to \wp(S)$ gives the states $\mathbf{s}(s, \mathfrak{a})$ accessible via $\mathfrak{a}$ at $s \in S$

$\mathbf{cost} : S \times A \to \mathbb{R}^+$ gives the cost $\mathbf{cost}(s, \mathfrak{a})$ of performing $\mathfrak{a}$ at $s \in S$

We say that the actions in $\mathbf{a}(s)$ are *executable* at s and that s *sees* a state $s' \in S$ iff $s' \in \mathbf{s}(s, \mathfrak{a})$ for some $\mathfrak{a}$ which is executable at s. Finally, a solution of a problem consists in a path from in to end that minimizes the costs.

5.1. A Basic Problem

A sequence of cells numbered from 1 to 6 (as in Figure 7) is given and we are placed in cell 1. We can move from cell to cell by either making a *walk*, which leads us from cell n to $n + 1$, provided that $n + 1$ exists, or by making a *jump*, which leads us from cell n to $2n$, provided that $2n$ exists. Finally, making a walk costs 1 unit of time, while making a jump costs 2 units of time. The problem is to find the sequence of actions that minimizes the total cost in terms of time. Let us present this problem as a tuple $(S, A, in, end, \mathbf{a}, \mathbf{s}, \mathbf{cost})$:

1. $S = \{1, 2, 3, 4, 5, 6\}$
2. $A = \{walk, jump\}$
3. $in = 1, end = 6$
4. $\mathbf{a} : W \to \wp(A)$ is such that
    (a) $\mathbf{a}(n) = \{walk, jump\}$, if $1 \leq n \leq 3$
    (b) $\mathbf{a}(n) = \{walk\}$, if $4 \leq n \leq 5$

1	2	3	4	5	6

Figure 7. Sequence of cells

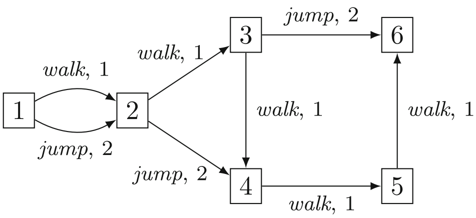


Figure 8. Problem graph

5. $\mathbf{s} : S \times A \to \wp(W)$ is such that
   (a) $\mathbf{s}(n, walk) = \{n+1\}$, if $1 \leq n \leq 5$
   (b) $\mathbf{s}(n, jump) = \{2n\}$, if $1 \leq n \leq 3$
6. $\mathbf{cost} : S \times A \to \mathbb{R}^+$ is such that
   (a) $\mathbf{cost}(n, walk) = 1$, if $1 \leq n \leq 5$
   (b) $\mathbf{cost}(n, jump) = 2$, if $1 \leq n \leq 3$

The problem can be pictured as a labeled directed graphs (Figure 8).

In this case the solution of the problem can be visualized without difficulty by explicitly considering all the possible paths from 1 to 6, namely by unraveling the problem graph (Figure 9). However, it is important to remark that, since the graph representing a search problem may include cycles, its unravelling may give rise to an infinite model. Here we use unravelling only for illustrating how an example of search problem can be encoded in our framework and such an encoding would work also in the presence of cycles in the initial graph.[10]

All in all, there are six paths with total costs: $5, 4, 5, 6, 5, 6$. The path minimizing the total cost is $(walk, walk, jump)$. So we can conclude that this path is the optimal one, thus solving our problem. Furthermore, we are in a position to specify the long run cost of each move by recursively computing the minimal total cost from each state to the end state (Figure 10).

The long run cost of walking in 5 is 1, since 1 is the minimal total cost to be paid to reach 6 from 5. In a similar way, the long run cost of walking in 3

[10]Tree-like models like the graph in Figure 9 are traditionally employed to interpret systems of *STIT*-logic. By contrast, in the present framework we mainly work with models like the graph in Figure 8. Furthermore, transitions between states are labelled in our models, whereas this is usually not the case in models for *STIT*. As the connection between Figures 8 and 9 suggests, by adapting the unravelling method from the model-theory of modal logic [2], one can transform each model in our framework into a tree-like model suitable for *STIT*-logic.

