Highlights

AI Alignment and Normative Reasoning:
Addressing Uncertainty through Deontic Logic

Ali Farjami

- Reinforced AI alignment with human values via deontic logic.
- Advanced AI alignment through integration of normative reasoning and preference-based approaches.
- Demonstrated complexity of ethical decisions in real-world scenarios.
- New algebraic system in LogiKEy used for complex ethical decision-making.
- Implemented input/output logic for normative reasoning under uncertainty.

AI Alignment and Normative Reasoning: Addressing Uncertainty through Deontic Logic

Ali Farjami[a]

[a]*Iran University of Science and Technology,*

Abstract

As artificial intelligence (AI) systems become more sophisticated and are increasingly used to make decisions that affect human lives, ensuring their alignment with human values and preferences is of paramount importance. One key aspect of AI alignment is addressing uncertainty, as AI systems that are highly certain about the outcomes of their actions may be less likely to make decisions in line with human values. In this paper, the role of normative reasoning in addressing uncertainty in AI alignment is explored, specifically examining how deontic logic can be used to guide the decision-making process of AI systems. The paper illustrates the significance of taking into account both normative reasoning and preference-based approaches in order to ensure AI alignment. It is examined the potential of deontic logic, a formal system that deals with rights and duties, to help AI systems better understand and make decisions under uncertainty in a way that is consistent with human preferences. The analysis has implications for the design and development of AI systems that can make decisions that are ethically responsible. In order to design and engineer ethical and legal reasoners and responsible systems, Benzmüller, Parent and van der Torre introduced the LogiKEy methodology, based on the semantical embedding of deontic logics into classic higher-order logic. This paper considerably extends the LogiKEy deontic logics and dataset using an algebraic approach, and develops a theory of input/output operations for normative reasoning on uncertainty.

Keywords: AI Alignment, Normative Reasoning, Reasoning under Uncertainty, Input/Output Logic, LogiKEy Framework, Higher-order Logic

Email address: farjami110@gmail.com (Ali Farjami)

1. Introduction

Ensuring that artificial intelligence (AI) systems are aligned with the values and goals of their human creators, known as AI alignment, is crucial as AI systems become more sophisticated and are used to make decisions that affect human lives. One way to approach AI alignment is to consider the following principles: the machine's only objective is to maximize the realization of human preferences, the machine is initially uncertain about what those preferences are, and the ultimate source of information about human preferences is human behavior [1]. The principles mentioned can serve as valuable guidelines for designing and developing AI systems that align with human values and goals.

Uncertainty can arise in various forms in AI systems, including incomplete or uncertain data, lack of knowledge about the environment, and the complexity of the decision-making process [2]. Addressing uncertainty is therefore an important aspect of AI alignment, as it helps to ensure that an AI system has a clear understanding of the outcomes of its actions and is able to make decisions that are aligned with the values and preferences of its human creators. One way to address uncertainty in AI alignment is through the use of normative reasoning, which involves the use of principles, values, and moral theories to guide decision-making [1, 3].

One field of study highly relevant to AI alignment is deontic logic, a formal system specifically designed to address rights and duties within normative reasoning. Deontic logic can be used to guide the decision-making process of AI systems in a way that is consistent with human preferences [4, 5]. This can be particularly useful when it comes to understanding and managing uncertainty [6, 7]. In this paper, the potential of deontic logic and normative reasoning to help address uncertainty in AI alignment is explored. The analysis has implications for the design and development of AI systems that are aligned with human values and goals and can make decisions that are ethically responsible.

In this paper, an algebraic formal framework for normative reasoning in uncertainty, inspired by input/output logic [8], is introduced. This framework is based on a set of given norms - such as social, ethical, or legal norms - and common conditional inference rules of conditional logic [9]. This logical framework for normative reasoning incorporates conditional preferences and employs the LogiKEy methodology. While the ultimate goal is to implement and experiment with these normative theories in machines, the present paper

lays the foundational groundwork for such future endeavors. Benzmüller, Parent and van der Torre [10] introduced the LogiKEy framework for the formalization and automation of new ethical reasoners, normative theories and deontic logics. The LogiKEy framework uses higher-order logic (HOL) as a metalogic to embed other logics. A logic embedded in HOL can thus be tracked by automated theorem provers (ATP), interactive automated provers (ITP) and HOL model finders. The LogiKEy methodology allows a user to simultaneously combine and experiment with underlying logics (and their combinations), ethico-legal domain theories, and concrete examples.

Earlier work presented semantical embedding of two traditions in deontic logic in the LogiKEy framework, namely Åqvist's dyadic deontic logic **E** [11] and Makinson and van der Torre's input/output (I/O) logic [12]. Subsequent work provided the Isabelle/HOL dataset for the LogiKEy workbench [13]. This paper considerably extends the LogiKEy deontic logics and dataset using an algebraic approach. In particular, it extends the theory of input/output operations [8] and corresponding proof systems on top of Boolean algebras and, more generally, abstract logics [14]. One advantage of building I/O operations over Boolean algebras is that the I/O logic can be directly embedded in HOL. Moreover, the adaptation of input/output logic to a wide range of base logics is beneficial for AI systems. Having a large class of logics available allows for accommodating the specific needs of AI systems.

The paper is structured as follows: Sections 2 and 3 introduce a new deontic consequence relation for reasoning under uncertainty and provide the soundness and completeness results of I/O operations for deriving permissions and obligations on top of Boolean algebras. Section 4 shows how I/O operations can be generalized over any abstract logic. Section 5 integrates a conditional theory into input/output logic and shows why it is crucial to consider both approaches based on norms and preferences when it comes to decision-making in the realm of aligning AI with human values. Section 6 introduces semantical embedding of I/O logic into HOL, including soundness and completeness (faithfulness). Section 7 discusses related work, and Section 8 concludes the paper. Appendix A shows how the semantical embedding described in Section 6 is implemented in the Isabelle/HOL proof assistant. Some experiments are provided to show that this logic implementation enables interactive and automated reasoning. All the proofs are in the appendices. Appendix B provides proofs relating to Sections 2 and 3, Appendix C proofs relating to Section 5 and Appendix D proofs relating to Section 6.

2. Permissive norms: input/output operations

Deontic logic is a branch of formal logic that deals with norms and values, and the logical relationships between them. It is frequently used to represent and reason about systems of moral norms and values, and to evaluate the logical consistency and coherence of these systems [15, 16]. Obligations in deontic logic are moral duties or legal requirements that must be fulfilled, while permissions are moral or legal allowances or authorizations for an action or state of affairs. There is a logical relationship between these two types of norms, as fulfilling an obligation may be necessary to be granted a permission, and exercising a permission may be necessary to fulfill an obligation. This relationship can be represented using the "implies" operator. However, the relationship between obligations and permissions is not always straightforward and may involve conflicting or competing demands. Permissions in deontic logic or normative systems can be classified in various ways, such as based on their source or origin (e.g., legal, moral, practical), scope or extent (e.g., global, local), or level of generality or specificity [17]. Four specific types of permissions that are often distinguished are weak permissions, which are derived from obligations and are the dual of obligations; static permissions, which are derived from strong permissions or explicit permissive norms in a normative system; dynamic permissions, which guide the legislator by describing the limits on what may be prohibited in a normative system; and exemptions, which are exceptions to prohibitions in a normative system [17, 18, 19]. The classification of permissions can be useful in evaluating the logical consistency and coherence of different norms and values, and in determining the appropriate course of action in a given situation. In this paper, a generic and abstract form of permission is considered by applying inference rules to a set of permissive norms. The relationship between different concepts of permission and obligation is explored in a separate work using subordination algebra and contact algebra [20, 21].

Normative systems that consider permission as primitive and define obligation as a derived concept typically view permissions as more fundamental or basic norms or values. In these systems, obligations are often defined in terms of permissions, such that the fulfillment of an obligation may be necessary in order to exercise a permission or to achieve a certain state of affairs. For example, an obligation to act in the best interest of others may be derived from a permission to pursue one's own interests, or an obligation to respect the rights of others may be derived from a permission to pursue one's own

interests or goals. This relationship between obligations and permissions can be used to evaluate the consistency and coherence of different norms and values within the system, and to determine the appropriate course of action in a given situation [17]. This approach to norms and values may not be specific to any particular category of permission or obligation, but rather applies more generally to the way in which these concepts are understood and related within the system.

2.1. Input/output logic

There are two main families of deontic logic: modal logic-based deontic logics and norm-based deontic logics. Modal logic-based deontic logics draw from the principles of modal logic, primarily engaging with the concepts of necessity and possibility. One of the most well-known modal logic-based deontic logics is standard deontic logic (SDL) [22]. Dyadic deontic logic (DDL) introduces a conditional operator to represent conditional obligation sentences dealing with norm violation, and it has been used to deal with contrary-to-duty reasoning [23] and prima facie obligations. Hansson [24], Åqvist [25], Kratzer [26], and, Carmo and Jones [27] are notable figures in the development of DDL. On the other hand, norm-based deontic logic [28] is a family of frameworks that analyze deontic modalities with reference to a set of explicitly given norms, rather than with reference to a set of possible worlds. The focus is on inferring which norms apply, given some input (e.g., a fact) and a set of explicit conditional norms (a normative system). Examples of norm-based deontic logics include input/output (I/O) logic, which uses operational semantics based on the concept of detachment and manipulates pairs of formulas rather than individual formulas [8], and Horty's theory of reasons [29], which is based on Reiter's default logic [30].

Input/output logic was initially introduced by Makinson and van der Torre [31] to study conditional norms viewed as relations between logical formulas. I/O logic provides a formal framework for reasoning about normative systems, which can be seen as sets of conditional obligations or permissions. These systems are used in a wide range of fields, from legal systems [32] and ethical frameworks [12] to automated decision-making processes [33] in artificial intelligence. One major advantage of I/O logic is its flexibility and clarity in modeling complex normative scenarios. It represents norms as pairs of conditions (inputs) and their associated actions or states (outputs). This allows for a clear, structured approach to representing norms, even when

the norms might be interconnected or conflicting. For example, in a self-driving car scenario, one conditional norm might be "if there is a pedestrian in the crosswalk (input), the car should stop (output)", while another might be "if the traffic light is green (input), the car should proceed (output)". I/O logic provides tools for reasoning about what to do when these norms conflict [34]. In this setting, the meaning of normative concepts is given in terms of a set of procedures yielding outputs for inputs. Let N^O denote a set of obligatory norms and N^P a set of permissive norms. The formal expression $(a, x) \in N^O$ means "given a, it is obligatory that x", while the formal expression $(a, x) \in N^P$ means "given a, it is permitted that x." The formal expression $x \in out(N^P, A)$ means "given normative system N^P and input set A (state of affairs), x (permission) is in the output". The output operations resemble inferences, where inputs need not be included among outputs, and outputs need not be reusable as inputs [31]. The proof system of an I/O logic is specified via a number of derivation rules acting on pairs (a, x) of formulas. Given a set N of pairs, $(a, x) \in derive_i(N)$ is written to say that (a, x) can be derived from N using these rules. The term "input/output logic" is used broadly to refer to a family of related systems such as *simple-minded*, *basic*, and *reusable* systems [31, 35]. This section uses similar terminology, and introduces some input/output systems for deriving permissions on top of Boolean algebras. Each derivation system is closed under a set of rules, including for instance the weakening of the output (WO) rule or the strengthening of the input (SI) rule. A bottom-up approach is used to characterize different derivation systems. The AND rule, for the output, is absent in the derivation systems presented in this section. In this section and the next, the division of input/output operations is based on the common distinction between possibility ($\Diamond$) and necessity ($\Box$) modal operators in modal logic. The possibility operator is not closed under AND. In the paper, there is no strong idea on whether the input/output operations of this section can or cannot be used for obligation, or vice versa. It depends on the context and application. Generally, the initial set of norms that mention either obligatory N^O or permissive N^P norms provides a better guidance for the purpose of the input/output systems in this paper.

The results in this section hold for any abstract logic, but Boolean algebras are used as the main algebraic structures to prove lemmas and theorems. One reason for using Boolean algebras is that they provide a more uniform formal framework for the purposes of the paper. In Section 5, Boolean algebras are used to model propositional logic and can be extended with pref-

erence relations over valuation functions or possible states. In Section 6, Boolean algebras form the basis of the semantical embedding for implementation in higher-order logic and proof assistants. Overall, the use of Boolean algebras allows for a more algebraic and formal analysis of normative systems, and can provide a basis for the development of more powerful and effective methods for reasoning about such systems. By considering multiple layers of abstraction, it may be possible to gain a more comprehensive view of the properties and behaviors of normative systems and to design systems that are more adaptable and applicable to a wide range of contexts.

Definition 1 (Boolean algebra). *A structure* $\mathcal{B} = \langle B, \wedge, \vee, \neg, 0, 1\rangle$ *is a Boolean algebra if and only if it satisfies the following identities:*[1]

- $x \vee y \approx y \vee x$, $x \wedge y \approx y \wedge x$
- $x \vee (y \vee z) \approx (x \vee y) \vee z$, $x \wedge (y \wedge z) \approx (x \wedge y) \wedge z$
- $x \vee 0 \approx x$, $x \wedge 1 \approx x$
- $x \vee \neg x \approx 1$, $x \wedge \neg x \approx 0$
- $x \vee (y \wedge z) \approx (x \vee y) \wedge (x \vee z)$, $x \wedge (y \vee z) \approx (x \wedge y) \vee (x \wedge z)$

Definition 2 (Syntax). *For a set of variables* X*, the set of Boolean terms defined over* X *is denoted by* $Ter(X)$ *as follows:*

$$Ter(X) = \bigcup_{n\in N} Ter_n(X)$$

where

$$Ter_0(X) = X \cup \{0, 1\}$$
$$Ter_{n+1}(X) = Ter_n(X) \cup \{a \wedge b, a \vee b, \neg a : a, b \in Ter_n(X)\}.$$

Given a Boolean algebra $\mathcal{B}$*, the elements of* $Ter(B)$ *are ordered as* $a \leq b$ *iff* $a \wedge b =_{\mathcal{B}} a$.[2] *Since* $\leq$ *is antisymmetric,* $a \leq b$ *and* $b \leq a$ *imply* $a =_{\mathcal{B}} b$.

Term algebras are a specific type of algebraic system that are used in the study of formal languages and logical systems. Term algebras are used to represent the syntactic structure of formal languages and to study the logical properties of those languages [36].

[1] An equation $t \approx t'$ holds in an algebra $\mathcal{A}$ if its universal closure $\forall x_0 ... x_n t \approx t'$ is a (first-order) sentence that is true in $\mathcal{A}$.

[2] The symbol "$=_{\mathcal{B}}$" is used to express that both sides name the same object in $\mathcal{B}$. The elements of the variable set (B) that are represented by different letters are supposed to be independent in the algebra $(\mathcal{B})$ w.r.t. $\leq$.

2.2. Deontic logic: a new consequence relation

Definition 3 (Upward-closed set). *Given a Boolean algebra $\mathcal{B}$, a set $A \subseteq Ter(B)$ is called upward-closed if it satisfies the following property:*

$$\textit{For all } x, y \in Ter(B), \textit{ if } x \leq y \textit{ and } x \in A, \textit{ then } y \in A.$$

The least upward-closed set that includes A is denoted by $Up(A)$. The Up operator satisfies the following properties:

- $A \subseteq Up(A)$ *(Inclusion)*
- $A \subseteq B \Rightarrow Up(A) \subseteq Up(B)$ *(Monotony)*
- $Up(A) = Up(Up(A))$ *(Idempotence)*

An operator that satisfies these properties is called a closure operator.

The "Up" operator, for a given set A, sees all the elements that are in a higher or equal position to the elements of A in terms of their ordering in Boolean algebra. Unlike the *propositional logic consequence relation* ("Cn") operator, the "Up" operator is not closed under conjunction so that we do not have $a \wedge \neg a \in Up(a, \neg a)$. The Up operator is defined as the union of the sets of all statements that follow from each individual member of the given set, or equivalently as the union of the sets of all statements that follow from the given set under the standard consequence relation (Cn):

$$Up(A) = \bigcup \{Cn(a) | a \in A\}$$

The following subsections provide motivation for the application of the Up operator in normative reasoning under uncertainty.

2.3. Evaluative uncertainty

Uncertainty, an inherent part of human cognition, is of different types depending on the nature of what is not known [37]. *Probabilistic uncertainty*, one of the most widely recognized, pertains to situations where the outcome is not deterministic but can be represented probabilistically [38]. For instance, the outcome of a dice roll or the weather forecast for the next day, which can be described by a set of possibilities each with a probability of occurrence. However, not all uncertainties are probabilistic. *Non-probabilistic uncertainty* arises in situations where the possible outcomes or their likelihoods cannot be clearly determined or quantified [39]. An example could be the uncertainty regarding the potential impacts of a newly emerged virus,

especially at its early stage of emergence when there isn't enough data for probabilistic assessments.

Beyond *empirical uncertainty*, which involves a lack of knowledge about factual elements of the world, we can encounter other varieties [40]. *Evaluative uncertainty* pertains to the difficulty of determining the value or worth of different options or outcomes. For instance, should you prioritize job security or job satisfaction while choosing a career? *Option uncertainty* refers to situations where one is unsure about the options that are available or that will be available. For example, an investor might be uncertain about future investment options due to unpredictable market conditions. *Modal uncertainty* pertains to the uncertainties regarding different possible worlds or the ways the world could be. For instance, contemplating the consequences of decisions in a world where climate change was addressed proactively versus one where it was ignored. This paper focuses on evaluative uncertainty.

Within evaluative uncertainty, we can discern two main types [40, 41]. *Uncertainty due to value assessments* represents the challenge of assigning value to specific outcomes. To illustrate, even when a person knows the complete specifications of two phones, they might struggle to decide which is better because they are unsure whether to prioritize battery life, camera quality, or brand reputation. Conversely, *normative uncertainty* refers to indecision about what ethical or moral principles to adhere to while making decisions. An example could be a person facing a dilemma between helping a friend in need, which aligns with the principle of beneficence, or maintaining their own well-being and personal commitments, which is an exercise of self-care and responsibility. The difference lies in the source of uncertainty, whether it's a problem of evaluating options or choosing between principles. Uncertainty due to value assessments refers to uncertainty about the subjective evaluations and preferences that individuals have regarding the desirability or value of different consequences or attributes. It involves uncertainty about how to prioritize and evaluate the properties of a consequence based on personal preferences. On the other hand, normative uncertainty relates to uncertainty about normative facts or principles, encompassing uncertainty about what is considered morally or ethically preferable.

Sections 2 and 3 explore input/output operations concerning normative uncertainty. Meanwhile, Section 5 brings utility functions and preference relations into the fold of a non-monotonic logical framework to confront evaluative uncertainty [40]. A critical challenge in AI value learning emerges from Stuart Armstrong's no-free-lunch result [42], which posits that numerous po-

tential utility functions can align with a specific dataset, making it nearly impossible to pinpoint an exact model of human values [6]. Even the application of Occam's Razor, which favors the simplest solution, fails to alleviate this predicament and could potentially lead AI systems astray. Representation Theorems, such as the Von Neumann-Morgenstern utility theorem, offer some solace, yet they require an impractically extensive evaluation of decisions. Moreover, relying solely on real-world human decisions overlooks potential preferences in unobserved scenarios, further complicating value extraction. For a detailed discussion, refer to "The Pointers Problem: Clarifications/Variations" by Abram Demski. Therefor, normative uncertainty often requires more nuanced philosophical deliberation and moral reasoning beyond the scope of utility functions, highlighting the need for comprehensive exploration and ethical discussions in addressing normative uncertainty.

2.4. Discursive input/output logic

In the original system of input/output logic [31], the Cn consequence relation is used to define input/output operations. This paper proposes using a new consequence relation, Up, instead of Cn. The main idea of this paper is to explore the possibilities of using Up in place of Cn for defining input/output operations. The specific benefits of using the Up consequence relation and how it can be applied in practice will be further discussed in the following.

Consideration of multiple viewpoints The $Up(A)$ operator does not allow for the derivation of the conjunctive formula $a \wedge b$ from the set $\{a, b\}$. This means that the consequence relation is non-adjunctive, as it does not allow for the combination of the two premises into a single conjunctive statement [43]. Instead, each premise is treated separately and individually. Non-adjunctive or discursive consequence relations can be seen as a way of dealing with uncertainty, as they allow for the consideration of multiple viewpoints and the possibility of conflicting premises. From the perspective of normative uncertainty, the significance of non-adjunctivity becomes apparent. Normative uncertainty arises when there are uncertainties about how to apply or prioritize different moral or ethical principles. The non-adjunctive nature of the consequence relation becomes valuable in this context. It avoids attempting to merge diverse moral assessments into a single conjunction statement. Instead, it treats each moral principle or norm independently, acknowledging their individual merits and respecting the subjective nature of moral assessments. By embedding multiple, possibly conflicting, ethical evaluations

within a unified logical framework, we enable an AI system to deliberate over these tensions—such as an autonomous vehicle making a decision between the safety of pedestrians and that of its passengers—rather than merely following a predefined ethical directive. In the subsequent discussion, we elucidate the distinction between using Up and Cn in addressing normative conflicts and handling *uncertainty in inputs and outputs*. Additionally, we delve into the *asymmetry between inputs and throughputs* when aggregating viewpoints.

Viewpoints are complex The Up operator presents a unique challenge due to its non-compact nature, especially when dealing with logical consequences arising from a set of sentences, such as $A = \{a_1, (a_1 \wedge a_2), (a_1 \wedge a_2 \wedge a_3), ...\}$. The non-compactness of the Up operator is highlighted with this set, which contains an uncountable number of elements. The Up operator is incapable of performing (a finite) conjunction of these elements, thereby rendering it not finitely axiomatizable. The use of Up operator can be related to uncertainty in the sense that it allows for the representation and reasoning about a set of propositions that are not fully captured by a finite set of axioms and is not deductively axiomatizable. In situations where values are unclear or open to interpretation, evaluative (normative) uncertainty emerges, requiring individuals to engage in subjective assessments and the construction of values. Yudkowsky's "Hidden Complexity of Wishes" suggests that our values are more complicated than they seem. It warns against oversimplifying goals in systems like AI. In this context, the non-compactness of the Up operator highlights the challenges in aligning AI with our true values and viewpoints.

Example 1. *Let's consider an AI system responsible for managing and moderating an online social platform. The AI's purpose is to create an environment that encourages respectful dialogue and reduces harm. Here, we can represent different ethical and community standards as atomic propositions: a_1 could represent respect for free speech, a_2 might stand for zero-tolerance for hate speech, a_3 might be about protection of privacy, a_4 might be about preventing misinformation, and so on. However, what constitutes "harm" or "respectful dialogue" might evolve with society's moral progress. For instance, certain behaviors or words that were considered acceptable a few years ago might be regarded as offensive or harmful now, and this necessitates an update in the AI's understanding and enforcement of community standards. In this context, we can define a set of propositions $Y = \{a_1, (a_1 \wedge a_2), (a_1 \wedge a_2 \wedge a_3), \ldots\}$, with each combination representing a different stage of*

moral progress and the corresponding changes in the enforcement of community standards. To deal with this evaluative uncertainty due to the progress in societal values, the AI system would need to continuously learn, adapt, and update its understanding of what constitutes harm, respectful dialogue, privacy, and misinformation, among other factors. It would also need to consider how to balance these values when they conflict with one another in specific situations. Thus, the challenge here is not only about accumulating more factual knowledge but also about understanding and adapting to evolving societal values and norms.

Definition 4 (Semantics). *In input/output logic, the main semantic construct for normative propositions is the output operation, which represents the set of normative propositions related to a normative system N, regarding state of affairs A, namely $out(N, A)$. A normative system N denotes a set of norms (a, x) in which the body and head are Boolean terms. Let $N(A) = \{x \mid (a, x) \in N \text{ for some } a \in A\}$. In a Boolean algebra $\mathcal{B}$, for $X \subseteq Ter(B)$, the set $Eq(X) = \{x \in Ter(B) | \exists y \in X, x = y\}$ is defined.*[3] *A set V is saturated in a Boolean algebra $\mathcal{B}$ if and only if, for all elements a and b in $\mathcal{B}$, if $a \in V$ and $b \geq a$, then $b \in V$, and if $a \vee b \in V$, then $a \in V$ or $b \in V$. Given a Boolean algebra $\mathcal{B}$, a normative system $N \subseteq Ter(B) \times Ter(B)$ and an input set $A \subseteq Ter(B)$, I/O Boolean operations are defined as follows:*

Zero Boolean I/O operation:

$$out_0^{\mathcal{B}}(N, A) = Eq(N(Eq(A)))$$

$$out_R^{\mathcal{B}}(N, A) = Eq(N(A)) \;\; out_L^{\mathcal{B}}(N, A) = N(Eq(A))$$

Simple-I Boolean I/O operation:

$$out_I^{\mathcal{B}}(N, A) = Eq(N(Up(A)))$$

Simple-II Boolean I/O operation:

$$out_{II}^{\mathcal{B}}(N, A) = Up(N(Eq(A)))$$

[3]Sometimes $Up(a, b, ...)(Eq(a, b, ...))$ is written instead of $Up(\{a, b, ...\})(Eq(\{a, b, ...\}))$ and $out(N, a)$ $(derive(N, a))$ is written instead of $out(N, \{a\})$ $(derive(N, \{a\}))$.

