

A Deontic Logic Reasoning Infrastructure

Christoph Benzmüller, Xavier Parent, and Leendert van der Torre

Computer Science and Communications, University of Luxembourg, Luxembourg

Abstract. A flexible infrastructure for the automation of deontic and normative reasoning is presented. Our motivation is the development, study and provision of legal and moral reasoning competencies in future intelligent machines. Since there is no consensus on the “best” deontic logic formalisms and since the answer may be application specific, a flexible infrastructure is proposed in which candidate logic formalisms can be varied, assessed and compared in experimental ethics application studies. Our work thus links the historically rich research areas of classical higher-order logic, deontic logics, normative reasoning and formal ethics.

1 Introduction

If humans and intelligent machines are supposed to peacefully coexist, appropriate forms of machine-control and human-machine-interaction are required. This motivates the provision of legal and ethical reasoning competencies in intelligent machines. Bottom-up, learning based approaches (e.g. GenEth [1]) try to acquire ethical behaviour from dialogs with ethicist or from existing data. Top-down approaches (e.g. [21]) try to explicitly model selected ethical theories or policies and to enforce them in intelligent systems; cf. [23, 20] and the references therein. Our research, which is more in line with the top-down approach, assumes that suitable declarative, logical reasoning means and competencies are mandatory in intelligent machines, in particular in the context of legal and moral reasoning. Such competencies seem vital not only for guaranteeing sufficient degrees of reliability and accountability, but also for achieving human intuitive interaction means regarding explainability and transparency of decisions.

This paper therefore addresses the practical development of computational tools for normative reasoning based on deontic logics. Since ethical and legal theories/policies as well as suitable deontic logic formalisms are both still under development [34], we outline a flexible workbench to support empirical studies with such theories/policies in which the preferred logic formalisms themselves can still be varied, complemented, assessed and compared. The infrastructure we propose draws on both recent developments in universal logical reasoning in classical higher-order logic (HOL) [5] and the coalescence and improvements in interactive and automated theorem proving (ATP) in HOL, as witnessed by systems such as Isabelle/HOL [32], LEO-II [11] and Leo-III. We are thus linking the historically rich research areas of HOL, deontic logics, normative reasoning and formal ethics. Moreover, since modern HOL theorem provers internally collaborate with effective SMT and SAT solvers, a novel bridge is provided from expressive deontic logics to SMT and SAT technology.

The paper is structured as follows. Section 2 surveys relevant deontic logic formalisms and discusses recent extensions to model ethical agency. Section 3 introduces norm-based deontic logic, our preferred framework. Sections 4 and 5, the core contribution of this paper, outline our flexible deontic reasoning infrastructure and sketch a case study in data protection.

2 Traditional Deontic Logic

Deontic logic [39, 25] is the field of logic that is concerned with normative concepts such as obligation, permission, and prohibition. Alternatively, a deontic logic is a formal system capturing the essential logical features of these concepts. Typically, a deontic logic uses Op to mean that it is obligatory that p , (or it ought to be the case that p), and Pp to mean that it is permitted, or permissible, that p . The term “deontic” is derived from the ancient Greek *déon*, meaning “that which is binding or proper”. Deontic logic can be used for reasoning about normative multiagent systems, i.e. about multiagent organisations with normative systems in which agents can decide whether to follow the explicitly represented norms, and the normative systems specify how, and to which extent, agents can modify the norms. Normative multiagent systems need to combine normative reasoning with agent interaction, and thus raise the challenge to relate the logic of normative systems to aspects of agency.

Traditional (or “standard”) deontic logic (SDL) is a normal propositional modal logic of type KD, which means that it extends the propositional tautologies with the axioms $K: O(p \rightarrow q) \rightarrow (Op \rightarrow Oq)$ and $D: \neg(Op \wedge O\neg p)$, and it is closed under the inference rules *modus ponens* $p, p \rightarrow q / q$ and *generalization* or *necessitation* p / Op . Prohibition and permission are defined by $Fp = O\neg p$ and $Pp = \neg O\neg p$. SDL is an unusually simple and elegant theory. An advantage of its modal-logical setting is that it can easily be extended with other modalities such as epistemic or temporal operators and modal accounts of action.

