Supercover Semantics for Deontic Action Logic

Karl Nygren

© The Author(s) 2019

Abstract
The semantics for a deontic action logic based on Boolean algebra is extended with an interpretation of action expressions in terms of sets of alternative actions, intended as a way to model choice. This results in a non-classical interpretation of action expressions, while sentences not in the scope of deontic operators are kept classical. A deontic structure based on Simons’ supercover semantics is used to interpret permission and obligation. It is argued that these constructions provide ways to handle various problems related to free choice permission. The main result is a sound and complete axiomatization of the semantics.

Keywords Action · Alternative semantics · Choice · Deontic logic · Free choice permission · Supercover semantics

1 Introduction

The standard approach to deontic logic, represented by standard deontic logic (SDL), suffers from a large number of so-called paradoxes, where the paradox of free choice permission is perhaps the most discussed in the literature. These are not paradoxes in any strict sense, but rather clashes between inferences valid in formal deontic logics, and intuitively valid and non-valid inferences as they occur in informal deontic reasoning.

SDL is the normal modal logic KD. Models are of the form \( \langle W, R, V \rangle \), where \( W \) is a non-empty set of possible worlds, \( R \) is a binary relation on \( W \), and \( V \) is a valuation function for the propositional variables of the language. The relation \( R \) has the intended meaning that \((w, v) \in R\) if and only if \( v \) is ideal from the point of view of \( w \). The set \( I^w = \{v \in W : (w, v) \in R\} \) is the set of ideal worlds from the point of view of \( w \). The relation \( R \) is assumed to be serial, i.e. for each \( w \in W \), there is some \( v \in W \) such that \((w, v) \in R\). A sentence \( P(\varphi) \)—where the intended reading of \( P \) is
it is permitted that…’—is then true at a world \( w \) if \( \varphi \) is true in some world that is ideal from the point of view of \( w \), and a sentence \( O(\varphi) \)—where \( O \) has the intended reading ‘it is obligatory that…’—is true at a world \( w \) if \( \varphi \) is true in all worlds that are ideal from the point of view of \( w \).

The problems of free choice permissions stem from the intuition that the sentences “Jane may go by bus or go by train” and “Jane may go by bus and Jane may go by train” seem to amount to more or less the same thing. One would then expect the following principle of free choice to be valid:

\[
P(\varphi \lor \psi) \leftrightarrow P(\varphi) \land P(\psi).
\]

This problem may be illustrated with the vegetarian free lunch example (Hansson 2013). If it is permitted to order a vegetarian meal, then, by (3), it is permitted to order a vegetarian meal and not pay for it. Even very weak assumptions allow the inference of (3) from (1), which shows that the problem of free choice is not that easy to avoid. According to Hansson, this “indicates that the free choice postulate may be faulty in itself, even if not combined with other deontic principles such as those of SDL.” (2013, p. 208).

Since the seminal work of von Wright (1951), deontic logics where the deontic operators attach to action expressions (names of actions) have been studied in different forms, for example in deontic variants of propositional dynamic logic (PDL) (e.g., Castro and Maibaum 2009; Meyer 1988; van der Meyden 1996), as well as in static deontic action logics (Bentzen 2014; Segerberg 1982; Trypuz and Kulicki 2009, 2015). The basic idea behind the semantics to be developed is that a disjunctive action expression such as ‘go by bus or go by train’ expresses a choice between different courses of action corresponding to each disjunct. As an example, consider the sentence “Jane may go by bus or by train”. This sentence can be taken to express a permission to choose between two different courses of action: the action of going by bus, and the action of going by train:
Jane may choose an action from the set $\{\text{go by bus}, \text{go by train}\}$, where $\text{go by bus}$ and $\text{go by train}$ are the actions described by the expressions ‘go by bus’ and ‘go by train’, respectively. In general, a sentence of the form ‘Permitted $\alpha$ or $\beta$’ can be analyzed as saying:

Permitted to choose an action from the set $\{[\alpha], [\beta]\}$.

Given certain natural assumptions, an analysis along these lines will produce the desired properties of free choice permissions. If it is permitted to choose between having pasta and having pizza, then it is also permitted to have pasta and permitted to have pizza. In general, if every action in a set of actions are permitted, then any (non-empty) subset of that set will contain permitted actions only. This idea is not new; according to Hansson,

\[ \text{Hansson does not develop the idea further. In the context of natural language semantics, the alternative semantics approach interprets disjunction as introducing sets of alternatives (see e.g., Alonso-Ovalle 2006; Ciardelli and Roelofsen 2011; Ciardelli et al. 2009; Groenendijk and Roelofsen 2009; Roelofsen 2013; Simons 2005a, b). The idea is, essentially, that a disjunctive statement tells you that at least one proposition from a larger set of propositions is true. One of the motivations for this kind of interpretation of disjunction is to account for free choice phenomena. The alternative analysis of disjunction allows interpreting permission modals as operating over sets of alternatives. A permitted disjunction, then, tells you that each proposition from the set of alternatives introduced by the disjunctive statement is permitted. Simons (2005a, b) develops this idea in more detail, introducing a precise notion of how deontic modals interact with a set of alternatives in terms of supercovers. However, her account makes certain counterintuitive predictions with regards to disjunctive statements not in the scope of deontic operators. In addition, Simons does not develop a full formal semantics suitable for a deontic logic.}

In this paper, I will combine Simons’ alternative semantics approach to disjunction and deontic modals with a static deontic action logic based on Boolean algebra. Specifically, I will be concerned with logics in the style of Segerberg (1982) and Trypuz and Kulicki (2009, 2015), and the action-theoretic layer of the logic of Castro and Maibaum (2009). In these logics, the arguments of deontic operators are names of actions, rather than propositional statements. Action expressions are given a semantics based on an underlying Boolean algebra, and permission is interpreted as an ideal on this algebra. This construction validates the free choice principle; however, substitution of Boolean algebra identities within the scope of the permission operator is a valid rule of inference, so the vegetarian free lunch problem arises. In addition, the distinction between action expressions and propositions is syntactically sharp, but semantically less so—the logical behavior of both types of expressions is essentially classical. When the semantics is extended with an interpretation of action expressions in terms of sets of action types, the logic of action expressions will deviate from the
Boolean algebra interpretation, resulting in a system where propositional formulas behave classically, while action expressions do not. I will argue that this move indeed offers ways to avoid the problems of free choice permission.

The structure of the paper is as follows. Section 2 introduces Simons’ supercover semantics and highlights some problems. In Sect. 3.1, basic action theoretic and deontic structures are defined; these structures are then used to construct a formal semantics in Sect. 3.2. I discuss the concepts of permission and obligation in Sect. 4. In Sect. 5, an axiomatization of the semantics is defined, and soundness and completeness results are obtained. In Sect. 6, I discuss certain aspects of the logic in more detail and compare the semantics to related work. Section 7 concludes the paper and discusses ideas for further research.

2 Supercover Semantics

Simons’ supercover semantics is an attempt to deal with free choice phenomena in the context of natural language semantics (Simons 2005a, b). Simons suggests supplementing the standard semantics for disjunction in the scope of deontic modals with a distribution requirement. This distribution requirement guarantees that $P(\phi \text{ or } \psi)$ is true only if there are ideal worlds in both the proposition expressed by $\phi$ and the proposition expressed by $\psi$, and that $O(\phi \text{ or } \psi)$ is true only if there are ideal worlds in both disjunct propositions, and the sum of the proposition expressed by $\phi$ and the proposition expressed by $\psi$ contains all ideal worlds.

Formally, this is accomplished in two stages. First, ‘or’ is treated as a set-formation operator, introducing sets containing propositions corresponding to the disjuncts. Second, Simons introduces the notion of a supercover. Let $U$ be some background set. A non-empty $S \subseteq \mathcal{P}(U)$ is a supercover of $A \subseteq U$ if and only if (i) every member of $S$ contains some member of $A$, and (ii) every member of $A$ belongs to some member of $S$ (Simons 2005a, p. 276). In other words, $S$ is a supercover of $A$ if the union of $S$ is a superset of $A$, and every element of $S$ has a non-empty intersection with $A$.

Denote the set of possible worlds where $\phi$ is true by $\parallel \phi \parallel$. Simons’ proposal for the interpretation of ‘or’-sentences is obtained by the clause (Simons 2005a, p. 292):

- $(\phi \text{ or } \psi)$ is true at $w \in W$ iff $\{\parallel \phi \parallel, \parallel \psi \parallel\}$ is a supercover of some $X \subseteq W$ with $w \in X$.

This clause expresses the idea that each disjunct in an ‘or’-sentence is non-vacuous: each disjunct contributes to the truth of the whole sentence in which it occurs (Simons 2005b, p. 212). Turning to deontic sentences, Simons suggests defining truth conditions for sentences of the form $P(\phi \text{ or } \psi)$ and $O(\phi \text{ or } \psi)$ as follows.

- $P(\phi \text{ or } \psi)$ is true at $w \in W$ iff $\{\parallel \phi \parallel, \parallel \psi \parallel\}$ is a supercover of some $X \subseteq I^w$.
- $O(\phi \text{ or } \psi)$ is true at $w \in W$ iff $\{\parallel \phi \parallel, \parallel \psi \parallel\}$ is a supercover of $I^w$.

---

1 The term ‘supercover semantics’ is used by Humberstone (2011, p. 811), but not, as far as I am aware, by Simons herself.

2 I have omitted any references to specific models.
I^w is the set of ideal worlds form the point of view of w. Given non-disjunctive and non-deontic \( \varphi \) and \( \psi \), the first clause gives the free choice equivalence between \( P(\varphi \lor \psi) \) and \( P(\varphi) \land P(\psi) \). In addition, Ross’ paradox (Ross 1941), which consists in the inference of \( O(\varphi \lor \psi) \) from \( O(\varphi) \), is blocked by the supercover condition in the second clause. On the other hand, the interpretation of ‘or’-sentences not embedded under deontic operators fails to account for classical disjunctions. Suppose that \( (\varphi \lor \psi) \) is true at \( w \) under the supercover semantics. This implies that \( w \) is in either \( \parallel \varphi \parallel \) or \( \parallel \psi \parallel \) (or both). The supercover semantics essentially agrees with Boolean disjunction in this aspect. As noted by Simons (2005a, p. 293, n. 33), the other direction—from \( \varphi \) infer \( (\varphi \lor \psi) \)—is much more problematic. The principle of disjunction introduction may be explicated as follows:

If \( \varphi \) is true at \( w \), then \( (\varphi \lor \psi) \) is true at \( w \).

In the usual Boolean semantics, disjunction introduction is a valid principle, since \( \parallel \varphi \lor \psi \parallel = \parallel \varphi \parallel \lor \parallel \psi \parallel = \parallel \varphi \parallel \cup \parallel \psi \parallel \). In supercover semantics, on the other hand, disjunction introduction is not valid. Consider the following example sentence given by Simons (2005a, p. 293, n. 33): “Jane dances or \( 2 + 2 = 5 \)”. Under the supercover semantics (provided that “\( 2 + 2 = 5 \)” is false at every possible world), this sentence will turn out false at all worlds, since a supercover cannot contain the empty set. Nevertheless, there is a strong intuition that if “Jane dances” is true, then “Jane dances or \( 2 + 2 = 5 \)” is true as well. In general, any disjunction with a logically impossible disjunct will inevitably be false under the supercover semantics. This constitutes a counter-example to the Boolean principle of disjunction introduction.

It is an open question whether it is possible to tinker with the supercover semantics in such a way as to make it agree with Boolean disjunction outside the scope of deontic modals. In this paper, I will take a different approach to this problem. With a careful distinction between action expressions and propositional statements, the usual Boolean story may be told for propositional statements, while a supercover semantics may be developed for action expressions. This makes it possible to deal with free choice phenomena while keeping classical disjunction in purely factual contexts outside the scope of deontic modals.

Are there independent reasons for giving up disjunction introduction for action expressions? Anglberger et al. (2014) argue that there are. One can for example appeal to normality, in the sense that action expressions are taken to refer to normal instances of actions (Pelletier and Asher 1997), or one may argue that action expressions are resource sensitive in the sense of Linear Logic (Girard 1987). Anglberger, Dong, and Roy give an example where action expressions display a kind of resource sensitivity (2014, pp. 24–25). Suppose that you are at a restaurant and are offered to order sushi. This does not imply that you are offered to choose between ordering sushi or pasta. The chef at the restaurant might not know how to cook pasta, or the restaurant might not have the right ingredients for making pasta, etc. In many cases, it seems that choosing between different alternatives is only possible if all the alternatives are live options.
3 A Deontic Logic Based on Supercover Semantics

In Segerberg’s seminal paper (1982), and in the subsequent extensions and variants (Castro and Maibaum 2009; Trypuz and Kulicki 2009, 2015), actions are interpreted as elements of a Boolean algebra, with permission and prohibition interpreted as (disjoint) ideals on this algebra. This means that these logics take notions of strong permission and prohibition as primitive. The model theoretic characterization of these logics interprets elements of the Boolean algebra of actions as sets of action tokens. The semantics to be developed in this paper extends this theory of action by introducing sets of sets of action tokens, and replaces the interpretation of permission by a deontic structure based on Simons’ supercover semantics. The supercover semantics is here developed in a different way compared to the presentation in the previous section. In particular, legal and required sets of sets of action tokens are collected in two sets satisfying certain conditions, similar to neighborhood semantics for modal logics (see e.g. Chellas 1980, Chapter 7).