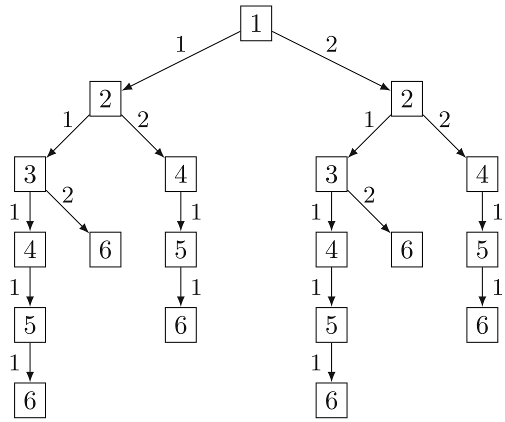


Figure 9. Unraveled problem graph

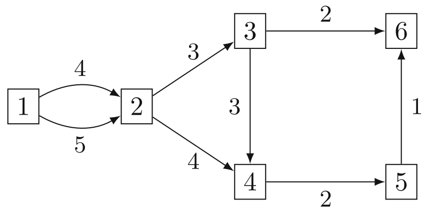


Figure 10. Long run cost of a move

is 3, since 3 is the minimal total cost to be paid to reach 6 if we walk, while the long run cost of jumping in 3 is 2, since 2 is the minimal total cost to be paid to reach 6 from 3 if we jump. Having determined the long run costs, we can assign a deontic value to each action at each state. In particular, walking is better than jumping at the first two states, since the long run cost of walking is 4 and 3, while the long run cost of jumping is 5 and 4; jumping is better than walking at state 3, since the long run cost of walking is 3, while the long run cost of jumping is 2. This final consideration allows us to understand how to construct a model from a search problem.

5.2. Modeling a Search Problem

The connection between searching problems and action deontic models is the following. A search problem is specified once the basic actions available at each state and the outcomes and costs of these actions are defined. Suppose that an action of an elementary type $\mathfrak{a}$ is available at w. Then, the agent is able to do $\mathfrak{a}$ at w, and so there is an alternative state from which a state where $\mathfrak{a}$ has been done is accessible. This provides a suitable way to represent a problem via an action deontic model and to shed light on the relation between action deontic models and transition systems at once.

The interesting part of the following definition concerns the specification of W, which contains worlds taking trace of the actions leading to them and the actions which are available in them. Accordingly, every world in W is a triple $(\mathfrak{a}, s, \mathfrak{b})$, where s is a state, $\mathfrak{a}$ is an elementary action leading to s, and $\mathfrak{b}$ is an elementary action available in s.

DEFINITION 12. Let $(S, A, in, end, \mathbf{a}, \mathbf{s}, \mathbf{cost})$ be a search problem. The action deontic model induced by this problem is the tuple $(W, R_i, A_i, D_i, \mathbb{D})$ where

1. W is the union of three sets:

   $\{(1, \mathsf{in}, \mathfrak{a}) : \mathfrak{a} \in \mathbf{a}(\mathsf{in})\}$
   $\{(\mathfrak{a}, \mathsf{end}, \mathfrak{a}) : \exists x(x \in S \text{ and } \mathsf{end} \in \mathbf{s}(x, \mathfrak{a}))\}$
   $\{(\mathfrak{a}, s, \mathfrak{b}) : \exists x(x \in S \text{ and } s \in \mathbf{s}(x, \mathfrak{s})), \exists y(y \in S \text{ and } y \in \mathbf{s}(s, \mathfrak{b}))\}$
2. R_i is such that $(\mathfrak{a}', s', \mathfrak{b}') \in R_i((\mathfrak{a}, s, \mathfrak{b}))$ iff $\mathfrak{a}' = \mathfrak{b}$ and $s' \in \mathbf{s}(s, \mathfrak{b})$
3. A_i is such that $(\mathfrak{a}', s', \mathfrak{b}') \in A_i((\mathfrak{a}, s, \mathfrak{b}))$ iff $s' = s$ and $\mathfrak{a}' = \mathfrak{a}$
4. D_i is inductively defined so that

   Base: $D_i(\mathfrak{a}) = \{(\mathfrak{a}', s', \mathfrak{b}') : \mathfrak{a}' = \mathfrak{a}\}$
   Step: $D_i(1) = W$
   $D_i(\overline{\alpha}) = W - D_i(\alpha)$
   $D_i(\alpha \sqcap \beta) = D_i(\alpha) \cap D_i(\beta)$
   $D_i(\alpha \sqcup \beta) = D_i(\alpha) \cup D_i(\beta)$
5. $\mathbb{D} = (DV, SV, \wedge_{\mathbb{D}}, c)$ is such that