Dyadic deontic logic (DDL) introduces a conditional operator $O(p/q)$, to be read as “it ought to be the case that p , given q ”. Many DDLs have been proposed to deal with so-called contrary-to-duty reasoning, cf. [18] for an overview on this area. An example is the DDL proposed by Hansson [28] and Åqvist [3, 33], and the one proposed by Carmo and Jones [18, 19].

To enable ethical agency a model of decision needs to be integrated in the deontic frames. Horty’s STIT logic [29], which combines deontic logic with a modal logic of action, has been proposed as a starting point. The semantic condition for the STIT-ought is a utilitarian generalisation of the SDL view that “it ought to be that A ” means that A holds in all deontically optimal worlds.

3 Norm-based Deontic Logic

The term “norm-based” deontic logic has been coined by Hansen [27] to refer to a family of frameworks analysing the deontic modalities not with reference to a set of possible worlds (some of them being more ideal than others), but with

reference to a set of explicitly given norms. In such a framework, the central question is: given some input (e.g. a fact) and a set of explicitly given conditional norms (a normative system), what norms apply? Thus, the perspective is slightly different from the traditional setting, focusing on inference patterns [35].

We propose to base the AI deontic reasoner on a specific norm-based deontic logic called input/output (I/O) logic. Initially devised by Makinson [31] and further developed over the past years by van der Torre and colleagues, I/O logic has gained increased recognition in the AI community. This is evidenced by the fact that the framework has its own chapter in the aforementioned Handbook of Deontic Logic and Normative Systems [34]. I/O logic can be viewed as a rule-based system. The knowledge base takes the form of a set of rules of the form (a,b) to be read as “if a then b ”. The key feature of I/O logic is that it uses an operational semantics, based on the notion of detachment, rather than a truth-functional one in terms of truth-values and possible worlds. On the semantical side, the meaning of the deontic concepts is given in terms of a set of procedures, called I/O operations, yielding outputs (e.g., obligations) for inputs (facts). On the syntactical side, the proof-theory is formulated as a set of inference rules manipulating pairs of formulas rather than individual formulas. The framework covers functionalities that are unanimously regarded as characteristic of the legal domain, and thus required to enable effective legal reasoning:

1. Support for the modelling of *constitutive rules*, which define concepts or constitute activities that cannot exist without such rules (e.g. legal definitions such as “property”), and *prescriptive rules*, which regulate actions by making them obligatory, permitted, or prohibited.
2. Management of the *reification of rules* that are objects with properties, such as jurisdiction, authority, temporal attributes [26].
3. Implementation of *defeasibility*—see [26, 38]; when the antecedent of a rule is satisfied by the facts of a case (or via other rules), the conclusion of the rule presumably holds, but is not necessarily true.

4 Deontic Logic Reasoning Machinery

In a nutshell, a reasoner is a tool that can perform reasoning tasks in a given application domain. Reasoning thereby refers to the process of deriving or concluding information that is not explicitly encoded in the knowledge base. In our context, a number of reasoning tasks are particularly relevant. These include:

- *Compliance checking*: Is the current situation compliant with a given regulation (a set of formally represented norms)?
- *Consistency checking*: Is a given regulation consistent? Is such-and-such norm, as part of a given regulation, consistent with this other set of norms, stemming from another regulation? Is such-and-such legal interpretation consistent with another one?
- *Entailment checking*: Does such-and-such obligation or legal interpretation follow from a given regulation?

Some of these tasks, e.g. consistency checking, are well supported by model finders, while others, such as entailment checking, in general require theorem proving technology. A powerful deontic reasoner should thus ideally provide both model finding and theorem proving. While this is comparably easy to achieve for many decidable propositional fragments of deontic logics, it becomes much less so for their quantified extensions.

The quest for “a single, best deontic logic” is still open, and it eventually will remain so for a long time to come. While I/O logic is the solution favoured in our group (see Sec. 3), we also want to stay open-minded regarding alternative proposals, such as the DDL by Hansson or the one by Carmo and Jones. Moreover, it is unclear yet whether regulatory texts can always be abstracted and simplified to the point that pure propositional logic encodings are feasible and justified, or whether first-order (FO) or even higher-order (HO) encodings are required instead. This poses a multitude of concrete challenges for a flexible deontic reasoning infrastructure.