3.1 Action, Choice and Deontic Status

Assume that a single agent in a single situation has available a (non-empty) set \( H = \{h_1, h_2, \ldots \} \) of action tokens. Action tokens can instantiate action types. Formally, an action type is a subset of \( H \). Let \( \mathcal{A} = \mathcal{P}(H) \) be the set of action types. Symbols \( s, t, \ldots \) will be used as variables ranging over action types from \( \mathcal{A} \). I will also introduce the notion of a choice set. Formally, a choice set is a set of action types. While the impossible action type is represented by the empty set, choice sets are always required to be non-empty. Let \( \mathcal{C} = \mathcal{P}(\mathcal{A}) \backslash \{\emptyset\} \) be the set of choice sets available to the agent in the situation. Symbols \( S, T, \ldots \) are used as variables ranging over choice sets from \( \mathcal{C} \).

Informally, a choice set represents choosing between the action types in the choice set. As an example, suppose that \( s_1 \) and \( s_2 \) are the action types denoted by the expressions ‘go by bus’ and ‘go by train’, respectively. The expression ‘go by bus or go by train’ can then be interpreted as denoting the choice set \( \{s_1, s_2\} \). The idea, then, is that the expression ‘go by bus or go by train’ has a reading on which it describes a choice between the action denoted by ‘go by bus’ and the action denoted by ‘go by train’.

Let \( G \subseteq H \) be a non-empty subset of the set of action tokens. \( G \) is supposed to represent the set of legal (right, good, deontically ideal, etc.) action tokens in the situation. This set is used to determine the deontic status of action types and choice sets. An action type is said to be legal if it contains some legal action token, i.e. if there is some legal way to perform the action type. A choice set is said to be legal if all the action types in it are legal—the agent is permitted to choose between different action types only if each one of them is legal. Formally, the set of legal choice sets is

\[ 3 \] The deontic status of action types and choice sets is thus reduced to the claim that certain action tokens are legal or not. I leave the legalness of action tokens as a primitive notion here. This notion could, for example, be analyzed in terms of sanctions or violations as done in some variants of dynamic deontic logic, e.g. Broersen (2004), Dignum et al. (1996) and Meyer (1988).
defined as

\[
\text{LEG} \overset{\text{def}}{=} \{ S \in \mathcal{C} : \text{for all } s \in S, s \cap G \neq \emptyset \}.
\]

As an example, take the sentence “Jane may go by bus or go by train”. The idea here is that this sentence has a reading on which it says that both the action type denoted by ‘go by bus’ and the action type denoted by ‘go by train’ are permitted. Again, let \( s_1 \) be the action type denoted by ‘go by bus’ and \( s_2 \) the action type denoted by ‘go by train’. The proposal, then, is that the sentence ‘Jane may go by bus or go by train’ is interpreted as saying that \( \{ s_1, s_2 \} \in \text{LEG} \). This, in turn, is true if and only if \( \{ s_1 \} \in \text{LEG} \) and \( \{ s_2 \} \in \text{LEG} \). This property of \text{LEG} generalizes: if a choice set \( S \) is legal, then every non-empty subset of \( S \) is legal, and if two choice sets \( S, T \) are legal, the union \( S \cup T \) is also legal. In this sense, the just introduced semantic constructs can be used to model free choice permission.

I will also define the set of \textit{required} choice sets, which will be used to interpret a concept of obligation. To do this, a slightly different construction is needed. A choice set is legal if it contains legal action types only. On the other hand, a choice set may be required even if it contains non-required action types. The following example given by Alchourrón and Bulygin illustrates this point:

There is e.g. the well known case of Sempronius who has an obligation to give a cow or a horse to Ticius, but he has not the obligation to give Ticius a cow nor has he the obligation to give him a horse. He can fulfill his obligation by giving either of the two things, since he must give one of the two, but he is not obliged to give either in particular. (1971, p. 157)

Thus, Sempronius is obliged to give a cow or a horse, but neither obliged to give a cow, nor obliged to give a horse. On the other hand, it may be said that Sempronius is obliged to choose one of the alternatives. I will interpret this as meaning that the sum of the action types in a required choice set contains all legal action tokens, and that each action type in a required choice set is legal. The last assumption seems quite natural, since if Sempronius can “fulfill his obligation by giving either of the two things”, it can hardly be the case that he is not permitted to, for example, give a cow. Formally, let

\[
\text{REQ} \overset{\text{def}}{=} \left\{ S \in \text{LEG} : G \subseteq \bigcup S \right\}.
\]

While the above example gives some intuitions regarding the particular modelling choices when it comes to obligation, the primary reason for introducing the \text{REQ} set is that it reflects Simons’ truth conditions for obligation in terms of supercovers. Indeed, the sets \text{LEG} and \text{REQ} can be formulated equivalently using the supercover notion.

\textbf{Theorem 1} \quad \text{Let } S \in \mathcal{C}. \text{ Then}

1. \( S \in \text{LEG} \) if and only if there is \( X \subseteq G \) such that \( S \) is a supercover of \( X \),

---

4 This set must be distinguished from the set \text{Leg} defined by Segerberg (1982), which corresponds to the set denoted by \( G \) here.
2. \( S \in \text{REQ} \) if and only if \( S \) is a supercover of \( G \).

**Proof** 1. (\( \Rightarrow \)) Suppose that \( S \in \text{LEG} \). Let \( X = \bigcup \{s \cap G : s \in S\} \). Clearly, \( X \subseteq G \). First, every element of \( S \) has a non-empty intersection with \( X \). Second, it holds that \( X \subseteq \bigcup S \); hence, \( S \) is a supercover of \( X \).

(\( \Leftarrow \)) Suppose that \( S \) is a supercover of some \( X \subseteq G \). Then, for every \( s \in S \), \( s \cap X \neq \emptyset \); and so \( s \cap G \neq \emptyset \). It follows that \( S \in \text{LEG} \).

2. (\( \Rightarrow \)) Suppose that \( S \in \text{REQ} \). By definition, \( G \subseteq \bigcup S \) and \( S \in \text{LEG} \), so every element of \( S \) has a non-empty intersection with \( G \). It follows that \( S \) is a supercover of \( G \).

(\( \Leftarrow \)) Suppose that \( S \) is a supercover of \( G \). It follows that \( G \subseteq \bigcup S \). By the right-to-left direction of the first item of the theorem, it holds that \( S \in \text{LEG} \). Hence, \( S \in \text{REQ} \). \( \square \)

### 3.2 Deontic Action/Choice Logic

#### 3.2.1 Language

The language for the deontic action/choice logic (DACL) to be developed is similar to the language used for the deontic logics based on Boolean algebra (Castro and Maibaum 2009; Segerberg 1982; Trypuz and Kulicki 2009, 2015). It is characteristically two-sorted, with a clear distinction between action expressions and propositional formulas. The language is given by the following recursive syntax rules:

\[
\varphi, \psi ::= \top \mid \alpha \equiv \beta \mid \alpha \Rightarrow \beta \mid P(\alpha) \mid \neg \varphi \mid (\varphi \land \psi)
\]

\[
\alpha, \beta ::= a_i \mid 1 \mid 0 \mid \overline{\alpha} \mid (\alpha \sqcup \beta) \mid (\alpha \sqcap \beta)
\]

where \( a_i \) belongs to a finite set \( \text{Act}_0 = \{a_1, a_2, \ldots, a_n\} \) of generators. Let \( \text{Act} \) be the set of terms defined by (5) for fixed \( \text{Act}_0 \). The missing operators of propositional logic \( \lor, \rightarrow, \) and \( \leftrightarrow \) are defined as usual. The term connectives \( -, \sqcup, \) and \( \sqcap \) will be referred to as action-negation, action-disjunction, and action-conjunction, respectively, or when there is no risk of confusion with the corresponding propositional connectives, they will be referred to simply as negation, disjunction, and conjunction. The designated term \( 0 \) represents the impossible (empty) action, while the designated term \( 1 \) represents the universal (necessary) action.

The language has three types of atomic formulas. First, formulas of the form \( \alpha \equiv \beta \) are used to express type-identity, with the intended meaning that \( \alpha \) and \( \beta \) refer to the same action type. Second, formulas of the form \( \alpha \Rightarrow \beta \) are used to express choice-identity, meaning that \( \alpha \) and \( \beta \) describe the same choices. Third comes deontic formulas of the form \( P(\alpha) \), intended to express that \( \alpha \) is permitted. The operators of propositional logic are then used to form complex statements about equivalence and deontic status of actions and choices.
### 3.2.2 Algebra of Actions

In the deontic action logics based on Boolean algebra (Castro and Maibaum 2009; Segerberg 1982; Trypuz and Kulicki 2009, 2015), the action-theoretic layer is described by an underlying Boolean algebra of terms from Act. A standard set of axioms for Boolean algebra is given below.

\[
\begin{align*}
\alpha \sqcup \beta &= \beta \sqcup \alpha, \quad \alpha \sqcap \beta = \beta \sqcap \alpha \tag{A1} \\
(\alpha \sqcup \beta) \sqcup \delta &= \alpha \sqcup (\beta \sqcup \delta), \quad (\alpha \sqcap \beta) \sqcap \delta = \alpha \sqcap (\beta \sqcap \delta) \tag{A2} \\
(\alpha \sqcap \beta) \sqcup \delta &= \beta, \quad \alpha \sqcap (\alpha \sqcap \beta) = \alpha \tag{A3} \\
\alpha \sqcap (\beta \sqcup \delta) &= (\alpha \sqcap \beta) \sqcup (\alpha \sqcap \delta) \tag{A4} \\
\alpha \sqcup \overline{\alpha} &= 1, \quad \alpha \sqcap \overline{\alpha} = 0 \tag{A5}
\end{align*}
\]

In Boolean algebras, a partial order \(\leq\) is induced by definition: \(\alpha \leq \beta\) if and only if \(\alpha = \alpha \sqcap \beta\).

Since fields of sets are Boolean algebras (Monk 1976, pp. 150–151), there is a straightforward way to construct a semantics based on the theory of action described in Sect. 3.1. Every term from Act is mapped to an element of the set of action types \(\mathcal{A}\) such that the designated terms 1 and 0 are mapped to \(H\) and \(\emptyset\), respectively, and complex terms involving term operations correspond to the set-theoretical operations of union, intersection and complement in the obvious way.

Recall that the set of generators \(Act_0\) defined in the previous section is assumed to be finite. As is well-known, a Boolean algebra generated by a finite set of generators is atomic. If there are \(n\) generators in \(Act_0\), every atom in the Boolean algebra generated by \(Act_0\) is equal to an element of the form

\[
\delta_1 \sqcap \cdots \sqcap \delta_n,
\]

where each \(\delta_i\) is one of the generators from \(Act_0\) or its negation. It should be pointed out that not every element that is equal to an element of this form is an atom, since it might be equal to the bottom element 0. Every element not equal to 0 of an atomic Boolean algebra is equal to a sum of atoms (Monk 1976). In particular, note that the top element 1 is equal to the sum of all atoms, while the bottom element 0 contains no atoms. Assuming that the set of generators is finite is integral to the proof of completeness, and it will be possible to make use of results obtained by Castro and Maibaum (2009) and Trypuz and Kulicki (2009). In particular, atoms are used to construct canonical models.

### 3.2.3 Deontic Action Models and Semantics

In this section, the basic action theoretic and deontic structures defined in Sect. 3.1 will be used to construct a formal semantics.

A deontic action model \(M\) is a structure \(\langle H, G, V \rangle\), where \(H\) is a non-empty set representing the available action tokens, \(G \subseteq H\) is a non-empty subset of \(H\) representing the legal action tokens, and \(V: Act_0 \to \mathcal{P}(H)\) is a valuation function
assigning an action type \( V(a_i) \subseteq H \) to every generator \( a_i \in Act_0 \). To get models for a Boolean algebra of terms from \( Act \), the function \( V \) is recursively extended to \( Act \) using the usual set-theoretical operations:

\[
\begin{align*}
V(1) &= H \\
V(0) &= \emptyset \\
V(\overline{\alpha}) &= H \setminus V(\alpha) \\
V(\alpha \sqcup \beta) &= V(\alpha) \cup V(\beta) \\
V(\alpha \sqcap \beta) &= V(\alpha) \cap V(\beta).
\end{align*}
\]

These models are similar to the models defined by Trypuz and Kulicki (2009, 2015). In their models, however, atoms of the underlying algebra of action are interpreted as singleton sets. This assumption makes it possible to consider legal closure principles at the semantic level. For my purposes here, this assumption is not necessary.

Every term \( \alpha \) is also associated with a choice set. To interpret \( \sqcap \) and \( \sim \), I will introduce two special operations on choice sets. For \( S, T \in \mathcal{C} \), define

\[
S \otimes T = \{ s \cap t : s \in S \text{ and } t \in T \},
\]

and

\[
\sim S = \text{Alt} \{ s \cap t : s \cap t = \emptyset \text{ for all } t \in S \},
\]

where the operator \( \text{Alt} \), originally introduced by Groenendijk and Roelofsen (2009), is defined as follows: for \( X \) a set of sets, \( \text{Alt} X = \{ x \in X : \text{ for no } y \in X, x \subset y \} \). The \( \text{Alt} \) operator picks out the maximal elements of a set \( X \) in the sense that it removes all elements that are proper subsets of some other element in \( X \).

