Deontic Logic for Normative Autonomous Systems

Colin Shea-Blymyer

September 2021

Abstract

As autonomous systems become more ubiquitous, the ability to represent and reason about their behaviors becomes more important. Normative autonomous systems are those systems that obey some set of norms, and deontic logic is a formal method for reasoning about norms. In this survey, I explore what normative autonomous systems are, why they are important, how they can be implemented, and what role deontic logic plays in their implementation. I also explore a particular form of deontic logic (Dominance Act Utilitarianism) and its strengths and weaknesses as a logic for normative autonomous systems.

1 Introduction

As self-driving cars take to the road, and autonomous delivery drones take to the skies, autonomous systems that behave by socially accepted norms will find it easier to inter-operate with humans in that society. It follows that the capability of autonomous systems to respect the ethical and social norms of the society they operate within will play a large role in their rate of adoption.

Any autonomous, embodied, system that obeys (or tries to obey) a set of ethical or social guidelines can be considered to be a *normative autonomous system* (NAS). These guidelines, or *norms*, are distinct from the system's mission, and modify and constrain the behavior of a NAS. In particular, normative systems are concerned with statements of *obligation*, *permission*, and *prohibition*.

A famous example of a normative autonomous system is Isaac Asimov's positronic robot that follows the Three Laws of Robotics [4]:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey orders given to it by human beings except where such orders would conflict with the First Law.
3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

These laws do not define a robot's mission (perhaps to drive from one city to another), but do constrain its behavior (preventing it from harming pedestrians). These laws are the norms by which the robotic system is governed.

I describe some properties of norms in the next section. In Section 3 I explore some approaches to imbuing autonomous systems with norms. Section 4 introduces the concepts of deontic logic. Section 5 introduces a specific flavor of deontic logic known as Dominance Act Utilitarianism (DAU). Through the lens of this logic, I approach some challenges in normative reasoning: contrary to duty reasoning (Section 6), and emergent norms in groups (Section 7). I then present some computational results regarding DAU in Section 8. Finally, I reflect on the use of deontic logic in normative autonomous systems and offer some exciting directions for future work in Section 9.

2 Norms

In general, norms may be considered to be the rules that govern the behavior of a system acting in a group or society [5].

Norms are often thought to be violable. For instance, a self-driving car might try to obey the norm to stop at stop signs, but will violate that norm if its brakes are not working, or if extenuating circumstances make it clearly unsafe to do so. It is also common to expect retribution in the case of a violated norm. This retribution may be exacted by an institution (such is the case when violating regulatory norms), or by other agents in the society (such is the case when violating social norms).[1] Norms are commonly held to be exogenous (originating from outside of an agent), but I will also consider norms that arise from and concern the same actor. Finally, as seen in Asimov's Three Laws of Robotics, norms may also be hierarchical.

3 Approaches to NAS

A natural question is how to represent norms and implement autonomous systems that obey them. One approach is to model norms implicitly. For example, inverse reinforcement learning systems that operate autonomous vehicles can learn to obey traffic laws. This approach relies on training data to impart norms into the system, and so it is difficult to enforce normative behavior directly. Further, it lacks a principled way to translate between natural language and the procedure for imparting norms. Importantly, this approach also lacks any clear method to reason about the norms of the system, and how they interact with each other. That is, how will changes to the system's training data affect its norms and behavior in the future?

Alternatively, norms can be modeled explicitly. Such representations seek to translate norms into mathematical rules that are then interpreted by the

[1] regulatory and social norms are sometimes referred to as *r-norms* and *s-norms*, respectively [24]. The former are determined by authority, and the latter are agreed upon (often implicitly) by the members of the society.

autonomous system. This approach is taken in [10], where ethical guidelines are incorporated as terms in an optimal controller's utility function and constraints. While this approach is straightforward to implement, and allows the system designer to impart norms directly to the system, it retains the other problems of implicit norm modeling.

Another way to explicitly model norms is to write them as specifications for the system. This approach is taken in [12] to describe and reason about legal systems, where the specifications are formalized in Linear Temporal Logic (LTL); and in [1] to define open organizations of agents, where organizational norms are formalized using first order logic.