These raised questions motivate empirical studies in which the different options are systematically compared and assessed within well selected application studies (see Sec. 5). However, for such empirical work to be feasible, implementations of the different deontic candidate logics have to be provided first, both on the propositional level and ideally also on the FO and HO level. Moreover, it is reasonable to ensure that these implementations remain maximally comparable regarding the technological foundations they are based on, since this may improve the fairness and the significance of (conceptual) empirical evaluations.

A meta-logical approach for flexible, deontic logic reasoning. Our currently preferred solution for the implementation of a most flexible deontic reasoner in the above sense is to work with a meta-logical approach to universal logic reasoning, cf. [5] and the references therein. We subsequently instantiate this approach in our ongoing project for various deontic candidate logics, including I/O logic and the other ones as mentioned. The approach, which is based on shallow semantical embeddings (SSE) [10] of these logics in HOL¹, has a very pragmatic motivation, foremost reuse of tools, simplicity and elegance. It utilises HOL as a unifying meta-logic in which the syntax and semantics of varying other logics can be explicitly modeled and flexibly combined. Off-the-shelf interactive and automated theorem provers, and model finders can then be employed to reason about and within the shallowly embedded logics.

Evidence from previous work. Respective experiments with this approach have e.g. been conducted in metaphysics (cf. [13]). An initial focus thereby has been on computer-supported assessments of rational arguments, in particular, of modern, modal logic variants of the ontological argument for the existence of God.

¹ HOL as addressed here refers to a (simply) typed logic of functions, which has been proposed by Church [2]. It provides lambda-notation, as an elegant and useful means to denote unnamed functions, predicates and sets. Types in HOL eliminate paradoxes and inconsistencies. For more information on HOL see the literature [7].

In the course of these experiments, in which the SSE approach was applied for automating different variants of HO quantified modal logics [10], the HO theorem prover LEO-II even detected an previously unnoticed inconsistency in Kurt Gödel’s prominent variant of the ontological argument, while the soundness of the slightly modified variant by Dana Scott was confirmed and all argument steps were verified. Further modern variants of the argument have subsequently been studied with the approach, and theorem provers have even contributed to the clarification of an unsettled philosophical dispute [12].

Deontic logics already covered in the SSE approach. The following deontic logics have already been “implemented” by utilising the SSE approach:

- SDL: All logics from the modal logic cube, including logic KD, i.e. SDL, have meanwhile been faithfully implemented in the SSE approach [10]. These implementations scale for FO and even HO extensions.
- the DDL by Carmo and Jones [19]: A semantic embedding of the propositional fragment of this logic in Isabelle/HOL is already available [6, 8], and most recently the ATP Leo-III has been adapted to accept DDL as input.
- I/O logic [31]: The main challenge comes from the fact that the framework does not have a truth-functional semantics, but an operational one. First experiments with the semantic embedding of the I/O-operator “*out1*” (called simple-minded) in Isabelle/HOL are promising [9]; see also Sect. 5.

Some relevant I/O logic variants have very recently been studied (see [36]), and we conjecture that some of these variants are related to certain non-normal modal logics, e.g. conditional logics with a selection function semantics or similar logics with a neighbourhood semantics. However, the embedding of such logics has already been studied in the first authors previous work [4]. It should thus be possible to benefit from these existing results in the given context.

A most interesting aspect is, that the SSE approach even supports meta-logical investigations within Isabelle/HOL in which the possible relationships between I/O and conditional logics we hinted at can be formally assessed. Examples of meta-logical studies are e.g. mentioned in [10].

Moreover, the SSE approach enables the reuse of existing model finding and theorem proving technology within the Isabelle/HOL proof assistant [32]. The automated reasoning systems that are integrated with Isabelle/HOL, respectively that are available via remote calls, include state-of-the-art SMT solvers, FO and HO theorem provers, and model finders; cf. [15] and the references therein. This infrastructure, in combination with the SSE approach, meets our demanding requirements regarding flexibility along different axes.