Simple-minded Boolean I/O operation:

$$out_1^{\mathcal{B}}(N, A) = Up(N(Up(A)))$$

Basic Boolean I/O operation:

$$out_2^{\mathcal{B}}(N, A) = \bigcap\{Up(N(V)), A \subseteq V, V \textit{is saturated}\}$$

Reusable Boolean I/O operation:

$$out_3^{\mathcal{B}}(N, A) = \bigcap\{Up(N(V)), A \subseteq V = Up(V) \supseteq N(V)\}$$

Put $out_i^{\mathcal{B}}(N) = \{(A, x) : x \in out_i^{\mathcal{B}}(N, A)\}$.

We turn to the proof theory. A derivation of a pair (a, x) from N, given a set X of rules, is understood to be a tree with (a, x) at the root, each non-leaf node related to its immediate parents by the inverse of a rule in X, and each leaf node an element of N.

Definition 5 (Proof system). *Given a Boolean algebra* $\mathcal{B}$ *and a normative system* $N \subseteq Ter(B) \times Ter(B)$*, it is defined that* $(a, x) \in derive_i^{\mathcal{B}}(N)$ *if and only if* (a, x) *is derivable from* N *using* EQI, EQO, SI, WO, OR, T *as follows:*[4]

$\text{derive}_i^{\mathcal{B}}$	*Rules*
$\text{derive}_R^{\mathcal{B}}$	$\{EQO\}$
$\text{derive}_L^{\mathcal{B}}$	$\{EQI\}$
$\text{derive}_0^{\mathcal{B}}$	$\{EQI,\ EQO\}$
$\text{derive}_I^{\mathcal{B}}$	$\{SI,\ EQO\}$
$\text{derive}_{II}^{\mathcal{B}}$	$\{WO,\ EQI\}$
$\text{derive}_1^{\mathcal{B}}$	$\{SI,\ WO\}$
$\text{derive}_2^{\mathcal{B}}$	$\{SI,\ WO,\ OR\}$
$\text{derive}_3^{\mathcal{B}}$	$\{SI,\ WO,\ T\}$

$$EQO\ \frac{(a, x) \qquad x =_{\mathcal{B}} y}{(a, y)} \qquad T\ \frac{(a, x) \qquad (x, y)}{(a, y)}$$

$$EQI\ \frac{(a, x) \qquad a =_{\mathcal{B}} b}{(b, x)} \qquad OR\ \frac{(a, x) \qquad (b, x)}{(a \vee b, x)}$$

$$SI\ \frac{(a, x) \qquad b \leq a}{(b, x)} \qquad WO\ \frac{(a, x) \qquad x \leq y}{(a, y)}$$

Given a set of $A \subseteq Ter(B)$*, then* $(A, x) \in derive_i^{\mathcal{B}}(N)$ *whenever* $(a, x) \in derive_i^{\mathcal{B}}(N)$ *for some* $a \in A$*. Put* $derive_i^{\mathcal{B}}(N, A) = \{x : (A, x) \in derive_i^{\mathcal{B}}(N)\}$.

[4]EQI stands for equivalence of the input, EQO for equivalence of the output, and T for transitivity.

Theorem 1 (Soundness and completeness). $out_i^{\mathcal{B}}(N) = derive_i^{\mathcal{B}}(N)$.

Example 2. *For the conditionals* $N = \{(1, a), (a, x)\}$ *and the input set* $A = \{\}$ *we have* $out_I^{\mathcal{B}}(N, A) = \{\}$*, and for the input set* $A = \{a\}$ *we have* $out_{II}^{\mathcal{B}}(N, A) = Up(x)$.

Example 3. *For the conditionals* $N = \{(a, x), (b, x), (x, y)\}$ *and the input set* $A = \{a \vee b\}$ *we have* $out_1^{\mathcal{B}}(N, A) = \{\}$*,* $out_2^{\mathcal{B}}(N, A) = Up(x)$*, and* $out_3^{\mathcal{B}}(N, A) = Up(x, y)$*. Moreover, for the input set* $A = \{a, b\}$ *we have* $out_1^{\mathcal{B}}(N, A) = out_2^{\mathcal{B}}(N, A) = Up(x)$.

Example 4. *For the conditionals* $N = \{(1, a), (a, x), (\neg a, \neg x), (b, y)\}$ *and the input set* $A = \{\neg a\}$*, since* $V = Up(\neg a, \neg x, a)$ *is the smallest set such that* $A \subseteq V$ *and* $V \supseteq N(V)$*, we have* $out_3^{\mathcal{B}}(N, A) = Up(a, x, \neg x)$*. Regarding the proof system, the following proof tree demonstrates why for instance* $a \in derive_3^{\mathcal{B}}(N, A)$.

$$WO\,\dfrac{\dfrac{(\neg a, \neg x) \qquad \neg x \leq 1}{T\,\dfrac{(\neg a, 1) \qquad\qquad (1, a)}{(\neg a, a)}}}{}$$

Uncertainty in inputs and outputs. The uncertainty here is not necessary related to norm conflict. For example, for the normative set $N = \{(a, x), (b, x), (a \wedge b, \neg x)\}$, the output of simple-minded in the original input/output operation for the input set $A = \{a, b\}$ is $out_1(N, A) = Cn(N(Cn(A))) = Cn(x, \neg x)$. In contrast, in the version defined in this paper, since the input/output operation does not join the input data, the output is $Up(N(Up(A))) = Up(x)$, indicating that there is no conflict in the output. The uncertainty here can be related to both inputs and outputs in both given intuition. For instance, non-adjunctive input handling can be used to handle uncertainty in situations where there are conflicting perspectives or opinions on a given issue. For example, in the normative systems $N = \{(a, x), (\neg a, x), (\bot, \bot)\}$ and for the input set $A = \{a, \neg a\}$, the output is $Up(x)$. The non-axiomatizability of output or normative propositions like our obligations and permissions is motivated as a result of normative uncertainty and moral insights.[5] In the context of artificial intelligence and AI alignment, this idea

[5]The non-axiomatizability of the Up operator and its application to moral insights was pointed out by Bas van Fraassen to me in his blog post. I would like to express my gratitude to him for this contribution. See: Deontic Logic and Consequence Relations

of normative uncertainty can have significant implications. A machine that assumes it has a complete understanding of objective moral truth will pursue it single-mindedly, ignoring any objections or concerns from humans. On the other hand, a machine that is uncertain about the true moral objective will exhibit a kind of humility and may defer to human preferences in its actions [1, 44]. This allows for a greater consideration of multiple viewpoints and the possibility of conflicting moral principles. Input/output operations consider both normative uncertainty (uncertainty about value itself) and empirical uncertainty (uncertainty about the empirical world) in a comprehensive manner [45].

The inference rules SI, WO, OR, AND, and CT (see Defintion 6) are motivated in the original system [31]. Here there are additional practical applications that motivate the use of the inference rules T, EQO, and EQI. The transitivity rule (T) is fundamental to logical reasoning, enabling agents to derive conclusions from chains of implications. The T rule, which facilitates logical connections between implications, is not without criticism in nonmonotonic reasoning contexts. Such critiques often advocate for the use of CT over T, as underscored by cases like Pearl's penguin conundrum [46]. In contrast, the CT rule necessitates aggregating premises, making T more streamlined in contexts where viewpoint aggregation isn't preferred. For example, from the premises $(a, \neg a)$ and $(\bot, \bot)$, one can infer $(a, \bot)$ using the CT rule. However, this derivation might be less compelling when aggregating a and $\neg a$ as distinct viewpoints. In the context of non-monotonicity, Section 5 integrates preference relations into normative reasoning. Moving on to the rule EQO, it allows for substitutions between logically equivalent terms within an inference. EQO addresses a distinct aspect of logical reasoning: maintaining coherence when terms are logically equivalent but not identical. However, it's essential to recognize that EQO is implied by the more comprehensive WO rule in I/O logic. Thus, the rationale for adopting EQO should articulate its relevance when WO is not desirable or required [47]. Similarly, the inference rule EQI, although weaker than SI, has its own usefulness, especially when the broader implications of SI are not required or could lead to undesired conclusions [48]. Regarding the introduction of new "derive" rules, EQO and T indeed appear in the text but are not entirely novel. T rule has a brief mention in Makinson and van der Torre [31], while EQO has been discussed in literature such as Straßer, Beirlaen, and van de Putte's work [49], and Parent and van der Torre's paper [50].

3. Obligatory norms: input/output operations

This section adds the AND and cumulative transitivity (CT) rules to the derivation systems introduced, with the aim of recovering the derivation systems introduced by Makinson and van der Torre [31] for deriving obligations.

Definition 6 (Proof system). *Given a Boolean algebra $\mathcal{B}$ and a normative system $N \subseteq Ter(B) \times Ter(B)$, it is defined that $(a, x) \in derive_i^X(N)$ if and only if (a, x) is derivable from N using $EQO, EQI, SI, WO, OR, AND, CT$ as follows:*

derive_i^X	*Rules*
derive_{II}^{AND}	$\{WO,\ EQI,\ AND\}$
derive_{1}^{AND}	$\{SI,\ WO,\ AND\}$
derive_{2}^{AND}	$\{SI,\ WO,\ OR,\ AND\}$
derive_{I}^{CT}	$\{SI,\ EQO,\ CT\}$
derive_{II}^{CT}	$\{WO,\ EQI,\ CT\}$
derive_{1}^{CT}	$\{SI,\ WO,\ CT\}$
$\text{derive}_{1}^{CT,AND}$	$\{SI,\ WO,\ CT,\ AND\}$

$$AND\ \frac{(a, x) \qquad (a, y)}{(a, x \wedge y)}$$

$$CT\ \frac{(a, x) \qquad (a \wedge x, y)}{(a, y)}$$

Given a set of $A \subseteq Ter(B)$, $(A, x) \in derive_i^X(N)$ whenever $(a, x) \in derive_i^X(N)$ for some $a \in A$. Put $derive_i^X(N, A) = \{x : (A, x) \in derive_i^X(N)\}$.

Makinson and van der Torre [31] noticed that in some cases, the order of application of two derivation rules is *reversible*. For instance, any application of AND followed by WO (SI) may be replaced by one in which WO (SI) is followed by AND. Based on this observation, new output operations are defined by, for example, rearranging the derivation (a, x) in the proof system $\{SI, WO, AND\}$ such that the AND rule applies only at the end. The $\{SI, WO\}$ system has been characterized as the *simple-minded I/O operation* $out_1^{\mathcal{B}}$. Now by applying (finite) successive rounds of AND on top of $out_1^{\mathcal{B}}$, a new output operation is defined that characterizes the proof system $\{SI, WO, AND\}$. Three kinds of such output operations are defined—out_i^{AND}, out_i^{CT}, and $out_i^{CT,AND}$ —that can characterize the proof systems introduced in Definition 6. Note that there are some non-reversible orders, such as the WO rule followed by the OR rule, for which no transformation appears to be available.

Definition 7 (Semantics out_i^{AND}). *Given a Boolean algebra $\mathcal{B}$, a normative system $N \subseteq Ter(B) \times Ter(B)$ and an input set $A \subseteq Ter(B)$, the AND operation is defined as follows:*

$$\begin{array}{ll} out_i^{AND^0}(N,A) & = out_i^{\mathcal{B}}(N,A) \\ out_i^{AND^{n+1}}(N,A) & = out_i^{AND^n}(N,A) \cup \\ & \{y \wedge z : y,z \in out_i^{AND^n}(N,\{a\}),\ \ a \in A\} \\ out_i^{AND}(N,A) & = \bigcup_{n\in N} out_i^{AND^n}(N,A) \end{array}$$

Put $out_i^{AND}(N) = \{(A,x) : x \in out_i^{AND}(N,A)\}$.

Definition 8 (Semantics out_i^{CT}). *Given a Boolean algebra $\mathcal{B}$, a normative system $N \subseteq Ter(B) \times Ter(B)$ and an input set $A \subseteq Ter(B)$, the CT operation is defined as follows:*

$$\begin{array}{ll} out_i^{CT^0}(N,A) & = out_i^{\mathcal{B}}(N,A) \\ out_i^{CT^{n+1}}(N,A) & = out_i^{CT^n}(N,A) \cup \\ & \{x : y \in out_i^{CT^n}(N,\{a\}) \, and \, x \in out_i^{\mathcal{B}}(N,\{a \wedge y\}),\ \ a \in A\} \\ out_i^{CT}(N,A) & = \bigcup_{n\in N} out_i^{CT^n}(N,A) \end{array}$$

Put $out_i^{CT}(N) = \{(A,x) : x \in out_i^{CT}(N,A)\}$.

Example 5. *For the conditionals $N = \{(a,b),(a,c),(a \wedge b \wedge c,d)\}$ and the input set $A = \{a\}$ we have:*

- *$out_1^{\mathcal{B}}(N,A) = Up(b,c)$ and therefore $out_1^{CT^0}(N,A) = Up(b,c)$.*
- *Similarly, $out_1^{CT^0}(N,a \wedge b) = out_1^{CT^0}(N,a \wedge c) = out_1^{CT^0}(N,A)$.*
- *Now, $out_1^{CT^1}(N,a) = Up(b,c) \cup out_1^{\mathcal{B}}(N,a \wedge b) \cup out_1^{\mathcal{B}}(N,a \wedge c) = Up(b,c)$.*
- *Also, $out_1^{CT^2}(N,a) = Up(b,c) \cup out_1^{\mathcal{B}}(N,a \wedge b) \cup out_1^{\mathcal{B}}(N,a \wedge c) = Up(b,c)$.*
- *So, $out_1^{CT}(N,a) = Up(b,c)$.*

Definition 9 (Semantics $out_i^{CT,AND}$). *Given a Boolean algebra $\mathcal{B}$, a normative system $N \subseteq Ter(B) \times Ter(B)$ and an input set $A \subseteq Ter(B)$, the CT,AND operation is defined as follows:*

$$\begin{array}{ll} out_i^{CT,AND^0}(N,A) & = out_i^{CT}(N,A) \\ out_i^{CT,AND^{n+1}}(N,A) & = out_i^{CT,AND^n}(N,A) \cup \\ & \{y \wedge z : y,z \in out_i^{CT,AND^n}(N,\{a\}),\ \ a \in A\} \\ out_i^{CT,AND}(N,A) & = \bigcup_{n\in N} out_i^{CT,AND^n}(N,A) \end{array}$$

Put $out_i^{CT,AND}(N) = \{(A,x) : x \in out_i^{CT,AND}(N,A)\}$.

Theorem 2. *Given a Boolean algebra $\mathcal{B}$, for every normative system $N \subseteq Ter(B) \times Ter(B)$ we have $out_i^{AND}(N) = derive_i^{AND}(N)$, $i \in \{II, 1, 2\}$; $out_i^{CT}(N) = derive_i^{CT}(N)$, $i \in \{I, II, 1\}$; and $out_1^{CT,AND}(N) = derive_1^{CT,AND}(N)$.*

Similarly, it is possible to define the $out_i^{OR}(N)$ operation and characterize some other proof systems:

$derive_i^X$	Rules
$derive_I^{OR}$	{SI, EQO, OR}
$derive_I^{CT,OR}$	{SI, EQO, CT, OR}
$derive_1^{CT,OR}$	{SI, WO, CT, OR}
$derive_1^{CT,OR,AND}$	{SI, WO, CT, OR, AND}

Makinson and van der Torre [31] introduced four I/O systems, based on the same inference rules used in $derive_1^{AND}$, $derive_2^{AND}$ (or $derive_1^{OR,AND}$), $derive_1^{CT,AND}$, and $derive_1^{CT,OR,AND}$, for reasoning about obligatory norms. These systems also include a rule for deriving tautologies. It is important to note that the derivation systems defined here, $derive(N, A)$ for a non-singleton set A, differ from the original system presented in their work [31]. In their work, $derive(N, A)$ is for some conjunction ($\bigwedge a_i$) of elements in A. In our system, we do not reason conjunctively with inputs:

$$derive(N, A) = \{x : (a, x) \in derive(N) \text{ for some } a \in A\}$$

Asymmetry of inputs and throughputs. In this paper, an asymmetry is introduced in the handling of inputs and throughputs (i.e., processed inputs) that does not exist in the original systems. For example, consider the original handling of a premise set $N = \{(\top, a), (\top, c), (a \wedge c, b)\}$ and $A = \{\top\}$, and $N' = \{(a \wedge c, b)\}$ with $A = \{a, c\}$. In both cases, b will be an output in the original reusable I/O system, since whether a and c are throughputs or direct inputs does not affect the triggering of $(a \wedge c, b)$. In contrast, in the approach presented in this paper (with conjunction), i.e., $derive_1^{CT,AND}$, the two cases are treated differently because throughputs are closed under conjunction, but inputs are not. In this approach, the handling of inputs and throughputs is asymmetrical, meaning that inputs and throughputs are treated differently. This asymmetry allows the approach to consider the potential consequences of different actions or events in a more systematic way. For example, if a and

c are throughputs (i.e., they have already been processed by the system), they will be closed under conjunction, meaning that they will be combined or linked together. In contrast, if a and c are inputs (i.e., they have not yet been processed by the system), they will not be closed under conjunction and will be treated separately. This asymmetry allows the approach to consider the potential consequences of different actions or events and to make more informed decisions about which actions or events are most appropriate in a given situation. Suppose we have an AI system that has been programmed to provide users with information they request. The inputs in this case could be various requests from users, say, user A asks for the fastest route to a location, while user B requests the most scenic route to the same location. The system will treat these inputs separately and will not combine them, since they reflect distinct individual preferences. On the other hand, consider the outputs or "throughputs" of this system which are the recommendations it generates based on its normative guidelines and the user's request. Let's say these guidelines include "respecting user's privacy" and "promoting environmental sustainability." When user A asks for the fastest route, the system generates an output, while also considering the throughput of "respecting user's privacy" - it doesn't share user A's location with other users. When user B asks for the most scenic route, it generates another output while considering the throughput of "promoting environmental sustainability" - it selects a route that involves less carbon emission. In this situation, these throughputs are closed under conjunction. This means that both of these principles can coexist in the system's operations and both are considered together in generating future outputs, thus demonstrating the conjunction of norms within the system's operation. This approach of treating inputs and throughputs differently (asymmetrically) allows the AI system to balance respecting individual user requests and overarching ethical norms. This asymmetry represents a philosophical approach to AI alignment, where balancing individual preferences entails evaluating disparate actions as distinct perspectives rather than amalgamating them, in alignment with community or societal norms.

4. Input/output operations over abstract logics

An abstract logic [14] is a pair $\mathcal{A} = \langle \mathcal{L}, C \rangle$ where $\mathcal{L} = \langle L, ... \rangle$ is an algebra and C is a closure operator, defined on the power set of its universe, that means that for all $A, B \subseteq L$:

- $A \subseteq C(A)$
- $A \subseteq B \Rightarrow C(A) \subseteq C(B)$
- $C(A) = C(C(A))$

The elements of an abstract logic can be ordered as $a \leq b$ if and only if $b \in C(\{a\})$.[6] Without loss of generality, the algebra of formulas (or terms in the algebraic context) is used where $\mathbf{Fm}(X) = \langle Fm(X), ...\rangle$ for a set of fixed variables X. Similar to Boolean algebras, the Eq and Up operators can be defined for $A \subseteq Fm(X)$.

Definition 10 (Semantics). *Given an abstract logic $\mathcal{A} = \langle \mathbf{Fm}(X), C\rangle$, a normative system $N \subseteq Fm(X) \times Fm(X)$ and an input set $A \subseteq Fm(X)$, the I/O operations are defined as follows:*

- $out_0^{\mathcal{A}}(N, A) = Eq(N(Eq(A)))$
- $out_I^{\mathcal{A}}(N, A) = Eq(N(Up(A)))$
- $out_{II}^{\mathcal{A}}(N, A) = Up(N(Eq(A)))$
- $out_1^{\mathcal{A}}(N, A) = Up(N(Up(A)))$
- $out_2^{\mathcal{A}}(N, A) = \bigcap\{Up(N(V)), A \subseteq V, V \textit{is saturated}\}$[7]
- $out_3^{\mathcal{A}}(N, A) = \bigcap\{Up(N(V)), A \subseteq V = Up(V) \supseteq N(V)\}$

Put $out_i^{\mathcal{A}}(N) = \{(A, x) : x \in out_i^{\mathcal{A}}(N, A)\}$.

Definition 11 (Proof system). *Given an abstract logic $\mathcal{A} = \langle \mathbf{Fm}(X), C\rangle$ and a normative system $N \subseteq Fm(X) \times Fm(X)$, it is defined that $(a, x) \in derive_i^{\mathcal{A}}(N)$ if and only if (a, x) is derivable from N using the rules $\{EQI, EQO\}$, $\{SI, EQO\}$, $\{WO, EQI\}$, $\{SI, WO\}$, $\{SI, WO, OR\}$ and $\{SI, WO, T\}$ for $i \in \{0, I, II, 1, 2, 3\}$ in turn. Given a set of $A \subseteq Fm(X)$, $(A, x) \in derive_i^{\mathcal{A}}(N)$ whenever $(a, x) \in derive_i^{\mathcal{A}}(N)$ for some $a \in A$. Put $derive_i^{\mathcal{A}}(N, A) = \{x : (A, x) \in derive_i^{\mathcal{A}}(N)\}$.*

Theorem 3 (Soundness and completeness). $out_i^{\mathcal{A}}(N) = derive_i^{\mathcal{A}}(N)$.

A logical system $\mathbf{L} = \langle L, \vdash_{\mathbf{L}}\rangle$ straightforwardly provides an equivalent abstract logic $\langle \mathbf{Fm}_L, C_{\vdash_L}\rangle$. Therefore, an I/O framework can be built over

[6] $a =_{\mathcal{A}} b$ if and only if $a \leq b$ and $b \leq a$.

[7] For this case, the abstract logic $\mathcal{A} = \langle \mathbf{Fm}(X), C\rangle$ should include $\vee$, that is a binary operation symbol, either primitive or defined by a term, and we then have $a \vee b, b \vee a \in C(\{a\})$ ($\vee$-Introduction) and if $c \in C(\{a\}) \cap C(\{b\})$ then $c \in C(a \vee b)$ and $c \in C(b \vee a)$ ($\vee$-Elimination).

different types of logics including first-order logic, simple type theory, description logic, as well as different kinds of modal logics that are expressive for intentional concepts such as belief and time.

Example 6. *In a modal logic system KT, for the conditionals* $N = \{(p, \Box q), (q, r), (s, t)\}$ *and the input set* $A = \{p\}$*, we have* $out_3^{KT}(N, A) = Up(\Box q, r)$*. The reflexivity axiom (T),* $\Box q \rightarrow q$*, guarantees that* r *can be detached, as* $V = Up(p, \Box q, r)$ *is the smallest set that satisfies both* $A \subseteq V$ *and* $V \supseteq N(V)$*.*

Moreover, other rules such as AND and CT can be added to the systems in the same way as in Section 3.

Theorem 4. *For a normative system* N*, every* $out_i^{\mathcal{B}}(N)$*, and* $out_i^{\mathcal{A}}(N)$ *operation is a closure operator.*

Nested input/output operations. Based on Theorem 4 and the results of building input/output operations on top of any abstract logic, it is possible to define nested input/output (I/O) operations. For any $N \subseteq Ter(B) \times Ter(B)$ and $M \subseteq (Ter(B) \times Ter(B)) \times (Ter(B) \times Ter(B))$, the operation $out_j^{\mathcal{A}}(M, out_i^{\mathcal{B}}(N))$ can be defined because $out_i^{\mathcal{B}}$ is a closure operator and in the abstract logic $\mathcal{A}$ we can take $L = N \times N$ and $C = out_i^{\mathcal{B}}$. In the abstract logic $\mathcal{A}$, this operation corresponds to $derive_j^{\mathcal{A}}(M, derive_i^{\mathcal{B}}(N))$. Similarly, it is possible to define nested operations of the form $out_j^{\mathcal{A}}(M, out_i^{\mathcal{A}}(N))$ for the abstract logic $\mathcal{A}$. Nested input/output (I/O) operations can be useful for combining regulative and constitutive norms [51].