Such an approach avoids the issues of implicit norm representation. Logics are developed with natural specification in mind, and determining a norm's consequences is a matter of formal reasoning. This raises the question of which logic should be used to specify norms. Logics of necessity and possibility (*alethic* logics) seem to be insufficient for expressing norms. To illustrate this, consider the norm "The car should stop at the stop sign". In an alethic logic, one might express this using Kripke semantics by stating "In all accessible worlds, the car stops at the stop sign" (styled '$\Box p$', where 'p' is the proposition that the car stops at the stop sign, and '$\Box$' is the necessity modality and the dual of possibility '$\Diamond$'). This is equivalent to saying "There is no accessible world in which the car does not stop at the stop sign". In such a case, it is clearly impossible to violate this norm, or any norm formulated as a matter of necessity. Since we understand norms to be violable (as discussed in Section 2), expressing norms in this manner does not capture all the essential features of norms.

The apparent unsuitability of alethic logics for expressing and reasoning about norms led to the development of deontic logic — a logic designed to reason about normative propositions.

4 Deontic Logic

Deontic logic can be considered as the formal study of norms and their interaction with each other [8]. In particular, deontic logic models norms in terms of *obligation*, *permission*, and *prohibition*. *Obligations* are positive normative imperatives, and are expressed by stating what ought to be the case. E.g. "It ought to be the case that the car stops at the stop sign" (styled '$\bigcirc p$'). *Permission* is usually defined as the dual of obligation, and something is *prohibited* (or forbidden) if it ought to be the case it does not hold ('$\bigcirc \neg p$').

While the foundations of deontic logic had been considered as far back as the tenth century [18], it wasn't until 1951 that philosopher Georg von Wright established it as a formal symbolic logic [26]. The logic that emerged from von Wright's work is known as Standard Deontic Logic (SDL).

SDL can be described in Kripke semantics by introducing a deontic accessibility relationship between worlds. That is, a world w is deontically accessible from world w' just in case world w is an ideal world — a world where all obligations hold. SDL is a normal S5 modal logic.

Deontic logics are built to express and reason about norms, so they are better suited for designing NAS than alethic logics. Deontic logic's status as a formal logic makes reasoning about norms possible.

In [8], the authors propose that deontic logic can be used to develop logic-based agents that can derive what they ought to do via deduction from a norm base. In [2], deontic logic is proposed as a method for formal verification and monitoring of normative properties in automata. In [23], the authors advance the verification proposal by developing a model-checking algorithm for deontic logic formulas. The authors of [20] use deontic logic to supervise a reinforcement learning agent.

SDL provides a fine starting point for the discussion of deontic logic, but it is plagued by paradoxes in practical scenarios, and lacks much of the context that makes normative reasoning special [8]. I now introduce Dominance Act Utilitarianism as a richer focal point for the study of deontic logic.

5 Dominance Act Utilitarianism

Dominance Act Utilitarianism (DAU) is a deontic logic that integrates modalities for agency, paths, states, and counterfactuals [16]. The modal operators introduced here make DAU a normal multi-S5 logic, but it can support non-monotonic modalities [19].

Syntax. Let *Agents* be a finite set of agents. The language of well-formed DAU formulas is given by:

$$A := \phi \mid \neg A \mid A \wedge A \mid \ [\alpha\, cstit\colon A] \ \mid \ \odot\,[\alpha\, cstit\colon A] \ \mid \ \odot\,([\alpha\, cstit\colon A]/\phi)$$

where $\alpha \in Agents$, $\wedge, \neg$ are the usual boolean connectives, and ϕ is a formula in a *branching time logic*. The *branching time logic* is used to describe an agent's mission and the states in the world. The informal description of *branching time logic* operators is given here. The reader can refer to [16] for formal semantics: the operator $\Box$ means "Historically Necessary" ($\Box A$ means A is true no matter which future is taken), $\Diamond$ means "Historically Possibe" ($\Diamond A$ means there is a future that leads to A). The formula $\mathsf{P}\,A$ means "there is a previous moment on this history where A holds", and $\mathsf{F}\,A$ means "there is a future moment on this history where A holds". The DAU-specific operators informally mean the following: $[\alpha\, cstit : A]$ is the agency operator and says that α sees to it, or ensures, that A is true; $\odot[\alpha\, cstit\colon A]$ is the obligation modality and says that α ought to ensure that A is true; finally, $\odot([\alpha\, cstit\colon A]/\phi)$ says that under the condition ϕ, α ought to ensure that A is true. The formal semantics of these deontic operators follow.