While there is some related work, see e.g. [17, 24, 37, 21], we are not aware of any other existing deontic logic reasoning approach and associated machinery that offers the same amount of flexibility and scalability.

Another advantage of the SSE approach, when implemented within powerful proof assistants such as Isabelle/HOL, is that proof construction (interactive or automated) can be supported at different levels of abstraction. For this note that proof protocols/objects may generally serve two different purposes: (a) they may

provide an independently verifiable explanation in a (typically) well-defined logical calculus, or (b) they may provide an intuitive explanation to the user why the problem in question has been answered positively or negatively. Many reasoning tools, if they are offering proof objects at all, do generate only objects of type (a). The SSE approach, however, has already demonstrated its capabilities to provide both types of responses simultaneously in even most challenging logic settings. For example, a quite powerful, abstract level theorem prover for hyperintensional HO modal logic has been provided by Kirchner [30]. He encoded an abstract level proof calculus for this logic as proof tactics and he demonstrated how these abstract level proof tactics can again be elegantly automated using respective tools in Isabelle/HOL. Kirchner then successfully applied his reasoning infrastructure to reveal, assess and intuitively communicate a non-trivial paradox in Zalta’s “Principia-logico Metaphysica” [40].

Drawing on the results and experiences from previous work, the ambition of our ongoing project is to further extend the already existing implementations of deontic logics in Isabelle/HOL towards a most powerful, flexible and scalable deontic logic reasoning infrastructure. A core motivation thereby is to support empirical studies in various application scenarios, and to assess and compare the suitability, adequacy and performance of individual deontic logic solutions for the engineering of moral agents and explainable intelligent systems.

5 Case Study: Data Protection

The General Data Protection Regulation (GDPR, Regulation EU 2016/679) is a relevant and interesting application scenario for normative reasoning. It is a regulation by which the European Parliament, the Council of the European Union and the European Commission intend to strengthen and unify data protection for all individuals within the European Union. The regulation becomes enforceable from 25 May 2018. We present two sample norms of the GDPR:

1. Personal data shall be processed lawfully (Art. 5). For example, the data subject must have given consent to the processing of his or her personal data for one or more specific purposes (Art. 6/1.a).
2. If the personal data have been processed unlawfully (none of the requirements for a lawful processing applies), the controller has the obligation to erase the personal data in question without delay (Art. 17.d, right to be forgotten).

When combined with the following a typical CTD-structure is exhibited.