The choice set associated with a term \( \alpha \in Act \) in a model \( M \), denoted \( [\alpha]_M \), is recursively defined as follows:\footnote{I will sometimes leave out the subscript \( M \) when there is no risk of misunderstandings.}

\[
\begin{align*}
[a_i]_M &= \{ V(a_i) \} \\
[1]_M &= \{ H \} \\
[0]_M &= \{ \emptyset \} \\
[\alpha \sqcup \beta]_M &= [\alpha]_M \cup [\beta]_M \\
[\alpha \sqcap \beta]_M &= [\alpha]_M \otimes [\beta]_M \\
[\overline{\alpha}]_M &= \sim [\alpha]_M.
\end{align*}
\]

The following result follows directly from the definition of choice sets.

**Theorem 2** For any \( \alpha \in Act \) and \( M = \langle H, G, V \rangle \), \( V(\alpha) = \bigcup [\alpha]_M \).

In light of this result, one can think of \( V(\alpha) \) as the range of \( \alpha \): the set of action tokens which are live options in the choice described by \( \alpha \).
A choice set $\llbracket \alpha \rrbracket$ should be thought of as the choices described by the term $\alpha$. A term whose choice set is a singleton describes a trivial choice; a choice with only one alternative. A term describes a genuine choice if its choice set consists of more than one alternative. It is clear that the only way to describe a genuine choice is by means of the $\sqcup$ operator.

The clause for conjunctive terms interprets $\alpha \sqcap \beta$ as the pairwise intersection of the choices described by $\alpha$ and the choices described by $\beta$. The intuition behind this interpretation can be illustrated with an example. Suppose that you are at a restaurant and you are told that “You may order wine or beer and pasta or sushi”. This sentence can be taken to introduce four permitted courses of action: the action of ordering wine and pasta, the action of ordering wine and sushi, the action of ordering beer and pasta, and the action of ordering beer and sushi. The choice set interpretation of conjunctive terms reflects this: simultaneously choose between ordering wine or beer and ordering pasta or sushi, then perform the two chosen actions together.

Following Ju and van Eijck (2016), I use a notion of ‘to do something else’ in the motivation for the choice set interpretation of $\overline{\alpha}$. To do something else than $\alpha$ is to do an action type that is different from the action types introduced by $\alpha$. Say that two action types are different if their intersection is empty. For a given choice set $S$ of action types, the set $\{s \in \mathcal{A} : s \cap t = \emptyset \text{ for all } t \in S\}$ contains all action types that are different from the ones in $S$. However, ‘doing something else than $\alpha$’ should not be understood as describing a choice between every possible action type different from the ones described by $\alpha$. Take a sentence such as “Jane may do something else than going to the movies”. What kind of things are Jane permitted to do? Combining the free choice interpretation of permission with the idea that ‘doing something else than going to the movies’ denotes a choice between every possible action type that is different from going to the movies has counterintuitive consequences. For example, among the action types different from that of going to the movies is the action type of robbing a bank (at least assume so for the sake of the argument). But it seems perfectly consistent that Jane is permitted to do something else than going to the movies, while not being permitted to rob a bank. For a given $\alpha$, there may be many action types that are different from the ones denoted by $\alpha$. In order to arrive at a precise interpretation of the choice set $\llbracket \overline{\alpha} \rrbracket$, the set operator $\text{Alt}$ is applied, picking out action types that are maximal in the sense that they are not properly included in any other action type in the set. It is readily verified that the set $\{s \in \mathcal{A} : s \cap t = \emptyset \text{ for all } t \in S\}$ contains a unique maximal action type which is equal to $H \setminus \bigcup \llbracket \alpha \rrbracket$. By Theorem 2, it follows that $\llbracket \overline{\alpha} \rrbracket = \{V(\overline{\alpha})\}$. Basically, then, negation collapses choice sets into singleton sets.

A formula $\varphi$ being true in a deontic action model $M$ (denoted $M \models \varphi$) is defined as follows, where $\text{LEG}$ is defined based on the particular elements $H$, $G$ in $M$ as in Sect. 3.1:

$$
M \models \top;
M \models \alpha = \beta \iff V(\alpha) = V(\beta);
M \models \alpha \Rightarrow \beta \iff \llbracket \alpha \rrbracket_M = \llbracket \beta \rrbracket_M;
M \models P(\alpha) \iff \llbracket \alpha \rrbracket_M \in \text{LEG};
M \models \neg \varphi \iff M \not\models \varphi;
M \models \varphi \land \psi \iff M \models \varphi \text{ and } M \models \psi.
$$
A formula is said to be satisfiable if it is true in some model. A formula $\varphi$ is valid (denoted $\models \varphi$) if it is true in all models. A set of formulas $\Sigma$ is true in a model (denoted $M \models \Sigma$) if $M \models \varphi$ for all $\varphi \in \Sigma$, and satisfiable if $\Sigma$ is true in some model. A formula $\varphi$ is a logical consequence of a set of formulas $\Sigma$ (denoted $\Sigma \models \varphi$) if, for any $M$, if $M \models \Sigma$, then $M \models \varphi$.

### 3.2.4 Some Properties and Abbreviations

The following properties are easily verified:

\[
\begin{align*}
M \models \alpha \overset{\equiv}{=} \beta & \iff M \models \alpha \equiv \beta, \text{ for } \alpha, \beta \text{ free from occurrences of } \sqcup; \\
M \models \overline{\alpha} \overset{\equiv}{=} \overline{\beta} & \iff M \models \overline{\alpha} \equiv \overline{\beta}; \\
M \models \alpha \overset{\equiv}{=} \beta & \iff M \models \overline{\alpha} \equiv \overline{\beta}.
\end{align*}
\]

In principle, then, the operator $\overset{\equiv}{=}$ could be introduced by definition: $\alpha \overset{\equiv}{=} \beta \overset{\text{def}}{=} \overline{\alpha} \equiv \overline{\beta}$.

It will also be convenient to introduce a couple of abbreviations:

\[
\begin{align*}
\alpha \leq \beta & \overset{\text{def}}{=} \alpha \overset{\equiv}{=} \alpha \sqcap \beta; \\
\alpha \sqsubseteq \beta & \overset{\text{def}}{=} \beta \overset{\equiv}{=} \alpha \sqcup \beta.
\end{align*}
\]

It can be verified that

\[
\begin{align*}
M \models \alpha \leq \beta & \iff V(\alpha) \subseteq V(\beta), \\
M \models \alpha \sqsubseteq \beta & \iff [\alpha]_M \subseteq [\beta]_M.
\end{align*}
\]

The abbreviation $\alpha \leq \beta$ corresponds to the standard partial order induced by a Boolean algebra of $\text{Act}$. A sentence of the form $\alpha \sqsubseteq \beta$ expresses that the choices described by $\alpha$ are included in the choices described by $\beta$, i.e. that the actions included in the choice set $[\alpha]$ are included in the choice set $[\beta]$.

A deontic operator for expressing obligation in accordance with the deontic structure defined in Sect. 3.1 can be introduced as an abbreviation:

\[
\text{O}(\alpha) \overset{\text{def}}{=} \text{P}(\alpha) \land \neg \text{P}(\overline{\alpha}).
\]

It is readily verified that formulas of the form $\text{O}(\alpha)$ have the following satisfiability condition:

\[
M \models \text{O}(\alpha) \iff [\alpha]_M \in \text{REQ},
\]

where $\text{REQ}$ is defined based on the particular elements $H, \ G$ in $M$ as in Sect. 3.1. On the intended reading of the operators, $\alpha$ is obligatory if and only if $\alpha$ is permitted and doing something else than $\alpha$ is not permitted.

A deontic operator for prohibition can be introduced as the negation of a permission. Such a concept says that a choice is prohibited when there is an illegal action type.
among the alternatives. One can also define a stronger concept of prohibition using a combination of external (propositional) negation and negation of terms:

\[
F(\alpha) \overset{\text{def}}{=} \neg P(\bar{\alpha}),
\]

Recall that negation collapses choice sets into singleton sets. Thus, a sentence \(F(\alpha)\) is true if and only if \([\alpha] = \{V(\bar{\alpha})\} = \{V(\alpha)\}\) does not belong to \(\text{LEG}\), i.e. precisely when there are no legal action tokens in \(V(\alpha)\). It is easily verified that

\[
M \vDash F(\alpha) \iff V(\alpha) \subseteq H \setminus G \iff G \subseteq V(\alpha).
\]

In this sense, the stronger prohibition concept says that a choice is prohibited when there is no legal alternative: it is forbidden (in the strong sense) to choose between going by bus and going by train if the action of going by bus and the action of going by train are both illegal.

4 Permission and Obligation

In this section, I will argue that the semantics just outlined deals quite naturally with the problems connected with free choice permissions, and then discuss the defined concept of obligation.

4.1 Permission in DACL

Suppose that Jane may go by bus or by train. Intuitively, it follows that Jane may go by bus, and that Jane may go by train; Jane is permitted to choose one of the options. In everyday discourse, it seems that the permission to do one thing or another is equivalent to both disjuncts being permitted. In the present semantics, the principle of free choice is valid.

**Proposition 1** The following is a validity:

\[
\vDash P(\alpha \sqcup \beta) \iff P(\alpha) \land P(\beta).
\]

**Proof** For the left-to-right direction, let \(M = \langle H, G, V \rangle\) be a deontic action model and suppose that \(M \vDash P(\alpha \sqcup \beta)\), i.e. \([\alpha \sqcup \beta]_M \in \text{LEG}\). By the clause defining the choice sets of disjunctive terms and properties of \(\text{LEG}\), it follows that \([\alpha]_M \in \text{LEG}\) and \([\beta]_M \in \text{LEG}\), which implies that \(M \vDash P(\alpha) \land P(\beta)\). The right-to-left direction follows directly from the fact that \(S, T \in \text{LEG}\) implies that \(S \cup T \in \text{LEG}\).

The deontic action logics based on Boolean algebra (Segerberg 1982; Trypuz and Kulicki 2009, 2015) allow inferring that \(\alpha \sqcap \beta\) is permitted from the permission of \(\alpha\). The reason for this is that these logics take the principle of free choice as an axiom, and allow for substitutions of Boolean algebra identities within the scope of the permission operator. In the semantics in this paper, substitution of Boolean equivalents is in general
not a valid rule of inference. This means that, for example, even though the classically valid equation \( a \sqcup \overline{a} \equiv b \sqcup \overline{b} \) is valid in the semantics, one can find models such that \( M \not\models P(a \sqcup \overline{a}) \iff P(b \sqcup \overline{b}) \). Given the free choice reading this is desirable since one may be permitted to choose between going by train or not going by train, while not being permitted to choose between robbing a bank or not robbing a bank. As another example, take the classically valid equation \( a \equiv (a \sqcap b) \sqcup (a \sqcap \overline{b}) \). Since \( a \) and \( (a \sqcap b) \sqcup (a \sqcap \overline{b}) \) are extensionally equivalent, i.e. denote the same action type, the equation is valid in the semantics. However, the two expressions may describe different choices. Indeed, \( a \) may not describe a genuine choice at all, while \( (a \sqcap b) \sqcup (a \sqcap \overline{b}) \) describes a choice between doing \( a \sqcap b \) and doing \( a \sqcap \overline{b} \). In terms of vegetarian lunches, one may take \( a \) to describe the action of ordering a vegetarian meal, \( a \sqcap b \) to describe the action of ordering a vegetarian meal and pay for it, and \( a \sqcap \overline{b} \) to describe the action of ordering a vegetarian meal and not pay for it.

**Proposition 2** The vegetarian free lunch problem is avoided, that is

\[ \not\models P(\alpha) \rightarrow P(\alpha \sqcap \beta). \]  

**Proof** Let \( M = \langle H, G, V \rangle \) be a deontic action model such that \( H = \{h_1, h_2\}, G = \{h_1\}, V(a_1) = \{h_1\} \) and \( V(a_2) = \{h_2\} \). Since \([a_1]_M = \{\{h_1\}\} \in \text{LEG}\), it holds that \( M \models P(a_1) \). But \([a_1 \sqcap a_2]_M = \emptyset \notin \text{LEG} \), so \( M \not\models P(a_1 \sqcap a_2) \). \( \square \)

Thus, DACL keeps the free choice postulate but avoids the vegetarian free lunch problems. Even though intuitively deeply problematic, one may argue that there is nothing wrong with vegetarian free lunches per se: one just needs to give permission operators for which the schema \( P(\varphi) \rightarrow P(\varphi \land \psi) \) are valid the right kind of informal reading. For example, one can interpret the permission operator in the strong or open sense (henceforth, I will refer to this interpretation as strong permission, and use \( P_s \) to refer to a strong permission operator). According to the strong permission account, \( P_s(\varphi) \) means that every way to perform the action denoted by \( \varphi \) is legal (Anglberger et al. 2014; Broersen 2004; Segerberg 1982; Trypuz and Kulicki 2009, 2015). This account should be contrasted with the weak reading of permission, where an action being weakly permitted means that at least some way of performing it is legal.\(^6\)

The strong permission account seems to offer an explanation of the problem of vegetarian free lunches: if every way to perform the action of ordering a vegetarian meal is legal, then it follows that ordering a vegetarian meal and not paying for it, which is indeed a way to order a vegetarian meal, must be legal as well. Given this interpretation, one can argue that the clash with intuition arises because of the incompleteness of informal normative discourse (Broersen 2004, pp. 163–164). When I permit you to order a vegetarian meal, I usually implicitly assume that further properties of the permitted action must be taken into account (e.g. that ordering a vegetarian meal requires paying for it). What was actually permitted in the first place, then, was the action of ordering a vegetarian meal and paying for it.