Branching time. Let *Tree* be a set of *moments* with an irreflexive, transitive ordering relation $<$ such that for any three moments m_1, m_2, m_3 in *Tree*, if $m_1 < m_3$ and $m_2 < m_3$ then either $m_1 < m_2$ or $m_2 < m_1$. There is a

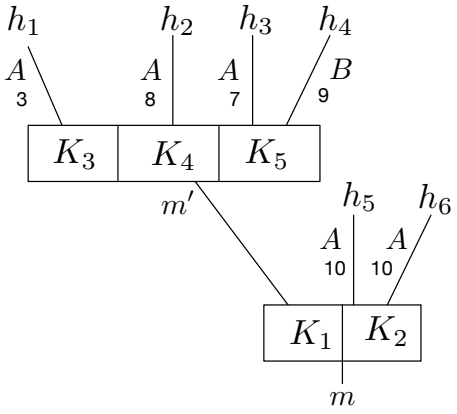

Figure 1: A utilitarian stit model for an agent α illustrating the main DAU definitions. Moments $m < m'$ with sets of histories $H_m = \{h_1, \ldots, h_6\}$ and $H_{m'} = \{h_1, \ldots, h_4\}$. Each moment is marked with the actions available to α at that moment: $Choice^m_\alpha = \{K_1, K_2\}$ and $Choice^{m'}_\alpha = \{K_3, K_4, K_5\}$. Action $K_2 = \{h_5, h_6\}$ and $K_4 = \{h_2\}$. Each history is marked with the formula(s) that it satisfies at m and with its value $Value(h)$, e.g., m/h_1 satisfies A and has value 3. $m/h_5 \models [\alpha\, cstit: A]$ since $Choice^m_\alpha(h_5) = K_2$, and both h_5 and h_6 satisfy A. On the other hand, $m/h_1 \not\models [\alpha\, cstit: A]$ since $Choice^m_\alpha(h_1) = K_1 = \{h_1, h_2, h_3, h_4\}$ and h_4 does not satisfy A. $Optimal^m_\alpha = \{K_2\}$ so $m/h_5 \models \odot[\alpha\, cstit: A]$. $Optimal^{m'}_\alpha = \{K_4, K_5\}$ and so α has no obligations at m' since there is no formula ϕ s.t. $|\phi|_{m'} \supseteq K_4 \cup K_5$ (See Def. 5.3).

unique *root moment* which is denoted by 0. A *history* is branch of the tree that extends infinitely into the future. Formally, it is a maximal, linearly ordered set of moments from *Tree*. Given a moment $m \in Tree$, H_m is the set of histories that go through m: $H_m := \{h \mid m \in h\}$. See Fig. 1. A pair of moments and histories is denoted by m/h, where $m \in Tree$ and $h \in H_m$.

Definition 5.1. [16, Def. 2.2] With AP a set of atomic propositions, a *branching time model* is a tuple $\mathcal{M} = (Tree, <, v)$ where *Tree* is a tree of moments with ordering $<$ and v is a function that maps m/h pairs in $\mathcal{M}$ to sets of atomic propositions from 2^{AP}, the set of subsets of AP.

A formula is considered to hold or not at an m/h pair. This is written: $\mathcal{M}, m/h \models \phi$, and it is granted that $h \in H_m$. The operators of branching time, F and $\Box$ are evaluated by:

$$\mathcal{M}, m/h \models \mathsf{F}\, A \text{ iff } \exists m' \in h \text{ s.t. } m < m' \text{ and } \mathcal{M}, m'/h \models A$$

$$\mathcal{M}, m/h \models \Box A \text{ iff } \forall h' \in H_m \mathcal{M}, m/h' \models A$$

The P operator is defined like F, but for moments $m' < m$, and the $\Diamond$ operator is the dual of $\Box$.