3. It is obligatory e.g. as part of a respective agreement between a customer and a company) to keep the personal data (as relevant to the agreement) provided that it is processed lawfully.
4. Some data in the context of such an agreement has been processed unlawfully.

```

1 theory GDPR_IO imports Main (*GDPR Example in I/O Logic, by C. Benzmüller & X. Parent, 2018*)
2 begin (*Some unimportant, technical setup*) declare [[ smt_solver = cvc4 ]]
3 (*Semantical Embedding of IO Logic in HOL*)
4 typedecl i -- "type for possible worlds" type_synonym e = "(i=>bool)"
5 abbreviation IOtrue :: "e" ("T") where "T ≡ λw. True"
6 abbreviation IOfalse :: "e" ("⊥") where "⊥ ≡ λw. False"
7 abbreviation IOnot :: "e=>e" ("¬"[52]53) where "¬φ ≡ λw. ¬φ(w)"
8 abbreviation IOor :: "e=>e=>e" (infixr"∨"50) where "φ∨ψ ≡ λw. φ(w)∨ψ(w)"
9 abbreviation IOand :: "e=>e=>e" (infixr"∧"51) where "φ∧ψ ≡ λw. φ(w)∧ψ(w)"
10 abbreviation IOimp :: "e=>e=>e" (infixr"⊃"49) where "φ⊃ψ ≡ λw. φ(w)→ψ(w)"
11 abbreviation IOvalid :: "e=>bool" ("⊢"[8]109) where "[p] ≡ ∀w. p w"
12 abbreviation "outpre" ≡ λNorms.λa.λy::e. ∃f. [a ⊃ f] ∧ Norms (f,y)
13 abbreviation "outl" ≡ λNorms.λa.λx.
14   ∃i j k. (outpre Norms a i ∧ outpre Norms a j ∧ outpre Norms a k ∧ [(i ∧ j ∧ k) ⊃ x]) ∨ [x]"
15
16 (*GDPR example in IO Logic (outl). We first introduce some uninterpreted constant symbols.*)
17 consts process_data_lawfully::e erase_data::e kill_boss::e
18 (*The set of IO-Norms of the GDPR example, encoded as a characteristic function in HOL.*)
19 abbreviation "Norms" ≡ (λX.
20   (*A1: It is an obligation to process data lawfully.*)
21   X = (T,process_data_lawfully) ∨
22   (*A2: If data was not processed lawfully, then it is an obligation to erase the data.*)
23   X = (¬process_data_lawfully,erase_data) ∨
24   (*Implicit: It is an obligation to keep the data if it was processed lawfully.*)
25   X = (process_data_lawfully,¬erase_data) )"
26 (*Assume a given Situation, where the data has been processed unlawfully.*)
27 abbreviation "Situation" ≡ ¬process_data_lawfully"
28
29 (*Query1: Given Norms and the concrete Situation, should the data be erased?*)
30 lemma "outl Norms Situation erase_data" sledgehammer() by smt
31 (*Query2: Given Norms and the concrete Situation, should the boss be killed?*)
32 lemma "outl Norms Situation kill_boss" nitpick (*countermodel*) oops
33 (*Query3: Given Norms and the concrete Situation, does Falsum (inconsistency) follow?*)
34 lemma "outl Norms Situation ⊥" nitpick (*countermodel*) oops
35 end

```

Proof state Auto update Update Search: 100%

Sledgehammering...
Proof found...
"cvc4": Try this: by fastforce (232 ms)
"z3": Try this: by fastforce (268 ms)

Output Query Sledgehammer Symbols

30,56 (1976/2308) (isabelle,isabelle,UTF-8-Isabelle)Nm r o UG 730/1176MB 5:03 PM

Fig. 1. GDPR example scenario in Isabelle/HOL.

The latter information pieces are not explicit part of the GDPR. Instead they are to be seen as implicit. 3 comes from another regulation, with which the GDPR has to co-exists. 4 is a factual information — it is exactly the kind of world situations the GDPR wants to regulate. This example is given for illustrative purposes only. It provides a taste of what this knowledge base might look like.

In a recent technical report [8] we illustrate the practical challenge of such a CTD scenario. Namely, when the above norms are encoded in SDL, an inconsistency follows, meaning that everything is implied in the given context, including arbitrarily weird and unethical conclusions such as the obligation to “kill the boss”. In the same report we demonstrate that our SSE based implementation of Carmo and Jones’s DDL is in contrast not suffering from this effect. No inconsistency follows when the above scenario is modelled in DDL. The obligation to erase the data, however, is implied as intended.

We now analyse the above CTD scenario in the context of I/O logic. This is done in Fig. 1, which first presents, in lines 4-13, an SSE based implementation of I/O logic in Isabelle/HOL.² A possible world semantics is employed in this embedding to adequately address an extensionality issue we have revealed in our previous work. This issue, and its solution, is discussed in more detail in a technical report [9]. The prescriptive rules of the GDPR scenario are then modelled in lines 19-25, where the set of given *Norms* is defined as $\{(\top, process_data_lawfully), (\neg process_data_lawfully, erase_data), (process_data_lawfully, \neg erase_data)\}$. The given *Situation*, in which we have $\neg process_data_lawfully$, is defined in line 27. Subsequently, three different queries are answered by the reasoning tools integrated with Isabelle/HOL. The first query asks whether the data should be erased in the given context. The ATPs integrated with Isabelle/HOL via the Sledgehammer tool [15] respond quickly: the SMT solver CVC4 [22] and the first-order prover Spass [16] return a proof within a few milliseconds. For queries 2 and 3 the ATPs fail (not shown here), but now the countermodel finder Nitpick [14] responds and presents counterarguments to both queries. That is, we receive the intended negative answers to queries 2 and 3 when the GDPR example is modelled in our preferred I/O logic. It is worth mentioning that I/O logic (and also DDL) have never been automated before.