Flexibility in AI alignment through input/output operations. In the sphere of AI alignment, creating a versatile framework of input/output operations applicable across various logical systems offers the distinct advantage of flexibility. This model moves away from the conventional notion of restricting AI systems to a single logical system and instead propounds a more adaptive and context-driven approach. Under this framework, the decision-making process of an AI system is no longer dictated by a fixed logical system. Instead, it can tap into a multitude of logical systems to better navigate the complexity and nuances of a given situation. For instance, in an ethical dilemma, the AI can leverage classical logic for its strong deductive power, or intuitionistic logic when there is incomplete information, each bringing a different perspective to the problem at hand. This open-ended structure paves the way for a more holistic decision-making process. Depending on the scenario,

an AI system can integrate diverse logical systems, whether it's dealing with moral questions, making risk assessments, or solving optimization problems. This approach seeks to expand the breadth of the AI's decision-making capabilities, aiming to produce decisions that are more robust and considered. Moreover, this model allows the AI system to adjust its logic based on the context and nature of the decision-making scenario. For example, in the fast-paced world of cybersecurity, the system could leverage classical logic to enforce stringent security measures but might switch to probabilistic logic when assessing the risk of potential threats. However, it's essential to address the potential complexities and uncertainties, especially when multiple logics might be applicable to a given situation. In conclusion, fostering an adaptable input/output operations model in AI alignment lays the groundwork for a dynamic decision-making engine. It accommodates a diverse range of logical systems, enhancing the AI system's flexibility, adaptability, and responsiveness to evolving scenarios. The system's capacity to adjust its logic according to the changing circumstances becomes its key strength, making it an indispensable tool in complex, evolving contexts.

5. Synthesizing normative reasoning and preferences

Input/output logic was originally developed on top of classical propositional logic [31]. This section demonstrates that the extension of propositional logic with a set of conditional norms is both sound and complete in relation to the class of Boolean algebras where the corresponding input/output operation is valid. The language of classical propositional logic consists of the connectives $\mathcal{L}_C = \{\wedge, \vee, \neg, \top, \bot\}$. Let X be a set of variables; as usual the set of formulas is defined over X and referred to as $Fm(X)$.[8] The algebra of formulas over X is a Boolean algebra as follows:

$$\mathbf{Fm}(X) = \langle Fm(X), \wedge^{\mathbf{Fm}(X)}, \vee^{\mathbf{Fm}(X)}, \neg^{\mathbf{Fm}(X)}, \top^{\mathbf{Fm}(X)}, \bot^{\mathbf{Fm}(X)} \rangle$$

where $\wedge^{\mathbf{Fm}(X)}(\varphi, \psi) = (\varphi \wedge \psi)$, $\vee^{\mathbf{Fm}(X)}(\varphi, \psi) = (\varphi \vee \psi)$, $\neg^{\mathbf{Fm}(X)}(\varphi) = \neg\varphi$, $\top^{\mathbf{Fm}(X)} = \top$, and $\bot^{\mathbf{Fm}(X)} = \bot$. Let $\varphi \vdash_C \psi$ if and only if $\varphi \leq \psi$, and $\varphi \dashv\vdash_C \phi$ if and only if $\varphi \leq \psi$ and $\psi \leq \varphi$. $\Gamma \vdash_C \psi$ if and only if there is a finite set $\{\gamma_1, \ldots, \gamma_n\} \subseteq \Gamma$ for which $(\gamma_1 \wedge \gamma_2 \wedge \ldots \wedge \gamma_n) \vdash_C \psi$.

[8] For the precise definition, the auxiliary symbols brackets), (are used. Apart from the use of brackets, the formulas over X are Boolean terms over X: $Ter(X)$.

Definition 12. *Let $N \subseteq Fm(X) \times Fm(X)$ where X is a set of propositional variables. It is defined that $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ if and only if (φ, ψ) is derivable from N using EQO, EQI, SI, WO, OR, T as follows:*

$\text{derive}_i^{\mathbf{Fm}(X)}$	*Rules*
$\text{derive}_R^{\mathbf{Fm}(X)}$	$\{EQO\}$
$\text{derive}_L^{\mathbf{Fm}(X)}$	$\{EQI\}$
$\text{derive}_0^{\mathbf{Fm}(X)}$	$\{EQI,\ EQO\}$
$\text{derive}_I^{\mathbf{Fm}(X)}$	$\{SI,\ EQO\}$
$\text{derive}_{II}^{\mathbf{Fm}(X)}$	$\{WO,\ EQI\}$
$\text{derive}_1^{\mathbf{Fm}(X)}$	$\{SI,\ WO\}$
$\text{derive}_2^{\mathbf{Fm}(X)}$	$\{SI,\ WO,\ OR\}$
$\text{derive}_3^{\mathbf{Fm}(X)}$	$\{SI,\ WO,\ T\}$

$$EQO\ \frac{(\varphi,\psi) \qquad \psi \dashv\vdash_C \phi}{(\varphi,\phi)} \qquad T\ \frac{(\varphi,\psi) \qquad (\psi,\phi)}{(\varphi,\phi)}$$

$$EQI\ \frac{(\varphi,\psi) \qquad \varphi \dashv\vdash_C \phi}{(\phi,\psi)} \qquad OR\ \frac{(\varphi,\psi) \qquad (\phi,\psi)}{(\varphi \vee \phi,\psi)}$$

$$SI\ \frac{(\varphi,\psi) \qquad \phi \vdash_C \varphi}{(\phi,\psi)} \qquad WO\ \frac{(\varphi,\psi) \qquad \psi \vdash_C \phi}{(\varphi,\phi)}$$

It is defined that $(\Gamma, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ if $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ for some $\varphi \in \Gamma \subseteq Fm(X)$. Put $derive_i^{\mathbf{Fm}(X)}(N, \Gamma) = \{\psi : (\Gamma, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)\}$.

Example 7. *For the conditionals $N = \{(\top, \varphi), (\varphi, \psi), (\psi, \gamma), (\gamma, \neg\varphi)\}$ and the input set $A = \{\gamma\}$, we have $out_3^{\mathbf{Fm}(X)}(N, A) = Up(\varphi, \psi, \gamma, \neg\varphi)$.*

Given $\langle \mathbf{Fm}(X), \vdash_C \rangle$, let $\mathcal{B}$ be a Boolean algebra and X be a set of propositional variables. A valuation on $\mathcal{B}$ is a function from X into the universe of $\mathcal{B}$. Any valuation on $\mathcal{B}$ can be extended in a unique way to a homomorphism from the algebra $\mathbf{Fm}(X)$ into $\mathcal{B}$. A valuation V on $\mathcal{B}$ satisfies a formula φ if $V(\varphi) = 1_{\mathcal{B}}$, and it satisfies a set of formulas Γ if $V(\gamma) = 1_{\mathcal{B}}$ for all $\gamma \in \Gamma$ [52].

Definition 13. *For any Boolean algebra $\mathcal{B}$, the consequence relation $\models_{\mathcal{B}}$ can be defined as follows:*

$$\Gamma \models_{\mathcal{B}} \varphi \text{ if and only if for any valuation on } \mathcal{B} \text{ that } V(\Gamma) = 1_{\mathcal{B}} \text{ then } V(\varphi) = 1_{\mathcal{B}}.$$

Definition 14. *Let* **BA** *be the class of all Boolean algebras. The consequence relation $\models_{\mathbf{BA}}$ can be defined as follows:*

$$\Gamma \models_{\mathbf{BA}} \varphi \text{ if and only if for any Boolean algebra } \mathcal{B},\ \Gamma \models_{\mathcal{B}} \varphi.$$

Theorem 5. *For every set of formulas Γ and every formula φ,*

$$\Gamma \vDash_{\mathbf{BA}} \varphi \text{ if and only if } \Gamma \vdash_C \varphi.$$

Theorem 6. *Let X be a set of propositional variables and $N \subseteq Fm(X) \times Fm(X)$. For a given Boolean algebra $\mathcal{B}$ and a valuation V on $\mathcal{B}$, it is defined that $N^V = \{(V(\varphi), V(\psi)) | (\varphi, \psi) \in N\}$. We have*

$$(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$$

if and only if

$$V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\}) \text{ for every } \mathcal{B} \in \mathbf{BA} \text{ and valuation } V.$$

The theorem can be extended for arbitrary input sets $\Gamma \subseteq Fm(X)$. Suppose that $(\Gamma, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$, then $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ for $\varphi \in \Gamma$. As above, we have $V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$ for every $\mathcal{B} \in \mathbf{BA}$ and valuation V, so that by definition of $out_i^{\mathcal{B}}$, it can be said that $V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi) | V(\varphi) \in V(\Gamma)\})$ for every $\mathcal{B} \in \mathbf{BA}$ and valuation V.

Theorem 7. *Let X be a set of propositional variables and $N \subseteq Fm(X) \times Fm(X)$. For a given Boolean algebra $\mathcal{B}$ and a valuation V on $\mathcal{B}$, it is defined that $N^V = \{(V(\varphi), V(\psi)) | (\varphi, \psi) \in N\}$. We have*

$$(\varphi, \psi) \in derive_i^{AND}(N)$$

if and only if

$$V(\psi) \in out_i^{AND}(N^V, \{V(\varphi)\}) \text{ for every } \mathcal{B} \in \mathbf{BA} \text{ and valuation } V.$$

5.1. Consistency check

Constraints can be added to the derivation systems such that the output set of formulas is consistent with the proposed constraint. For example, one constraint that could be added to the derivation system in the context of AI alignment is a constraint that ensures that the AI system only generates output formulas that are consistent with certain moral principles, such as the principle of non-maleficence or the principle of autonomy. By adding such a constraint, the AI system can be made to prioritize the importance of these

principles in its decision-making processes, and thus helps to align the AI system with human moral values.

Definition 15. *Let X be a set of propositional variables and $N \subseteq Fm(X) \times Fm(X)$. Given the constraint Con that is a set of formulas $Con \subseteq Fm(X)$, it is defined that $(\varphi, \psi) \in derive_i^{Con}(N)$ if and only if*

$$(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N) \text{ and } Con, \psi \nvdash_C \bot.$$

Given a set of $\Gamma \subseteq Fm(X)$, it is defined that $(\Gamma, \psi) \in derive_i^{Con}(N)$ if $(\varphi, \psi) \in derive_i^{Con}(N)$ for some $\varphi \in \Gamma$.

Theorem 8. *Let X be a set of propositional variables, $N \subseteq Fm(X) \times Fm(X)$, and $Con \subseteq Fm(X)$. For a given Boolean algebra $\mathcal{B}$ and a valuation V on $\mathcal{B}$, it is defined that $N^V = \{(V(\varphi), V(\psi)) | (\varphi, \psi) \in N\}$. We have*

$$(\varphi, \psi) \in derive_i^{Con}(N)$$

if and only if

$V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$ *for every* $\mathcal{B} \in \mathbf{BA}$ *and valuation* V

and

for some $\mathcal{B} \in \mathbf{BA}$*, there is a valuation* V *such that* $\forall \delta \in Con, V(\delta \wedge \psi) = 1_{\mathcal{B}}$.

Reducing norm sets. The idea of pruning the set of norms is explored in constrained input/output logic [34]. The concept involves reducing the set of norms to a level just below the point where it becomes excessive, and examining the outcome that follows. This is accomplished by identifying the largest subsets of norms, denoted as $N' \subseteq N$, where the output $out(N', A)$ remains consistent. These subsets are referred to as the maxfamily of A, while the corresponding outputs $out(N', A)$ form the outfamily of A. For example, consider the conditionals $N = \{(\top, \varphi), (\neg\varphi, \psi), (\varphi, \neg\psi)\}$. The maxfamily of $A = \{\neg\varphi\}$ in this case is $\{\{(\top, \varphi), (\neg\varphi, \psi)\}, \{(\top, \varphi), (\varphi, \neg\psi)\}, \{(\neg\varphi, \psi), (\varphi, \neg\psi)\}\}$, while the outfamily of A is $\{Cn(\varphi, \psi), Cn(\varphi, \neg\psi), Cn(\psi)\}$. Therefore, employing a skeptical approach, we can derive $Cn(\varphi \vee \psi)$, and adopting a credulous stance leads us to derive $Cn(\bot)$ [35]. In our approach, when $Con = \{\}$,

we have $derive_i^{Con}(N) = derive_i^{\mathbf{Fm}(X)}(N)$ in this example. Although this approach might appear simpler than constrained input/output logic, it aims to lessen the focus on the intricate interactions of norms. The constraints here are pruning the set of norms based on the valuation functions and their associated models. This reduction paves the way for introducing more intricate concepts, such as preferences, into the I/O logic framework in the subsequent sections.

5.2. Preferences: an AI alignment outline

In summary, the principles of AI alignment proposed with Russell [1] involve ensuring that the objectives and behaviors of artificial intelligence systems are aligned with the preferences and values of humans. This may involve considering the uncertainty that AI systems may have about what those preferences and values are, and using human behavior as a guide for determining and adapting to those preferences and values over time. Human preferences can be understood in several different senses, depending on the context in which they are used [53]:

- In economic and decision-theoretic contexts, preferences typically refer to an individual's or an AI system's ranking or ordering of different alternatives or options, based on their relative desirability or utility. For example, an individual might have a preference for a particular brand of coffee over others, or an AI system might have a preference for a particular course of action over others, based on the expected outcomes or consequences of each option. This preference is closely linked to uncertainty arising from value assessments.
- In moral or normative contexts, preferences may refer to an individual's or an AI system's values, goals, or principles, which guide their decision-making and determine the acceptable or desirable outcomes or actions in a given context. For example, an individual might have a preference for fairness or equality, or an AI system might have a preference for maximizing the well-being of all sentient beings, which would influence their moral decisions and actions. The preference described is closely associated with normative uncertainty.

In both of these senses, preferences are subjective and personal, and may differ from one individual or AI system to another. They are also often uncertain or incompletely known, and may change over time or in response to new information or experiences. In AI alignment, both forms of preferences

can play a role. For example, an AI system may need to consider the preferences of individual users or stakeholders in order to make decisions that are aligned with their goals or values. At the same time, the AI system may also need to consider moral or normative considerations, such as the potential consequences of its actions for society as a whole, or the ethical implications of different courses of action. In such cases, the AI system may need to use both forms of human preferences to make informed and ethical decisions.

In this paper, preferences are used in two different senses and formal settings. The first sense is represented by utility functions in the decision-theoretic framework or preference relations in the logical and non-monotonic setting. These preferences are used to evaluate the desirability of different outcomes or actions based on their expected utility. The second sense of preferences is represented by a set of conditional norms, which are used to guide moral and normative decision-making in situations where there is uncertainty about the relevant norms and values or about the likely consequences of different actions or choices.

In their paper "The Off-Switch Game," Dylan Hadfield-Menell, Anca Dragan, Pieter Abbeel, and Stuart Russell [54] explore the importance of incorporating uncertainty into the utility functions of AI systems. They propose a model where the AI, in a game with a human, has an off-switch that can be disabled by the AI itself. Traditional AI systems, which accept their reward functions without question, often disable the off-switch to ensure self-preservation and maximize utility. However, when the AI is uncertain about its utility function and views human actions as valuable insights into the "true" objective, it is less likely to disable the off-switch. By introducing uncertainty into the utility function, the AI system is encouraged to observe and learn from human behavior rather than instinctively pursue self-preservation. This results in safer AI designs and promotes the cooperation and co-learning of AI systems and their human counterparts.

Example 8. *Consider a healthcare robot, RoboNurse, embroiled in an "off-switch" game with a patient. RoboNurse's utility function entails objectives such as providing optimal patient care, preserving its operational status (avoiding being shut down), and aligning with patient preferences. Here, we grapple with evaluative uncertainty, specifically uncertainty due to value assessments, arising from these conflicting values. RoboNurse might confront dilemmas like:*

- *Should it always prioritize patient safety, even if this means opposing*

a patient's attempt to deactivate it?

- *Does its commitment to its own operational continuity outweigh the importance of respecting the patient's autonomy and decision to halt its functions?*
- *How should it balance the significance of adhering to patient wishes, particularly when these might result in its deactivation and thus halt its caregiving duties?*

RoboNurse faces evaluative uncertainties while balancing its utility function objectives: patient care, operational continuity, and patient preferences. It reassesses these priorities based on real-time interactions and clinical contexts. For example, while it might initially emphasize its operational continuity, RoboNurse could shift focus to respect patient autonomy, especially if a patient persistently attempts deactivation. This capability to adjust priorities in response to evolving scenarios ensures that RoboNurse makes informed, ethical decisions and maintains a harmonious interaction with patients, effectively managing care and respecting patient choices.

Utility function $\Longrightarrow$ *Preference relation.* A preference relation can be derived from a utility function by comparing utility values of various options or outcomes. For instance, a higher utility value for option A than B signifies a preference for A. While a common approach is the "more is better" assumption, where higher utility values are favored, some decision-makers might prioritize based on risk factors. It's crucial to understand that these preferences are subjective, influenced by the individual's goals, values, and specific circumstances. To make informed decisions, one should weigh consequences, uncertainties, risks, and ethical considerations of each option.

5.3. Preferences and normative reasoning

In this section, the input/output framework for normative reasoning in uncertainty is extended to include a preference relation induced for instance from a utility function. By combining normative reasoning and preference reasoning in this way, it is possible to make more informed decisions in situations where there is uncertainty about the relevant norms and values or about the likely consequences of different actions or choices. Overall, this paper demonstrates the importance of considering both normative and preference-based approaches to decision-making in the context of AI alignment.

In the normative systems proposed, it is possible to add a preference relation over the set of valuations and define a new conditional theory. Conditional obligation sentences are analyzed which have the form $\varphi \hookrightarrow \bigcirc\psi$,

where $\hookrightarrow$ is a (preferential) conditional connective [24, 55]. Given the set of obligatory norms N^O, the new conditionals are defined as follows:

$$\varphi \hookrightarrow \bigcirc\psi \text{ holds iff } (\varphi, \psi) \in derive_i(N^O) \text{ and } \varphi \hookrightarrow \psi \text{ holds}$$

where $derive_i(N^O)$ is an appropriate derivation system for obligation. Intuitively, the modal translation of $\varphi \rightarrow \bigcirc\psi$ for $(\varphi, \psi) \in derive_i(N^O)$ [12, 31] is considered, where a comprehensive theory of conditional obligation requires the integration of two distinct components [56, 57]: a theory of the conditional ($\hookrightarrow$) and a theory of obligation ($derive_i(N^O)$). This approach provides a compositional definition of monadic obligation operators and conditionals. The formula $\varphi \hookrightarrow \psi$ can be intrepreted as "if φ is the case, then ψ is the case." Similarly, the formula $\varphi \hookrightarrow \bigcirc\psi$ can be intrepreted as "if φ is the case, then ψ is obligatory." In the context of primary and secondary obligations [58], this is better suited for secondary obligations.

For a given set of permissive norms N^P, by choosing a appropriate derivation system for permission— $derive_i(N^P)$ —that is similar to the definition of a conditional obligation, then conditional permission can be defined as

$$\varphi \hookrightarrow P\psi \text{ holds iff } (\varphi, \psi) \in derive_i(N^P) \text{ and } \neg(\varphi \hookrightarrow \neg\psi) \text{ holds}$$

where $\neg(\varphi \hookrightarrow \neg\psi)$ is the conditional dual of $\varphi \hookrightarrow \psi$. The set of new conditional obligations is denoted by $derive_i^O$ and the set of new conditional permissions is denoted by $derive_i^P$. Henceforth, reference to the subscripts or superscripts of the normative system or derivation systems is omitted, whenever they are clear from the context or do not affect our discussion.

As for the comparison with Parent's work on priority/preference relations in the I/O formalism, the focus here diverges. Parent's work [59] primarily explores preferences within the I/O formalism as a method to handle conflicting obligations or permissions, utilizing priority relations directly within the I/O framework. However, in our context, preferences are derived separately and applied at a higher level of reasoning, augmenting the I/O framework rather than modifying it internally. They serve a different purpose guiding decision-making under uncertainty, rather than conflict resolution within the formalism itself. This more nuanced interpretation does not extend the I/O formalism itself, but rather complements it with an additional layer of decision-making apparatus, providing a more comprehensive tool for dealing with normative reasoning in uncertainty.

Definition 16. *Let X be a set of propositional variables and $MaxC$ the set of all the maximal consistent subsets of $Fm(X)$. Let $f \subseteq MaxC \times MaxC$ be a relation over elements of $MaxC$ and $\mathrm{opt}_f(\varphi) = \{M \in MaxC \mid \varphi \in M, \forall K\,(\varphi \in K \rightarrow (M, K) \in f)\}$. It is defined that $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$ if and only if*

$$(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N) \text{ and } \forall M \in \mathrm{opt}_f(\varphi)\,(\psi \in M).$$

Given a set of $\Gamma \subseteq Fm(X)$, it is defined that $\Gamma \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$ if $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$ for some $\varphi \in \Gamma$.

Definition 17. *Let X be a set of propositional variables and $f \subseteq MaxC \times MaxC$. A preference Boolean algebra for $\mathbf{Fm}(X)$ is a structure $\langle \mathcal{B}, \mathcal{V}, \succeq_f \rangle$ where:*

- *$\mathcal{B}$ is a Boolean algebra,*
- *$\mathcal{V} = \{V_i\}_{i \in I}$ is the set of valuations from $\mathbf{Fm}(X)$ on $\mathcal{B}$,*
- *$\succeq_f \subseteq \mathcal{V} \times \mathcal{V}$: $\succeq_f$ is a betterness or comparative goodness relation over valuations from $\mathbf{Fm}(X)$ to $\mathcal{B}$ such that $V_i \succeq_f V_j$ iff $(\{\varphi | V_i(\varphi) = 1_\mathcal{B}\}, \{\psi | V_j(\psi) = 1_\mathcal{B}\}) \in f$.*

For sake of generality, no specific properties (like reflexivity or transitivity) are considered for the betterness relation. The choice of including or excluding such properties is informed by specific contexts and objectives. For a deeper understanding of when and why these properties are deemed appropriate, see [48]. For a given preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_f \rangle$, it is defined that $opt_{\succeq_f}(\varphi) = \{V_i \in \mathcal{V} \mid V_i(\varphi) = 1_\mathcal{B}, \forall V_j (V_j(\varphi) = 1_\mathcal{B} \rightarrow V_i \succeq_f V_j)\}$. $opt_{\succeq_f}(\varphi)$ might be empty for non-reflexive relations f.

Theorem 9. *Let X be a set of propositional variables, where $N \subseteq Fm(X) \times Fm(X)$, and $f \subseteq MaxC \times MaxC$. For a given Boolean algebra $\mathcal{B}$ and a valuation V on $\mathcal{B}$, it is defined that $N^V = \{(V(\varphi), V(\psi)) | (\varphi, \psi) \in N\}$. We have*

$$\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$$

if and only if

$$V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\}) \text{ for every } \mathcal{B} \in \mathbf{BA} \text{ and valuation } V,$$

and

for every preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_f \rangle$,
for every valuation $V_i \in opt_{\succeq_f}(\varphi)$,
it is the case that $V_i(\psi) = 1_{\mathcal{B}}$.

The theorem can also be rewritten as follows:[9]

$$\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$$

if and only if

$\psi \in out_i^{\mathbf{Fm}(X)}(N, \{\varphi\})$ and in $\langle \mathbf{2}, \mathcal{V}, \succeq_f \rangle$,
for every valuation $V_i \in opt_{\succeq_f}(\varphi)$, we have $V_i(\psi) = 1_{\mathcal{B}}$.

Example 9. *In a modern healthcare facility, a robot nurse, aptly named RoboNurse, has been designed to provide patient care. Equipped with artificial intelligence, it operates based on two core conditional norms that prioritize patient safety and autonomy.*

- N_1*: Should Mr. Smith request RoboNurse's deactivation, the robot should comply to respect his autonomy:*

$$(request, deactivate)$$

- N_2*: However, if deactivation jeopardizes Mr. Smith's safety, RoboNurse should resist:*

$$(threat, \neg deactivate)$$

[9] $\mathbf{2}$ is the two-element Boolean algebra.

During one of his shifts, Mr. Smith, in a semi-coherent state, attempts to turn off RoboNurse, activating the conditions set by N_1 and N_2. The challenge RoboNurse faces is the conflicting requirements of these norms: should it prioritize autonomy and allow Mr. Smith to turn it off, or should it prioritize Mr. Smith's safety and resist the action?