---

\(^6\) A different way of characterising the difference between strong and weak permission is to say that an action is strongly permitted if no way of executing it leads to a violation state; a weakly permitted action is such that at least some way of executing it does not lead to a violation state (Broersen 2004; Dignum et al. 1996).
Dignum et al. (1996) offer a formalization of this idea by introducing an operator on action expressions that allows referring to actions performed in isolation, and special action expressions that, apart from naming actions, also refer to contexts in which the actions can be performed. A contextual action expression $\alpha_C$ means a performance of the action named $\alpha$, possibly in combination with actions in the context $C$. The contextual action expressions are used to refer to a restricted set of all possible ways to perform an action. A permission $P_s(\alpha_C)$, then, means that every way to perform the action $\alpha$ in combination with actions in the context $C$ is legal. In the system of Dignum et al. (1996), it is possible to have $P_s(\alpha_C)$ without having $P_s(\alpha)$. The correct way of translating the sentence “It is permitted to order a vegetarian meal” into formal language, they argue, is to translate it into $P_s(\alpha_C)$, where $\alpha$ denotes the action of ordering a vegetarian meal and $C$ is some context specifying actions that are ‘safe’ to perform in combination with the action of ordering a vegetarian meal. In this way, one need not say that every possible way to order a vegetarian meal must be legal, but only those ways which are expressed explicitly using some appropriate context.

Even though I agree that vegetarian free lunches are less problematic under the strong reading of permission, I do not think that this account necessarily is the most natural one or the only account worth investigating when it comes to interpreting free choice permissions. For example, Jane may very well choose freely between driving her Volvo to work and driving her BMW to work. The fact that Jane ought to be sober while driving does not cancel her free choice permission: it seems strange to say that the only free choice Jane is permitted to make is that between driving her Volvo sober and driving her BMW sober. Giordani and Canavotto (2016) make a similar point, arguing that “ordinary choices can be risky: we are ordinarily allowed to choose between alternative actions even if there are ways of performing such actions that lead to a violation of the law.” (2016, p. 89). To capture this property, strong permission is too strong, and weak permission is too weak (cf. van der Meyden 1996, p. 470). Even the contextual permission approach of Dignum et al. (1996) fails to account for permissions of risky choices. The present approach, on the other hand, offers a way to account for free choice permissions without resorting to strong permission. The operator $P$ is a kind of mix between the strong and weak reading of permission: $P(\alpha)$ means that every alternative action in the choice described by $\alpha$ is weakly permitted. Thus, I accept the idea of a free choice permission being a permission in which every alternative action to choose from is legal, but I reject the idea that an action is legal only if every way to perform it is acceptable. Free choice permission is permission applied to choices, rather than directly to actions.

That being said, the present framework provides a straightforward way to combine two kinds of deontic operators: one referring to action types and the other referring to choice sets. For example, the $P$ and $O$ operators refer to choices, while the $F$ operator is technically equivalent to an operator applying directly to action types. It is not possible to introduce strong permission in DACL as a syntactic abbreviation, but it is straightforward to introduce it as a primitive notion. Adding strong permission to DACL results in an extension of Segerberg’s B.C.D. system (1982).

Turning back to the free choice principle itself, it can be noted that the right-to-left direction of it is also valid in any deontic logic where the permission operator is analyzed as a normal modal diamond, that is, as a weak permission operator. However,
such logics also validate a stronger principle: \( P(\varphi) \rightarrow P(\varphi \lor \psi) \). This can be seen as a permission variant of Ross’ paradox, admitting the following counterintuitive instance: “If you may post the letter, then you may post the letter or burn it”. This time, the culprit is closure of permission under logical consequence: from the tautology \( \varphi \rightarrow \varphi \lor \psi \), apply closure of permission under logical consequence to obtain \( P(\varphi) \rightarrow P(\varphi \lor \psi) \).

In the present framework, permission is not even closed under logical equivalence (valid equations of Boolean algebra), and \( P(\alpha) \rightarrow P(\alpha \sqcup \beta) \) is not valid.

**Proposition 3** The following properties hold:

\[
\vdash P(\alpha \cap \beta) \rightarrow P(\alpha) \land P(\beta) \tag{8}
\]

\[
\vdash P(\alpha \cap (\beta \sqcup \delta)) \rightarrow P(\alpha \cap \beta) \land P(\alpha \cap \delta) \tag{9}
\]

\[
\not\vdash P(\alpha) \rightarrow P(\alpha \sqcup \beta) \tag{10}
\]

\[
\not\vdash P(\alpha) \lor P(\overline{\alpha}) \tag{11}
\]

**Proof** For a proof of (8), let \( M = \langle H, G, V \rangle \) be a model and assume that \( M \models P(\alpha \cap \beta) \), i.e. \( [\alpha \cap \beta]_M \in \text{LEG} \). It follows that \( t \cap G \neq \emptyset \) for each \( t \in [\alpha]_M = \{ s \cap s' : s \in [\alpha]_M \text{ and } s' \in [\beta]_M \} \), which implies that \( s \cap G \neq \emptyset \) and \( s' \cap G \neq \emptyset \) for every \( s \in [\alpha]_M \text{ and } s' \in [\beta]_M \). This, in turn, implies that \( [\alpha]_M \in \text{LEG} \) and \( [\beta]_M \in \text{LEG} \), i.e. \( M \models P(\alpha) \land P(\beta) \). For a proof of (9), let \( M = \langle H, G, V \rangle \) be a model such that \( M \models P(\alpha \cap (\beta \sqcup \delta)) \), i.e. \( [\alpha \cap (\beta \sqcup \delta)]_M \in \text{LEG} \). It holds that

\[
[\alpha \cap (\beta \sqcup \delta)]_M = \{ s \cap t : s \in [\alpha]_M \text{ and } t \in [\beta \sqcup \delta]_M \}
\]

\[
= \{ s \cap t : s \in [\alpha]_M \text{ and } t \in [\beta]_M \cup [\delta]_M \}
\]

\[
= \{ s \cap t : s \in [\alpha]_M \text{ and } t \in [\beta]_M \cup [\delta]_M \}
\]

By properties of \( \text{LEG} \), it follows that \( [\alpha \cap \beta]_M \in \text{LEG} \) and \( [\alpha \cap \delta]_M \in \text{LEG} \), i.e. \( M \models P(\alpha \cap \beta) \land P(\alpha \cap \delta) \). For (11), consider a model \( M = \langle H, G, V \rangle \) such that \( H = \{ h_1, h_2 \}, G = \{ h_1 \}, V(a_1) = \{ h_1 \} \) and \( V(a_2) = \{ h_2 \} \). In this model, \( M \models \neg P(a_1 \sqcup a_2) \) and \( M \models \neg P(a_1 \sqcup a_2). \)

These properties nicely illustrate the double nature of permission in DACL: (8) says that permitted conjunctions can be weakened, which is a characteristic property of weak permission. Another characteristic property of weak permission is closure: for every action, either it or its negation is permitted. This property fails for the operator \( P \) as shown by (11).

I will close this discussion with some remarks regarding (9). In everyday discourse, it seems that free choice phenomena are preserved when ‘or’ is embedded under ‘and’. Suppose that \( a_1 \) means ‘sit on the sofa’, \( a_2 \) means ‘read a book’, and \( a_3 \) means ‘watch TV’. Then, the term \( a_1 \cap (a_2 \sqcup a_3) \) means ‘sit on the sofa and read a book or watch TV’. Now, consider the following informal inference:

Jane may sit on the sofa and read a book or watch TV. Therefore, Jane may sit on the sofa and read a book and Jane may sit on the sofa and watch TV.
The inference seems intuitively valid, and illustrates that free choice effects are not canceled when ‘or’ is embedded under ‘and’. This property is reflected in the validity (9) of Proposition 3.

### 4.2 Obligation in DACL

Regarding the interaction of permission and obligation, it can be noted that the O and P operators satisfy the following principle, which follows directly from the fact that \( \text{REQ} \subseteq \text{LEG} \).

**Proposition 4** ‘Ought implies may’, that is,

\[
\models O(\alpha) \rightarrow P(\alpha). \tag{12}
\]

Note that a consequence of the above principle is that \( O(\alpha \sqcup \beta) \rightarrow P(\alpha) \land P(\beta) \) is valid. This is in accordance with the justification for the \( \text{REQ} \) set given in Sect. 3.1.

**Proposition 5** Ross’ paradox is avoided, that is,

\[
\not\models O(\alpha) \rightarrow O(\alpha \sqcup \beta). \tag{13}
\]

**Proof** Let \( M = \langle H, G, V \rangle \) be a deontic action model such that \( H = \{h_1, h_2\}, G = \{h_1\}, V(a_1) = \{h_1\} \) and \( V(a_2) = \{h_2\} \). Since \( \llbracket a_1 \rrbracket_M = \{h_1\} \in \text{REQ} \), it holds that \( M \models O(a_1) \). However, \( \llbracket a_1 \sqcup a_2 \rrbracket_M = \{h_1, h_2\} \notin \text{REQ} \), so \( M \not\models O(a_1 \sqcup a_2) \). □

The semantics also validates the well-known principles of obligation aggregation, obligation weakening, the impossibility of conflicting obligations and a principle saying that choosing to do an impossible action is never obligatory.

**Proposition 6** The following validities hold:

\[
\models O(\alpha) \land O(\beta) \rightarrow O(\alpha \cap \beta) \tag{14}
\]

\[
\models O(\alpha \cap \beta) \rightarrow O(\alpha) \land O(\beta) \tag{15}
\]

\[
\models \neg (O(\alpha) \land O(\overline{\alpha})) \tag{16}
\]

\[
\models \neg O(\emptyset). \tag{17}
\]

**Proof** The proof of (14) is left to the reader. Here is a proof of (15). Let \( M = \langle H, G, V \rangle \) be a model and assume that \( M \models O(\alpha \cap \beta) \), i.e. \( \llbracket \alpha \cap \beta \rrbracket_M \in \text{REQ} \). By (12) of Proposition 4, it holds that \( M \models P(\alpha \cap \beta) \), so by (8) of Proposition 3 \( M \models P(\alpha) \), i.e. \( \llbracket \alpha \rrbracket_M \in \text{LEG} \). By the definition of \( \text{REQ} \) and properties of choice sets, it holds that \( G \subseteq \bigcup \llbracket \alpha \cap \beta \rrbracket_M \subseteq \bigcup \llbracket \alpha \rrbracket_M \). Taken together, this implies that \( \llbracket \alpha \rrbracket_M \in \text{REQ} \), i.e. \( M \models O(\alpha) \). (16) follows from (18) of Proposition 7 below. (17) follows from the syntactic definition of \( O \) and the fact that \( \models \neg P(\emptyset) \), the latter being valid because \( \{\emptyset\} \notin \text{LEG} \). □

That \( O(\alpha) \) and \( O(\overline{\alpha}) \) are not jointly satisfiable guarantees what is known as deontic consistency: one and the same choice can never be both obligatory and prohibited. In
fact, Theorem 2 together with the definition of $\text{REQ}$ shows that conflicting obligations require accepting models where $G = \emptyset$, i.e. models where there are no legal action tokens available.

Regarding the interaction of obligation and prohibition, the following validities are shown to hold.

**Proposition 7** The following validities hold:

\[
\models O(\alpha) \rightarrow F(\alpha) \quad (18)
\]
\[
\models F(\alpha) \rightarrow O(\alpha) \quad (19)
\]
\[
\models O(\alpha) \land F(\beta) \rightarrow O(\alpha \land \beta). \quad (20)
\]

*Proof* (18) is immediate from the syntactic definition of $O$ and $F$. For (19), let $M = \langle H, G, V \rangle$ be a deontic action model and assume that $M \models F(\alpha)$. Then $G \subseteq V(\alpha)$. Since $[\alpha]_M = \{V(\beta)\}$, it immediately follows that $[\alpha]_M \in \text{REQ}$. (20) is a consequence of (19) and (14). $\square$

Trypuz and Kulicki (2015, p. 1254) take the principles of obligation economy (18) and obligation trimming (20) together with (17) as axioms characterizing a minimal concept of obligation. The $O$ operator is stronger than this; for example, obligations can be weakened according to (15). In the basic system of Trypuz and Kulicki (2015) where obligation is axiomatized by (17), (18), and (20), it is not possible to derive (15).

Castro and Maibaum (2009) introduce an obligation operator in terms of strong permission and prohibition, which in the present framework corresponds to the abbreviation $O_P(\alpha) \overset{\text{def}}{=} P_s(\alpha) \land F(\alpha)$ (where $P_s$ is strong permission with truth conditions $M \models P_s(\alpha)$ iff $V(\alpha) \subseteq G$). A consequence of this definition is that there is only one obligatory action type: $O_P(\alpha) \land O_P(\beta) \rightarrow \alpha \equiv \beta$ is valid. As observed by Trypuz and Kulicki (2015, p. 1253), this property makes many of the intuitively desirable properties of obligation trivially valid. The $O$ operator of DACL is introduced using the $P$ operator instead of $P_s$, which guarantees that there may be several different obligations. Consider, for example, a model $M = \langle H, G, V \rangle$ where $G = \{h_1, h_2\}$ and $V(a_1) = \{h_1\}$, $V(a_2) = \{h_2, h_3\}$ and $V(a_3) = \{h_1, h_2\}$. In this model, $M \models O(a_1 \sqcup a_2)$ and $M \models O(a_3)$, but $M \not\models a_1 \sqcup a_2 \equiv a_3$ and $M \not\models a_1 \sqcup a_2 \equiv a_3$.