Given a DAU statement A, the *proposition* it expresses at moment m is the set of histories where it holds starting at m

$$|A|_m^{\mathcal{M}} := \{h \in H_m \mid \mathcal{M}, m/h \models A\} \tag{1}$$

$\mathcal{M}$ is omitted from the notation where there is no risk of ambiguity; writing instead, e.g. $|A|_m$, or $m/h \models A$.

Choice. Consider an agent $\alpha \in Agents$. Formally, at m, an action K is a subset of H_m: this is the subset of histories that are still realizable after taking the action. At every moment m, α is faced with a choice of actions which is denoted by $Choice_\alpha^m$. So $Choice_\alpha^m \subset 2^{H_m}$. See actions in Fig. 1. $Choice_\alpha^m$ must obey certain constraints given in the Supplementary material. In what follows, $Choice_\alpha^m$ is assumed finite for every α and m.

Agency. Agency is defined via the 'Chellas sees to it' operator *cstit*, named after Brian Chellas [9]. This operator can be used on its own to define a '*stit*-logic', whose product with computational tree logic (CTL) approaches the semantics of DAU [6]. Intuitively, an agent *sees to it*, or ensures, that A holds at m/h if it takes an action K s.t., whatever other history h' could've resulted from K, A is true at m/h' as well. I.e., the non-determinism does not prevent α from guaranteeing A.

Definition 5.2 (Chellas cstit)**.** [16, Def. 2.7] With agent α and DAU statement A, let $Choice_\alpha^m(h)$ be the unique action that contains h. Then

$$\mathcal{M}, m/h \models [\alpha\, cstit : A] \text{ iff } Choice_\alpha^m(h) \subseteq |A|_m^{\mathcal{M}}$$

If $K \subseteq |A|_m$ we say K *guarantees* A. See Fig. 1.

Optimal actions. An agent's obligations in DAU are defined in terms of an agent's 'optimal actions' — those actions that bring about an ideal state of affairs. Let $Value : H_0 \to \mathbb{R}$ be a *value function* that maps histories of $\mathcal{M}$ to *utility values* from the real line $\mathbb{R}$. This value represents the utility associated by all the agents to this common history. Two sets of histories Z and Y are ordered as

$$Z \leq Y \text{ iff } Value(h) \leq Value(h') \quad \forall\, h \in Z, h' \in Y \tag{2}$$

Let $State_\alpha^m := Choice_{Agents\setminus\{\alpha\}}^m$ be the set of *background states* against which α's decisions are to be evaluated. These are the choices of action available to other agents. Given two actions K, K' in $Choice_\alpha^m$, $K \preceq K'$ iff $K \cap S \leq K' \cap S$ for all $S \in State_\alpha^m$. That is, K' dominates K iff it is preferable to it regardless of what the other agents do (known as *sure-thing reasoning*). Strict inequalities are naturally defined. Optimal actions are given by

$$Optimal_\alpha^m := \{K \in Choice_\alpha^m \mid \nexists K' \in Choice_\alpha^m \text{ s.t. } K \prec K'\} \tag{3}$$

$Optimal_\alpha^m$ is non-empty in models with finite $Choice_\alpha^m$ [16, Thm. 4.10].

Dominance Ought. Obligations can now be defined. Intuitively, at moment m, agent α *ought to see to it* that A iff A is a necessary condition of all the histories considered ideal at moment m. This is formalized in the following *dominance Ought operator*, which is pronounced "α ought to see to it that A holds".

Definition 5.3 (Dominance Ought)**.** With α an agent and A an obligation in a model $\mathcal{M}$,

$$\mathcal{M}, m/h \models \odot[\alpha\, cstit\colon A] \text{ iff } K \subseteq |A|^{\mathcal{M}}_m \quad \text{for all } K \in Optimal^m_\alpha \tag{4}$$

See Fig. 1 for examples. The dominance ought satisfies a number of intuitive logical properties; notably, it is a normal S5 modal operator. The reader may refer to [16, Ch. 4] for more details.