   $DV = \{\mathbf{o}, \mathbf{i}, \mathbf{f}\}$
   $SV = \{\mathbf{o}\}$
   $\wedge_{\mathbb{D}}$ is defined by the matrix

	f	**i**	**o**
f	**f**	**f**	**f**
i	**f**	**i**	**i**
o	**f**	**i**	**o**

   c is such that: $c(w, \alpha) = \mathbf{o}$ iff the long run cost of doing α at w is minimum; $c(w, \alpha) = \mathbf{i}$ iff the long run cost of doing α at w is only minimal, but not the minimum; $c(w, \alpha) = \mathbf{f}$ otherwise.

As an example of how the transition system underlying a search problem is represented, let us consider the graph in Figure 11:

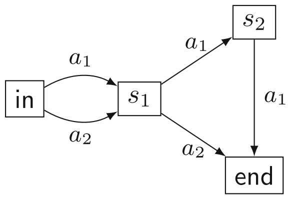


Figure 11. A simple transition system

This case is general enough to illustrate the idea of the previous definition and it allows us to better understand the connection between transition systems and action deontic frames. The graph contains four states and two actions:

(i) if different arrows *leave a state*, then different actions are available to the agent and therefore the state has to be split in different copies linked by the relation A_i;

(ii) if different arrows *lead to a state*, then again the state has to be split in different copies taking trace of the action leading to them, since states are not memoryless.

The states and relations in the corresponding action deontic frame are then as in Figure 12.

We now show that the induced frame is indeed an action deontic frame.

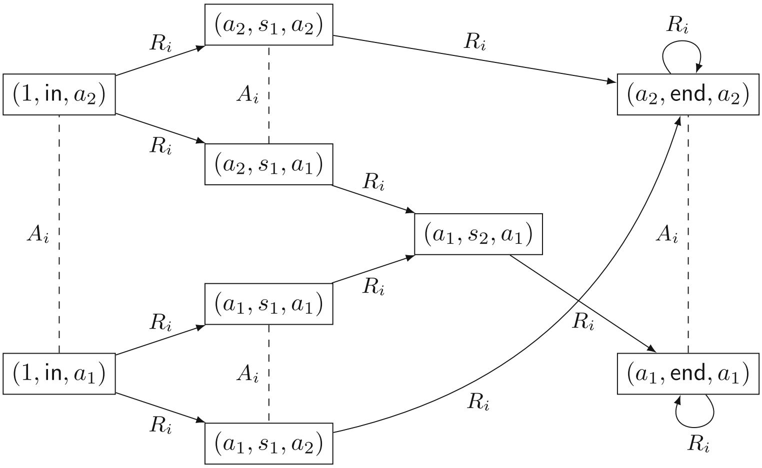


Figure 12. Representation of the previous transition system

THEOREM 8. *The frame induced by a search problem is an action deontic frame.*

PROOF. We show that R_i, A_i, D_i and $\mathbb{D}$ satisfy the relevant conditions.

R_i is such that $R_i(w) \neq \varnothing$ for every $w \in W$. Since $w \in W$, by definition, w has the form $(\mathfrak{a}, s, \mathfrak{b})$ for some $s \in S$ and some actions $\mathfrak{a}, \mathfrak{b}$. Furthermore, by construction, every state, except possibly end, sees another state and end sees itself. Therefore we conclude that $R_i(w) \neq \varnothing$ for every $w \in W$.

A_i satisfies conditions $A1$ and $A2$.

As to $A1$, by definition, $x \in A_i(w)$ iff the first two elements of x coincide with the first two elements of w. Suppose that $v \in A_i(w)$. If $x \in A_i(w)$ or $x \in A_i(v)$, then both the first two elements of x and w and the first two elements of x and v are the same. Thus, the first two elements of v and w are the same, and so $A_i(v) = A_i(w)$. In addition, again by definition, $v \in A_i(v)$, and thus $A_i(v) = A_i(w)$ implies $v \in A_i(w)$. It follows that $v \in A_i(w)$ iff $A_i(v) = A_i(w)$.