## 5 Axiomatization

In this section, the logic DACL is presented axiomatically. The axiom system is shown to be sound and complete with respect to the semantics defined in previous sections. For the rest of what follows, assume a fixed set $\text{Act}_0$ of generators.

DACL is axiomatized by the following axioms and rules.

- A complete set of axioms for Propositional Logic (PL).
- A complete set of axioms for Boolean algebra in terms of $\equiv$ [e.g. axioms (A1)–(A5), with $=$ replaced by $\equiv$].
Supercovers Semantics for Deontic Action Logic

– Axioms for \( \equiv \):

\[
\begin{align*}
\alpha \sqcup \alpha & \equiv \alpha & \text{(A6)} \\
\alpha \sqcup \beta & \equiv \beta \sqcup \alpha, \alpha \sqcap \beta & \equiv \beta \sqcap \alpha & \text{(A7)} \\
(\alpha \sqcup \beta) \sqcup \delta & \equiv \alpha \sqcup (\beta \sqcup \delta), (\alpha \sqcap \beta) \sqcap \delta & \equiv \alpha \sqcap (\beta \sqcap \delta) & \text{(A8)} \\
\alpha \sqcap (\beta \sqcup \delta) & \equiv (\alpha \sqcap \beta) \sqcup (\alpha \sqcap \delta) & \text{(A9)} \\
\overline{\alpha \sqcup \beta} & \equiv \overline{\alpha} \sqcap \overline{\beta} & \text{(A10)} \\
\overline{a_i} & \equiv a_i \text{ for all } a_i \in \mathit{Act}_0 & \text{(A11)} \\
\overline{1} & \equiv 1 & \text{(A12)} \\
\overline{0} & \equiv 0 & \text{(A13)} \\
\overline{\alpha} \sqsubseteq \beta \sqcup \delta & \rightarrow \overline{\alpha} \sqsubseteq \beta \lor \overline{\alpha} \sqsubseteq \delta & \text{(A14)} \\
\alpha \sqsubseteq \overline{\beta} & \rightarrow \alpha \equiv \overline{\beta} & \text{(A15)} \\
\end{align*}
\]

– Identity axioms:

\[
\begin{align*}
\alpha \doteq \alpha & \quad \text{(A16)} \\
\alpha \doteq \alpha & \quad \text{(A17)} \\
\alpha \doteq \beta \rightarrow \theta(\alpha) \doteq \theta(\beta) & \quad \text{(A18)} \\
\alpha \doteq \beta \rightarrow \theta(\alpha) \doteq \theta(\beta) & \quad \text{(A19)} \\
\end{align*}
\]

where \( \theta(\alpha) \) is some term with a number of occurrences of \( \alpha \), and \( \theta(\beta) \) is the term obtained from \( \theta(\alpha) \) by replacing some or all occurrences of \( \alpha \) with \( \beta \).

– Bridge axioms:

\[
\begin{align*}
\alpha \doteq \beta & \rightarrow \alpha \doteq \beta & \text{(A20)} \\
\alpha \doteq \beta & \rightarrow \overline{\alpha} \doteq \overline{\beta} & \text{(A21)}
\end{align*}
\]

– Deontic axioms:

\[
\begin{align*}
P(\overline{\alpha \sqcup \beta}) & \leftrightarrow P(\overline{\alpha}) \lor P(\overline{\beta}) & \text{(A22)} \\
P(\alpha \sqcup \beta) & \leftrightarrow P(\alpha) \land P(\beta) & \text{(A23)} \\
P(1) & \quad \text{(A24)} \\
\neg P(0) & \quad \text{(A25)} \\
P(\alpha) & \rightarrow P(\overline{\alpha}) & \text{(A26)} \\
\alpha \doteq \beta & \rightarrow (P(\alpha) \leftrightarrow P(\beta)) & \text{(A27)}
\end{align*}
\]

– The sole rule of inference is \textit{modus ponens} (\( \vdash \varphi \) means that \( \varphi \) is a theorem of the axiom system):

\( (\text{MP}) \) If \( \vdash \varphi \) and \( \vdash \varphi \rightarrow \psi \), then \( \vdash \psi \).

\( \copyright \) Springer
$\Sigma \vdash \varphi$ means that $\varphi$ is derivable (in the axiom system) from $\Sigma$; this notion is defined as usual.

### 5.1 Some Theorems

The following theorems provide a syntactic characterization of the concept of prohibition, as well as the interaction between prohibitions and permissions.

**Lemma 1** The following are theorems of DACL.

\[
\begin{align*}
1. & \quad \vdash F(\alpha \sqcup \beta) \leftrightarrow F(\alpha) \land F(\beta) \\
2. & \quad \vdash F(0) \\
3. & \quad \vdash \neg F(1) \\
4. & \quad \vdash P(\alpha) \rightarrow \neg F(\alpha) \\
5. & \quad \vdash \neg P(\beta) \rightarrow F(\beta) \\
6. & \quad \vdash (\alpha \sqsubseteq \beta) \leftrightarrow (\neg F(\alpha) \leftrightarrow F(\beta)) \\
7. & \quad \vdash F(\beta) \land \alpha \leq \beta \rightarrow F(\alpha) 
\end{align*}
\]

*Proof* These theorems are straightforward consequences of the deontic axioms and the definition of $F$. Here is a proof of Theorem (27).

1. $F(\beta) \land \alpha \leq \beta$
2. $F(\beta)$ from 1 by propositional logic
3. $\alpha \leq \beta$ from 1 by propositional logic
4. $\alpha \sqsubseteq \beta \sqcap \beta$ from 3 by the definition of $\leq$
5. $\beta \sqsubseteq \alpha \sqcap \beta$ from 4 by Boolean algebra properties
6. $F(\alpha \sqcup \beta)$ from 2 by theorem (26)
7. $F(\beta)$ from 6 by theorem (21)

The following theorems give a flavor of the logical behavior of choices.

**Lemma 2** The following are theorems of DACL.

\[
\begin{align*}
1. & \quad \vdash \alpha \subseteq \alpha \\
2. & \quad \vdash \alpha \subseteq \alpha \sqcup \beta \\
3. & \quad \vdash \alpha \subseteq \beta \rightarrow \alpha \subseteq \beta \sqcup \delta \\
4. & \quad \vdash \alpha \subseteq \beta \sqcap \delta \sqsubseteq \delta \rightarrow \alpha \subseteq \delta \\
5. & \quad \vdash \alpha \sqsubseteq \beta \sqcap \delta \sqsubseteq \delta \rightarrow \alpha \sqsubseteq \beta \\
6. & \quad \vdash \alpha \subseteq \beta \sqcup \delta \sqsubseteq \delta \rightarrow \alpha \sqsubseteq \beta \\
7. & \quad \vdash \alpha \sqsubseteq \beta \rightarrow \alpha \subseteq \beta \sqcup \beta \sqsubseteq \alpha \\
8. & \quad \vdash \alpha \sqsubseteq \beta \leftrightarrow \alpha \sqsubseteq \beta
\end{align*}
\]
The proofs of these are straightforward. Below I provide proofs of theorem (29) and (33).

**Theorem (29):**

1. \( \alpha \sqcup \beta \iff \alpha \sqcup \beta \) \hspace{1cm} \text{axiom (A17)}
2. \( \alpha \iff \alpha \sqcup \alpha \) \hspace{1cm} \text{axiom (A6)}
3. \( \alpha \sqcup \beta \iff (\alpha \sqcup \alpha) \sqcup \beta \) \hspace{0.5cm} \text{from 1, 2 by axiom (A19)}
4. \( \alpha \sqcup \beta \iff \alpha \sqcup (\alpha \sqcup \beta) \) \hspace{0.5cm} \text{from 3 by axiom (A8)}
5. \( \alpha \sqsubseteq \alpha \sqcup \beta \) \hspace{1cm} \text{by the definition of } \sqsubseteq

**Theorem (33):**

1. \( \alpha \sqsubseteq \beta \land \beta \sqsubseteq \alpha \) \hspace{1cm} \text{from 1 by propositional logic and the definition of } \sqsubseteq
2. \( \beta \iff \alpha \sqcup \beta \) \hspace{1cm} \text{from 1 by propositional logic and the definition of } \sqsubseteq
3. \( \alpha \iff \beta \sqcup \alpha \) \hspace{1cm} \text{from 1 by propositional logic and the definition of } \sqsubseteq
4. \( \alpha \iff \alpha \sqcup \beta \) \hspace{1cm} \text{from 3 by axioms (A7) and (A19)}
5. \( \alpha \iff \beta \) \hspace{1cm} \text{from 2, 4 by axiom (A19)}

\( \Box \)

### 5.2 Soundness

**Theorem 3** (Soundness theorem) The axiom system is sound with respect to the semantics, that is, if \( \Sigma \vdash \varphi \), then \( \Sigma \models \varphi \).

**Proof** It must be shown that each axiom is valid, and that the rules of inference preserve validity. I prove some representative cases below.

Axiom (A14). Let \( M = \langle H, G, V \rangle \) be a deontic action model such that \( M \models \alpha \sqsubseteq \beta \sqcup \delta \). This means that \( \llbracket \alpha \rrbracket_M = \{V(\alpha)\} \subseteq \llbracket \beta \sqcup \delta \rrbracket_M = \llbracket \beta \rrbracket_M \cup \llbracket \delta \rrbracket_M \). Hence, \( \llbracket \alpha \rrbracket_M \subseteq \llbracket \beta \rrbracket_M \) or \( \llbracket \alpha \rrbracket_M \subseteq \llbracket \delta \rrbracket_M \). It follows that \( M \models \alpha \sqsubseteq \beta \lor \alpha \sqsubseteq \delta \).

Axiom (A15). Assume that \( M = \langle H, G, V \rangle \) is a deontic action model where \( M \models \alpha \sqsubseteq \beta \), i.e. \( \llbracket \alpha \rrbracket_M \subseteq \llbracket \beta \rrbracket_M = \{V(\beta)\} \). This is only possible if \( \llbracket \alpha \rrbracket_M = \emptyset \) or every element of \( \llbracket \alpha \rrbracket_M \) is equal to \( V(\beta) \). Since the first case is ruled out by definition, it follows that \( \llbracket \alpha \rrbracket_M = \llbracket \beta \rrbracket_M \), i.e. \( M \models \alpha \iff \beta \).

Axiom (A22). Assume that \( M = \langle H, G, V \rangle \) is a deontic action model where \( M \models P(\alpha \sqcup \beta) \), i.e. \( \llbracket \alpha \sqcup \beta \rrbracket_M \in \text{LEG} \). Since \( \llbracket \alpha \sqcup \beta \rrbracket_M = \{V(\alpha \sqcup \beta)\} = \{V(\alpha) \cup V(\beta)\} \), it holds that there is some \( h \in G \) such that \( h \in V(\alpha \sqcup \beta) = V(\alpha) \cup V(\beta) \). It follows that \( h \in V(\alpha) = V(\alpha) \) or \( h \in V(\beta) = V(\beta) \). This, in turn, shows that \( \llbracket \alpha \rrbracket_M \in \text{LEG} \) or \( \llbracket \beta \rrbracket_M \in \text{LEG} \), i.e. \( M \models P(\alpha) \lor P(\beta) \). The other direction is similar. \( \Box \)

### 5.3 Disjunctive Negative Translation

The completeness proof utilizes the fact that every term has a certain normal form. First, I will define the *disjunctive negative translation* of a term \( \alpha \), denoted \( \text{DNT}(\alpha) \). This translation is similar to the one defined by Ciardelli and Roelofsen (2011, p. 69).
in the context of inquisitive logic. If \( X \) is a set of terms, then \( \bigcup X \) is a term that is a disjunction of the terms in \( X \).

**Definition 1** (*Disjunctive negative translation*)

- \( \text{DNT}(a_i) = \overline{a_i} \), for \( a_i \in \text{Act}_0 \).
- \( \text{DNT}(1) = \overline{1} \).
- \( \text{DNT}(0) = \overline{0} \).
- \( \text{DNT}(\alpha \sqcup \beta) = \text{DNT}(\alpha) \sqcup \text{DNT}(\beta) \).
- \( \text{DNT}(\alpha \sqcap \beta) = \bigsqcup \{ \overline{\alpha_i} \sqcup \overline{\beta_j} : 1 \leq i \leq k, 1 \leq j \leq l \} \), where
  - \( \text{DNT}(\alpha) = \overline{\alpha_1} \sqcup \cdots \sqcup \overline{\alpha_k} \).
  - \( \text{DNT}(\beta) = \overline{\beta_1} \sqcup \cdots \sqcup \overline{\beta_l} \).
- \( \text{DNT}(\overline{\alpha}) = \text{DNT}(\alpha) \).

Say that two terms \( \alpha, \beta \in \text{Act} \) are *choice-identical* if \( [\alpha]_M = [\beta]_M \) is true for any model \( M \).