Consider the scenario where RoboNurse operates under a straightforward decision-making mechanism, represented as $Up(N(Up(A)))$. In this context:

- *The input set comprises two distinct signals: $Up(request, threat)$. Here, request is activated when Mr. Smith intends to deactivate RoboNurse, while threat is triggered upon identifying a potential risk to Mr. Smith's well-being if the action is permitted.*

- *Correspondingly, the output set is defined by RoboNurse's subsequent actions: $Up(deactivate, \neg deactivate)$. The deactivate output is generated when RoboNurse accedes to the deactivation, whereas $\neg$deactivate signifies RoboNurse's decision to continue its operation in the interest of patient safety.*

In the original input/output system, $Cn(N(Cn(A)))$, uncertainties in both inputs and outputs cannot be adequately addressed. For example, we deduce $\bot \in Cn(N(Cn(request, threat))))$. This system's limitation becomes particularly evident in scenarios with normative and evaluative uncertainties.

Uncertainties encountered:

1. ***Normative uncertainty:*** *RoboNurse could face uncertainty about which normative action to take, even when the values have been clearly defined. Given that Mr. Smith is in a semi-coherent state, RoboNurse might be unsure about whether his request truly reflects his wishes, or if he might regret the decision once he's in a more lucid state. Turning itself off based on an unclear request could endanger Mr. Smith, while not turning off might be seen as overstepping and not respecting patient autonomy.*

2. ***Uncertainty due to value assessments:*** *This pertains to the challenge of weighing the value of safety against autonomy. Different stakeholders might have varied opinions on which is more important. For instance, the healthcare facility might lean more towards patient safety due to legal and ethical reasons, while a patient rights advocate might*

prioritize autonomy. This creates a dilemma for the AI, especially when its normative rules don't provide a clear hierarchy between these values.

Possible value preferences:

1. ***Safety first:*** *Prioritizes Mr. Smith's well-being over his current wishes. While this may seem paternalistic, it leans towards the hospital's ethical and legal obligations.*

   For the conditionals $N = \{(request, deactivate), (threat, \neg deactivate)\}$*, based on the safety priority, the maximal consistent sets (scenarios) can be ordered as follows: in the first type, labelled as* s_1*, there's a Threat and the patient has Requested deactivation. In the slightly less optimal scenario, labelled* s_2*, not only is there a Threat and a Request, but the Deactivation has occurred.*

$$\begin{array}{c} best \quad s_1 \bullet Threat,\ Request \\ - \\ 2nd\ best \quad s_2 \bullet Threat,\ Request,\ Deactivation \end{array}$$

   Given that $(threat, \neg deactivate) \in derive_1^{\mathbf{Fm}(X)}(N)$ *and* $f = \{(s_1, s_2)\}$*, since* $\forall M \in \mathrm{opt}_f(threat)\,(\neg deactivate \in M)$*, we have*

$$threat \hookrightarrow \bigcirc \neg deactivate \in derive_1^{O^H}(N).$$

2. ***Respect for autonomy:*** *Prioritizes Mr. Smith's immediate wishes, echoing the principle of patient self-determination, even if it might lead to potential harm.*

   Similar to the first case based on autonomy priority, we can order the maximal consistent sets as follows:

$$\begin{array}{c} best \quad s_2 \bullet Threat,\ Request,\ Deactivation \\ - \\ 2nd\ best \quad s_1 \bullet Threat,\ Request \end{array}$$

   Given that $(request, deactivate) \in derive_1^{\mathbf{Fm}(X)}(N)$ *and* $f = \{(s_2, s_1)\}$*, since* $\forall M \in \mathrm{opt}_f(request)\,(deactivate \in M)$*, we have*

$$request \hookrightarrow \bigcirc deactivate \in derive_1^{O^H}(N).$$

In this example, normative reasoning establishes ethical limits for AI, ensuring it avoids harmful actions. Meanwhile, preference-based methods let the AI adapt to user behavior, promoting personalization. After alignment, some norms might be deemed irrelevant in certain contexts, which is represented as the non-reflexivity of the input/output consequence relation. Refer to Example 10 for further details. This scenario can be examined through the lens of constrained input/output logic [34], wherein RoboNurse encounters two maximal consistent norm sets: $\{(request, deactivate)\}$ *and* $\{(threat, \neg deactivate)\}$*. Depending on whether safety or autonomy takes precedence, RoboNurse may prioritize accordingly. The constrained input/output logic approach addresses nonmonotonicity by adopting maximally consistent sets of norms and situating norms at a meta-level. This method, however, precludes the norms from being directly embedded within the object language. As a result, it curtails the system's capacity to introspectively reason about norms and, consequently, to formulate explanations (preferences) inherently within the confines of the logic system itself [60].*

This paper explores a non-monotonic reasoning approach grounded in preference-based logic [24, 48]. It is proposed that if a certain condition, φ, is present, then it would typically lead to an outcome, ψ, aligned with the current preferences. However, this inferential relationship is dynamic; the introduction of new information, ϕ, can modify the scenario. Consequently, when both φ and ϕ are considered, it is no longer certain that ψ will be the outcome. In summery, from $\varphi \hookrightarrow \psi$, it is not necessary that $\varphi \wedge \phi \hookrightarrow \psi$. This dynamic is central to the fluidity of the reasoning model being discussed, where inferences are continually adapted in light of new information. Nonetheless, it is noteworthy that the current system does not support meta-level reasoning about preferences. For instance, in the context of the RoboNurse scenario, the framework does not facilitate a shift in RoboNurse's priorities from safety to autonomy on its own. This limitation underscores a potential area for future expansion: evolving the logical systems to include mechanisms for the dynamic updating of preferences. Such advancements would markedly increase the system's adaptability and relevance in complex, real-life situations.

5.4. Incorporating preferences through premise sets and human behavior

The application of premise semantics in the pursuit of aligning AI systems with human preferences promises a rewarding path. This field, significantly

influenced by the seminal works of David Lewis and Angelika Kratzer on ordering semantics, involves the examination of the premise sets in arguments or discourses, which plays a critical role in determining the overall meaning of a text or statement [61, 62]. Language serves as a primary mode of expressing our values, goals, and aspirations, providing substantial information about human behavior, which is central to AI alignment [1]. By studying these premises that people use to communicate, premise semantics offers insights into the nuances of how individuals articulate their preferences and how these preferences can be shaped by the language and reasoning applied. Transitioning to the domain of corrigibility, the importance of premise semantics becomes even more pronounced. Corrigibility, the property of AI systems to accept and adapt to feedback without objection, is of paramount concern for AI safety and alignment [63]. Given that humans might not always articulate their preferences perfectly and that these preferences can change over time, AI systems need to be designed in a way that they can interpret, adapt, and realign based on new information or corrections. This is where premise semantics becomes invaluable. By understanding the underlying premises of human communication, AI can better interpret the intent behind feedback, making corrections more effective and ensuring that the AI remains aligned with evolving human values. Thus, embedding premise semantics in the core design of AI models can be a cornerstone in building corrigible systems that genuinely understand and evolve with human intentions. In the following, the conditionals are integrated through a preference induced by a premise set [61, 62]. This ensures that scenarios are evaluated and ranked based on their alignment with the given set of premises.

Definition 18. *Let X be a set of propositional variables and $MaxC$ the set of all maximal consistent subsets of $Fm(X)$. For $A \subseteq Fm(X)$, let $f^A \subseteq MaxC \times MaxC$ such that $f^A = \{(K, M) | \forall \varphi \in A, (\varphi \in M \rightarrow \varphi \in K)\}$ is a relation over elements of $MaxC$. Let $opt_{f^A}(\varphi) = \{M \in MaxC \mid \varphi \in M, \forall K\,(\varphi \in K \rightarrow (M, K) \in f^A)\}$. It is defined that $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$ if and only if*

$$(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N) \text{ and } \forall M \in opt_{f^A}(\varphi)\,(\psi \in M).$$

Given a set of $\Gamma \subseteq Fm(X)$, it is defined that $\Gamma \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$ if $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$ for some $\varphi \in \Gamma$.

Definition 19. *Let X be a set of propositional variables and $A \subseteq Fm(X)$. A factual-preference Boolean algebra for $\mathbf{Fm}(X)$ is a structure $\langle \mathcal{B}, \mathcal{V}, \succeq_A \rangle$, where:*

- *$\mathcal{B}$ is a Boolean algebra,*
- *$\mathcal{V} = \{V_i\}_{i \in I}$ is the set of valuations from $\mathbf{Fm}(X)$ on $\mathcal{B}$,*
- *$\succeq_A \subseteq \mathcal{V} \times \mathcal{V}$ such that ($V_i \succeq_A V_j$ iff $\forall \varphi \in A \quad (V_j(\varphi) = 1_\mathcal{B} \rightarrow V_i(\varphi) = 1_\mathcal{B})$).*

Here, the betterness relation is reflexive and transitive by definition. For a given preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_A \rangle$, it is defined that $opt_{\succeq_A}(\varphi) = \{V_i \in \mathcal{V} \mid V_i(\varphi) = 1_\mathcal{B}, \forall V_j (V_j(\varphi) = 1_\mathcal{B} \rightarrow V_i \succeq_A V_j)\}$.

Theorem 10. *Let X be a set of propositional variables, where $N \subseteq Fm(X) \times Fm(X)$, and $A \subseteq Fm(X)$. For a given Boolean algebra $\mathcal{B}$ and a valuation V on $\mathcal{B}$, it is defined that $N^V = \{(V(\varphi), V(\psi) | (\varphi, \psi) \in N\}$. We have*

$$\varphi \hookrightarrow \bigcirc \psi \in derive_i^{O^K}(N)$$

if and only if

$$V(\psi) \in out_i^\mathcal{B}(N^V, \{V(\varphi)\}) \text{ for every } \mathcal{B} \in \mathbf{BA} \text{ and valuation } V$$

and

for every factual-preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_A \rangle$,
for every valuation $V_i \in opt_{\succeq_A}(\varphi)$,
it is the case that $V_i(\psi) = 1_\mathcal{B}$.

The theorem can be rewritten as follows:

$$\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$$

if and only if

$$\psi \in out_i^{\mathbf{Fm}(X)}(N, \{\varphi\}), \text{ and for } \langle \mathbf{2}, \mathcal{V}, \succeq_A \rangle,$$
$$\text{for every valuation } V_i \in opt_{\succeq_A}(\varphi), \text{ we have } V_i(\psi) = 1_{\mathcal{B}}$$

or

$$\psi \in out_i^{\mathbf{Fm}(X)}(N, \{\varphi\}), \text{ and if } \varphi \text{ is consistent with } A$$
$$\text{then } A, \varphi \vdash \psi, \text{ and if } \varphi \text{ is inconsistent with } A, \text{ then } \varphi \vdash \psi.$$

The results for the constrained assumptions and preferences can be extended for the other systems introduced, for instance $derive_i^{AND}(N)$.

Example 10. *Consider the premise set $A = \{\neg\varphi \vee \psi, \neg\varphi \vee \neg\psi\}$. The following table shows the possible states for the variables φ and ψ, along with the corresponding values for $\neg\varphi \vee \psi$ and $\neg\varphi \vee \neg\psi$:*

state	φ	ψ	$\neg\varphi \vee \psi$	$\neg\varphi \vee \neg\psi$
s_1	*0*	*0*	*1*	*1*
s_2	*1*	*0*	*0*	*1*
s_3	*0*	*1*	*1*	*1*
s_4	*1*	*1*	*1*	*0*

From this, we can see that $f_A = \{(s_1, s_2), (s_1, s_4), (s_3, s_2), (s_3, s_4), (s_1, s_3), (s_3, s_1)\} \cup \{(s_i, s_i) | i = 1, \ldots, 4\}$. However, this leads to $opt_{f_A}(\varphi) = \emptyset$, since the "best" φ-states s_2 and s_4 are incomparable. Now, suppose we have $N = \{(\varphi, \varphi)\}$. Then $\varphi \hookrightarrow \bigcirc\varphi \notin derive_i^{O^K}(N)$. This shows that the new inference consequence relation, which arises from the combination of input/output operations and a preference relation, is not reflexive. In other words, if $(\varphi, \psi) \in N$, it is not necessarily the case that $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$. Input/output operations are not necessarily reflexive for input sets, meaning $A \nsubseteq out(N, A)$, but are reflexive for norms, meaning $N \subseteq out(N)$. However, this new consequence relation is not reflexive for the set of norms. After AI alignment, it may be determined that some norms are not applicable or relevant in certain situations. In such cases, specific conditional norms may be removed from the set that the AI system follows.

Example 11. *For the conditionals* $N = \{(\top, \varphi), (\varphi, \psi), (\neg\varphi, \neg\psi)\}$ *and the premise set* $A = \{\neg\varphi, \neg\varphi \to \neg\psi\}$*, the best maximal consistent sets have* $\{\neg\varphi, \neg\psi\}$ *(type* s_1*). The second-best maximal consistent sets are those that have either* $\{\varphi, \neg\psi\}$ *(type* s_2*),* $\{\varphi, \psi\}$ *(type* s_3*), or* $\{\neg\varphi, \psi\}$ *(type* s_4*).*

$$\begin{array}{ccccc} & & \textit{best} & s_1\bullet & \\ & - - - - - - - - - - - - - - - - - & & & \\ \textit{2nd best} & s_2 \bullet \varphi & & s_3 \bullet \varphi, \psi & s_4 \bullet \psi \end{array}$$

Since $\forall M \in \mathit{opt}_{f^A}(\neg\varphi)\,(\neg\psi \in M)$*, we have* $\neg\varphi \hookrightarrow \bigcirc\neg\psi \in \mathit{derive}_i^{O^K}(N)$*. The states satisfying* $\neg\varphi$ *are* s_1 *and* s_2*. Within the context of* A*, state* s_1 *is ranked higher than* s_2 *since it fulfills more premises from* A*. In this approach, maximal consistent sets are ordered based on the number of premises they satisfy from* A*.*

Permissive norms and preferences. It is straightforward to rewrite the theorems for conditional permissions.

$\varphi \hookrightarrow P\psi \in \mathit{derive}_i^{P^K}(N)$ if and only if $(\varphi, \psi) \in \mathit{derive}_i^{\mathbf{Fm}(X)}(N)$ and for every factual-preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_A \rangle$, there is a valuation $V_i \in \mathit{opt}_{\succeq_A}(\varphi)$ such that $V_i(\psi) = 1_{\mathcal{B}}$.	$\varphi \hookrightarrow P\psi \in \mathit{derive}_i^{P^H}(N)$ if and only if $(\varphi, \psi) \in \mathit{derive}_i^{\mathbf{Fm}(X)}(N)$ and for every preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_f \rangle$, there is a valuation $V_i \in \mathit{opt}_{\succeq_f}(\varphi)$ such that $V_i(\psi) = 1_{\mathcal{B}}$.

Combining preferences and permissive norms can help to ensure that AI systems make decisions that are both desirable and ethical. For example, consider the case of an access control system for a building [64]. The preferences of the building owner might be to allow as many people as possible to enter the building, while the permissive norms might be to only allow entry to those who have the proper authorization. By combining these preferences and norms, the access control system can make decisions that are both desirable (maximizing the number of people allowed to enter) and ethical (only allowing those with proper authorization). Other examples of combining preferences and permissive norms in the context of AI alignment might include an AI assistant that helps to schedule meetings, a self-driving car that makes decisions about routes and speeds, or a healthcare system that recommends treatment options [65].

6. Semantical embedding of input/output logic into HOL

The LogiKEy framework [10] is a tool for designing and engineering ethical reasoners, normative theories, and deontic logics. It is specifically designed to be used in the control and governance of intelligent autonomous systems. The framework is based on semantical embeddings of deontic logics, logic combinations, and ethico-legal domain theories in expressive classic higher-order logic (HOL). This meta-logical approach allows for powerful tool support in LogiKEy, including the use of off-the-shelf theorem provers and model finders for HOL. These tools allow for flexible experimentation with underlying logics and their combinations, with ethico-legal domain theories, and with concrete examples. The LogiKEy methodology is divided into three layers: logics and logic combinations (L1), ethico-legal domain theories (L2), and applications (L3). In this paper, the focus is on the semantical embeddings in HOL for L1. The embeddings have been implemented in Isabelle/HOL and tested with some logical tasks. While the ultimate goal is to use the LogiKEy framework to encode informal examples using L2 and L3, this work has been postponed for future papers due to space constraints.

The simple type theory developed by Church [66], also known as classical higher-order logic (HOL), is a powerful language for representing mathematical structures. The syntax and semantics of HOL are well understood [67, 68] (for a brief introduction to HOL see [12]). It has roots in Frege's book [69] and Russell's ramified theory of types [70]. The so-called *shallow semantical embedding* approach was developed by Benzmüller [71] for translating (the semantics of) classical and non-classical logics into HOL. Examples include propositional and quantified multimodal logics [72, 73] and dyadic deontic logics [11, 74].

Benzmüller et al. [12] devised an indirect approach to embedding two I/O operations in modal logic and consequently into HOL. One advantage of building I/O operations over Boolean algebras is that the I/O logic can be directly embedded in HOL. For normative system N, the structure $\mathcal{N} = \langle \mathcal{B}, V, N^V \rangle$ is called a Boolean normative model, where V is a valuation from $\mathbf{Fm}(X)$ to $\mathcal{B}$. The semantical embedding of I/O logic is based on Theorem 6, which states that $(\varphi, \psi) \in \mathit{derive}_i^{\mathbf{Fm}(X)}(N)$ holds if and only if $V(\psi) \in \mathit{out}_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$ holds in all Boolean normative models.

The remainder of this section shows how the embedding works, abbreviating type $i \to o$ as τ. The HOL signature is assumed to contain the constant symbols $N_{i\to\tau}$, $\neg_{i\to i}$, $\vee_{i\to i\to i}$, $\wedge_{i\to i\to i}$, $\top_i$ and $\bot_i$. Moreover, for each atomic

propositional symbol $p^j \in X$ of $\mathbf{Fm}(X)$, the HOL signature must contain a respective constant symbol p_i^j. Without loss of generality, it is assumed that besides those symbols and the primitive logical connectives of HOL, no other constant symbols are given in the signature of HOL.

The mapping $\lfloor\cdot\rfloor$ translates element $\varphi \in \mathbf{Fm}(X)$ into HOL terms $\lfloor\varphi\rfloor$ of type i. The mapping is recursively defined:

$$\begin{array}{lcll}
\lfloor p^j \rfloor & = & p_i^j & p^j \in X \\
\lfloor \top \rfloor & = & \top_i & \\
\lfloor \bot \rfloor & = & \bot_i & \\
\lfloor \neg\varphi \rfloor & = & \neg_{i\to i}(\lfloor\varphi\rfloor) & \\
\lfloor \varphi \vee \psi \rfloor & = & \vee_{i\to i\to i}\lfloor\varphi\rfloor\lfloor\psi\rfloor & \\
\lfloor \varphi \wedge \psi \rfloor & = & \wedge_{i\to i\to i}\lfloor\varphi\rfloor\lfloor\psi\rfloor & \\
\lfloor d_i(N)(\varphi,\psi)^{10} \rfloor & = & (\bigcirc_i(N)_{\tau\to\tau}\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor &
\end{array}$$

$\bigcirc_I(N)_{\tau\to\tau}$, $\bigcirc_{II}(N)_{\tau\to\tau}$, $\bigcirc_1(N)_{\tau\to\tau}$, $\bigcirc_2(N)_{\tau\to\tau}$ and $\bigcirc_3(N)_{\tau\to\tau}$ are thereby abbreviated HOL terms:

$$\begin{array}{ll}
\bigcirc_I(N)_{\tau\to\tau} & = \lambda A_\tau \lambda X_i(\exists U\,(\exists Y\,(\exists Z\,(A\,Z \wedge Z = Y \wedge N\,Y\,U \wedge U \leq X)))) \\
\bigcirc_{II}(N)_{\tau\to\tau} & = \lambda A_\tau \lambda X_i(\exists U\,(\exists Y\,(\exists Z\,(A\,Z \wedge Z \leq Y \wedge N\,Y\,U \wedge U = X)))) \\
\bigcirc_1(N)_{\tau\to\tau} & = \lambda A_\tau \lambda X_i(\exists U\,(\exists Y\,(\exists Z\,(A\,Z \wedge Z \leq Y \wedge N\,Y\,U \wedge U \leq X))))
\end{array}$$

$$\begin{array}{ll}
\bigcirc_2(N)_{\tau\to\tau} & = \lambda A_\tau \lambda X_i(\forall V\,(\mathit{Saturated}\,V \wedge \forall U(A\,U \to V\,U) \\
& \to \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y))))
\end{array}$$

$$\begin{array}{ll}
\bigcirc_3(N)_{\tau\to\tau} & = \lambda A_\tau \lambda X_i(\forall V\,(\forall U(A\,U \to V\,U) \wedge V = \mathit{Up}\,V \\
& \wedge \forall W(\exists Y(V\,Y \wedge N\,Y\,W) \to V\,W) \\
& \to \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y))))
\end{array}$$

where

$$\begin{array}{ll}
\leq & = \lambda X_i \lambda Y_i(X \wedge_{i\to i\to i} Y = X) \\
\mathit{Saturated} & = \lambda A_\tau(\forall X\,\forall Y((A\,(X \vee_{i\to i\to i} Y) \to A\,X \vee A\,Y) \\
& \wedge (A\,X \wedge X \leq Y \to A\,Y))) \\
\mathit{Up} & = \lambda A_\tau \lambda X_i(\exists Z(A\,Z \wedge Z \leq X)).
\end{array}$$

No further specification is needed for $N_{i\to\tau}$, $\neg_{i\to i}$, $\vee_{i\to i\to i}$, $\wedge_{i\to i\to i}$, $\top_i$ and $\bot_i$.

[10] $d_i(N)(\varphi,\psi)$ is an abbreviation of $(\varphi,\psi) \in \mathit{derive}_i^{\mathbf{Fm}(X)}(N)$.

6.1. Soundness and completeness

To prove the soundness and completeness, that is, faithfulness, of the above embedding, a mapping from Boolean normative models into Henkin models [75] is employed.

Definition 20 (Henkin model $H^{\mathcal{N}}$ for Boolean normative model $\mathcal{N}$). *For any Boolean normative model $\mathcal{N} = \langle \mathcal{B}, V, N^V \rangle$, a corresponding Henkin model $H^{\mathcal{N}}$ is defined. Thus, let a Boolean normative model $\mathcal{N} = \langle \mathcal{B}, V, N^V \rangle$ be given. Moreover, assume that the finite set $X = \{p^1, , ..., p^m\}$, for $m \geq 1$, are the only atomic symbols in $\mathbf{Fm}(X)$. The embedding requires the corresponding signature of HOL to provide constant symbols p_i^j such that $\lfloor p^j \rfloor = p_i^j$.*

A Henkin model $H^{\mathcal{N}} = \langle \{D_\alpha\}_{\alpha \in T}, I \rangle$ for $\mathcal{N}$ is now defined as follows: D_i is chosen as the set of B; all other sets $D_{\alpha\to\beta}$ are chosen as (not necessarily full) sets of functions from D_α to D_β. For all $D_{\alpha\to\beta}$, the rule that every term $t_{\alpha\to\beta}$ must be denoted in $D_{\alpha\to\beta}$ must be obeyed (henceforth referred to as Denotatpflicht). In particular, it is required that D_i, $D_{i\to i}$, $D_{i\to i\to i}$ and $D_{i\to\tau}$ should contain the elements Ip_i^j, $I\top_i$, $I\bot_i$, $I\neg_{i\to i}$, $I\vee_{i\to i\to i}$, $I\wedge_{i\to i\to i}$ and $IN_{i\to\tau}$. The interpretation function I of $H^{\mathcal{N}}$ is defined as follows:

1. *For $j = 1, ..., m$: $Ip_i^j \in D_i$ is chosen such that $Ip_i^j = V(p^j)$ in $\mathcal{N}$.*
2. *$I\top_i \in D_i$ is chosen such that $I\top_i = V(\top)$ in $\mathcal{N}$.*
3. *$I\bot_i \in D_i$ is chosen such that $I\bot_i = V(\bot)$ in $\mathcal{N}$.*
4. *$I\neg_{i\to i} \in D_{i\to i}$ is chosen such that $I(\neg_{i\to i}\varphi) = \psi$ iff $\neg V(\varphi) = V(\psi)$ in $\mathcal{N}$.*
5. *$I\vee_{i\to i\to i} \in D_{i\to i\to i}$ is chosen such that $I\vee_{i\to i\to i}\varphi\psi = \phi$ iff $V(\varphi) \vee V(\psi) = V(\phi)$ in $\mathcal{N}$.*
6. *$I\wedge_{i\to i\to i} \in D_{i\to i\to i}$ is chosen such that $I\wedge_{i\to i\to i}\varphi\psi = \phi$ iff $V(\varphi) \wedge V(\psi) = V(\phi)$ in $\mathcal{N}$.*
7. *$IN_{i\to\tau} \in D_{i\to\tau}$ is chosen such that $IN_{i\to\tau}\varphi\psi = T$ iff $(V(\varphi), V(\psi)) \in N^V$ in $\mathcal{N}$.*
8. *For the logical connectives $\neg$,$\wedge$, $\vee$, Π and $=$ of HOL, the interpretation function I is defined as usual (see Appendix D).*

The existence of valuation V, which is a Boolean homomorphism from the Boolean algebra $\mathbf{Fm}(X)$ into the Boolean algebra $\mathcal{B}$, guarantees the existence

of I and its above-mentioned requirements. Since it is assumed that there are no other symbols (apart from $\top_i$, $\bot_i$, $\neg_{i\to i}$, $\vee_{i\to i\to i}$, $\wedge_{i\to i\to i}$, $N_{i\to\tau}$ as well as $\neg$, $\vee$, $\prod$ and $=$) in the signature of HOL, I is a total function. Moreover, the above construction guarantees that $H^{\mathcal{N}}$ is a Henkin model: $\langle D, I\rangle$ is a frame, and the choice of I in combination with the Denotatpflicht ensures that for arbitrary assignments, g, $\|.\|^{H^M,g}$ is a total evaluation function.