As to $A2$, suppose that $v \in R_i(w)$ and $x \in A_i(v)$. Let $w = (\mathfrak{a}, s, \mathfrak{b})$. Then, v has the form $(\mathfrak{b}, s', \mathfrak{b}')$, for some action $\mathfrak{b}'$, and x has the form $(\mathfrak{b}, s', \mathfrak{b}'')$, for some action $\mathfrak{b}''$, which implies that $x \in R_i(w)$.

D_i satisfies conditions $D1$ to $D4$: straightforward, by the definition of D_i.

$\mathbb{D}$ is a deontic structure: straightforward, by the definition of $\mathbb{D}$. ■

Let us close this section by showing how language $\mathcal{L}_0$ can be used to describe some aspects of a search problem. The basic idea is that one can use a formula of the form $\mathbf{O}\mathfrak{a}$ to mean that performing an action of type $\mathfrak{a}$ has a minimal long run cost for the reasoning agent. So, given the definition of truth in models for $\mathcal{L}_0$, we conclude that $M, w \models \mathbf{O}\mathfrak{a}$ precisely when $\mathfrak{a}$ is one of the actions with a minimum long run cost at w. Notice that, since the actions that are classified as obligatory in the context of a search problem are always among those available to the reasoning agent, the following principle, which incorporates a version of the Ought-implies-Can thesis, should be assumed as valid in all models representing a search problem: $\mathbf{O}\mathfrak{a} \rightarrow \Diamond_i \langle i \rangle done_i(\mathfrak{a})$. From this it also follows that, in a model for a search problem, the state *end* can be described in a purely normative way: it is the only state where every formula of the form $\mathbf{O}\mathfrak{a}$ is false. Also, some of the structural aspects of a problem can be precisely described in $\mathcal{L}_0$, once an appropriate valuation function is selected. Specifically: suppose that a problem has states $S = \{1, 2, ..., n\}$; select V such that $V(p_s) = \{s\}$ for $s \in S$. Then there are formulas corresponding to the fact that an action is executable at a state and to the fact that the current state can see another state. In fact, if M is the model based on the frame induced by such a problem:

$M, w \models \Diamond_i done_i(\mathfrak{a})$ iff $\mathfrak{a} \in \mathbf{a}(w)$

$M, w \models \Diamond_i(done_i(\mathfrak{a}) \wedge p_s)$ iff $s \in \mathbf{s}(w, \mathfrak{a})$

This shows at least some of the potentialities deriving from the representation obtained above.

6. Final Remarks

The approach to action deontic logic introduced in this article was shown to be expressively rich and very flexible with respect to the representation of normative reasoning. In particular, its main strength is an encoding of fundamental relations between action tokens, action types and deontic values; an encoding that makes room for combining and generalizing alternative formal languages employed in the literature.

Several directions could be followed in order to extend our work. Here we list three important ones, that are directly connected to issues mentioned within our exposition. First, building on the analysis of the three systems by Kulicki and Trypuz [13] that we provided in Section 4.2, one could show how to capture other systems of normative reasoning available in the literature, shedding further light on the expressiveness of our proposal. In this direction, we are working on some results that show how standard deontic logic, von Wright's original action logic ([24,25]), and Segerberg's basic system of action deontic logic, together with some of its extensions ([3,22,23]), emerge as fragments of the present system. Second, one could explore the definition of dyadic operators for conditional norms, starting with the ideas presented in Section 2.3, and address some of the well-known paradoxes related to the notion of conditional obligation. An interesting issue in this regard is determining the fragment of a logical system associated with a dyadic operator, i.e., for a given system S and dyadic operator $\mathbf{O}$, axiomatizing the set of theorems of S in which $\mathbf{O}$ is the main operator. Third, one could exploit the deontic matrices used in our semantics to capture a hierarchy among normative sources, so as to always assign top values in the matrix to norms coming from prioritized sources, thus contributing to the debate concerning the use of priority orderings (either on norms or on reasons) in deontic logic (as exploited, e.g., in [19,20]).

Acknowledgements. Matteo Pascucci was supported by the *Štefan Schwarz Fund* for the project "A fine-grained analysis of Hohfeldian concepts" (2020–2022).

Author contributions The article is the result of a joint research work of the two authors.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

[1] Anglberger, A., Dynamic deontic logic and its paradoxes, *Studia Logica* 89:807–827, 2008.

[2] van Benthem, J., *Modal Logic and Classical Logic*, Bibliopolis, Naples, 1983.