**Theorem 4** For any \( \alpha \in \text{Act} \), \( \alpha \) is choice-identical to \( \text{DNT}(\alpha) \).

**Proof** The proof is by induction over \( \alpha \). Let \( \langle H, G, V \rangle \) be an arbitrary model. For the induction base, note that for \( \alpha = a_i, \alpha = 1, \) and \( \alpha = 0 \) it holds by definition that \( [\alpha]_M = \{ V(\alpha) \} = \{ V(\overline{\alpha}) \} = [\text{DNT}(\alpha)]_M \). For the induction step, assume that the statement is true for terms \( \alpha', \alpha'' \). There are three cases to consider.

- \( \alpha = \alpha' \sqcup \alpha'' \). By the induction hypothesis, \( [\alpha']_M = [\text{DNT}(\alpha')]_M \) and \( [\alpha'']_M = [\text{DNT}(\alpha'')]_M \). It follows by the definition of choice sets that \( [\alpha' \sqcup \alpha'']_M = [\alpha']_M \sqcup [\alpha'']_M = \text{DNT}(\alpha' \sqcup \alpha'') \).
- \( \alpha = \alpha' \sqcap \alpha'' \). It holds that \( [\alpha' \sqcap \alpha'']_M = \{ V(\gamma) \cap V(\delta) : \gamma \in [\alpha']_M, \delta \in [\alpha'']_M \} \). Assuming that \( \text{DNT}(\alpha') = \overline{\alpha_1} \sqcup \cdots \sqcup \overline{\alpha_k} \) and \( \text{DNT}(\alpha'') = \overline{\alpha_1'} \sqcup \cdots \sqcup \overline{\alpha_l} \), it follows by the induction hypothesis that \( [\alpha' \sqcap \alpha'']_M = \{ V(\overline{\alpha_i}) \cap V(\overline{\alpha_j}) : 1 \leq i \leq k, 1 \leq j \leq l \} = [\text{DNT}(\alpha' \sqcap \alpha'')]_M \).
- \( \alpha = \overline{\alpha'} \). It holds that \( [\overline{\alpha'}]_M = \{ V(\overline{\alpha'}) \} \) by definition. By the induction hypothesis, \( [\overline{\alpha'}]_M = [\text{DNT}(\overline{\alpha'})]_M \), so by Theorem 2, \( V(\alpha') = V(\text{DNT}(\alpha')) \). It follows that \( [\overline{\alpha'}]_M = \{ V(\overline{\alpha'}) \} = [H \setminus V(\alpha')] = [H \setminus V(\text{DNT}(\alpha'))] = [V(\text{DNT}(\alpha'))] = [\text{DNT}(\overline{\alpha'})]_M \).

The mapping \( \text{DNT} \) always results in a term which is a disjunction of negations. Using Theorem 4, every choice set can be expressed in a standardized way. For any term \( \alpha \in \text{Act} \) and any model \( \langle H, G, V \rangle \), it holds that \( [\alpha]_M = [\text{DNT}(\alpha)]_M = [\overline{\alpha_1}]_M \sqcup \cdots \sqcup [\overline{\alpha_k}]_M = \{ V(\overline{\alpha_i}) : 1 \leq i \leq k \} \), where \( \text{DNT}(\alpha) = \overline{\alpha_1} \sqcup \cdots \sqcup \overline{\alpha_k} \).

Two terms \( \alpha \) and \( \beta \) are said to be *provably choice-identical*, denoted \( \vdash \alpha = \beta \), if \( \vdash \alpha \models \beta \). It is shown that any term is provably choice-identical to its disjunctive negative translation.

**Theorem 5** For any \( \alpha \in \text{Act} \), \( \alpha \) is provably choice-identical to \( \text{DNT}(\alpha) \).
Proof The proof is by induction over $\alpha$. For the induction base, note that for $\alpha = a_i$, $\alpha = 1$, and $\alpha = 0$, axioms (A11), (A12), and (A13) give that $\alpha \vdash \text{DNT}(\alpha)$. For the induction step, assume that the statement holds true for terms $\alpha', \alpha''$. There are three cases to consider.

- $\alpha = \alpha' \cup \alpha''$. By the induction hypothesis, $\alpha' \vdash \text{DNT}(\alpha')$ and $\alpha'' \vdash \text{DNT}(\alpha'')$. Now, starting from $\alpha' \cup \alpha'' \vdash \alpha' \cup \alpha''$, axiom (A19) implies that $\alpha' \cup \alpha'' \vdash \text{DNT}(\alpha') \cup \text{DNT}(\alpha'')$; since $\text{DNT}(\alpha') \cup \text{DNT}(\alpha'') = \text{DNT}(\alpha' \cup \alpha'')$, it holds that $\alpha' \cup \alpha'' \vdash \text{DNT}(\alpha' \cup \alpha'')$.

- $\alpha = \alpha' \cap \alpha''$. By the induction hypothesis, $\alpha' \vdash \text{DNT}(\alpha')$ and $\alpha'' \vdash \text{DNT}(\alpha'')$. Starting from $\alpha' \cap \alpha'' \vdash \alpha' \cap \alpha''$, it follows by axiom (A19) that $\alpha' \cap \alpha'' \vdash \text{DNT}(\alpha') \cap \text{DNT}(\alpha'')$. Assuming that $\text{DNT}(\alpha') = \overline{\alpha'_1} \cup \cdots \cup \overline{\alpha'_k}$ and $\text{DNT}(\alpha'') = \overline{\alpha''_1} \cup \cdots \cup \overline{\alpha''_l}$, axiom (A9) implies that $\alpha' \cap \alpha'' \vdash \bigwedge \{\overline{\alpha'_i \cap \alpha''_j} : 1 \leq i \leq k, 1 \leq j \leq l\}$. By axiom (A10), $\overline{\alpha'_i \cap \alpha''_j} \vdash \overline{\alpha'_i} \cup \overline{\alpha''_j}$ for $1 \leq i \leq k, 1 \leq j \leq l$. By axiom (A19), it follows that $\alpha' \cap \alpha'' \vdash \text{DNT}(\alpha' \cap \alpha'')$.

- $\alpha = \overline{\alpha'}$. By the induction hypothesis, $\alpha' \vdash \text{DNT}(\alpha')$. From $\text{DNT}(\alpha') \vdash \overline{\alpha'}$, axiom (A19) implies that $\alpha' \vdash \text{DNT}(\alpha')$, i.e. $\overline{\alpha'} \vdash \text{DNT}(\overline{\alpha'})$.

\[\Box\]

5.4 Completeness

The completeness proof follows the same structure as the corresponding proofs given by Castro and Maibaum (2009) and Trypuz and Kulicki (2009). Completeness is proved by proving the equivalent result that each consistent set of formulas has a model.

Definition 2 (Canonical model) Let $\Phi$ be a maximally consistent set of formulas with respect to the axiomatization of DACL, and let $[\alpha]_{\perp}$ be the equivalence class of $\alpha$ for $\alpha \in \text{Act}$, i.e. $[\alpha]_{\perp} = \{\beta : \alpha \equiv \beta \in \Phi\}$. A canonical model $M^\Phi = \langle H^\Phi, G^\Phi, V^\Phi \rangle$ has the following form:

- $H^\Phi = \{[\gamma]_{\perp} : \gamma$ is an atom of Act$\}$;
- $V^\Phi(a_i) = \{[\gamma]_{\perp} \in H^\Phi : \gamma \leq a_i \in \Phi\}$;
- $G^\Phi = H^\Phi \cup \{V^\Phi(\alpha) : F(\alpha) \in \Phi\}$.

The following two lemmas are proved along the same lines as done by Castro and Maibaum (2009, p. 448) (the proofs are stated here for completeness of presentation).

Lemma 3 For any atom $\gamma$ of Act, $\gamma \leq \alpha \in \Phi$ if and only if $[\gamma]_{\perp} \in V^\Phi(\alpha)$.

Proof The proof is by induction over $\alpha$. First, for $\alpha = a_i$, the claim is true by definition of $V^\Phi$. For $\alpha = 0$, the claim holds since $V^\Phi(0) = \emptyset$ and $\gamma \leq 0 \notin \Phi$ for every atom $\gamma$. For the induction step, there are three cases to consider.

- $\alpha = \alpha' \cup \alpha''$. Suppose that $[\gamma]_{\perp} \in V^\Phi(\alpha' \cup \alpha'') = V^\Phi(\alpha') \cup V^\Phi(\alpha'')$. By the induction hypothesis, $\gamma \leq \alpha' \in \Phi$ or $\gamma \leq \alpha'' \in \Phi$. By properties of Boolean
algebra, it follows that \( \gamma \leq \alpha' \cup \alpha'' \in \Phi \). Suppose that \( \gamma \leq \alpha' \cup \alpha'' \in \Phi \). By properties of Boolean algebra, it follows that \( \gamma \leq \alpha' \in \Phi \) or \( \gamma \leq \alpha'' \in \Phi \).

By induction hypothesis, it holds that \([\gamma]_\bot \in V^\Phi(\alpha')\) or \([\gamma]_\bot \in V^\Phi(\alpha'')\), i.e. \([\gamma]_\bot \in V^\Phi(\alpha') \cup V^\Phi(\alpha'') = V^\Phi(\alpha' \cup \alpha'')\).

- \( \alpha = \alpha' \cup \alpha'' \). Suppose that \([\gamma]_\bot \in V^\Phi(\alpha')\). Then \([\gamma]_\bot \notin V^\Phi(\alpha'')\), so by the induction hypothesis, \( \gamma \leq \alpha' \notin \Phi \). Since \( \gamma \) is an atom, properties of Boolean algebra imply that \( \gamma \leq \alpha' \notin \Phi \). Suppose that \( \gamma \leq \alpha' \notin \Phi \). By properties of Boolean algebra, it follows that \( \gamma \leq \alpha' \notin \Phi \), so by the induction hypothesis, \([\gamma]_\bot \notin V^\Phi(\alpha')\), i.e. \([\gamma]_\bot \notin V^\Phi(\alpha')\).

**Lemma 4** \( \alpha \models \beta \in \Phi \) if and only if \( V^\Phi(\alpha) = V^\Phi(\beta) \).

**Proof** (\( \Rightarrow \)) Assume that \( \alpha \models \beta \in \Phi \). It follows that \( \alpha \leq \beta \in \Phi \). This, in turn, means that for every atom \( \gamma \), if \( \gamma \leq \alpha \in \Phi \), then \( \gamma \leq \beta \in \Phi \). By Lemma 3, it follows that \( V^\Phi(\alpha) \subseteq V^\Phi(\beta) \). If \( V^\Phi(\alpha) = V^\Phi(\beta) \), then \( V^\Phi(\alpha) = V^\Phi(\beta) \). Hence, \( V^\Phi(\alpha) = V^\Phi(\beta) \).

(\( \Leftarrow \)) Assume that \( V^\Phi(\alpha) = V^\Phi(\beta) \). If \( V^\Phi(\alpha) = V^\Phi(\beta) = \emptyset \), then there is no atom \( \gamma \) such that \( \gamma \leq \alpha \in \Phi \) or \( \gamma \leq \beta \in \Phi \). This means that \( \alpha \models 0 \in \Phi \) and \( \beta \models 0 \in \Phi \), from which it follows that \( \alpha \models \beta \in \Phi \). For \( V^\Phi(\alpha) = V^\Phi(\beta) \), let \( \gamma \) be the sum of all atoms \( \gamma \) for which \([\gamma]_\bot \in V^\Phi(\alpha) = V^\Phi(\beta) \). Then, by Lemma 3 and properties of Boolean algebra, \( \gamma \models \alpha \in \Phi \) and \( \gamma \models \beta \in \Phi \); hence, \( \alpha \models \beta \in \Phi \).

The two lemmas above show that the interpretation function \( V^\Phi \) has the right behavior, and so:

**Lemma 5** \( M^\Phi = \langle H^\Phi, G^\Phi, V^\Phi \rangle \) is a deontic action model.

**Proof** First, note that \( G^\Phi \subseteq H^\Phi \). It is shown that \( G^\Phi \) is non-empty. Suppose that there are \( n \) atoms of Act. Suppose, for a proof by contradiction, that for every atom \( \gamma_i \), \( 1 \leq i \leq n \), it holds that \([\gamma_i]_\bot \in \bigcup\{V^\Phi(\alpha): F(\alpha) = \Phi\} \). This implies that \( F(\gamma_i) \in \Phi \), \( 1 \leq i \leq n \) by (27). By (21), it follows that \( F(\gamma_1 \cup \cdots \cup \gamma_n) \in \Phi \). But \( \gamma_1 \cup \cdots \cup \gamma_n \models 1 \in \Phi \), so \( F(1) \in \Phi \) by (26). This is, however, a contradiction since it holds that \( \neg F(1) \in \Phi \) by (23).

Using these results, the completeness proofs by Castro and Maibaum (2009) and Trypuz and Kulicki (2009) can be extended to cover identity of choice sets and deontic operators. First, I will state a number of useful lemmas.

**Lemma 6** \( \overline{\alpha} \models \overline{\beta} \in \Phi \) if and only if \([\overline{\alpha}]_{M^\Phi} = [\overline{\beta}]_{M^\Phi} \).