Lemma 1. *Let $H^M = \langle \{D_\alpha\}_{\alpha\in T}, I\rangle$ be a Henkin model for Boolean normative model $\mathcal{N}$. We have $H^{\mathcal{N}} \models^{HOL} \Sigma$ for all $\Sigma \in \{COM\vee, COM\wedge, ASS\vee, ASS\wedge, IDE\vee, IDE\wedge, COMP\vee, COMP\wedge, Dis\vee\wedge, Dis\wedge\vee\}$, where:*

$COM\vee$	*is*	$\forall X_i\, Y_i\ (X \vee Y = Y \vee X)$
$COM\wedge$	*is*	$\forall X_i\, Y_i\ (X \wedge Y = Y \wedge X)$
$ASS\vee$	*is*	$\forall X_i\, Y_i\, Z_i\ (X \vee (Y \vee Z) = (X \vee Y) \vee Z)$
$ASS\wedge$	*is*	$\forall X_i\, Y_i\, Z_i\ (X \wedge (Y \wedge Z) = (X \wedge Y) \wedge Z)$
$IDE\vee$	*is*	$\forall X_i\ (X \vee \bot = X)$
$IDE\wedge$	*is*	$\forall X_i\ (X \wedge \top = X)$
$COMP\vee$	*is*	$\forall X_i\ (X \vee \neg X = \top)$
$COMP\wedge$	*is*	$\forall X_i\ (X \wedge \neg X = \bot)$
$Dis\vee\wedge$	*is*	$\forall X_i\, Y_i\, Z_i\ (X \vee (Y \wedge Z) = (X \vee Y) \wedge (X \vee Z))$
$Dis\wedge\vee$	*is*	$\forall X_i\, Y_i\, Z_i\ (X \wedge (Y \vee Z) = (X \wedge Y) \vee (X \wedge Z))$

Lemma 2. *Let $H^{\mathcal{N}}$ be a Henkin model for Boolean normative model $\mathcal{N} = \langle \mathcal{B}, V, N^V\rangle$. For all conditional norms (φ, ψ) with arbitrary variable assignments g, it holds that $V(\psi) \in out_i^{\mathcal{B}}(N, \{V(\varphi)\})$ if and only if $\|\lfloor d_i(N)(\varphi,\psi)\rfloor\|^{H^{\mathcal{N}},g} = T$.*

Lemma 3. *For every Henkin model $H = \langle \{D_\alpha\}_{\alpha\in T}, I\rangle$ such that $H \models^{HOL} \Sigma$ for all $\Sigma \in \{COM\vee, COM\wedge, ASS\vee, ASS\wedge, IDE\vee, IDE\wedge, COMP\vee, COMP\wedge, Dis\vee\wedge, Dis\wedge\vee\}$, there exists a corresponding Boolean normative model $\mathcal{N}$. Corresponding means that for all conditional norms (φ, ψ) and for all g assignments, then $\|\lfloor d_i(N)(\varphi,\psi)\rfloor\|^{H,g} = T$ if and only if $V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$.*

Theorem 11 (Soundness and completeness of the embedding).

For every Boolean normative model $\mathcal{N}$, $\quad V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$

if and only if

$$\{COM\vee, ..., Dis\wedge\vee\} \models^{HOL} \lfloor d_i(N)(\varphi,\psi)\rfloor.$$

7. Related work

Value alignment. The field of value alignment in artificial intelligence has received considerable attention over the past few years. Amodei et al. [76] discuss a set of problems related to AI safety and propose research directions to address these concerns. Our work extends this research by incorporating logical frameworks, specifically deontic logic, as a tool to navigate potential safety pitfalls. Hadfield-Menell et al. [77] and [78] propose a reward-based approach for value alignment, using inverse reward design and *cooperative inverse reinforcement learning*. While reward-based systems are undoubtedly a crucial aspect of alignment, our approach provides a more structured and theoretical framework for understanding complex ethical situations through logical reasoning. In terms of ethical considerations in AI, Vincent C. Müller's work [79] provides an insightful analysis. However, our approach seeks to operationalize some of these ethical considerations using logical systems, taking theory into potential application. Some work, such as that of Lake et al. [80], and Leike et al. [81], has been done on making machine learning more interpretable and adaptable, focusing on creating AI systems that learn and think like humans or modeling agent behavior via reward. Our logical approach compliments this by providing a framework for AI to reason about ethical situations in a way that can be easily interpreted by humans. In the context of *cooperative inverse reinforcement learning* [77], integrating normative reasoning and preference-based approaches could lead to robust and effective AI alignment. Normative reasoning sets ethical and operational boundaries, providing hard limits to AI behavior and preventing it from undertaking harmful or unethical actions. On the other hand, preference-based approaches allow the AI to learn and adapt based on user behavior and inferred desires, thus enhancing its ability to cater to individual user needs and promote personalization [80]. The combination of these approaches ensures that an AI can adapt to a user's unique requirements while maintaining a consistent ethical standard. Yudkowsky's work [82] further provides an excellent foundation for why alignment is hard and where research should be started. Our approach is an attempt to operationalize some of the research directions suggested by Yudkowsky by using deontic logic to handle complex and uncertain ethical situations in a robust and flexible manner. In summary, while the related works have advanced the field of AI value alignment significantly, our approach provides a new perspective. We aim to integrate logical reasoning, particularly deontic logic, into the conversation, provid-

ing a structured approach to dealing with ethical and safety challenges in AI. Compared to earlier attempts at motivating deontic logic in a hybrid approach to value alignment [4], this paper offers a comprehensive framework for applying deontic logic in AI alignment. It achieves this through a detailed logical formalization, characterization, and implementation drawn from the deontic literature. Additionally, the framework is strengthened by incorporating uncertainty into normative reasoning, making it more robust and adaptable to real-world scenarios. Particularly, an inspiring work for future research is a neural-symbolic implementation of input/output logic, which is designed to handle uncertainty in dynamic normative contexts [83]. This approach combines neural networks and symbolic reasoning to provide a powerful framework for dealing with uncertainty. By leveraging the strengths of both neural and symbolic techniques, it offers a promising avenue for addressing uncertainty in normative reasoning.

Uncertainty and choice in deontic logic. STIT (Seeing To It That) logic, developed by John Horty [84], is a model for examining how agents handle and act in situations marked by uncertainty. This uncertainty can be categorized into two primary types: uncertainty in an indeterministic timeline where the future can unfold in numerous ways, and uncertainty about the precise outcome of an action, where the agent doesn't have full control over future developments. Consider the "gambling problem", where an agent faces the decision of whether to gamble five dollars. If they choose to gamble, they could either gain ten dollars or lose the initial five. Conversely, if they abstain from gambling, they keep the original sum, irrespective of external events. This scenario introduces a third type of uncertainty: where the expected value does not guide the agent's choice, thus leading to indecision. Both options present the same expected value, further complicating the decision-making process. The current scope of STIT logic focuses more on how agents manage uncertainty using their preferences and utility functions, rather than incorporating moral values or considerations. However, this focus may not sufficiently address scenarios involving moral uncertainty, where moral values significantly influence decision-making processes and may not be easily encapsulated by utility functions or preferences. It is essential to note, though, that there are various extensions to STIT logic in the literature, like deontic STIT logics using violation constants [85], and deontic STIT logics based on relational semantics [86, 87], that are not founded on utilities and can better represent moral uncertainties. Hence, while traditional STIT logic may

have its limitations in certain scenarios, these extensions offer more nuanced and comprehensive approaches for handling moral uncertainties. Moreover, there are both decision-theoretic extensions of deontic logic [88] and game-theoretic approaches [89] that address uncertain outcomes in normative reasoning, rather than uncertainty in normative reasoning itself. The approach in this paper allows us to represent the agent's uncertainty and the different ways in which the situation could unfold, depending on the norms or moral insights that the agent chooses to adopt.

Input/output logic and joining systems. Gabbay, Parent and van der Torre [90] proposed building an I/O framework on top of lattices. They have results only for the simple-minded output operation. This paper has shown that for an input set A, by using the *upward-closed set of A* operator instead of the *upward-closed set of the infimum of A* [90], many new and old derivation systems can be built over Boolean algebras, Heyting algebras, and generally any abstract logic. The algebrization of the I/O framework shows more similarity with the theory of joining-systems [91], an algebraic approach for the study of normative systems over Boolean algebras. Norms in the I/O framework play the same role as joining in the theory of Lindahl and Odelstal [91, 92]. There are important similarities between input/output logic and the theory of joining-systems, such as studying normative systems as deductive systems and representing norms as ordered pairs. Moreover, both frameworks can generally be built on top of algebraic structures such as Boolean algebras and lattices. While the focus in input/output logic is deontic and factual detachment, the central themes of the theory of joining-systems are intermediate concepts and representing normative systems as a network of subsystems and their inter-relationships (for more details, see [91]). Sun [92] built Boolean joining systems that characterize I/O logic in a sense that a norm is derivable from a set of norms if and only if it is in the set of norms algebraically generated in the Lindenbaum-Tarski algebra for propositional logic. As in the Bochman approach [93], the work of Sun [92] and Domenico et al. [20] has no direct connection to input/output operations. In this paper, algebraic I/O operations were built directly over Boolean algebras and, more generally, abstract logics. There is a similar result for building the simple-minded I/O operation over Tarskian consequence relations in [94] (see the discussion about abstract input/output logic in [95]).

Modal logic and normative reasoning. This paper defines two groups of operations similar to the possible world semantics characterization of "box"

and "diamond." In this characterization, "box" is closed under conjunction (represented as $(\Box\varphi \wedge \Box\psi) \rightarrow \Box(\varphi \wedge \psi)$), while "diamond" is not.

Derivations systems that do not admit the AND rule: In the main literature of input/output logic developed by Makinson and van der Torre [31], Parent, Gabbay, and van der Torre [96], Parent and van der Torre [50, 97, 98], and Stolpe [99, 100, 101], at least one form of the AND inference rule is present. Sun [95] analyzed norm derivation rules of input/output logic in isolation. Still, it is not clear how to combine them and build new logical systems, specifically systems that do not admit the AND rule. This paper has shown how to remove the AND rule from the proof system and build new I/O operations to produce permissible propositions. Unlike minimal deontic logics [9, 58], and similar approaches such as that of Ciabattoni, Gulisano and Lellmann [102] that do not have deontic aggregation principles, the approach presented in this paper validates deontic and factual detachment.

Derivations systems that admit the AND rule: In accordance with the reversibility of inference rules in the I/O proof systems, this paper has shown how it is possible to add AND and other rules required for obligation [31] to the proof systems, and find I/O operations for them.

There are other abstract approaches: I/O operations over semigroups [103], which do not admit AND, and a detachment mechanism over an arbitrary set [104] that admits a kind of AND, called cumulative aggregation. These approaches may have certain limitations or may not fit easily within the framework of formal logic and logical reasoning

Prioritizing norms in normative reasoning. The theory of reasons [29], developed by John Horty, is a framework for understanding how different norms or moral principles can conflict with one another and how these conflicts can be resolved. In this theory, norms are ranked according to their priority, with some norms taking precedence over others in certain circumstances. When two norms conflict, the norm with the higher priority is said to defeat the norm with the lower priority. This process of norm defeat allows for the resolution of moral conflicts and helps to determine which norm should be followed in a given situation. Constrained input/output logic [34, 59] is a mechanism that allows for the addition of priority to norms. In this paper, a preference relation is introduced in the object language to discuss the ranking of norms. Input/output operations are studied in the object language using subordination algebras and duality techniques [20]. This paper motivates to represent and combine input/output operations as tools for normative

reasoning and preference relations to rank and prioritize norms in the object language. There is an abstract and fruitful approach for value alignment that involves an explicit connection between norms and values [105]. However, it is not always clear how norms correspond to values in this method. Furthermore, this approach is more general and shares some techniques, such as lifting, that have been previously explored in input/output logic [59]. However, this approach neglects the conditional structure of norms. A preference relation was integrated in this paper as a non-monotonic means between the heads and bodies of norms in order to produce a linguistic and compositional conditional structure [106] for selecting norms with values.

8. Conclusion

This paper has shown the importance of considering both normative reasoning and preference-based approaches in the alignment of AI systems with human values and preferences. By taking into account a range of ethical and legal norms and values, and using them in combination to make informed and ethical decisions, AI systems can better align with human preferences and values, even in situations of uncertainty. This can contribute to the advancement of AI alignment and the development of more ethical and responsible AI systems.

This paper presented new algebraic systems developed in the LogiKEy normative reasoning framework. A dataset of semantical embeddings of deontic logics in HOL is available (see Appendix A). The dataset can be used for ethical and legal reasoning tasks. In summary, this paper characterizes a class of proof systems over Boolean algebras for a set of explicitly given conditional norms as follows:

$derive_i^{\mathcal{B}}$	Rules
$derive_R^{\mathcal{B}}$	{EQO}
$derive_L^{\mathcal{B}}$	{EQI}
$derive_0^{\mathcal{B}}$	{EQI, EQO}
$derive_I^{\mathcal{B}}$	{SI, EQO}
$derive_{II}^{\mathcal{B}}$	{WO, EQI}
$derive_1^{\mathcal{B}}$	{SI, WO}
$derive_2^{\mathcal{B}}$	{SI, WO, OR}
$derive_3^{\mathcal{B}}$	{SI, WO, T}

$$\text{EQO}\ \frac{(a,x) \qquad x=y}{(a,y)} \qquad \text{T}\ \frac{(a,x) \qquad (x,y)}{(a,y)}$$

$$\text{EQI}\ \frac{(a,x) \qquad a=b}{(b,x)} \qquad \text{OR}\ \frac{(a,x) \qquad (b,x)}{(a \vee b,x)}$$

$$\text{SI}\ \frac{(a,x) \qquad b \leq a}{(b,x)} \qquad \text{WO}\ \frac{(a,x) \qquad x \leq y}{(a,y)}$$

Each proof system is sound and complete for an input/output (I/O) operation. The I/O operations resemble inferences, where inputs need not be included among outputs, and outputs need not be reusable as inputs [31]. Moreover, this paper has shown how to add the two rules AND and CT to the proof systems and find corresponding I/O operations for them.

$derive_i^X$	Rules
$derive_{II}^{AND}$	{WO, EQI, AND}
$derive_1^{AND}$	{SI, WO, AND}
$derive_2^{AND}$	{SI, WO, OR, AND}
$derive_I^{CT}$	{SI, EQO, CT}
$derive_{II}^{CT}$	{WO, EQI, CT}
$derive_1^{CT}$	{SI, WO, CT}
$derive_1^{CT,AND}$	{SI, WO, CT, AND}
$derive_I^{OR}$	{SI, EQO, OR}
$derive_I^{CT,OR}$	{SI, EQO, CT, OR}
$derive_1^{CT,OR}$	{SI, WO, CT, OR}
$derive_1^{CT,OR,AND}$	{SI, WO, CT, OR, AND}

$$\text{AND}\ \frac{(a,x) \qquad (a,y)}{(a, x \wedge y)}$$

$$\text{CT}\ \frac{(a,x) \qquad (a \wedge x, y)}{(a,y)}$$

The input/output logic is inspired by a view of logic as a *secretarial assistant* tasked with preparing inputs before they go into the motor engine and are unpacked as outputs, rather than logic as an *inference motor* [31]. Input/output logic can be based on a wide range of base logics [20, 60]. In the existing literature, the investigated input/output operations are primarily built upon classical propositional logic and intuitionist propositional logic [96]. However, the presented algebraic construction demonstrates the possibility of constructing input/output operations on top of any abstract logic.

Finally, this paper has proved that the extension of propositional logic with a set of conditional norms is both sound and complete in relation to the class of Boolean algebras where the corresponding input/output operation is valid. Based on this result, a conditional theory has been integrated into input/output logic. This paper presented an extension of the input/output framework for normative reasoning in uncertain situations, incorporating a preference relation. By combining normative and preference reasoning, more informed decisions can be made when there is uncertainty about relevant norms and values or about the likely consequences of different actions. This research has emphasized the pivotal role of amalgamating both normative

(deontic) prescriptions and quantitative, utility-based preferences within the scope of AI alignment. Acknowledging the necessity for a potential trade-off between these two forms of reasoning, this work recognizes the benefits of a diversified approach in the realm of AI alignment.

Acknowledgments

I would like to express my gratitude to the anonymous referees for their valuable comments. I would like to thank Bas van Fraassen, Dov Gabbay, Leon van der Torre, Christian Straßer, Christoph Benzmüller, Xavier Parent, Majid Alizadeh, Morteza Ansarinia, and Llio Humphreys for comments that greatly improved the manuscript.

Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author used the ChatGPT AI tool in order to enhance the language in the paper. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the publication.

References

[1] S. Russell, Human compatible: Artificial intelligence and the problem of control, Penguin, 2019.

[2] B. Christian, The alignment problem: Machine learning and human values, WW Norton & Company, 2020.

[3] I. Gabriel, Artificial intelligence, values, and alignment, Minds and Machines 30 (3) (2020) 411–437.

[4] T. W. Kim, J. Hooker, T. Donaldson, Taking principles seriously: A hybrid approach to value alignment in artificial intelligence, Journal of Artificial Intelligence Research 70 (2021) 871–890.

[5] F. J. McDonald, AI, alignment, and the categorical imperative, AI and Ethics (2022) 1–8.

[6] P. Eckersley, Impossibility and uncertainty theorems in AI value alignment (or why your AGI should not have a utility function), arXiv preprint arXiv:1901.00064 (2018).

[7] S. Lazar, Duty and doubt, Journal of Practical Ethics 8 (1) (2020).

[8] X. Parent, L. van der Torre, Input/output logic, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of Deontic Logic, Vol. 1, College Publications, 2013, pp. 499–544.

[9] B. F. Chellas, Modal logic, Cambridge University Press, 1980.

[10] C. Benzmüller, X. Parent, L. van der Torre, Designing normative theories for ethical and legal reasoning: LogiKEy framework, methodology, and tool support, Artificial Intelligence 237 (2020) 103348.

[11] C. Benzmüller, A. Farjami, X. Parent, Åqvist's dyadic deontic logic E in HOL, Journal of Applied Logics – IfCoLoG Journal of Logics and their Applications (Special Issue on Reasoning for Legal AI) 6 (5) (2019) 733–755.

[12] C. Benzmüller, A. Farjami, P. Meder, X. Parent, I/O logic in HOL, Journal of Applied Logics – IfCoLoG Journal of Logics and their Applications (Special Issue on Reasoning for Legal AI) 6 (5) (2019) 715–732.

[13] C. Benzmüller, A. Farjami, D. Fuenmayor, P. Meder, X. Parent, A. Steen, L. van der Torre, V. Zahoransky, LogiKEy workbench: Deontic logics, logic combinations and expressive ethical and legal reasoning (Isabelle/HOL dataset), Data in Brief 33 (2020) 106409.

[14] J. M. Font, R. Jansana, A general algebraic semantics for sentential logics, Cambridge University Press, 2017.

[15] P. McNamara, Deontic logic, in: E. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Winter 2014 Edition, Metaphysics Research Lab, Stanford University, 2014.

[16] S. Nair, Deontic logic and ethics, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of Deontic Logic, Vol. 2, College Publications, 2021, pp. 549–656.

[17] S. O. Hansson, The varieties of permissions, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of deontic logic and normative systems., Vol. 1, College Publications, 2013, pp. 195–240.

[18] D. Makinson, L. van der Torre, Permission from an input/output perspective, Journal of Philosophical Logic 32 (4) (2003) 391–416.

[19] A. Stolpe, A theory of permission based on the notion of derogation, Journal of Applied Logic 8 (1) (2010) 97–113.

[20] A. D. Domenico, A. Farjami, K. Manoorkar, A. Palmigiano, M. Panettiere, X. Wang, Subordination algebras as semantic environment of input/output logic, in: A. Ciabattoni, E. Pimentel, R. J. G. B. de Queiroz (Eds.), Logic, Language, Information, and Computation - 28th International Workshop, WoLLIC 2022, Iaşi, Romania, September 20-23, 2022, Proceedings, Vol. 13468 of Lecture Notes in Computer Science, Springer, 2022, pp. 326–343.

[21] A. D. Domenico, A. Farjami, K. Manoorkar, A. Palmigiano, M. Panettiere, X. Wang, Obligations and permissions, algebraically, Unpublished (2023).

[22] G. H. von Wright, Deontic logic, Mind 60 (237) (1951) 1–15.

[23] R. M. Chisholm, Contrary-to-duty imperatives and deontic logic, Analysis 24 (1963) 33–36.

[24] B. Hansson, An analysis of some deontic logics, Noûs (1969) 373–398.

[25] L. Åqvist, Deontic logic, in: D. Gabbay, F. Guenthner (Eds.), Handbook of Philosophical Logic: Volume 8, Springer, 2002, pp. 147–264.

[26] A. Kratzer, The notional category of modality, in: H. J. Eikmeyer, H. Rieser (Eds.), Words, Worlds, and Contexts: New Approaches in Word Semantics, De Gruyter, 1981, pp. 38–74.

[27] J. Carmo, A. Jones, Completeness and decidability results for a logic of contrary-to-duty conditionals, Journal of Logic and Computation 23 (3) (2013) 585–626.

[28] J. Hansen, Reasoning about permission and obligation, in: S. O. Hansson (Ed.), David Makinson on Classical Methods for Non-classical Problems, Springer, 2014, pp. 287–333.

[29] J. F. Horty, Reasons as defaults, Oxford University Press, 2012.

[30] R. Reiter, A logic for default reasoning, Artificial intelligence 13 (1) (1980) 81–132.

[31] D. Makinson, L. van der Torre, Input/output logics, Journal of Philosophical Logic 29 (4) (2000) 383–408.

[32] L. Robaldo, C. Bartolini, M. Palmirani, A. Rossi, M. Martoni, G. Lenzini, Formalizing GDPR provisions in reified I/O logic: the DAPRECO knowledge base, Journal of Logic, Language and Information 29 (2019) 401–449.

[33] A. Steen, Goal-directed decision procedures for input/output logics, in: F. Liu, A. Marra, P. Portner, F. van de Putte (Eds.), Deontic Logic and Normative Systems — 14th International Conference, DEON2020/21, Munich, Germany, 21-24 July, 2021, College Publications, 2021, pp. 414–426.

[34] D. Makinson, L. van der Torre, Constraints for input/output logics, Journal of Philosophical Logic 30 (2) (2001) 155–185.

[35] X. Parent, L. van der Torre, Introduction to deontic logic and normative systems, College Publications, 2018.

[36] S. Burris, H. P. Sankappanavar, A course in universal algebra, Vol. 78 of Graduate Texts Math, Springer, 1981.

[37] J. Y. Halpern, Reasoning about uncertainty, MIT press, 2017.

[38] M. Oaksford, N. Chater, Rationality in an uncertain world: Essays on the cognitive science of human reasoning, Psychology Press, 1998.

[39] K. Stenning, M. van Lambalgen, Human reasoning and cognitive science, MIT Press, 2012.

[40] R. Bradley, Decision theory with a human face, Cambridge University Press, 2017.

[41] R. Bradley, M. Drechsler, Types of uncertainty, Erkenntnis 79 (2014) 1225–1248.

[42] S. Armstrong, S. Mindermann, Occam's razor is insufficient to infer the preferences of irrational agents, in: S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada, 2018, pp. 5603–5614.