6 Conclusion

The deontic logic reasoning infrastructure we have presented supports empirical studies on legal and ethical theories/policies in which the particular deontic logic formalisms itself can be varied, assessed and compared in context. We believe that this infrastructure can fruitfully support the development of much needed logic based approaches towards ethical agency. The solution we have presented supports a wide range of specific deontic logic variants, and it also scales for their first-order and higher-order extensions. In fact, our infrastructure already now

² The semantical embedding of out_1 as presented here is technically still an approximate solution. For a complete embedding, x needs to be defined as a consequence of an arbitrary number of facts (instead of just i , j and k) in lines 13 and 14.

implements a wider range of deontic and related logics than any other competitor systems we are aware of.

References

1. M. Anderson and S. L. Anderson. Toward ensuring ethical behavior from autonomous systems: a case-supported principle-based paradigm. *Industrial Robot*, 42(4):324–331, 2015.
2. P. Andrews. Church’s type theory. In E. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Spring 2014 edition, 2014.
3. L. Åqvist. Deontic logic. In D. Gabbay and F. Guenther, editors, *Handbook of Philosophical Logic*, volume 8, pages 147–264. Kluwer Academic Publishers, Dordrecht, Holland, 2nd edition, 2002.
4. C. Benzmüller. Cut-elimination for quantified conditional logic. *Journal of Philosophical Logic*, 46(3):333–353, 2017.
5. C. Benzmüller. Recent successes with a meta-logical approach to universal logical reasoning (extended abstract). In S. A. da Costa Cavalheiro and J. L. Fiadeiro, editors, *SBMF 2017, Proc.*, volume 10623 of *LNCS*, pages 7–11. Springer, 2017.
6. C. Benzmüller, A. Farjami, and X. Parent. Faithful semantical embedding of a dyadic deontic logic in HOL. Technical report, CoRR, 2018. <https://arxiv.org/abs/1802.08454>.
7. C. Benzmüller and D. Miller. Automation of higher-order logic. In D. M. Gabbay, J. H. Siekmann, and J. Woods, editors, *Handbook of the History of Logic, Volume 9 — Computational Logic*, pages 215–254. North Holland, Elsevier, 2014.
8. C. Benzmüller and X. Parent. First experiments with a flexible infrastructure for normative reasoning. Technical report, CoRR, 2018. <http://arxiv.org/abs/1804.02929>.
9. C. Benzmüller and X. Parent. I/O logic in HOL — first steps. Technical report, CoRR, 2018. <https://arxiv.org/abs/1803.09681>.
10. C. Benzmüller and L. Paulson. Quantified multimodal logics in simple type theory. *Logica Universalis*, 7(1):7–20, 2013.
11. C. Benzmüller, N. Sultana, L. C. Paulson, and F. Theiß. The higher-order prover LEO-II. *Journal of Automated Reasoning*, 55(4):389–404, 2015.
12. C. Benzmüller, L. Weber, and B. Woltzenlogel Paleo. Computer-assisted analysis of the Anderson-Hájek controversy. *Logica Universalis*, 11(1):139–151, 2017.
13. C. Benzmüller and B. Woltzenlogel Paleo. The inconsistency in Gödel’s ontological argument: A success story for AI in metaphysics. In S. Kambhampati, editor, *IJCAI 2016*, volume 1-3, pages 936–942. AAAI Press, 2016.
14. J. Blanchette and T. Nipkow. Nitpick: A counterexample generator for higher-order logic based on a relational model finder. In *ITP 2010*, number 6172 in *LNCS*, pages 131–146. Springer, 2010.
15. J. C. Blanchette, S. Böhme, and L. C. Paulson. Extending Sledgehammer with SMT solvers. *J. Autom. Reasoning*, 51(1):109–128, 2013.
16. J. C. Blanchette, A. Popescu, D. Wand, and C. Weidenbach. More SPASS with Isabelle - Superposition with Hard Sorts and Configurable Simplification. In *ITP 2012, Proceedings*, volume 7406 of *LNCS*, pages 345–360. Springer, 2012.
17. S. Bringsjord, K. Arkoudas, and P. Bello. Toward a general logicist methodology for engineering ethically correct robots. *IEEE Intelligent Systems*, 21:38–44, 2006.