Conditional obligation. It is often necessary to say that an obligation is imposed only under certain conditions. Let X be a proposition, i.e. $X = |\phi|_m$ for some ϕ. The choice of actions available to α at m under the condition that X holds is defined as $Choice^m_\alpha/X := \{K \in Choice^m_\alpha \mid K \cap X \neq \emptyset\}$. This is the right definition because non-determinism might make it impossible to have $K \subseteq X$ (i.e., an action that guarantees X), but future actions might still ensure the finally realized history will satisfy X. Thus in Fig. 1 $Choice^m_\alpha/B = \{K_1\}$. Conditional dominance is then defined by comparing only histories that satisfy ϕ: for two actions K, K' from $Choice^m_\alpha$, $K \preceq_X K'$ iff $K \cap S \cap X \leq K' \cap S \cap X$ for all $S \in State^m_\alpha$. The *conditionally optimal actions* are then

$$Optimal^m_\alpha/X := \{K \in Choice^m_\alpha/X \mid \nexists K' \in Choice^m_\alpha/X \text{ s.t. } K \prec_X K'\} \tag{5}$$

Finally, where A is an obligation and ϕ a formula in the underlying temporal logic, the conditional Ought is defined by

$$\mathcal{M}, m/h \models \odot([\alpha\, cstit\colon A]/\phi) \text{ iff } K \subseteq |A|^{\mathcal{M}}_m \ \forall K \in Optimal^m_\alpha/|\phi|^{\mathcal{M}}_m. \tag{6}$$

Notably, conditional obligation is *not* the same as $\phi \implies \odot[\alpha\, cstit : \phi]$. Conditional obligation only compares ϕ-satisfying histories, while this latter formula still compares all histories.

6 Contrary to Duty Reasoning

Conditional obligations are key in DAU for dealing with so-called "contrary to duty" (CTD) obligations. These CTD obligations are concerned with what ought to be the case when an obligation is violated [21]. Since norms are violable, it is important to be able to reason about what an agent ought to do in the contrary to duty case.

An example CTD scenario presented in [21] regarding the appearance of holiday cottages is as follows:

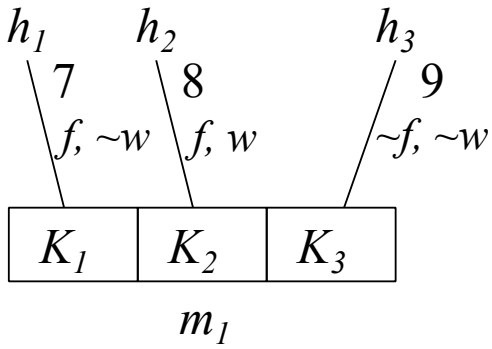

Figure 2: A utilitarian stit model for agent α illustrating a valid model for Example 6. Here $m_1/h_1 \models f$ as does m_1/h_2. It is clear that $[\alpha\, cstit\colon f]$ holds for m_1/h_1 and m_1/h_2 because both $Choice^m_\alpha(h1) = K_1$ and $Choice^m_\alpha(h2) = K_2$ guarantee f. $\odot[\alpha\, cstit : \neg f]$ also holds because K_3 dominates all other actions (and is therefore the only member of $Optimal^m_\alpha$), and K_3 guarantees $\neg f$. $\odot([\alpha\, cstit\colon w]/f)$ holds because $Optimal^m_\alpha/f = K_2$, and K_2 guarantees f.

CTD Example 1.

1. There must be no fence.
2. If there is a fence then it must be a white fence.
3. There is a fence.

This is troublesome to formulate in SDL. The natural approach would be to state "There ought not be a fence, and a fence implies it ought to be a white fence, and there is a fence": $\bigcirc\neg f \wedge (f \implies \bigcirc w) \wedge f$. However, in SDL (and, indeed, in any normal deontic logic) $f \implies \bigcirc f$ by necessitation. Thus we have $\bigcirc\neg f \wedge \bigcirc f$, from which we can derive any obligation due to the principle of explosion.