[43] H. A. Costa, Non-adjunctive inference and classical modalities, Journal of Philosophical Logic 34 (5/6) (2005) 581–605.

[44] N. Bostrom, Superintelligence, Dunod, 2017.

[45] F. Dietrich, B. Jabarian, Decision under normative uncertainty, Economics & Philosophy 38 (3) (2022) 372–394.

[46] J. Pearl, Probabilistic semantics for nonmonotonic reasoning: A survey, University of California (Los Angeles). Computer Science Department, 1989.

[47] X. Parent, L. van der Torre, Input/output logics without weakening, Filosofiska Notiser 6 (1) (2019) 189–208.

[48] X. Parent, Preference semantics for Hansson-type dyadic deontic logic: a survey of results, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of Deontic Logic, Vol. 2, College Publications, 2021, pp. 7–70.

[49] C. Straßer, M. Beirlaen, F. van de Putte, Adaptive logic characterizations of input/output logic, Studia Logica 104 (5) (2016) 869–916.

[50] X. Parent, L. van der Torre, The pragmatic oddity in norm-based deontic logics, in: G. Governatori, J. Keppens (Eds.), Proceedings of the 16th edition of the International Conference on Articial Intelligence and Law, ACM-SIAM, 2017, pp. 169–178.

[51] X. Sun, L. van der Torre, Combining constitutive and regulative norms in input/output logic, in: F. Cariani, D. Grossi, J. Meheus, X. Parent (Eds.), Deontic Logic and Normative Systems — 12th International

Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014, Springer, 2014, pp. 241–257.

[52] R. Jansana, Algebraic propositional logic, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Winter 2016 Edition, Metaphysics Research Lab, Stanford University, 2016.

[53] S. O. Hansson, T. Grüne-Yanoff, Preferences, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Spring 2022 Edition, Metaphysics Research Lab, Stanford University, 2022.

[54] D. Hadfield-Menell, A. D. Dragan, P. Abbeel, S. Russell, The off-switch game, in: The Workshops of the The Thirty-First AAAI Conference on Artificial Intelligence, Saturday, February 4-9, 2017, San Francisco, California, USA, Vol. WS-17 of AAAI Technical Report, AAAI Press, 2017.

[55] D. Lewis, Counterfactuals, Blackwell, Oxford, 1973.

[56] R. H. Thomason, Deontic logic as founded on tense logic, in: R. Hilpinen (Ed.), New Studies in Deontic Logic, Springer, 1981, pp. 165–176.

[57] D. Bonevac, Against conditional obligation, Noûs 32 (1) (1998) 37–53.

[58] L. Goble, Prima facie norms, normative conflicts, and dilemmas, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of Deontic Logic, Vol. 1, College Publications, 2013, pp. 499–544.

[59] X. Parent, Moral particularism in the light of deontic logic, Artificial Intelligence and Law 19 (2) (2011) 75–98.

[60] K. van Berkel, C. Straßer, Reasoning with and about norms in logical argumentation, in: F. Toni, S. Polberg, R. Booth (Eds.), Computational Models of Argument: Proceedings of COMMA 2022, Vol. 353, IOS Press, 2022, pp. 332–343.

[61] D. Lewis, Ordering semantics and premise semantics for counterfactuals, Journal of Philosophical Logic 10 (2) (1981) 217–234.

[62] A. Kratzer, Modals and conditionals: New and revised perspectives, Vol. 36, Oxford University Press, 2012.

[63] N. Soares, B. Fallenstein, S. Armstrong, E. Yudkowsky, Corrigibility, in: T. Walsh (Ed.), Artificial Intelligence and Ethics, Papers from the 2015 AAAI Workshop, Austin, Texas, USA, January 25, 2015, Vol. WS-15-02 of AAAI Technical Report, AAAI Press, 2015.

[64] W. Serrano, iBuilding: Artificial intelligence in intelligent buildings, Neural Computing and Applications 34 (2) (2022) 875–897.

[65] A. K. Sahoo, S. Mallik, C. Pradhan, B. S. P. Mishra, R. K. Barik, H. Das, Intelligence-based health recommendation system using big data analytics, in: N. Dey, H. Das, B. Naik, H. S. Behera (Eds.), Big Data Analytics for Intelligent Healthcare Management, Elsevier, 2019, pp. 227–246.

[66] A. Church, A formulation of the simple theory of types, Journal of Symbolic Logic 5 (2) (1940) 56–68.

[67] C. Benzmüller, C. Brown, M. Kohlhase, Higher-order semantics and extensionality, Journal of Symbolic Logic 69 (4) (2004) 1027–1088.

[68] C. Benzmüller, P. Andrews, Church's type theory, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Summer 2019 Edition, Metaphysics Research Lab, Stanford University, 2019.

[69] G. Frege, Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens, Halle, 1879.

[70] B. Russell, Mathematical logic as based on the theory of types, American Journal of Mathematics 30 (3) (1908) 222–262.

[71] C. Benzmüller, Universal (meta-) logical reasoning: Recent successes, Science of Computer Programming 172 (2019) 48–62.

[72] C. Benzmüller, L. Paulson, Quantified multimodal logics in simple type theory, Logica Universalis (Special Issue on Multimodal Logics) 7 (1) (2013) 7–20.

[73] C. Benzmüller, Automating quantified conditional logics in HOL, in: F. Rossi (Ed.), 23rd International Joint Conference on Artificial Intelligence (IJCAI-13), AAAI Press, Beijing, China, 2013, pp. 746–753.

[74] C. Benzmüller, A. Farjami, X. Parent, A dyadic deontic logic in HOL, in: J. Broersen, C. Condoravdi, S. Nair, G. Pigozzi (Eds.), Deontic Logic and Normative Systems — 14th International Conference, DEON 2018, Utrecht, The Netherlands, 3-6 July, 2018, College Publications, 2018, pp. 33–50.

[75] L. Henkin, Completeness in the theory of types, Journal of Symbolic Logic 15 (2) (1950) 81–91.

[76] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, D. Mané, Concrete problems in AI safety, arXiv preprint arXiv:1606.06565 (2016).

[77] D. Hadfield-Menell, S. Russell, P. Abbeel, A. D. Dragan, Cooperative inverse reinforcement learning, in: D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (Eds.), Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, NeurIPS 2016, December 5-10, 2016, Barcelona, Spain, 2016, pp. 3909–3917.

[78] D. Hadfield-Menell, S. Milli, P. Abbeel, S. J. Russell, A. D. Dragan, Inverse reward design, in: I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, R. Garnett (Eds.), Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, NeurIPS 2017, December 4-9, 2017, Long Beach, CA, USA, 2017, pp. 6765–6774.

[79] V. C. Müller, Ethics of artificial intelligence and robotics, in: E. N. Zalta (Ed.), The Stanford Encyclopedia of Philosophy, Summer 2021 Edition, Metaphysics Research Lab, Stanford University, 2021.

[80] B. M. Lake, T. D. Ullman, J. B. Tenenbaum, S. J. Gershman, Building machines that learn and think like people, Behavioral and Brain Sciences 40 (2017) e253.

[81] J. Leike, D. Krueger, T. Everitt, M. Martic, V. Maini, S. Legg, Scalable agent alignment via reward modeling: a research direction, arXiv preprint arXiv:1811.07871 (2018).

[82] E. Yudkowsky, The AI alignment problem: why it is hard, and where to start, Symbolic Systems Distinguished Speaker (2016).

[83] T. R. Besold, A. d. Garcez, K. Stenning, L. van der Torre, M. van Lambalgen, Reasoning in non-probabilistic uncertainty: Logic programming and neural-symbolic computing as examples, Minds and Machines 27 (2017) 37–77.

[84] J. F. Horty, Agency and deontic logic, Oxford University Press, 2001.

[85] J. Broersen, Deontic epistemic STIT logic distinguishing modes of mens rea, Journal of Applied Logic 9 (2) (2011) 137–152.

[86] E. Lorini, Temporal logic and its application to normative reasoning, Journal of Applied Non-Classical Logics 23 (4) (2013) 372–399.

[87] K. van Berkel, T. Lyon, The varieties of ought-implies-can and deontic stit logic, in: F. Liu, A. Marra, P. Portner, F. van de Putte (Eds.), Deontic Logic and Normative Systems — 14th International Conference, DEON2020/21, Munich, Germany, 21-24 July, 2021, College Publications, 2021, pp. 55–76.

[88] L. van der Torre, Y.-H. Tan, Diagnosis and decision making in normative reasoning, Artificial Intelligence and Law 7 (1) (1999) 51–67.

[89] O. Roy, Deontic logic and game theory, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of deontic logic and normative systems., Vol. 2, College Publications, 2021, pp. 765–790.

[90] D. Gabbay, X. Parent, L. van der Torre, A geometrical view of I/O logic, arXiv preprint arXiv:1911.12837 (2019).

[91] L. Lindahl, J. Odelstad, The theory of joining-systems, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of deontic logic and normative systems., Vol. 1, College Publications, 2013, pp. 545–634.

[92] X. Sun, Proof theory, semantics and algebra for normative systems, Journal of Logic and Computation 28 (8) (2015) 1757–1779.

[93] A. Bochman, Explanatory nonmonotonic reasoning, World scientific, 2005.

[94] W. Carnielli, M. E. Coniglio, L. van der Torre, Input/output consequence relations: Reasoning with intensional contexts, Unpublished report (2009).

[95] X. Sun, Logic and games of norms: a computational perspective, Ph.D. thesis, University of Luxembourg, Luxembourg (2016).

[96] X. Parent, D. Gabbay, L. van der Torre, Intuitionistic basis for input/output logic, in: S. O. Hansson (Ed.), David Makinson on Classical Methods for Non-Classical Problems, Springer, 2014, pp. 263–286.

[97] X. Parent, L. van der Torre, Sing and dance!, in: F. Cariani, D. Grossi, J. Meheus, X. Parent (Eds.), Deontic Logic and Normative Systems — 12th International Conference, DEON 2014, Ghent, Belgium, July 12-15, 2014, Springer, 2014, pp. 149–165.

[98] X. Parent, L. van der Torre, I/O logics with a consistency check., in: J. Broersen, C. Condoravdi, S. Nair, G. Pigozzi (Eds.), Deontic Logic and Normative Systems — 14th International Conference, DEON 2018, Utrecht, The Netherlands, 3-6 July, 2018, College Publications, 2018, pp. 285–299.

[99] A. Stolpe, Normative consequence: The problem of keeping it whilst giving it up, in: R. van der Meyden, L. van der Torre (Eds.), Deontic Logic and Normative Systems — 9th International Conference, DEON 2008, Luxembourg, Luxembourg, July 15-18, 2008, Springer, 2008, pp. 174–188.

[100] A. Stolpe, Norms and norm-system dynamics, Ph.D. thesis, Department of Philosophy, University of Bergen, Norway (2008).

[101] A. Stolpe, A concept approach to input/output logic, Journal of Applied Logic 13 (3) (2015) 239–258.

[102] A. Ciabattoni, F. Gulisano, B. Lellmann, Resolving conflicting obligations in Mīmāṃsā: a sequent-based approach, in: J. Broersen, C. Condoravdi, S. Nair, G. Pigozzi (Eds.), Deontic Logic and Normative Systems — 14th International Conference, DEON 2018, Utrecht, The Netherlands, 3-6 July, 2018, College Publications, 2018, pp. 91–109.

[103] S. C. Tosatto, G. Boella, L. van der Torre, S. Villata, Abstract normative systems: Semantics and proof theory, in: G. Brewka, T. Eiter, S. A. Mcllraith (Eds.), Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning, AAAI Press, 2012, pp. 358–368.

[104] D. A. Ambrossio, Non-monotonic logics for access control: Delegation revocation and distributed policies, Ph.D. thesis, University of Luxembourg, Luxembourg (2017).

[105] M. Serramia, M. López-Sánchez, S. Moretti, J. A. Rodríguez-Aguilar, On the dominant set selection problem and its application to value alignment, Autonomous Agents and Multi-Agent Systems 35 (2) (2021) 1–38.

[106] A. Farjami, Discursive input/output logic: Deontic modals, and computation, Ph.D. thesis, University of Luxembourg, Luxembourg (2020).

[107] T. Nipkow, L. Paulson, M. Wenzel, Isabelle/HOL — A proof assistant for higher-order logic, Vol. 2283 of Lecture Notes in Computer Science, Springer, 2002.

[108] J. Blanchette, T. Nipkow, Nitpick: A counterexample generator for higher-order logic based on a relational model finder, in: M. Kaufmann, L. C. Paulson (Eds.), First International Conference on Interactive Theorem Proving, no. 6172 in Lecture Notes in Computer Science, Springer, 2010, pp. 131–146.

[109] J. C. Blanchette, S. Böhme, L. C. Paulson, Extending sledgehammer with SMT solvers, Journal of Automated Reasoning 51 (1) (2013) 109–128.

[110] W. J. Blok, D. Pigozzi, Algebraizable logics, Vol. 77, American Mathematical Society, 1989.

Appendix A

The semantical embedding outlined in Section 6 has been implemented in the higher-order proof assistant Isabelle/HOL [107]. Figures 1 and 2 display their respective encoding. Figure 1, after introducing type i for representing the elements of Boolean algebra, introduces the algebraic operators as constants in higher-order logic. The algebraic operators are also characterized in accordance with the definition of Boolean algebra.

Figure 1: Semantical embedding of Boolean algebra in Isabelle/HOL

```
theory IOBoolean
  imports Main

begin

typedecl i (* type for boolean elements *)
type_synonym τ = "(i⇒bool)"
type_synonym α = "(i⇒i⇒bool)"
consts   N :: "i⇒i⇒bool" ("N")   (* Normative system *)
consts  dis :: "i⇒i⇒i" (infixr"∨"50)
consts  con :: "i⇒i⇒i"  (infixr"∧"60)
consts  neg :: "i⇒i"  ("¬_"[52]53)
consts  top :: i ("⊤")
consts  bot :: i ("⊥")

axiomatization where
  COMdis  : "∀X. ∀Y. (X ∨ Y) = (Y ∨ X)" and
  COMcon  : "∀X. ∀Y. (X ∧ Y) = (Y ∧ X)" and
  ASSdis  : "∀X. ∀Y. ∀ Z. (X ∨ (Y ∨ Z)) = (X ∨ (Y ∨ Z))" and
  ASScon  : "∀X. ∀Y. ∀ Z. (X ∧ (Y ∧ Z)) = (X ∧ (Y ∧ Z))" and
  IDEdis  : "∀X. (X ∨ ⊥) = X" and
  IDEcon  : "∀X. (X ∧ ⊤) = X" and
  COMPdis : "∀X. (X ∨ ¬X) = ⊤" and
  COMPcon : "∀X. (X ∧ (¬X) ) = ⊥" and
  Ddiscon : "∀X. ∀Y. ∀ Z. (X ∨ (Y ∧ Z)) = ((X ∨ Y) ∧ (X ∨ Z))" and
  Dcondis : "∀X. ∀Y. ∀ Z. (X ∧ (Y ∨ Z)) = ((X ∧ Y) ∨ (X ∧ Z))"
```

Figure 2 displays the semantical embedding of I/O operations (out_i) in HOL, including the definition of the upward-closed set operator and saturated set.

Figure 3 shows some experiments via the model and countermodel finder Nitpick [108], and prove some facts about I/O operations using automatic theorem provers (*auto* and *meson*) via the Sledgehammer tool [109].

Figure 2: Semantical embedding of out_i in Isabelle/HOL

```
definition ordeIOB :: "i⇒τ" (infixr"≤"80) where "X ≤ Y ≡ ((X ∧ Y) = X)"
definition satuIOB :: "τ ⇒ bool" ("Saturated")  where
"Saturated V ≡ ∀X. ∀Y. (((V (X ∨ Y))⟶ (V X ∨ V Y)) ∧ ((V X ∧ (X≤Y)) ⟶ V Y))"
definition UpwardIOB :: "τ ⇒ τ" ("Up") where " Up V ≡ λX. (∃Z . (V Z ∧ Z ≤ X))"


definition outI :: "α ⇒ τ ⇒ τ" ("○I<_;_>")
  where "○I<M;A> ≡ λX. ∃U. (∃Y. (∃Z. (A Z ∧ (Z=Y) ∧ M Y U ∧ (U ≤ X) ) ) )"

definition outII :: "α ⇒ τ ⇒ τ" ("○II<_;_>")
  where "○II<M;A> ≡ λX. ∃U. (∃Y. (∃Z. (A Z ∧ (Z≤Y) ∧ M Y U ∧ (U = X) ) ) )"

definition out1 :: "α ⇒ τ ⇒ τ" ("○1<_;_>")
  where "○1<M;A> ≡ λX. ∃U. (∃Y. (∃Z. (A Z ∧ (Z≤Y) ∧ M Y U ∧ (U ≤ X) ) ) )"

definition out2 :: "α ⇒τ ⇒ τ" ("○2<_;_>")
  where "○2<M;A> ≡ λX. (∀V. ( (Saturated V) ∧ (∀U. (A U ⟶ V U ))
                    ⟶ (∃Y. (∃Z. ( (V Y) ∧ (M Y Z) ∧ (Z≤X)) )) ))"

definition out3 :: "α ⇒ τ ⇒ τ" ("○3<_;_>")
  where "○3<M;A> ≡ λX. (∀V. ( ((V = Up V) ∧ (∀U. (A U ⟶ V U ))∧(∀W.(∃Y.(V Y ∧ (M Y W)))⟶V W))
                    ⟶ (∃Y. (∃Z. ((Z≤ X)∧ N Y Z ∧ V Y) )) ))"
```

Figure 3: Some experiments on out_i in Isabelle/HOL

```
consts a :: i
consts b :: i
consts c :: i
consts W :: τ

lemma "○1<N;((λX. X = a))> a" nitpick [user_axioms,expect=genuine,show_all] oops

lemma "(○1<N;((λX. X = a))> x ∧ (b ≤ a)) ⟶ ○1<N;((λX. X = b))> x"
  nitpick [satisfy,user_axioms,show_all,expect=genuine,card=4] oops

lemma "(○1<N;((λX. X= a))> x ∧ ○1<N;((λX. X= b))> x) ⟶ ○1<N;((λX. X= (a∨b)))> x"
  nitpick [user_axioms,expect=genuine,show_all] oops

lemma "(○1<N;((λX. X= a))> x ∧ ○1<N;((λX. X= a))> y) ⟶ ○1<N;((λX. X= a))> (x ∧ y)"
  nitpick [user_axioms,expect=genuine,show_all] oops

lemma "(○3<N;((λX. X= a))> x) ⟶ (○2<N;((λX. X= a ))> x)"
  nitpick [user_axioms,expect=genuine,show_all] oops

lemma "(○1<N;((λX. X= a))> x) ⟶ (○2<N;((λX. X= a))> x)" unfolding Defs by auto

lemma "(○1<N;((λX. X= a))> x) ⟶ (○3<N;((λX. X= a))> x)" unfolding Defs by meson

lemma "(○3<N;((λX. X= a))> x) ⟶ (○1<N;((λX. X= a))> x)"
  nitpick [user_axioms,expect=genuine,show_all] oops
```

In Figure 4, the first two lemmas prove the soundness of out_1. The next two lemmas show the factual detachment of this output operation. The last

two lemmas illustrate the soundness of out_I and out_{II} where the depth of inference is one.

Figure 4: Soundness of out_1 in Isabelle/HOL

```
(*Soundness for Out1*)
lemma "(○₁<N;((λX. X= a))> x ∧ (x ≤ y)) ⟶ ○₁<N;((λX. X= a))> y"  unfolding Defs
  by (metis COMPcon COMPdis COMcon COMdis Dcondis Ddiscon IDEcon IDEdis ordeIOB_def)
lemma "(○₁<N;((λX. X= a))> x ∧ (b ≤ a)) ⟶ ○₁<N;((λX. X= b))> x" unfolding Defs
  by (metis COMPcon COMPdis COMcon COMdis Dcondis Ddiscon IDEcon IDEdis)

lemma "(N a b) ⟶ (○₁<N;((λX. X= a))> b)"
  unfolding Defs
  by (metis COMPcon COMcon COMdis Dcondis Ddiscon IDEdis)

lemma "(N a b ∧ W a) ⟶ (○₁<N;W> b)"
  unfolding Defs
  by (metis COMPcon COMcon COMdis Dcondis Ddiscon IDEdis)

lemma "((N a b ∧ (b ≤ c)) ⟶ (○₁<N;((λX. X= a))> c))"
  unfolding Defs
  by (metis COMPdis COMcon COMdis Dcondis Ddiscon IDEcon)

lemma "((N a b ∧ (c ≤ a)) ⟶ (○₁<N;((λX. X= c))> b))"
  unfolding Defs
  by (metis COMPcon COMcon COMdis Dcondis Ddiscon  IDEdis)

lemma "((N a b ∧ (b ≤ c)) ⟶ (○I<N;((λX. X= a))> c))"
  unfolding Defs using ordeIOB_def outI_def by auto

 lemma "((N a b ∧ (c ≤ a)) ⟶ (○II<N;((λX. X= c))> b))"
  unfolding Defs using ordeIOB_def outII_def by auto
```

Figure 5 shows the soundness of out_2 and out_3 for a depth of one. The input/output operations introduced by Makinson and van der Torre [31] are implemented in Figure 6. The implementations are based on the *reversibility of rules in the derivation systems*. The four input/output operations introduced in [31] were built over the *simple-minded output operation* (see Section 3).

Figure 5: Soundness of out_2 and out_3 in Isabelle/HOL

```
(* out2 depth1-soundness *)
lemma "((N a b ∧ (b ≤ c)) ⟶ (○₂<N;((λX. X= a))> c))"
  unfolding Defs by auto

lemma "((N a b ∧ (c ≤ a)) ⟶ (○₂<N;((λX. X= c))> b))"
  unfolding Defs
  by (metis COMPcon COMcon COMdis Dcondis Ddiscon IDEdis)

lemma "(N a b ∧ N c b) ⟶ (○₂<N;((λX. X= (a ∨ c)))> b)"
  unfolding Defs
  by (metis COMPdis COMcon COMdis Dcondis Ddiscon IDEcon)

(* out3 depth1-soundness *)
lemma "((N a b ∧ (b ≤ c)) ⟶ (○₃<N;((λX. X= a))> c))"
  unfolding Defs by auto

lemma "((N a b ∧ (c ≤ a)) ⟶ (○₃<N;((λX. X= c))> b))"
  unfolding Defs
  by (metis COMPdis COMcon COMdis Dcondis Ddiscon IDEcon)

lemma "(N a b ∧ N b c) ⟶ (○₃<N;((λX. X= a))> c)"
  unfolding Defs
  by (metis COMPcon COMcon COMdis Dcondis Ddiscon IDEdis)
```

Figure 6: Semantical embedding of output operations in Isabelle/HOL

```
theory outoperation imports IOBoolean
begin

definition Rout :: "α ⇒ τ⇒i⇒i⇒bool" ("Rout<_;_>")
  where "Rout<M;A> ≡ λZ. λX. ∃U. (∃Y. ( (A Z ∧ (Z≤Y) ∧ M Y U ∧ (U ≤ X)) ) )"
definition Sub_rel :: "α⇒α⇒bool" where "Sub_rel R Q ≡ ∀ u v. R u v ⟶ Q u v"

(* OUT1 orginal *)
definition Close_AND :: "α ⇒ bool" where "Close_AND Q ≡ ∀u v w.(Q u v ∧ Q u w ⟶ (Q u (v ∧ w)))"
definition TCAND :: "α ⇒ α" where "TCAND R ≡ λ X Y. ∀ Q. Close_AND Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition outAND :: "α⇒ τ ⇒ τ" ("○AND<_;_>") where "○AND<M;A> ≡ λX. ∃Y. TCAND (Rout<M;A>) Y X"
(* OUT2 orginal *)
definition Close_OR :: "α ⇒ bool" where "Close_OR Q ≡ ∀ u v w. (Q v u ∧ Q w u ⟶ (Q (v ∨ w) u))"
definition TCOR :: "α ⇒ α" where "TCOR R ≡ λ X Y. ∀ Q. Close_OR Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition outOR :: "α⇒ τ ⇒ τ" ("○OR<_;_>") where "○OR<M;A> ≡ λX. ∃Y. TCOR (Rout<M;A>) Y X"
definition outORAND :: "α⇒ τ ⇒ τ" ("○ORAND<_;_>")
  where "○ORAND<M;A> ≡ λX. ∃Y. TCAND (TCOR (Rout<M;A>))  Y X "
(* OUT3 orginal *)
definition Close_CT :: "α ⇒ bool" where "Close_CT Q ≡ ∀ u v w. (Q v u ∧ Q (v ∧ u) w ⟶ (Q v w))"
definition TCCT :: "α ⇒ α" where "TCCT R ≡ λ X Y. ∀ Q. Close_CT Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition outCT :: "α⇒ τ ⇒ τ" ("○CT<_;_>") where "○CT<M;A> ≡ λX. ∃Y. TCCT (Rout<M;A>) Y X"
definition outCTAND :: "α⇒ τ ⇒ τ" ("○CTAND<_;_>")
  where "○CTAND<M;A> ≡ λX. ∃Y. TCAND (TCCT (Rout<M;A>)) Y X"
(* OUT4 orginal *)
definition outCTORAND :: "α⇒ τ ⇒ τ" ("○CTORAND<_;_>")
  where "○CTORAND<M;A> ≡ λX. ∃Y. TCAND (TCOR (TCCT (Rout<M;A>))) Y X"
```

The following lemmas (see Fig. 7) show the automation capability of implemented output operations for the *simple-minded output operation* ($out_1^{AND}(N)$) as introduced by Makinson and van der Torre [31].