**Proof** (\( \Rightarrow \)) Assume that \( \overline{\alpha} \models \overline{\beta} \in \Phi \). By (35), \( \overline{\alpha} \models \overline{\beta} \in \Phi \) holds. By Lemma 4, it holds that \( V^\Phi(\overline{\alpha}) = V^\Phi(\overline{\beta}) \), and so \([\overline{\alpha}]_{M^\Phi} = [\overline{\beta}]_{M^\Phi} \) by the definition of choice sets.

(\( \Leftarrow \)) Assume that \([\overline{\alpha}]_{M^\Phi} = [\overline{\beta}]_{M^\Phi} \). It follows that \( V^\Phi(\overline{\alpha}) = V^\Phi(\overline{\beta}) \), from which it follows by Lemma 4 that \( \overline{\alpha} \models \overline{\beta} \in \Phi \). By (35), \( \overline{\alpha} \models \overline{\beta} \in \Phi \).

**Lemma 7** For all \( \text{DNT}(\alpha) \in \text{Act} \) and all \( V^\Phi(\overline{\beta}) \in [\text{DNT}(\alpha)]_{M^\Phi} \), it holds that \( \overline{\beta} \subseteq \text{DNT}(\alpha) \in \Phi \).
Proof The proof is by induction over DNT(α). For the induction base, note that for
DNT(α) = \overline{α}', it holds that \[\overline{α}']_{Mφ} = \{Vφ(\overline{α}')\}.\] By axiom (A6), it holds that \[\overline{α}'' \subseteq \overline{α}' \in Φ,\] so the claim in the lemma holds.

For the induction step, assume that the statement holds true for DNT(α'), DNT(α'').
Consider the term DNT(α) = DNT(α') \cup DNT(α''). Assume that Vφ(β) ∈ \[DNT(α') \cup DNT(α'')\] Mφ. It follows by definition that Vφ( \overline{β}) ∈ \[DNT(α') \cup DNT(α'')\] Mφ, and so by the induction hypothesis, \[\overline{β} \subseteq DNT(α') \in Φ or \overline{β} \subseteq DNT(α'') \in Φ.\] By (30), therefore, \[\overline{β} \subseteq DNT(α') \cup DNT(α'') \in Φ, i.e. \overline{β} \subseteq DNT(α) \in Φ.\] □

Lemma 8 \(α \models β \in Φ\) if and only if \[\[α\]_{Mφ} \models \[β\]_{Mφ}.\]

Proof (⇒) Assume that \(α \models β \in Φ\). By Theorem 5, \(α \models DNT(α) \in Φ\) and \(β \models DNT(β) \in Φ\), and so DNT(α) = DNT(β) ∈ Φ. Assuming that DNT(α) = \(\overline{α}_1 \cup \cdots \cup \overline{α}_k\) and DNT(β) = \(\overline{β}_1 \cup \cdots \cup \overline{β}_l\), it holds by (29) that \(\overline{α}_i \subseteq DNT(α) \in Φ,\) for \(1 \leq i \leq k.\) By the initial assumption and axiom (A19), it holds that \(\overline{α}_i \subseteq DNT(β) \in Φ\) for each \(\overline{α}_i\) occurring in DNT(α), \(1 \leq i \leq k.\) By axioms (A14) and (A15), it follows that for each \(\overline{α}_i\) occurring in DNT(α), \(1 \leq i \leq k,\) there is some \(\overline{β}_j\) occurring in DNT(β), \(1 \leq j \leq l,\) such that \(\overline{α}_i \models \overline{β}_j \in Φ.\) By Lemma 6, it follows that \([DNT(α)]_{Mφ} \subseteq [DNT(β)]_{Mφ}.\) By the same reasoning, it is shown that \([DNT(β)]_{Mφ} \subseteq [DNT(α)]_{Mφ}.\) This together with Theorem 4 implies that \([α]_{Mφ} \models [β]_{Mφ} = [DNT(α)]_{Mφ} = [DNT(β)]_{Mφ} = [β]_{Mφ}.\)

(⇐) Assume that \([α]_{Mφ} \models [β]_{Mφ}.\) By Theorem 4, \([α]_{Mφ} \models [DNT(α)]_{Mφ} \models [DNT(β)]_{Mφ} = [β]_{Mφ}.\) Assuming that DNT(α) = \(\overline{α}_1 \cup \cdots \cup \overline{α}_k\) and DNT(β) = \(\overline{β}_1 \cup \cdots \cup \overline{β}_l,\) Lemma 7 implies that for each \(Vφ(\overline{β}_i) \in [DNT(β)]_{Mφ} = [DNT(α)]_{Mφ},\) \(1 \leq i \leq l,\) it holds that \(\overline{β}_i \subseteq DNT(α) \in Φ.\) It follows that DNT(β) ⊆ DNT(α) ∈ Φ by (32). By the same reasoning, it holds that DNT(α) ⊆ DNT(β) ∈ Φ. It follows that DNT(α) = DNT(β) ∈ Φ by (33). By Theorem 5, \(α \models β \in Φ.\)

Due to the definition of the operator F, the proof of the next lemma is essentially the same as the corresponding proof by Castro and Maibaum (2009). Let

\[\text{LEG}^{φ} = \{S ∈ \mathcal{P}(\mathcal{P}(Hφ)) \setminus \{0\} : \text{for all } s ∈ S, s \cap Gφ \neq \emptyset\}.\]

Lemma 9 \(F(α) ∈ Φ\) if and only if \(Vφ(α) ∩ Gφ = \emptyset.\)

Proof (⇒) Assume that \(F(α) ∈ Φ.\) It follows that \(Vφ(α) \subseteq \bigcup\{Vφ(β) : F(β) ∈ Φ\},\) so \(Vφ(α) ∩ Gφ = \emptyset\) by the definition of Gφ.

(⇐) Assume that \(Vφ(α) ∩ Gφ = \emptyset.\) This implies that \(Vφ(α) \subseteq \bigcup\{Vφ(β) : F(β) ∈ Φ\}\) by the definition of Gφ. It follows that for each \(γ ∈ Vφ(α),\) there is some \(β\) such that \(γ \models β \in Φ\) and \(F(β) ∈ Φ.\) Therefore by Lemma 3, \(γ ≤ β \in Φ\) holds. By (27), it can be inferred that \(F(γ) \in Φ.\) Let \(χ\) be the disjunction of all γ for which \(γ \models Vφ(α).\) It then holds that \(χ \models α \in Φ.\) By (21), it follows that \(F(χ) \in Φ,\) and hence \(F(α) ∈ Φ\) by (26). □

Lemma 10 \(P(α) ∈ Φ\) if and only if \([α]_{Mφ} ∈ \text{LEG}^{φ}.\)

Proof (⇒) Assume that \(P(α) ∈ Φ.\) Assume that DNT(α) = \(\overline{α}_1 \cup \cdots \cup \overline{α}_k\). By Theorem 5, it holds that \(P(DNT(α)) ∈ Φ,\) so by axiom (A23) it follows that \(P(\overline{α}_i) ∈ Φ,\) \(1 ≤ i ≤ k.\) By (24) it is guaranteed that \(\neg F(\overline{α}_i) ∈ Φ,\) and hence \(F(\overline{α}_i) \notin Φ\) for
1 \leq i \leq k. By Lemma 9, it follows that \( V^\Phi(\overline{a_i}) \cap G^\Phi \neq \emptyset \), 1 \leq i \leq k. This, together with Theorem 4, implies that

\[ [\alpha]_{M^\Phi} = [\text{DNT}(\alpha)]_{M^\Phi} = \{ V^\Phi(\overline{a_1}), \ldots, V^\Phi(\overline{a_k}) \} \in \text{LEG}^\Phi. \]

(\Rightarrow) Assume that \([\alpha]_{M^\Phi} \in \text{LEG}^\Phi\). Assume that \( \text{DNT}(\alpha) = \overline{a_1} \sqcup \cdots \sqcup \overline{a_k} \). Since

\[ [\alpha]_{M^\Phi} = [\text{DNT}(\alpha)]_{M^\Phi} = \{ V^\Phi(\overline{a_1}), \ldots, V^\Phi(\overline{a_k}) \} \]

by Theorem 4, it holds by the initial assumption that \( V^\Phi(\overline{a_i}) \cap G^\Phi \neq \emptyset \), 1 \leq i \leq k. It follows by Lemma 9 that \( \text{F}(\overline{a_i}) \notin \Phi \), i.e. \( \neg \text{F}(\overline{a_i}) \in \Phi \), so by (25) it holds that \( \text{P}(\overline{a_i}) \in \Phi \) for 1 \leq i \leq k. By axiom (A23), it follows that \( \text{P}(\text{DNT}(\alpha)) \in \Phi \). Hence, by Theorem 5 and axiom (A27), \( \text{P}(\alpha) \in \Phi \).

\[ \square \]

Lemma 11 (Truth lemma) \( \varphi \in \Phi \) if and only if \( M^\Phi \models \varphi \).

**Proof** The proof is inductive. For the PL operators, the proof is standard. All the remaining cases follow from Lemmas 4, 8, and 10. \[ \square \]

Lemma 11 shows that the canonical model has the desired behavior, and completeness is then proved by a routine argument.

**Theorem 6** (Completeness theorem) For each consistent set \( \Sigma \) of formulas of DACL, there is a model which satisfies it.

**Proof** If \( \Sigma \) is consistent, then there exists a maximally consistent set \( \Sigma' \) which is an extension of \( \Sigma \). By the definition of the canonical model and Lemma 11, it follows that \( M^{\Sigma'} \models \Sigma \), so \( \Sigma \) has a model which satisfies it. This concludes the proof. \[ \square \]

### 6 Discussion

In this section, I will discuss some aspects of the approach in this paper that deserve further attention. First, I will briefly compare choice sets with the interpretation of propositions in inquisitive semantics. Second follows a discussion of the algebraic properties of choice sets, and finally, I will make a comparison with the conceptually similar Action Type Deontic Logic of Bentzen (2014).

#### 6.1 Alternativeness in Inquisitive Semantics

It should be stressed that arbitrary sets of action types count as choice sets under the present action theory. One may reasonably think that the notion of a choice set should be much more restricted. For example, a standard assumption in decision theoretic settings is that alternative actions are mutually exclusive.

In the literature on inquisitive semantics (Ciardelli and Roelofsen 2011; Ciardelli et al. 2009; Groenendijk and Roelofsen 2009; Roelofsen 2013), propositions are interpreted as sets of possibilities, where possibilities, in turn, are sets of possible worlds.

\[ \square \] Springer
Weak forms of inquisitive semantics (for example the system Inq∅; Ciardelli et al. 2009) do not put any restrictions on propositions (conceived of as sets of possibilities), and the resulting semantics comes very close to how choice sets are characterized in this paper. In stronger forms of inquisitive semantics (for example the system InqB; Ciardelli and Roelofsen 2011), two possibilities are said to be *alternatives* if neither of them is a proper subset of the other. In a similar spirit, why not restrict the notion of choice set so as they contain alternative (in the sense used above for possibilities) action types only? In the present action theoretic context, a restricted version of the concept of a choice set may be defined as follows: a choice set $S$ is *restricted* if for all $s \in S$, there is no $s' \in S$ such that $s \subset s'$. In other words, the interpretation of terms as restricted choice sets is obtained by applying the operator $\text{Alt}$, defined in Sect. 3.2.3, to the unrestricted choice sets of terms of any form, not only those of the form $\alpha$.

The reason for not making this move is the use of Simons’ supercover semantics. By using supercover semantics for permission and obligation in terms of the sets LEG and REQ, information will be lost by moving to restricted choice sets. For example, if $S = \{s\}$, $T = \{t\}$ and $t \subset s$, it follows that $\text{Alt}(S \cup T) = \{s\}$. If $s \cap G \neq \emptyset$ and $t \cap G = \emptyset$, it follows that $\text{Alt}(S \cup T) \in \text{LEG}$, even though $\text{Alt}T \notin \text{LEG}$. In order to keep the free choice principle, there is a point in allowing a more liberal definition of choice sets, even though this makes the algebraic properties of choice sets more complex.\footnote{Propositions in inquisitive semantics defined in terms of alternative possibilities (in the sense used above) form a Heyting algebra (Roelofsen 2013).}

### 6.2 Algebraic Considerations

The behavior of choices in DACL deviates from the Boolean interpretation of actions. Indeed, the algebraic structure defined on choice sets does not even form a lattice, since absorption is missing. On the other hand, a straightforward extension of DACL turns out to have an underlying semiring structure.

As developed in this paper, DACL is only concerned with non-empty choice sets. This is in part motivated by its relation to Simons’ supercover semantics. One may wish to extend the semantics with an empty choice set. This extension is straightforward, but interesting for technical reasons since it gives a semiring structure with $\emptyset$ as the identity element of the union of choice sets. The structure

$$\langle \mathcal{P}(\mathcal{P}(H)), \cup, \otimes, \sim, \emptyset, \{\emptyset\}, \{H\} \rangle$$

is a special kind of commutative idempotent semiring:

- $\langle \mathcal{P}(\mathcal{P}(H)), \cup, \emptyset \rangle$ is a join-semilattice;
- $\langle \mathcal{P}(\mathcal{P}(H)), \otimes, \{H\} \rangle$ is a commutative monoid;
- $\otimes$ distributes over $\cup$ and $\emptyset$ is an annihilator for $\otimes$;
- $\langle B, \otimes, \sim, \{\emptyset\}, \{H\} \rangle$ is a Boolean algebra, where $B$ is the set of singleton choice sets;
– ∼ is an operation mapping every choice set to the set containing the complement of its union.