In DAU these norms may be formalized as $\odot[\alpha\, cstit : \neg f] \wedge \odot([\alpha\, cstit : w]/f) \wedge f$. The proposition f may hold in m'/h', but that does not necessarily imply that the obligation is violated — the optimal action may not include h'. Therefore I can devise a model in which these statements hold (see Figure 2), so this formalization is consistent. If the third statement in the example is interpreted not just as something that holds true in one possible history, but as something the agent sees to, the example may be formalized as $\odot[\alpha\, cstit\colon \neg f] \wedge \odot([\alpha\, cstit\colon w]/f) \wedge [\alpha\, cstit\colon f]$. This case may still be satisfiable provided some action available to the agent can guarantee f (see Figure 2). However, if statement 3 is interpreted as a statement of necessity then we are left with: $\odot[\alpha\, cstit\colon \neg f] \wedge \odot([\alpha\, cstit\colon w]/f) \wedge \Box f$. Because DAU is normal (and thus $\Box f \implies \odot[\alpha\, cstit\colon f]$), we can deduce $\odot[\alpha\, cstit\colon \neg f] \wedge \odot[\alpha\, cstit\colon f]$; again invoking the principle of explosion.

This last case exposes some flaws in the formulation of DAU that are common to other deontic logics [22]. First, DAU can not effectively express obligations that can not possibly be fulfilled. Though this may seem like the right conclusion, it seems to be that humans are capable of reasoning about norms that can not be met. In the given example a human would ideally paint their cottage fences white, even if all cottages had fences. Second, DAU can not effectively express true moral dilemmas. It does not seem like desired behavior to accept all obligations when faced with two ideal, but mutually exclusive worlds.

One proposed solution to these problems is the use of defeasible logic [10]. For the first problem, a defeat mechanism could be implemented to release the agent from an obligation that can not be met. However, it has been argued that such a mechanism would make it impossible to determine if a norm has been truly violated, or if a permission had been granted to ignore that norm [21]. For the second problem, non-monotonic reasoning could assign one obligation more weight than the other, but DAU already incorporates a preference relation, so such a solution would only change the modality of the dilemma. Non-monotonic reasoning does, however, provide a useful framework for describing hierarchies of norms, and norm change. Another solution involves introducing paraconsistency to the logic [11]. This would avoid explosion in the logic, but would trade away commonly accepted principles of the logic.

Now I will explore CTD reasoning with a temporal dimension in a second example[2] from [21]:

CTD Example 2.

1. On Monday you ought to help your friend on Tuesday.
2. On Monday you ought not apologize to your friend on Wednesday if you help them on Tuesday.
3. On Monday you ought to apologize to your friend on Wednesday if you do not help them on Tuesday.
4. On Monday you will not help your friend on Tuesday.

In [21] it is shown that expressing this problem in SDL leads to the 'pragmatic oddity' where you ought to help your friend, *and* apologize to them for not helping. Avoiding the pragmatic oddity is a goal of successful CTD reasoning. In [7] the authors begin with a product of LTL and SDL, but approach the semantics of DAU as they attempt to represent a similar problem. In [22], a deontic logic with counterfactual and temporal modalities is used to solve this problem, but struggles to solve a problem like CTD Example 1. This failure may be related to the use of accessibility relations in the counterfactual modality as opposed to a preference ordering, as DAU and [7] do.

[2]Though originally introduced in [21], I am adapting the temporal presentation of this problem from [22]

An open area of research in CTD and alethic-deontic logics is how obligations propagate. Detachment is the method by which secondary, or conditional norms become primary norms [21]. Some work on propagation regarding the alethic modalities of DAU has been presented in [23], but it does not explore the counterfactual modalities. Another area of research in CTD is the nature of CTD obligations in multiagent games [25].

7 Group Agency

STIT logics, like the agency logic embedded in DAU, express an agent's ability to ensure some state through performing some action. These logics can also express the interaction between two or more agents, or represent the interactions between coalitions of agents. This allows deontic logics built on stit models to reason about how agents ought to interact with each other. This is also essential to multiagent system designers for reasoning about how choices of norms could impact the system [6].