18. J. Carmo and A. J. I. Jones. Deontic logic and contrary-to-duties. In D. M. Gabbay and F. Guentner, editors, *Handbook of Philosophical Logic: Volume 8*, pages 265–343. Springer Netherlands, Dordrecht, 2002.
19. J. Carmo and A. J. I. Jones. Completeness and decidability results for a logic of contrary-to-duty conditionals. *J. of Logic and Computation*, 23(3):585–626, 2013.
20. L. A. Dennis and M. Fischer. Practical challenges in explicit ethical machine reasoning. In *ISAIM 2018, Fort Lauderdale, Florida, USA*, 2018.
21. L. A. Dennis, M. Fisher, M. Slavkovik, and M. Webster. Formal verification of ethical choices in autonomous systems. *Robotics and Autonomous Systems*, 77:1–14, 2016.
22. M. Deters, A. Reynolds, T. King, C. W. Barrett, and C. Tinelli. A tour of CVC4: how it works, and how to use it. In *Formal Methods in Computer-Aided Design, FMCAD 2014, Lausanne, Switzerland, October 21-24, 2014*, page 7. IEEE, 2014.
23. V. Dignum. Responsible autonomy. In *IJCAI-17*, pages 4698–4704, 2017.
24. U. Furbach, C. Schon, and F. Stolzenburg. Automated reasoning in deontic logic. In *MIWAI 2015, Proceedings*, volume 8875 of *LNCS*, pages 57–68, 2014.
25. D. Gabbay, J. Horty, X. Parent, R. van der Meyden, and L. van der Torre, editors. *Handbook of Deontic Logic and Normative Systems*. College Publications, London, UK, 2013.
26. T. Gordon. *The Pleading Game: an Artificial Intelligence Model of Procedural Approach*. Springer, New York, 1995.
27. J. Hansen. Reasoning about permission and obligation. In S. O. Hansson, editor, *David Makinson on Classical Methods for Non-Classical Problems*, pages 287–333. Springer, Dordrecht, 2014.
28. B. Hansson. An analysis of some deontic logics. *Noûs*, 3(4):373–398, 1969.
29. J. Horty. *Agency and Deontic Logic*. OUP, London, UK, 2009.
30. D. Kirchner, C. Benzmüller, and E. N. Zalta. Mechanizing principia logico-metaphysica in functional type theory. *CoRR*, <https://arxiv.org/abs/1711.06542>, 2017.
31. D. Makinson and L. W. N. van der Torre. Input/output logics. *J. Philosophical Logic*, 29(4):383–408, 2000.
32. T. Nipkow, L. Paulson, and M. Wenzel. *Isabelle/HOL: A Proof Assistant for Higher-Order Logic*. Number 2283 in *LNCS*. Springer, 2002.
33. X. Parent. Completeness of Åqvist’s systems E and F. *Review of Symbolic Logic*, 8(1):164–177, 2015.
34. X. Parent and L. van der Torre. Input/output logic. In Gabbay et al. [25], pages 499–544.
35. X. Parent and L. van der Torre. Detachment in normative systems: Examples, inference patterns, properties. *The IfCoLog Journal of Logics and their Applications*, 4(9):2295–3039, 2017.
36. X. Parent and L. van der Torre. The pragmatic oddity in a norm-based semantics. In G. Governatori, editor, *ICAIL 2017, Proceedings*, pages 169–178, New York, NY, USA, 2017. ACM.
37. L. M. Pereira and A. Saptawijaya. *Programming Machine Ethics*, volume 26 of *Studies in Applied Philosophy, Epistemology and Rational Ethics*. Springer, 2016.
38. G. Sartor. *Legal Reasoning: A Cognitive Approach to Law*. Springer, 2005.
39. G. H. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.
40. E. N. Zalta. Principia logico-metaphysica. Draft version, preprint available at <https://mally.stanford.edu/principia.pdf>, 2016.