The language and axiomatics of DACL must be modified slightly to account for the addition of the empty choice set. A new designated term, say \( f \), whose action type interpretation and choice set interpretation both equals the empty set, should be added.\(^8\)

Adding axioms

\[
\begin{align*}
\alpha \sqcup f & \models \alpha & (36) \\
\alpha \sqcap f & \models f & (37) \\
f & \models 0 & (38)
\end{align*}
\]

to the non-deontic axioms of DACL and replacing axiom \((A15)\) by

\[
\alpha \sqsubseteq \beta \rightarrow \alpha \models \beta \lor \alpha \models f
\]

result in a complete axiom system for the extended version of the non-deontic fragment of DACL. The proof of this claim is similar to the completeness proof in Sects. 5.3 and 5.4. Note that adding (38) allows inferring

\[
\bar{f} \models 1.
\]

Algebraic considerations suggest that any finite idempotent commutative semiring with an additional Boolean algebra structure on its join-irreducible elements and a suitable ‘negation’ operation satisfying axioms corresponding to \((A10)\), \((A14)\), \((39)\) and \((40)\) is isomorphic to a semiring \(\mathcal{P}(\mathcal{P}(H))\) for \(H\) some finite set, with operations \(\otimes\), set union \(\cup\) and \(\sim\). The argument for this claim is similar to corresponding proofs of representation theorems for finite distributive lattices and Boolean algebras (see e.g. Davey and Priestley 2002, Chapter 5). In the infinite case, things are less clear. Note, however, that the syntactic restriction on the set of generators \(\text{Act}_0\) already presupposes finiteness, and that the construction of the canonical models in the completeness proof shows that any satisfiable DACL formula is satisfiable in a model with a finite domain. However, this is not the place to discuss this in detail, and I will leave the issue for further research.

6.3 Bentzen’s Action Type Deontic Logic

A deontic action logic that is conceptually related to the present approach is presented by Bentzen (2014). Bentzen uses a distinction between action expressions (terms) and propositional statements, and introduces a kind of distribution requirement in the interpretation of disjunctive action expressions. The semantics is intended to solve many of the problems with the standard approach to deontic logic.

In order to account for the special interaction between deontic operators and disjunction, Bentzen introduces a disjunctive term operator which is interpreted so as to

\(^8\) Note that \(f\) must be introduced as a designated term in order to keep functional completeness: no combination of terms of the original DACL language has the empty set as its choice set interpretation.
be non-empty: the action type denoted by the disjunctive term \((\alpha \text{ or } \beta)\) is instantiated by an action token \(h\) if and only if \(h\) instantiates the action type denoted by \(\alpha\) or \(h\) instantiates the action type denoted by \(\beta\) and there are action tokens instantiating the action type denoted by \(\alpha\) and action tokens instantiating the action type denoted by \(\beta\). Bentzen interprets this non-emptiness condition as a “a criterion of relevance or availability” (2014, p. 405).

The models of the semantics are structures \(M = \langle G, V \rangle\): \(G\) is a non-empty set representing the legal action tokens available to an agent in a situation, and \(V\) is function assigning subsets of \(G\) to every term. Formally, the relevance criterion is captured by interpreting disjunctive terms \(\alpha \cup \beta\) as follows:

\[
V(\alpha \cup \beta) = \begin{cases} 
\emptyset & \text{if } V(\alpha) = \emptyset \text{ or } V(\beta) = \emptyset, \\
V(\alpha) \cup V(\beta) & \text{otherwise.} 
\end{cases}
\]

The satisfiability conditions for permission and obligation sentences are defined as

\[
M \models \text{may}[\alpha] \iff V(\alpha) \cap G \neq \emptyset,
\]

\[
M \models \text{must}[\alpha] \iff V(\alpha) = G.
\]

Conceptually, the approach in this paper is closely related to Bentzen’s, even though the formal implementation is quite different. Bentzen motivates his semantics by considering several benchmark cases. These consist of principles that in some way or another indicate problems with other deontic logics, and are divided into principles that should be valid and principles that should not be valid. Interestingly, DACL agrees with Bentzen’s semantics on all of these principles except for two. In Bentzen’s semantics, permission and obligation are duals, and a closure property of permission, stating that for every action either it or its negation is permitted, is valid. As is clear from the syntactic definition of \(O\), the duality property does not hold in DACL. Item (11) of Proposition 3 shows that there is no closure of permission in DACL. However, note that DACL preserves free choice effects when disjunction is embedded under conjunction [item (9) of Proposition 3]. This property fails for Bentzen’s semantics.

An interesting difference between DACL and Bentzen’s semantics is that the latter only model ideal behavior in the sense that only legal action tokens are included in the domains of the models: “it is presupposed that the agent chooses an acceptable action.” (Bentzen 2014, p. 406). This assumption is a limitation of Bentzen’s approach. The concept of norm-violation is often seen as an integral part of deontic logic, for example when modeling contrary-to-duties or in applications to agent-regulation. Some even claim that the possibility of norm-violations is inherent to the definition of a normative system. I interpret Carmo and Jones (2002) as holding this view. They argue that if it can be assumed that agents always behave as they should,
the normative dimension ceases to be of interest: the actual does not depart from the ideal, so nothing is lost by merely describing what agents in fact do. (Carmo and Jones 2002, p. 265)

In the semantics presented in this paper, the possibility of agents having illegal action tokens available is included. Whether it is possible to extend Bentzen’s semantics in order to account for norm-violations is a question that will be left open here: suffice it to note that the approach in the present paper represents an alternative formal implementation of the ideas underlying Bentzen’s semantics.

7 Conclusion

I have presented an approach to deontic action logic where action expressions have two kinds of formal interpretations: as action types, and as choice sets. This distinction is then utilized when interpreting deontic concepts. A formal deontic logic based on the logics of Trypuz and Kulicki (2009, 2015) was developed. In this logic, permissions are sensitive to the choice meaning of action expressions, and substitution of logical equivalents within the scope of the permission operator is not in general a valid rule of inference. The main result is the completeness of an axiomatization of the logic.

The logic is intended to capture properties of informal deontic reasoning, and in particular provide ways to handle the various problems related to the interaction of permission and disjunction. The present approach is conceptually similar to Bentzen’s Action Type Deontic Logic (2014), but the formal implementation is different. Unlike Bentzen’s approach, it allows for the modeling of non-ideal actions. This opens up possibilities for modeling norm-violations, for example contrary-to-duty obligations (Carmo and Jones 2002). This is an interesting topic for further research.

There are additional requirements one may put on choice sets. For example, the notion of alternative could be made more restrictive as done in inquisitive semantics and in decision theory in general. What kind of logic such different restrictions give rise to is a topic for further research. In the approach taken here, the interpretation of action-negation essentially collapses the choice sets of negated terms into singleton choice sets. This makes it possible to utilize the disjunctive negative translation in the completeness proof. On the other hand, one might think that this interpretation lacks sufficient philosophical motivation.

At present, the logic is quite limited in its scope. It cannot be used to talk about performances and consequence of actions and choices. It would be interesting to consider the approach developed here in a dynamic setting (see e.g. Castro and Maibaum 2009; Meyer 1988), thus arriving at a considerably more expressive theory. Neither are there any resources for reasoning about sequential composition of actions, and I did not consider contrary-to-duty normative reasoning. These are all topics for further inquiry.

Finally, the algebraic foundations of DACL deserve closer attention. I argued in Sect. 6.2 that a straightforward extension gives rise to a special kind of semiring structure. Because of the close connection with inquisitive semantics, research in this direction may provide results not only on the algebraic properties of choice sets, but
also regarding the algebraic foundations of versions of inquisitive semantics with non-maximal possibilities.

Acknowledgements I wish to thank Valentin Goranko, Magnus Hjelmblom and Piotr Kulicki for helpful discussions and comments on earlier drafts of this work, and the three anonymous reviewers for their helpful suggestions. Parts of this work were carried out while I was affiliated with the Decision, Risk, and Policy Analysis group at the Faculty of Engineering and Sustainable Development, University of Gävle, Sweden.

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

References

Alchourrón, C. E., & Bulygin, E. (1971). *Normative systems*. New York: Springer.

Alonso-Ovalle, L. (2006). *Disjunction in alternative semantics*. Ph.D. Dissertation, University of Massachusetts Amherst.

Anglberger, A. J. J., Dong, H., & Roy, O. (2014). Open reading without free choice. In F. Cariani, D. Grossi, J. Meheus, & X. Parent (Eds.), *Deontic logic and normative systems* (pp. 19–32). Heidelberg: Springer.

Asher, N., & Bonevac, D. (2005). Free choice permission is strong permission. *Synthese, 145*(3), 303–323.

Bentzen, M. M. (2014). Action type deontic logic. *Journal of Logic, Language and Information, 23*(4), 397–414.

Broersen, J. (2004). Action negation and alternative reductions for dynamic deontic logics. *Journal of Applied Logic, 2*, 153–168.

Carmo, J., & Jones, A. J. I. (2002). Deontic logic and contrary-to-duties. In D. M. Gabbay & F. Guenthner (Eds.), *Handbook of philosophical logic* (Vol. 8, pp. 265–343). Dordrecht: Kluwer.

Castro, P. F., & Maibaum, T. S. E. (2009). Deontic action logic, atomic Boolean algebras and fault-tolerance. *Journal of Applied Logic, 7*(4), 441–466.

Chellas, B. F. (1980). *Modal logic: An introduction*. Cambridge: Cambridge University Press.

Ciardelli, I., Groenendijk, J., & Roelofsen, F. (2009). Attention! might in inquisitive semantics. In S. Ito & E. Corman (Eds.), *Proceedings of semantics and linguistic theory* (Vol. XIX, pp. 91–108). Ithaca: Cornell University, CLC Publications.

Ciardelli, I., & Roelofsen, F. (2011). Inquisitive logic. *Journal of Philosophical Logic, 40*(1), 55–94.

Davey, B., & Priestley, H. (2002). *Introduction to lattices and order*. Cambridge: Cambridge University Press.

Dignum, F., Meyer, J.-J. C., & Wierenga, R. J. (1996). Free choice and contextually permitted actions. *Studia Logica, 57*, 193–220.

Giordani, A., & Canavotto, I. (2016). Basic action deontic logic. In O. Roy, A. Tamminga, & M. Willer (Eds.), *Deontic logic and normative systems* (pp. 80–92). London: College Publications.

Girard, J.-Y. (1987). Linear logic. *Theoretical Computer Science, 50*(1), 1–101.

Groenendijk, J., & Roelofsen, F. (2009). Inquisitive semantics and pragmatics. In J. M. Larrazabal & L. Zubeldia (Eds.), *Meaning, content, and argument: Proceedings of the ILCLI international workshop on semantics, pragmatics, and rhetoric* (pp. 41–72). San Sebastián: University of the Basque Country Publication Service.

Hansson, S. O. (2013). The varieties of permission. In D. Gabbay, J. Horry, X. Parent, R. van der Meyden, & L. van der Torre (Eds.), *Handbook of deontic logic and normative systems* (pp. 195–240). London: College Publications.

Hilpinen, R. (1982). Disjunctive permissions and conditionals with disjunctive antecedents. *Acta Philosophica Fennica, 35*, 175–194.

Humberstone, L. (2011). *The connectives*. Cambridge, MA: MIT Press.

Ju, F., & van Eijck, J. (2016). To do something else. In O. Roy, A. Tamminga, & M. Willer (Eds.), *Deontic logic and normative systems* (pp. 109–122). London: College Publications.

Kamp, H. (1973). Free choice permission. *Proceedings of the Aristotelian Society, 74*, 57–74.
Meyer, J-J Ch. (1988). A different approach to deontic logic: Deontic logic viewed as a variant of dynamic logic. Notre Dame Journal of Formal Logic, 29(1), 109–136.

Monk, J. D. (1976). Mathematical logic. New York: Springer.

Pelletier, F. J., & Asher, N. (1997). Generics and defaults. In J. van Benthem & A. G. B. ter Meulen (Eds.), Handbook of logic and language (pp. 1125–1177). Amsterdam: Elsevier.

Roelofsen, F. (2013). Algebraic foundations for the semantic treatment of inquisitive content. Synthese, 190, 79–102.

Ross, A. (1941). Imperatives and logic. Theoria, 7, 53–71.

Segerberg, K. (1982). A deontic logic of action. Studia Logica, 41(2–3), 269–282.

Simons, M. (2005a). Dividing things up: The semantics of or and the modal/or interaction. Natural Language Semantics, 13(3), 271–316.

Simons, M. (2005b). Semantics and pragmatics in the interpretation of or. In E. Georgala & J. Howell (Eds.), Proceedings of semantics and linguistic theory (Vol. XV, pp. 205–222). Ithaca: Cornell University, CLC Publications.

Trypuz, R., & Kulicki, P. (2009). A systematics of deontic action logics based on Boolean algebra. Logic and Logical Philosophy, 18(3–4), 253–270.

Trypuz, R., & Kulicki, P. (2015). On deontic action logics based on Boolean algebra. Journal of Logic and Computation, 25(5), 1241–1260.

van der Meyden, R. (1996). The dynamic logic of permission. Journal of Logic and Computation, 6(3), 465–479.

von Wright, G. H. (1951). Deontic logic. Mind, 60, 1–15.