In [25], the authors propose the "left and right shoes" game in which two agents are each given two shoes, not of a pair (i.e. two left shoes or two right shoes). The agents must independently determine how many shoes they would like to give to the other agent with the knowledge that a matching pair of shoes is worth more than two not of a pair. An agent can give away 0, one, or both their shoes ($K_{\alpha,0}, K_{\alpha,1}$, and $K_{\alpha,2}$ respectively). The utility they receive is determined in part by the action the other agent takes (see Table 1 for utility assignment). The authors formalize the game in a close relative to DAU, and

j, i	$K_{i,0}$	$K_{i,1}$	$K_{i,2}$
$K_{j,0}$	$2, 2$	$4, 1$	$6, 0$
$K_{j,1}$	$1, 4$	$3, 3$	$4, 1$
$K_{j,2}$	$0, 6$	$1, 4$	$2, 2$

Table 1: Utilities for left and right shoes agents i and j given their choices of action.

show that the optimal action available to each agent is related to that agent's dominant strategy. They also remark that if each agent acts according to their obligations (that is, to not give away any shoes), an equilibrium is obtained, but it is not an optimal outcome. However, if the agents act contrary to duty, then an optimal outcome is achieved.

Group action in DAU was presented in [16] for groups of agents receiving the same utility from the intersection of their actions. In [17], semantics are introduced for determining optimal actions for groups with knowledge of future moments. With this, the author shows that individual agents acting from a future perspective reach an equilibrium for the group, but not necessarily a group optimum if there are multiple such equilibria. If the agents act cooperatively, however, then they are capable of ensuring an optimal outcome. If the agents do not act from a future perspective (and therefore do not have knowledge of

what other agents may do), then they can not guarantee cooperation, optimal outcomes, or equilibrium (specifically in the absence of static equilibria). This demonstrates the importance of information on the success of groups of agents, and further epistemic frames are developed for DAU in [15] and [14].

8 Computational Details of DAU

Modeling multiagent systems is a strength of stit logics, however it comes at a computational cost. It has been shown that satisfiability of atemporal stit formulas that model two or more agents is undecidable [13]. [13] also shows that such logics are not finitely axiomatizable; that is, there is no finite set of axioms from which all the theorems of the logic could be derived. Reducing the agent count to one allows the logic to be decidable in nondeterministic exponential time, and an axiomatization has been given in [27]. However, [6] introduces a flow-product of CTL and stit, which avoids some of these issues; implying there may be interesting temporal fragments of stit that are axiomatizable and decidable.

Regarding DAU more directly, [19] shows that an interpretation of the logic without temporal or counterfactual operators is decidable and finitely axiomatizable. This simplified version of the logic has been mechanized as a natural deduction calculus [3]. However, it is argued in [19] that this simplified logic is essentially the same as SDL, and so shares in its paradoxes.

9 Conclusion

Deontic logic allows designers of autonomous systems to reason formally about the norms of those systems. With the development of more powerful tools in deontic logic, designers could better prevent the violation of rules, develop agents that internalize and reason about explicitly specified social conduct, explain why an agent reasoned a move was optimal, or deduce the impacts of introducing a competitor to a society. Though the number of modalities that seem to be critical to effective deontic reasoning make the subject computationally complex to explore, the number of contexts deontic logics are able to capture make them highly expressive. Balancing this trade off between expressiveness and complexity is important. So, too, is exploring the interactions between the different modalities of deontic logic, and better characterizing obligation propagation and CTD obligation.

Deontic logic could also benefit from some practical applications. Its use in game theory is a promising start, and could be followed by control synthesis in decision problems and beyond. I also advocate for the use of deontic logic in model checking, where modal logic tools are already well developed. Finally, learning obligations from data is a compelling method to integrate deontic logic with common modern systems.

References

[1] Aldewereld, H., and Dignum, V. Operetta: Organization-oriented development environment. In *International Workshop on Languages, Methodologies and Development Tools for Multi-Agent Systems* (2010), Springer, pp. 1–18.