Figure 7: Semantical embedding of output operations in Isabelle/HOL

```
86 lemma imp : "○₁<N;((λX. X= a)) > b ⟶  ○AND<N;((λX. X= a)) > b"
87   using out1_def  Rout_def Sub_rel_def Close_AND_def  TCAND_def unfolding Defst outAND_def
88   by auto
89
90 lemma "(N a b) ⟶ (○AND<N;((λX. X= a))> b)"
91   using imp  Rout_def Sub_rel_def Close_AND_def  TCAND_def unfolding  Defst outAND_def
92   by (metis COMPcon COMPdis COMcon COMdis Dcondis IDEcon IDEdis ordeIOB_def )
93
94 lemma "(N a b ∧ N a c) ⟶ (○AND<N;((λX. X= a))> (b ∧ c))"
95   using imp   Rout_def Sub_rel_def Close_AND_def  TCAND_def
96   unfolding  Defst  outAND_def TCAND_def
97   by (metis COMPcon COMPdis Dcondis IDEcon IDEdis ordeIOB_def)
98
99 lemma imp2 : "○₁<N;((λX. X= a))> b ⟶ ○AND<N;((λX. X= a))> b"
100  using out1_def  Rout_def Sub_rel_def Close_OR_def  TCOR_def unfolding Defst outOR_def
101  by auto
102
103 lemma "((○₁<N;((λX. X= a ))> b) ∧  (○₁<N;((λX. X= a))> c)) ⟶ ○AND<N;((λX. X= a))> (b ∧ c)"
104  unfolding  Defst  outAND_def TCAND_def Close_AND_def out1_def
105  by metis
```

The proof system of input/output logic can be implemented directly in Isabelle/HOL—see Fig. 8 and 9. The idea is based on an (universal) order of rules in a derivation. The ordering of rules and closure operation are the main ways of defining the derivation systems (for more details, see Section 3.) For example, in line 27 of Fig.9, `derSIEQO` introduces the derivation system $derive_I$ with the rules in $\{SI, EQO\}$ and in lines 51–52, `derSIWOCTORAND` introduce the derivation system $derive_4$ with the rules in $\{SI, WO, CT, OR, AND\}$.

Figure 8: Semantical embedding of I/O proof systems in Isabelle/HOL

```
theory outsystems imports IOBoolean
begin

definition Close_EQO :: "α ⇒ bool" where "Close_EQO Q ≡ ∀ u v w.(Q u v ∧ (v = w) ⟶ (Q u w))"
definition Close_EQI :: "α ⇒ bool" where "Close_EQI Q ≡ ∀ u v w.(Q u v ∧ (u = w) ⟶ (Q w v))"
definition Close_SI :: "α ⇒ bool" where "Close_SI Q ≡ ∀ u v w.(Q u v ∧ (w ≤ u) ⟶ (Q w v))"
definition Close_WO :: "α ⇒ bool" where "Close_WO Q ≡ ∀ u v w.(Q u v ∧ (v ≤ w) ⟶ (Q u w))"
definition Close_AND :: "α ⇒ bool" where "Close_AND Q ≡ ∀ u v w.(Q u v ∧ Q u w ⟶ (Q u (v ∧ w)))"
definition Close_OR :: "α ⇒ bool" where "Close_OR Q ≡ ∀ u v w.(Q v u ∧ Q w u ⟶ (Q (v ∨ w) u))"
definition Close_CT :: "α ⇒ bool" where "Close_CT Q ≡ ∀ u v w.(Q v u ∧ Q (v ∧ u) w ⟶ (Q v w))"

definition Sub_rel :: "α⇒α⇒bool" where "Sub_rel R Q ≡ ∀ u v. R u v ⟶ Q u v"
definition TCEQO :: "α ⇒ α" where "TCEQO R ≡ λ X Y. ∀ Q. Close_EQO Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCEQI :: "α ⇒ α" where "TCEQI R ≡ λ X Y. ∀ Q. Close_EQI Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCSI :: "α ⇒ α" where "TCSI R ≡ λ X Y. ∀ Q. Close_SI Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCWO :: "α ⇒ α" where "TCWO R ≡ λ X Y. ∀ Q. Close_WO Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCAND :: "α ⇒ α" where "TCAND R ≡ λ X Y. ∀ Q. Close_AND Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCOR :: "α ⇒ α" where "TCOR R ≡ λ X Y. ∀ Q. Close_OR Q ⟶ (Sub_rel R Q ⟶ Q X Y)"
definition TCCT :: "α ⇒ α" where "TCCT R ≡ λ X Y. ∀ Q. Close_CT Q ⟶ (Sub_rel R Q ⟶ Q X Y)"

definition derSI :: "α⇒α" ("derSI<_>") where "derSI<M> ≡ TCSI (M)"
definition derWO :: "α⇒α" ("derWO<_>") where "derWO<M> ≡ TCWO (M)"
definition derAND :: "α⇒α" ("derAND<_>") where "derAND<M> ≡ TCAND (M)"
definition derOR :: "α⇒α" ("derOR<_>") where "derOR<M> ≡ TCOR (M)"
definition derCT :: "α⇒α" ("derCT<_>") where "derCT<M> ≡ TCCT (M)"
```

Figure 9: Semantical embedding of I/O proof systems in Isabelle/HOL

```
definition derSIEQO :: "α⇒α" ("derSIEQO<_>") where "derSIEQO<M> ≡ TCSI (TCEQO (M))"
definition derWOEQI :: "α⇒α" ("derWOEQI<_>") where "derWOEQI<M> ≡ TCWO (TCEQI (M))"

(*Derive1-Per*)
definition derSIWO :: "α⇒α" ("derSIWO<_>") where "derSIWO<M> ≡ TCWO (TCSI (M))"
definition derWOSI :: "α⇒α" ("derWOSI<_>") where "derWOSI<M> ≡TCSI (TCWO (M))"

(*Derive1-Ob*)
definition derSIWOAND :: "α⇒α" ("derSIWOAND<_>") where "derSIWOAND<M> ≡ TCAND (TCWO (TCSI (M)))"

(*Derive2-Per*)
definition derSIWOOR :: "α⇒α" ("derSIWOOR<_>") where "derSIWOOR<M> ≡ TCOR (TCWO (TCSI (M)))"

(*Derive2-Ob*)
definition derSIWOORAND :: "α⇒α" ("derSIWOORAND<_>")
  where "derSIWOORAND<M> ≡ TCAND (TCOR (TCWO (TCSI (M))))"

(*Derive3-Ob*)
definition derSIWOCT :: "α⇒α" ("derSIWOCT<_>")
  where "derSIWOCT<M> ≡ TCCT (TCWO (TCSI (M)))"
definition derSIWOCTAND :: "α⇒α" ("derSIWOCTAND<_>")
  where "derSIWOCTAND<M> ≡ TCAND (TCCT (TCWO (TCSI (M))))"

(*Derive4-Ob*)
definition derSIWOCTORAND :: "α⇒α" ("derSIWOCTORAND<_>")
  where "derSIWOCTORAND<M> ≡ TCAND (TCOR (TCCT (TCWO (TCSI (M)))))"
```

One advantage of implementing the proof system of I/O logic, besides the output operations, is that completeness theorems can be checked. For example, the completeness of *out*1, as shown in Fig. 10, is checked in lines 70–73. Lines 61 and 62 show the AND closure. Lines 64–67 demonstrate automation of the implementation for a normative system M.

Figure 10: Completeness checking of *out*1 in Isabelle/HOL

```
61 lemma "Close_AND (TCAND N)" unfolding Defst TCAND_def
62   by metis
63
64 lemma "(M a b ∨ (∃ y. M y b ∧ (a ≤ y))) ⟶ derSI<M> a b"
65 using   Sub_rel_def Close_SI_def  TCSI_def
66   unfolding Defst and Defs  derSI_def
67   by metis
68
69 (*OUT1  completness*)
70 lemma "(○₁<N;((λX. X = a))> y ⟶ derSIWO<N> a y)"
71   using   Sub_rel_def Close_SI_def Close_WO_def  TCSI_def   TCWO_def
72   unfolding Defst and Defs  derSI_def Sub_rel_def TCWO_def TCSI_def
73   by metis
```

The proof theoretical difference of different I/O systems can be examined (cf. Fig. 11). For example, lines 81–85 show that the implemented derivation system `derSIWOOR` ($derive_2$) is sound for the OR rule for a depth of one.

Figure 11: Some experiments on I/O proof systems in Isabelle/HOL

```
75 lemma "((N a b ∧ N a c) ∧ (N x y ⟶ (N a b ∨ N a c)))
76       ⟶  derSIWOAND<N> a (b∧c)" (* AND Closed *)
77   using   Sub_rel_def Close_SI_def  TCSI_def   TCWO_def
78   unfolding Defst and Defs  derSI_def Sub_rel_def TCWO_def
79   by auto
80
81 lemma "((N a b ∧ N c b) ∧ (N x y ⟶ (N a b ∨ N c a)))
82       ⟶  derSIWOOR<N> (a ∨ c) b" (*OR Closed *)
83   using   Sub_rel_def Close_SI_def  TCSI_def   TCWO_def
84   unfolding Defst and Defs  derSI_def Sub_rel_def TCWO_def
85   by auto
86
87 lemma "((N a b ∧ N (a∧b) c) ∧ (N x y ⟶ (N a b ∨ N(a∧b) c)))
88       ⟶  derSIWOCT<N> a c" (*CT Closed *)
89   using   Sub_rel_def Close_SI_def  TCSI_def   TCWO_def
90   unfolding Defst and Defs  derSI_def Sub_rel_def TCWO_def
91   by (smt Close_CT_def Sub_rel_def TCCT_def TCWO_def derSIWOCT_def)
92
93 lemma "((N a b ∧ N (a∧b) c) ∧ (N x y ⟶ (N a b ∨ N(a∧b) c)))
94       ⟶  derSIWOCTAND<N> a c" (*CT Closed *)
95   using Close_CT_def Sub_rel_def TCCT_def TCWO_def derSIWOCT_def
96   unfolding Defst and Defs  derSI_def Sub_rel_def TCWO_def TCOR_def
97    by (metis (no_types, hide_lams) Sub_rel_def TCSI_def)
```

Appendix B

Proof for Theorem 1: Zero Boolean I/O operation

Outline of proof for soundness: for the input set $A \subseteq Ter(B)$, it is shown that if $(A, x) \in derive_0^{\mathcal{B}}(N)$, then $x \in out_0^{\mathcal{B}}(N, A)$. By definition, $(A, x) \in derive_0^{\mathcal{B}}(N)$ iff $(a, x) \in derive_0^{\mathcal{B}}(N)$ for some $a \in A$. By induction on the length of the derivation and since $out_0^{\mathcal{B}}(N)$ validates EQI and EQO, if $(a, x) \in derive_0^{\mathcal{B}}(N)$, then $x \in out_0^{\mathcal{B}}(N, \{a\})$. Thus, by definition of $out_0^{\mathcal{B}}$, we have $x \in out_0^{\mathcal{B}}(N, A)$. If $A = \{\}$, then by definition $(A, x) \notin derive_0^{\mathcal{B}}(N)$. The outline works for the soundness of other systems presented in this appendix as well.

Soundness: $out_0^{\mathcal{B}}(N)$ validates EQI and EQO.

EQI: It needs to be shown that

$$\text{EQI}\ \frac{x \in Eq(N(Eq(a))) \qquad a =_{\mathcal{B}} b}{x \in Eq(N(Eq(b)))}$$

If $x \in Eq(N(Eq(a)))$, then there are t_1 and t_2 such that $t_1 =_{\mathcal{B}} a$ and $t_2 =_{\mathcal{B}} x$ and $(t_1, t_2) \in N$. If $a =_{\mathcal{B}} b$ then $t_1 =_{\mathcal{B}} b$. Hence, by definition, $x \in Eq(N(Eq(b)))$.

EQO: It needs to be shown that

$$\text{EQO}\ \frac{x \in Eq(N(Eq(a))) \qquad x =_{\mathcal{B}} y}{y \in Eq(N(Eq(a)))}$$

If $x \in Eq(N(Eq(a)))$, then there are t_1 and t_2 such that $t_1 =_{\mathcal{B}} a$ and $t_2 =_{\mathcal{B}} x$ and $(t_1, t_2) \in N$. If $x =_{\mathcal{B}} y$ then $t_2 =_{\mathcal{B}} y$. Hence, by definition, $y \in Eq(N(Eq(a)))$.

Completeness: $out_0^{\mathcal{B}}(N) \subseteq derive_0^{\mathcal{B}}(N)$.[11]

It is shown that if $x \in Eq(N(Eq(A)))$, then $(A, x) \in derive_0^{\mathcal{B}}(N)$. Suppose that $x \in Eq(N(Eq(A)))$, then there are t_1 and t_2 such that $t_1 =_{\mathcal{B}} a$ and $a \in A$, and $t_2 =_{\mathcal{B}} x$ such that $(t_1, t_2) \in N$.

[11] For the completeness proofs, if $A = \{\}$, then by definition of $Eq(\{\}) = \{\}$ and $Up(\{\}) = \{\}$, we have $x \notin out_i^{\mathcal{B}}(N, \{\}) = \{\}$.

$$EQO\ \dfrac{\dfrac{(t_1,t_2)\qquad t_2 =_{\mathcal{B}} x}{(t_1,x)}\qquad t_1 =_{\mathcal{B}} a}{(a,x)}$$

Thus, $x \in derive_0^{\mathcal{B}}(N,a)$ and then $x \in derive_0^{\mathcal{B}}(N,A)$.

Proof for Theorem 1: Simple-I Boolean I/O operation

Soundness: $out_I^{\mathcal{B}}(N)$ validates SI and EQO.

SI: It needs to be shown that

$$\mathrm{SI}\ \dfrac{x \in Eq(N(Up(a)))\qquad b \le a}{x \in Eq(N(Up(b)))}$$

If $x \in Eq(N(Up(a)))$, then $\exists t_1$ such that $a \le t_1$ and $(t_1,x) \in N$ or $((t_1,y) \in N$ and $y =_{\mathcal{B}} x)$. Hence, if $b \le a$, we have $b \le t_1$ and then $x \in Eq(N(Up(b)))$.

EQO: It needs to be shown that

$$\mathrm{EQO}\ \dfrac{x \in Eq(N(Up(a)))\qquad x =_{\mathcal{B}} y}{y \in Eq(N(Up(a)))}$$

If $x \in Eq(N(Up(a)))$, then by definition of $Eq(X)$, if $x =_{\mathcal{B}} y$, we have $y \in Eq(N(Up(a)))$.

Completeness: $out_I^{\mathcal{B}}(N) \subseteq derive_I^{\mathcal{B}}(N)$.

It is shown that if $x \in Eq(N(Up(A)))$, then $(A,x) \in derive_I^{\mathcal{B}}(N)$. Suppose that $x \in Eq(N(Up(A)))$, then there is t_1 such that $a \le t_1$ and $(t_1,x) \in N$ or $((t_1,y) \in N$ and $y =_{\mathcal{B}} x)$ for $a \in A$. There are two cases:

$$SI\ \dfrac{(t_1,x)\qquad a \le t_1}{(a,x)}\qquad\qquad EQO\ \dfrac{(t_1,y)\qquad y =_{\mathcal{B}} x}{SI\ \dfrac{(t_1,x)\qquad a \le t_1}{(a,x)}}$$

Thus, $x \in derive_I^{\mathcal{B}}(N,a)$ and then $x \in derive_I^{\mathcal{B}}(N,A)$.

Proof for Theorem 1: Simple-II Boolean I/O operation

Soundness: $out^{\mathcal{B}}_{II}(N)$ validates WO and EQI.

WO: It needs to be shown that

$$\text{WO}\,\frac{x \in Up(N(Eq(a))) \qquad x \leq y}{y \in Up(N(Eq(a)))}$$

If $x \in Up(N(Eq(a)))$, then there is t_1 such that $t_1 \leq x$ and $(a, t_1) \in N$ or $((b, t_1) \in N$ and $a =_{\mathcal{B}} b)$. If $x \leq y$, then $t_1 \leq y$ and we have $y \in Up(N(Eq(a)))$.

EQI: It needs to be shown that

$$\text{EQI}\,\frac{x \in Up(N(Eq(a))) \qquad a =_{\mathcal{B}} b}{x \in Up(N(Eq(b)))}$$

If $x \in Up(N(Eq(a)))$, then there is t_1 such that $t_1 \leq x$ and $(a, t_1) \in N$ or $((c, t_1) \in N$ and $a =_{\mathcal{B}} c)$. Hence, if $a =_{\mathcal{B}} b$, then by definition $x \in Up(N(Eq(b)))$.

Completeness: $out^{\mathcal{B}}_{II}(N) \subseteq derive^{\mathcal{B}}_{II}(N)$.

It is shown that if $x \in Up(N(Eq(A)))$, then $(A, x) \in derive^{\mathcal{B}}_{II}(N)$. Suppose that $x \in Up(N(Eq(A)))$, then there is t_1 such that $t_1 \leq x$ and $(a, t_1) \in N$ or $((b, t_1) \in N$ and $a =_{\mathcal{B}} b)$ for $a \in A$. There are two cases:

$$WO\,\frac{(a, t_1) \qquad t_1 \leq x}{(a, x)} \qquad\qquad EQI\,\frac{(b, t_1) \qquad a =_{\mathcal{B}} b}{WO\,\dfrac{(a, t_1) \qquad\qquad t_1 \leq x}{(a, x)}}$$

Thus, $x \in derive^{\mathcal{B}}_{II}(N, a)$ and then $x \in derive^{\mathcal{B}}_{II}(N, A)$.

Proof for Theorem 1: Simple-minded Boolean I/O operation

Soundness: $out^{\mathcal{B}}_{1}(N)$ validates SI and WO.

SI: It needs to be shown that

$$\text{SI}\,\frac{x \in Up(N(Up(a))) \qquad b \leq a}{x \in Up(N(Up(b)))}$$

Since $b \leq a$ we have $Up(a) \subseteq Up(b)$. Hence, $N(Up(a)) \subseteq N(Up(b))$ and therefore $Up(N(Up(a))) \subseteq Up(N(Up(b)))$.

WO: It needs to be show shown that

$$\text{WO}\,\frac{x \in Up(N(Up(a))) \qquad x \leq y}{y \in Up(N(Up(a)))}$$

Since $Up(N(Up(a)))$ is upward-closed and $x \leq y$, we have $y \in Up(N(Up(a)))$.

Completeness: $out_1^{\mathcal{B}}(N) \subseteq derive_1^{\mathcal{B}}(N)$.

It is shown that if $x \in Up(N(Up(A)))$, then $(A, x) \in derive_1^{\mathcal{B}}(N)$. Suppose that $x \in Up(N(Up(A)))$, then there is y_1 such that $y_1 \in N(Up(A))$, $y_1 \leq x$, and there is t_1 such that $(t_1, y_1) \in N$ and $a \leq t_1$ for $a \in A$.

$$SI\,\frac{a \leq t_1 \qquad WO\,\dfrac{(t_1, y_1) \quad y_1 \leq x}{(t_1, x)}}{(a, x)}$$

Thus, $x \in derive_1^{\mathcal{B}}(N, a)$ and then $x \in derive_1^{\mathcal{B}}(N, A)$.

Proof for Theorem 1: Basic Boolean I/O operation

Soundness: $out_2^{\mathcal{B}}(N)$ *validates* SI, WO *and* OR.

OR: It needs to be shown that

$$\text{OR}\,\frac{x \in out_2^{\mathcal{B}}(N, \{a\}) \qquad x \in out_2^{\mathcal{B}}(N, \{b\})}{x \in out_2^{\mathcal{B}}(N, \{a \vee b\})}$$

Suppose that $\{a \vee b\} \subseteq V$, since V is saturated we have $a \in V$ or $b \in V$. Suppose that $a \in V$, in this case since $out_2^{\mathcal{B}}(N, \{a\}) \subseteq Up(N(V))$, we have $x \in out_2^{\mathcal{B}}(N, \{a \vee b\})$.

Completeness: $out_2^{\mathcal{B}}(N) \subseteq derive_2^{\mathcal{B}}(N)$.

Suppose that $x \notin derive_2^{\mathcal{B}}(N, A)$, then by monotony of the derivability operation, there is a maximal set V such that $A \subseteq V$ and $x \notin derive_2^{\mathcal{B}}(N, V)$.[12] V is saturated because:

(a) Suppose that $a \in V$ and $a \leq b$, by definition of V we have $(a, x) \notin derive_2^{\mathcal{B}}(N)$. It needs to be shown that $x \notin derive_2^{\mathcal{B}}(N, b)$ and since V is maximal, we have $b \in V$. Suppose that $(b, x) \in derive_2^{\mathcal{B}}(N)$. We have

$$\text{SI}\,\frac{(b, x) \qquad a \leq b}{(a, x)}$$

That is a contradiction of $(a, x) \notin derive_2^{\mathcal{B}}(N)$.

(b) Suppose that $a \vee b \in V$, by definition of V we have $x \notin derive_2^{\mathcal{B}}(N, a \vee b)$. It needs to be shown that $x \notin derive_2^{\mathcal{B}}(N, a)$ or $x \notin derive_2^{\mathcal{B}}(N, b)$. Suppose that $x \in derive_2^{\mathcal{B}}(N, a)$ and $x \in derive_2^{\mathcal{B}}(N, b)$, then we have

$$OR\,\frac{(a, x) \qquad (b, x)}{(a \vee b, x)}$$

That is a contradiction of $x \notin derive_2^{\mathcal{B}}(N, a \vee b)$.

Therefore, we have $x \notin Up(N(V))$ (that is equal to $x \notin out_1^{\mathcal{B}}(N, V)$) and so $x \notin out_2^{\mathcal{B}}(N, A)$.

Proof for Theorem 1: Reusable Boolean I/O operation
Soundness: $out_3^{\mathcal{B}}(N)$ *validates* SI, WO *and* T.

T: It needs to be shown that

$$\text{T}\,\frac{x \in out_3^{\mathcal{B}}(N, \{a\}) \qquad y \in out_3^{\mathcal{B}}(N, \{x\})}{y \in out_3^{\mathcal{B}}(N, \{a\})}$$

Suppose that X is the smallest set such that $\{a\} \subseteq X = Up(X) \supseteq N(X)$. Since $x \in out_3^{\mathcal{B}}(N, \{a\})$ we have $x \in X$, and from $y \in out_3^{\mathcal{B}}(N, \{x\})$ we have $y \in X$. Thus, $y \in out_3^{\mathcal{B}}(N, \{a\})$.

[12] Consider the set $E = \{V : A \subseteq V \text{ and } x \notin deriv(G, V)\}$. This set is a partially ordered set which is ordered by the monotony property of derivation. Every chain (any set linearly ordered by set-theoretic inclusion) has an upper bound (the union of the sets) in E. So set E has at least a maximal element by Zorn's lemma.

Completeness: $out_3^{\mathcal{B}}(N) \subseteq derive_3^{\mathcal{B}}(N)$.