[3] Castro, P. F., and P. Kulicki, Deontic logics based on Boolean algebra, in R. Trypuz, (ed.), *Krister Segerberg on Logic of Actions*, Springer, 2014, pp. 85–117.

[4] Chellas, B. F., Time and modality in the logic of agency, *Studia Logica* 51:485–517, 1992.

[5] Giordani, A., Ability and responsibility in general action logic, in J. Broersen, C. Condoravdi, S. Nair, and G. Pigozzi, (eds.), *Deontic Logic and Normative Systems*, Proceedings of DEON, 2018, pp. 121–138.

[6] Glavaničová, D., and M. Pascucci, A realistic view on normative conflicts, *Logic and Logical Philosophy* 29:447–462, 2020.

[7] Herzig, A., and E. Lorini, A dynamic logic of agency I: STIT, capabilities and powers, *Journal of Logic, Language and Information* 19:89–122, 2010.

[8] Horty, J., *Agency and Deontic Logic*, Oxford University Press, Oxford, 2001.

[9] Horty, J., and E. Pacuit, Action types in stit semantics, *Review of Symbolic Logic* 10:617–637, 2017.

[10] Hughes, G. E., and M. J. Cresswell, *A New Introduction to Modal Logic*, Routledge, London, 1996.

[11] Hurbans, R., *Grokking Artificial Intelligence Algorithms*, Manning Publications, New York, 2020.

[12] Jones, A. J. I., and I. Pörn, Ideality, sub-ideality and deontic logic, *Synthese* 65:275–290, 1985.

[13] Kulicki, P., and R. Trypuz, Multivalued logics for conflicting norms, in O. Roy, A. Tamminga, and M. Willer, (eds.), *Proceedings of DEON*, 2016, pp. 123–138.

[14] Lorini, E., and A. Herzig, A logic of intention and attempt, *Synthese* 163:45–77, 2008.

[15] McNamaraR., P., Praise, blame, obligation, and DWE: toward a framework for classical supererogation and kin. *Journal of Applied Logic* 9:153–170, 2011.

[16] Meyden, R. van der, The dynamic logic of permission, *Journal of Logic and Computation* 6:465–479, 1996.

[17] Meyer, J. J. C., A different approach to deontic logic: deontic logic viewed as a variant of dynamic logic, *Notre Dame Journal of Formal Logic* 29:109–136, 1987.

[18] Meyer, J. J. C, J. Broersen, and A. Herzig, BDI Logics, in H. van Ditmarsch, J. Y. Halpern, W. van der Hoek, and B. Kooi, (eds.), *Handbook of Logics for Knowledge and Belief*, College Publications, 2015, pp. 453–498.

[19] Nair, S., Conflicting reasons, unconflicting 'ought's, *Philosophical Studies* 173:629–663, 2016.

[20] Nair, S., and J. Horty, The logic of reasons, in D. Star, (ed.), *The Oxford Handbook of Reasons and Normativity* Oxford University Press, 2018 pp. 67–85.

[21] Russell, S. J., and P. Norvig, *Artificial Intelligence. A Modern Approach,* Prentice Hall, New Jersey, 2010.

[22] Segerberg, K., A deontic logic of action, *Studia Logica* 41:269–282, 1982.

[23] Trypuz, R., and P. Kulicki, On deontic action logics based on Boolean algebra, *Journal of Logic and Computation* 25:1241–1260, 2015.

[24] von Wright, G.H., Deontic logic, *Mind* 237:1–15, 1951.

[25] von Wright, G. H., *Norm and Action: A Logical Inquiry,* Routledge and Kegan Paul, London, 1963.

[26] von Wright, G. H., Deontic logic and the theory of conditions, in R. Hilpinen, (ed.), *Deontic Logic: Introductory and Systematic Readings*, Reidel, 1971 pp. 159–177.

[27] von Wright, G. H., Problems and prospects of deontic logic: a survey, in E. Agazzi, (ed.), *Modern Logic. A Survey* 1980 pp. 399–423.

A. Giordani
Dipartimento di Filosofia
Università Cattolica del Sacro Cuore di Milano
Milano
Italy
alessandro.giordani@unicatt.it

M. Pascucci
Department of Analytic Philosophy, Institute of Philosophy
Slovak Academy of Sciences, V.V.I.
Bratislava
Slovakia
matteopascucci.academia@gmail.com