[2] Alechina, N., Bassiliades, N., Dastani, M., De Vos, M., Logan, B., Mera, S., Morris-Martin, A., and Schapachnik, F. Computational models for normative multi-agent systems. In *Dagstuhl Follow-Ups* (2013), vol. 4, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[3] Arkoudas, K., Bringsjord, S., and Bello, P. Toward ethical robots via mechanized deontic logic. In *AAAI fall symposium on machine ethics* (2005), The AAAI Press Menlo Park, CA, pp. 17–23.

[4] Asimov, I. I, robot, 1950.

[5] Bicchieri, C., Muldoon, R., and Sontuoso, A. Social Norms. In *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed., Winter 2018 ed. Metaphysics Research Lab, Stanford University, 2018.

[6] Broersen, J. Ctl. stit: enhancing atl to express important multi-agent system verification properties. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems: volume 1-Volume 1* (2010), pp. 683–690.

[7] Broersen, J., and Brunel, J. 'what i fail to do today, i have to do tomorrow': a logical study of the propagation of obligations. In *International Workshop on Computational Logic in Multi-Agent Systems* (2007), Springer, pp. 82–99.

[8] Broersen, J., Cranefield, S., Elrakaiby, Y., Gabbay, D., Grossi, D., Lorini, E., Parent, X., van der Torre, L. W., Tummolini, L., Turrini, P., et al. Normative reasoning and consequence. In *Dagstuhl Follow-Ups* (2013), vol. 4, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[9] Chellas, B. *The Logical Form of Imperatives*. Department of Philosophy, Stanford University, 1968.

[10] Gerdes, J. C., and Thornton, S. M. Implementable ethics for autonomous vehicles. In *Autonomes fahren*. Springer, 2015, pp. 87–102.

[11] Goble, L. A logic for deontic dilemmas. *Journal of Applied Logic 3*, 3-4 (2005), 461–483.

[12] Gorín, D., Mera, S., and Schapachnik, F. A software tool for legal drafting. *arXiv preprint arXiv:1109.2658* (2011).

[13] Herzig, A., and Schwarzentruber, F. Properties of logics of individual and group agency. *Advances in modal logic 7* (2008), 133–149.

[14] Horty, J. *Epistemic oughts in stit semantics*. Ann Arbor, MI: Michigan Publishing, University of Michigan Library, 2019.

[15] Horty, J., and Pacuit, E. Action types in stit semantics. *The Review of Symbolic Logic 10*, 4 (2017), 617–637.

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

[17] Horty, J. F. Perspectival act utilitarianism. In *Dynamic Formal Epistemology*. Springer, 2011, pp. 197–221.

[18] McNamara, P., and Van De Putte, F. Deontic Logic. In *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed., Spring 2021 ed. Metaphysics Research Lab, Stanford University, 2021.

[19] Murakami, Y. Utilitarian deontic logic. *AiML-2004: Advances in Modal Logic 287* (2004), 287–302.

[20] Neufeld, E., Bartocci, E., Ciabattoni, A., and Governatori, G. A normative supervisor for reinforcement learning agents. *Automated Deduction–CADE 28* (2021), 565.

[21] Prakken, H., and Sergot, M. Contrary-to-duty obligations. *Studia Logica 57*, 1 (1996), 91–115.

[22] Rönnedal, D. Contrary-to-duty paradoxes and counterfactual deontic logic. *Philosophia 47*, 4 (2019), 1247–1282.

[23] Shea-Blymyer, C., and Abbas, H. Algorithmic ethics: Formalization and verification of autonomous vehicle obligations. *arXiv preprint arXiv:2105.02851* (2021).

[24] Tuomela, R., and Bonnevier-Tuomela, M. Norms and agreement. In *Social Ontology in the Making*. De Gruyter, 2020, pp. 283–292.

[25] Turrini, P., Parent, X., Van Der Torre, L., and Tosatto, S. C. Contrary-to-duties in games. In *Logic Programs, Norms and Action*. Springer, 2012, pp. 329–348.

[26] von Wright, G. H. Deontic logic. *Mind 60*, 237 (January 1951).

[27] Xu, M. Axioms for deliberative stit. *Journal of Philosophical Logic 27*, 5 (1998), 505–552.