Suppose that $x \notin derive_3^{\mathcal{B}}(N,a)$. It is necessary to find B such that $a \in B = Up(B) \supseteq N(B)$ and $x \notin Up(N(B))$. Put $B = Up(\{a\} \cup derive_3^{\mathcal{B}}(N,a))$. It is shown that $N(B) \subseteq B$. Suppose that $y \in N(B)$, then there is $b \in B$ such that $(b,y) \in N$. It is shown that $y \in B$. Since $b \in B$, there are two cases:

- $b \geq a$: in this case we have $(a,y) \in derive_3^{\mathcal{B}}(N)$ since $(b,y) \in derive_3^{\mathcal{B}}(N)$ and we have

$$SI\ \frac{(b,y) \qquad a \leq b}{(a,y)}$$

- $\exists z \in derive_3^{\mathcal{B}}(N,a), b \geq z$: in this case we have

$$T\ \frac{(a,z) \qquad SI\ \dfrac{(b,y) \qquad z \leq b}{(z,y)}}{(a,y)}$$

It only needs to shown that $x \notin Up(N(B)) = out_1^{\mathcal{B}}(N, \{a\} \cup derive_3^{\mathcal{B}}(N,a))$. Suppose that $x \in Up(N(B))$, then there is y_1 such that $x \geq y_1$ and $\exists t_1$, $(t_1, y_1) \in N$ and $t_1 \in Up(\{a\} \cup derive_3^{\mathcal{B}}(N,a))$. There are two cases:

- $t_1 \geq a$: in this case we have

$$SI\ \frac{(t_1,y_1) \qquad a \leq t_1}{WO\ \dfrac{(a,y_1) \qquad y_1 \leq x}{(a,x)}}$$

- $\exists z_1 \in derive_3^{\mathcal{B}}(N,a), z_1 \leq t_1$: in this case we have

$$T\ \frac{(a,z_1) \qquad SI\ \dfrac{(t_1,y_1) \qquad z_1 \leq t_1}{(z_1,y_1)}}{WO\ \dfrac{(a,y_1) \qquad y_1 \leq x}{(a,x)}}$$

Thus, in both cases, $(a,x) \in derive_3^{\mathcal{B}}(N)$ and then $x \in derive_3^{\mathcal{B}}(N,a)$, and that is a contradiction.

Proof for Theorem 2

The proof is based on the reversibility of inference rules, as studied by Makinson and van der Torre [31].

Lemma 4. *Let D be any derivation using at most EQI, SI, WO, OR, AND, CT. Then, there is a derivation D' of the same root from a subset of leaves that applies AND only at the end.*

Proof 1. *See Observation 18 [31].*

The main point of the observation is that it is possible to reverse the order of rules AND, WO to WO, AND; AND, SI to SI, AND; AND, OR to OR, AND and finally AND, CT to SI, CT or CT, AND. It is also possible to reverse the order of the AND and EQI rules as follows:

$$AND\frac{EQI\dfrac{\dfrac{(a,x)\qquad(a,y)}{(a,x\wedge y)}\qquad a=_{\mathcal{B}} b}{(b,x\wedge y)}}{} \qquad EQI\frac{AND\dfrac{\dfrac{(a,x)\ a=_{\mathcal{B}} b}{(b,x)}\qquad EQI\dfrac{(a,y)\ a=_{\mathcal{B}} b}{(b,y)}}{(b,x\wedge y)}}{}$$

Hence, in each system of $\{WO, EQI, AND\}$, $\{SI, WO, AND\}$ and $\{SI, WO, OR, AND\}$, the AND rule can be applied only at the end. Thus, it is possible to characterize $deriv_i^{AND}(N)$ using the fact $deriv_i^{\mathcal{B}}(N) = out_i^{\mathcal{B}}(N)$ and the iterations of AND.

It is easy to check that CT can be reversed with SI, EQO, WO, and EQI by the fact that it is similarly possible to characterize $deriv_i^{CT}(N)$.

Finally, since AND can be reversed with SI, WO and CT, it is possible to characterize $deriv_1^{CT,AND}(N)$ by applying (finite) iterations of AND over $out_1^{CT}(N)$ that means $out_1^{CT,AND}(N)$.

Proof for Theorem 3

The proofs are the same as the soundness and completeness proofs in Theorem 1.

Proof for Theorem 4

This only looks at I/O operations over Boolean algebras since the argument for abstract logics is similar. It needs to be shown that

- $N \subseteq out_i^{\mathcal{B}}(N)$
- $N \subseteq M \Rightarrow out_i^{\mathcal{B}}(N) \subseteq out_i^{\mathcal{B}}(M)$

- $out_i^{\mathcal{B}}(N) = out_i^{\mathcal{B}}(out_i^{\mathcal{B}}(N))$

By the soundness and completeness theorems, we have $out_i^{\mathcal{B}}(N) = derive_i^{\mathcal{B}}(N)$. So $derive_i^{\mathcal{B}}(N)$ is studied, which is more simple than $out_i^{\mathcal{B}}(N)$. The first two properties are clear from the definition of $derive_i^{\mathcal{B}}$. For the last property, it needs to be shown that $derive_i^{\mathcal{B}}(N) = derive_i^{\mathcal{B}}(derive_i^{\mathcal{B}}(N))$. We have $derive_i^{\mathcal{B}}(derive_i^{\mathcal{B}}(N)) = derive_i^{\mathcal{B}}(\{(A,x)|(a,x) \in derive_i^{\mathcal{B}}(N)$ for some $a \in A\}) = \{(A,x)|(a,x) \in derive_i^{\mathcal{B}}(N)$ for some $a \in A\}$ since $N \subseteq \{(A,x)|(a,x) \in derive_i^{\mathcal{B}}(N)$ for some $a \in A\}$ and the same rules apply over $derive_i^{\mathcal{B}}(N)$. Actually, it needs to be shown that if $(a,x) \in derive_i^{\mathcal{B}}(N)$, then $derive_i^{\mathcal{B}}(N) = derive_i^{\mathcal{B}}(N \cup \{(a,x)\})$ holds for $derive_i^{\mathcal{B}}$.

Appendix C

Proof for Theorem 5

See [110, 52].

Proof for Theorem 6

Here is the proof for the case of $i = 1$.

- Suppose that $(\varphi, \psi) \in derive_1^{\mathbf{Fm}(X)}(N)$. For an arbitrary valuation V and arbitrary Boolean algebra $\mathcal{B} \in \mathbf{BA}$, it needs to be shown that $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$. The proof is by induction on the length of the proof $(\varphi, \psi) \in derive_1^{\mathbf{Fm}(X)}(N)$.
  Base case: If $(\varphi, \psi) \in N$, then $(V(\varphi), V(\psi)) \in N^V$ by definition, and we have $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$.
  Inductive step: It is shown that for $n > 0$, if $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$ holds for n, then $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$ also holds for $n+1$.

  Suppose that the length of proof $(\varphi, \psi) \in derive_1^{\mathbf{Fm}(X)}(N)$ is $n+1$. There are two possibilities:

  - *Using SI in the last step:* There is ϕ such that $(\phi, \psi) \in derive_1^{\mathbf{Fm}(X)}(N)$ and $\varphi \vdash_C \phi$. In this case, by the induction step we have $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\phi)\})$, and by the completeness of the simple-minded operation we have $(V(\phi), V(\psi)) \in derive_1^{\mathcal{B}}(N)$. Since $\varphi \vdash_C \phi$, then by Theorem 5 we have $\varphi \vDash_{\mathbf{BA}} \phi$. So $V(\varphi) \wedge V(\phi) = V(\varphi)$. Then from $(V(\phi), V(\psi)) \in derive_1^{\mathcal{B}}(N)$ and $V(\varphi) \leq V(\phi)$ using the SI rule we have $(V(\varphi), V(\psi)) \in$

$derive_1^{\mathcal{B}}(N)$, and by the soundness of the simple-minded operation we have $V(\psi) \in out_1^{B}(N^V, \{V(\varphi)\})$.

- *Using WO in the last step:* There is ϕ such that $(\varphi, \phi) \in derive_1^{\mathbf{Fm}(X)}(N)$ and $\phi \vdash_C \psi$. In this case, by the induction step we have $V(\phi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$, and by the completeness of the simple-minded operation we have $(V(\varphi), V(\phi)) \in derive_1^{\mathcal{B}}(N)$. Since $\phi \vdash_C \psi$, then by Theorem 5 we have $\phi \vDash_{\mathbf{BA}} \psi$. So $V(\phi) \wedge V(\psi) = V(\phi)$. Then from $(V(\varphi), V(\phi)) \in derive_1^{\mathcal{B}}(N)$ and $V(\phi) \leq V(\psi)$ using the WO rule we have $(V(\varphi), V(\psi)) \in derive_1^{\mathcal{B}}(N)$, and by the soundness of the simple-minded operation we have $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$.

- The proof in the other direction is by contraposition. Suppose that $(\varphi, \psi) \notin derive_1^{\mathbf{Fm}(X)}(N)$, if $\mathbf{Fm}(X)$ is taken as a Boolean algebra, then by the completeness of $derive_1^{\mathbf{Fm}(X)}(N)$, we have $\psi \notin out_1^{\mathbf{Fm}(X)}(N, \{\varphi\})$. Then it is enough that the valuation function is put as the identity function on the Boolean algebra $Fm(X)$ which means $\psi \notin out_1^{\mathcal{B}=\mathbf{Fm}(X)}(N, \{\varphi\})$.

The proof is similar for the other derivation systems: $derive_R^{\mathbf{Fm}(X)}(N)$, $derive_L^{\mathbf{Fm}(X)}(N)$, $derive_I^{\mathbf{Fm}(X)}(N)$, $derive_{II}^{\mathbf{Fm}(X)}(N)$, $derive_2^{\mathbf{Fm}(X)}(N)$, and $derive_3^{Fm(X)}(N)$.

Proof for Theorem 7

- The proof from right to left is similar to Theorem 6. It just needs to be checked that for the case when AND is the last step of the derivation, that there are δ_1 and δ_2 such that $(\varphi, \delta_1), (\varphi, \delta_2) \in derive_i^{AND}(N)$ and $\psi = \delta_1 \wedge \delta_2$. In this case, by the induction step we have $V(\delta_1) \in out_i^{AND}(N^V, \{V(\varphi)\})$ and $V(\delta_2) \in out_i^{AND}(N^V, \{V(\varphi)\})$. By Theorem 2, we have $(\varphi, \delta_1) \in derive_i^{AND}(N)$ and $(\varphi, \delta_2) \in derive_i^{AND}(N)$. Then by using the AND rule, we have $(\varphi, \delta_1 \wedge \delta_2) \in derive_i^{AND}(N)$, and then by Theorem 2, we have $V(\psi) \in out_i^{AND}(N^V, \{V(\varphi)\})$.

- The proof in the other direction is by contraposition. Suppose that $(\varphi, \psi) \notin derive_1^{AND}(N)$, if $\mathbf{Fm}(X)$ is taken as a Boolean algebra, then by Theorem 2, we have $\psi \notin out_1^{AND}(N, \{\varphi\})$, then if the valuation function is put as the identity function on the algebra $Fm(X)$, we have $\psi \notin out_1^{AND}(N, \{\varphi\})$.

It is possible to extend the proof for the arbitrary input set $\Gamma \subseteq Fm(X)$ and to extend this theorem for other addition rule operators.

Proof for Theorem 8

- From left to right: Suppose that $(\varphi, \psi) \in derive_i^{Con}(N)$, then by definition, $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ and $Con, \psi \nvdash_C \bot$. From Theorem 6 we have "$V(\psi) \in out_i^{\mathcal{B}}\ (N^V,\ \{V(\varphi)\})$ for every $\mathcal{B} \in \mathbf{BA}$ and valuation V", and from Theorem 5 there is a Boolean algebra $\mathcal{B}$ such that $Con, \psi \nvDash_{\mathcal{B}} \bot$. So there is a valuation V on $\mathcal{B}$ such that $\forall \delta \in Con$, $V(\delta \wedge \psi) = 1_{\mathcal{B}}$.

- The proof from right to left is similar.

By the definition of $derive_i^{Con}(N)$, it is possible to extend the theorem for the case of $(\Gamma, \psi) \in derive_i^{Con}(N)$ where $\Gamma \subseteq Fm(X)$.

Proof for Theorem 9

- From left to right: Suppose that $\varphi \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$, by definition, $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ and from Theorem 6, we have "$V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$ for every $\mathcal{B} \in \mathbf{BA}$ and valuation V". For the second part, notice that every maximal consistent subset defines a valuation and vice versa. So "$\forall M \in \mathrm{opt}_f(\varphi)(\psi \in M)$" is equivalent to that for any valuation $V_i \in opt_{\succeq_f}(\varphi)$, so that we have $V_i(\psi) = 1_{\mathcal{B}}$ and vice versa.

- From right to left, the proof is similar.

By the definition of $derive_i^{O^H}(N)$, it is possible to extend the theorem for the case of $\Gamma \hookrightarrow \bigcirc\psi \in derive_i^{O^H}(N)$ where $\Gamma \subseteq Fm(X)$.

For another formulation of the theorem, it's important to note that if $V_i \in opt_{\succeq_f}(\varphi)$ in $\langle \mathbf{2}, \mathcal{V}, \succeq_f \rangle$, then we have $V_i \in opt_{\succeq_f}(\varphi)$ in every preference Boolean algebra $\langle \mathcal{B}, \mathcal{V}, \succeq_f \rangle$.

Proof for Theorem 10

- From left to right: Suppose that $(\varphi, \psi) \in derive_i^{O^K}(N)$, by definition, $(\varphi, \psi) \in derive_i^{\mathbf{Fm}(X)}(N)$ and from Theorem 6 we have "$V(\psi) \in out_i^{B}(N^V, \{V(\varphi)\})$ for every $\mathcal{B} \in \mathbf{BA}$ and valuation V". For the second part, notice that every maximal consistent subset defines a valuation and vice versa. So "$\forall M \in opt_{f^A}(\varphi)(\psi \in M)$" is equivalent to that for any valuation $V_i \in opt_{\succeq_A}(\varphi)$, so that we have $V_i(\psi) = 1_{\mathcal{B}}$ and vice versa.

- From right to left, the proof is similar.

By the definition of $derive_i^{O^K}(N)$, it is possible to extend the theorem for the case of $\Gamma \hookrightarrow \bigcirc\psi \in derive_i^{O^K}(N)$ where $\Gamma \subseteq Fm(X)$.

For another formulation of the theorem, it's important to note that according to the definition of $opt_{f^A}(\varphi)$, ψ must be included in all maximal consistent subsets that include both A and φ, or if φ is inconsistent with A, in all maximal consistent subsets that include φ.

Appendix D

Proof for Lemma 1

The proof is straightforward. For example, for COM$\vee$ we have the following: COM$\vee$:

For all $a, b \in D_i$: $I \vee_{i\to i\to i} a\, b = I \vee_{i\to i\to i} b\, a$
(from the definition of $I\vee_{i\to i\to i}$ and $\vee$)
$\Leftrightarrow$ For all assignments g, for all $a, b \in D_i$
$\|X \vee Y = Y \vee X\|^{H^M, g[a/X_i][b/Y_i]} = T$
$\Leftrightarrow$ For all g, we have $\|\forall X \forall Y (X \vee Y = Y \vee X)\|^{H^{\mathcal{N}}, g} = T$
$\Leftrightarrow$ $H^{\mathcal{N}} \models^{\text{HOL}} \text{COM}\vee$

Proof for Lemma 2

Fact: notice that for all $\varphi \in \mathbf{Fm}(X)$ and for all assignments g by induction on the structure of φ, we have $\|\lfloor\varphi\rfloor\|^{H^{\mathcal{N}}, g} = V(\varphi)$.

For simplification, the term abbreviations are used for the saturated set, the $\leq$ ordering and upward set. It is easy to see that these terms abbreviations have the same corresponding sets in the corresponding Henkin model as in the Boolean algebra.

Here then is the proof:

$(d_1(N))$

$\|\lfloor d_1(N)(\varphi, \psi)\rfloor\|^{H^{\mathcal{N}}, g} = T$
$\Leftrightarrow$ $\|(\bigcirc_1(N)_{\tau\to\tau}\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}}, g} = T$
$\Leftrightarrow$ $\|(\lambda A_\tau \lambda X_i (\exists U\, (\exists Y\, (\exists Z\, (A\, Z \wedge Z \leq Y$
$\wedge N\, Y\, U \wedge U \leq X))))\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}}, g} = T$
$\Leftrightarrow$ $\|(\lambda X_i (\exists U\, (\exists Y\, (\exists Z\, (\{\lfloor\varphi\rfloor\}\, Z \wedge Z \leq Y$
$\wedge N\, Y\, U \wedge U \leq X)))))\lfloor\psi\rfloor\|^{H^{\mathcal{N}}, g} = T$

$\Leftrightarrow$ $\|\exists U\,(\exists Y\,(\exists Z\,(\{\lfloor\varphi\rfloor\}\,Z \wedge Z \leq Y \wedge N\,Y\,U \wedge U \leq \lfloor\psi\rfloor)))\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|\exists U\,(\exists Y\,(\,\lfloor\varphi\rfloor \leq Y \wedge N\,Y\,U \wedge U \leq \lfloor\psi\rfloor))\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ There are elements b and c such that $b, c \in D_i$ and
$\|\lfloor\varphi\rfloor \leq Y \wedge N\,Y\,U \wedge U \leq \lfloor\psi\rfloor\|^{H^{M},g[b/U_i][c/Y_i]} = T$

$\Leftrightarrow$ There are elements $b, c \in B$ such that
$V(\varphi) \leq c \wedge N^V\,c\,b \wedge b \leq V(\psi)$

$\Leftrightarrow$ $V(\psi) \in Up(N^V(Up(\{V(\varphi)\})))$

$\Leftrightarrow$ $V(\psi) \in out_1^{\mathcal{B}}(N^V, \{V(\varphi)\})$

$(d_2(N))$

$\|\lfloor d_2(N)(\varphi,\psi)\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\bigcirc_2(N)_{\tau\to\tau}\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\lambda A_\tau\lambda X_i(\forall V\,(Saturated\,V \wedge \forall U(A\,U \to V\,U)$
$\to \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y))))\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\lambda X_i(\forall V\,(Saturated\,V \wedge \forall U(\{\lfloor\varphi\rfloor\}\,U \to V\,U)$
$\to \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y)))))\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|\forall V\,(Saturated\,V \wedge \forall U(\{\lfloor\varphi\rfloor\}\,U \to V\,U)$
$\to \exists Y\,(\exists Z\,(Z \leq \lfloor\psi\rfloor \wedge N\,Y\,Z \wedge V\,Y)))\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ There are elements b and c such that $b, c \in D_i$ and
$\|\forall V\,(Saturated\,V \wedge \forall U(\{\lfloor\varphi\rfloor\}\,U \to V\,U)$
$\to (Z \leq \lfloor\psi\rfloor \wedge N\,Y\,Z \wedge V\,Y))\|^{H^{\mathcal{N}},g[b/Y_i][c/Z_i]} = T$

$\Leftrightarrow$ For every saturated set V that $\{V(\varphi)\} \subseteq V$,
there are elements $b, c \in B$ such that
$c \leq V(\psi) \wedge N^V\,b\,c \wedge V\,b$

$\Leftrightarrow$ For every saturated set V such that $\{V(\varphi)\} \subseteq V$,
we have $V(\psi) \in Up(N^V(V))$

$\Leftrightarrow$ $V(\psi) \in out_2^{\mathcal{B}}(N^V, \{V(\varphi)\})$

$(d_3(N))$

$\|\lfloor d_3(N)(\varphi,\psi)\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\bigcirc_3(N)_{\tau\to\tau}\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\lambda A_\tau\lambda X_i(\forall V(\forall U(A\,U \to V\,U) \wedge V = Up\,V$
$\wedge\forall W(\exists Y(V\,Y \wedge N\,Y\,W) \to V\,W)$
$\to \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y))))\{\lfloor\varphi\rfloor\})\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|(\lambda X_i(\forall V(\forall U(\{\lfloor\varphi\rfloor\}\,U \to V\,U) \wedge V = Up\,V$

$\wedge\forall W(\exists Y(V\,Y \wedge N\,Y\,W) \rightarrow V\,W)$
$\rightarrow \exists Y\,(\exists Z\,(Z \leq X \wedge N\,Y\,Z \wedge V\,Y)))))\lfloor\psi\rfloor\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ $\|\forall V(\forall U(\{\lfloor\varphi\rfloor\}\,U \rightarrow V\,U) \wedge V = Up\,V$
$\wedge\forall W(\exists Y(V\,Y \wedge N\,Y\,W) \rightarrow V\,W)$
$\rightarrow \exists Y\,(\exists Z\,(Z \leq \lfloor\psi\rfloor \wedge N\,Y\,Z \wedge V\,Y)))\|^{H^{\mathcal{N}},g} = T$

$\Leftrightarrow$ There are elements b and c such that $b, c \in D_i$ and
$\|\forall V(\forall U(\{\lfloor\varphi\rfloor\}\,U \rightarrow V\,U) \wedge V = Up\,V$
$\wedge\forall W(\exists Y(V\,Y \wedge N\,Y\,W) \rightarrow V\,W)$
$\rightarrow (Z \leq \lfloor\psi\rfloor \wedge N\,Y\,Z \wedge V\,Y))\|^{H^{\mathcal{N}},g[b/Y_i][c/Z_i]} = T$

$\Leftrightarrow$ For every set V that $Up(V) = V$, $\{V(\varphi)\} \subseteq V$ and $N^V(V) \subseteq V$, there are elements $b, c \in B$ such that $c \leq V(\psi) \wedge N^V\,b\,c \wedge V\,b$

$\Leftrightarrow$ For every set V that $Up(V) = V$, $\{V(\varphi)\} \subseteq V$ and $N^V(V) \subseteq V$, we have $V(\psi) \in Up(N^V(V))$

$\Leftrightarrow$ $V(\psi) \in out_3^{\mathcal{B}}(N, \{V(\varphi)\})$

Proof for Lemma 3

Suppose that $H = \langle\{D_\alpha\}_{\alpha\in T}, I\rangle$ is a Henkin model such that $H \models^{\mathrm{HOL}} \Sigma$ for all $\Sigma \in \{COM\vee, ..., Dis\wedge\vee\}$. Without loss of generality, it can be assumed that the domains of H are denumerable [75]. The corresponding Boolean normative model $\mathcal{N}$ is constructed as follows:

- $B = D_i$.
- $1 = I\top_i$.
- $0 = I\bot_i$.
- $a \vee b = c$ for $a, b, c \in B$ iff $I \vee_{i\rightarrow i\rightarrow i} ab = c$.
- $a \wedge b = c$ for $a, b, c \in B$ iff $I \wedge_{i\rightarrow i\rightarrow i} ab = c$.
- $a = \neg b$ for $a, b \in B$ iff $I\neg_{i\rightarrow i}a = b$.
- The valuation on $\mathcal{B}$ is defined such that for all $p^j \in X$, $V(p^j) = I(p_i^j)$.
- $(a, b) \in N^V$ for $a, b \in B$ iff $IN_{i\rightarrow\tau}ab = T$.

Since $H \models^{\mathrm{HOL}} \Sigma$ for all $\Sigma \in \{COM\vee, ..., Dis\wedge\vee\}$, it is straightforward (but tedious) to verify that $\wedge$, $\vee$, $\neg$, 0 and 1 satisfy the conditions required for a Boolean algebra.

Moreover, the above construction ensures that H is a Henkin model $H^{\mathcal{N}}$

for Boolean normative model $\mathcal{N}$. Hence, Lemma 2 applies. This ensures that for all conditional norms (φ, ψ), and for all assignment g, we have:

$$\|\lfloor d_i(N)(\varphi, \psi)\rfloor\|^{H,g} = T \text{ if and only if } V(\psi) \in out_i^{\mathcal{B}}(N^V, \{V(\varphi)\}).$$

Proof for Theorem 11

Soundness

The proof is by contraposition. Suppose that for a Boolean normative model $\langle \mathcal{B}, V, N^V \rangle$, we have $V(\psi) \notin out_i^{\mathcal{B}}(N^V, \{V(\psi)\})$. Now let $H^{\mathcal{N}}$ be a Henkin model for Boolean normative model $\mathcal{N}$. Then by Lemma 2 for an arbitrary assignment g, it is held that $\|\lfloor d_i(N)(\varphi, \psi)\rfloor\|^{H^{\mathcal{N}},g} = F$, but $\|COM \vee\|^{H^{\mathcal{N}},g} = T$, ..., $\|Dis \wedge \vee\|^{H^{\mathcal{N}},g} = T$, and that is a contradiction.

Completeness

The proof is again by contraposition. If it is assumed that $\{COM\vee, ..., Dis\wedge\vee\} \not\models^{\text{HOL}} \lfloor d_i(N)(\varphi, \psi)\rfloor$, then there is a Henkin model $H = \langle \{D_\alpha\}_{\alpha \in T}, I\rangle$ such that $H \models^{\text{HOL}} \Sigma$ for all $\Sigma \in \{COM\vee, ..., Dis\wedge\vee\}$, but $\|\lfloor d_i(N)(\varphi, \psi)\rfloor\|^{H,g} = F$ for some assignment g. By Lemma 3, there is a Boolean normative model $\mathcal{N}$ such that $V(\psi) \notin out_i^{\mathcal{B}}(N^V, \{V(\varphi)\})$, and that is a contradiction.