

Logic and Games for Ethical Agents

Xin Sun
University of Luxembourg

April 18, 2016

- 1 Introduction
- 2 Background
 - Boolean game
 - Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic
- 3 Ethical agents
- 4 Complexity results
 - Complexity of deontic norm logic
 - High complexity
 - Low complexity
 - Intermediate complexity
- 5 Related work
- 6 Conclusion and future work

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

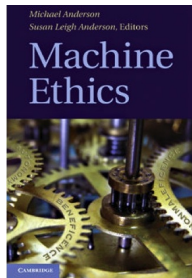
6 Conclusion and future work

The dawn of ethical agents

Ethical agents have been extensively studied in **moral philosophy** and in **economics**, and is identified as one of the thorniest challenges in **artificial intelligence**.

The intersection of these areas is the new field of **machine ethics**.

- Boer Deng. Machine ethics: the robot's dilemma. *Nature*, 523, 24-26 (02 July 2015)



Nagenborg identifies an artificial **moral agents** as an artificial agent guided by norms.

- Michael Nagenborg. *Artificial moral agents: An intercultural perspective*. *International Review of Information Ethics*, 7(9): 129-133, 2007.

In the BOID (belief-obligation-intention-desire) architecture [3]: a **selfish agent** is an agent whose desires override obligations; a **social agent** is an agent whose obligations override desires.

In this talk, we develop a formal model to characterize ethical agents of different types including selfish, social and moral agents.

Different agents have different reactions when there are conflicts between their obligations (moral value) and desires (utility). At least the following types of agents exist or should be constructed.

- 1 An *amoral* agent prefers actions with higher utility, and ignores the moral aspect of his actions.
- 2 A *moral* agent prefers actions with higher moral value and ignores the utility of his actions.
- 3 A *selfish* agent first prefers actions of higher utility. For two actions of the same utility, the agent prefers the one with higher moral value.
- 4 A *social* agent first prefers actions of higher moral value. For two actions of the same moral value, he prefers the action with higher utility.
- 5 A *negatively impartial* agent first classifies actions into illegal category and non-illegal category. Then he ranks his actions using utility within these two categories.
- 6 A *positively impartial* agent first classifies strategies into legal category and non-legal category. Then he ranks his actions using utility within these two categories.

Research question

How to formally characterize different types of ethical agents?

Methodology

We adopt a Boolean game + deontic logic approach to the construction of ethical agents.

success criteria

Our success criteria is to build formal models of ethical agents such that

- norms play an important role in agents' decision-making procedure,
- such procedures are decidable in general and computationally tractable under certain reasonable restrictions.

- 1 In the Boolean game theoretical setting, each agents controls a set of propositional variables.
- 2 An action/strategy of an agent is a truth assignment to the variables he controls.
- 3 Norms are used to classify actions as moral, legal or illegal.
- 4 We use deontic logic to represent norms and define the normative status of actions.
- 5 The preference relation in Boolean games are changed by the normative status of actions.
- 6 Agents of different types use different deontic logic for normative reasoning and have different procedures of preference change.
- 7 The deontic logic and the procedure of preference change characterizes different types of ethical agents.

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Definition (weighted boolean game)

A **weighted boolean game** is a 4-tuple $(Agent, \mathbb{P}, \pi, Goal)$, where

- 1 $Agent = \{1, \dots, n\}$ is a set of agents.
- 2 \mathbb{P} is a finite set of propositional variables.
- 3 $\pi : Agent \mapsto 2^{\mathbb{P}}$ is a control assignment function such that $\{\pi(1), \dots, \pi(n)\}$ forms a partition of \mathbb{P} . For each agent i , $2^{\pi(i)}$ is the set of all actions of i .
- 4 $Goal = \langle Goal_1, \dots, Goal_n \rangle$ is a sequence of sets of weighted formulas of $L_{\mathbb{P}}$. That is, each $Goal_i$ is a finite set $\{\langle x_1, m_1 \rangle, \dots, \langle x_k, m_k \rangle\}$ where $x_j \in L_{\mathbb{P}}$ and m_j is a real number.

Utilities function: for every action profiles S ,

$$u_i(S) = \sum \{m_j : \langle x_j, m_j \rangle \in Goal_i, S \models x_j\}$$

Agent's preference: $S \leq_i S'$ iff $u_i(S) \leq u_i(S')$.

Example

Let $G = (\text{Agent}, \mathbb{P}, \pi, \text{Goal})$ where

- $\text{Agent} = \{1, 2\}$,
- $\mathbb{P} = \{p, q, s\}$,
- $\pi(1) = \{p\}$, $\pi(2) = \{q, s\}$,
- $\text{Goal}_1 = \{\langle p \leftrightarrow q, 1 \rangle, \langle s, 2 \rangle\}$, $\text{Goal}_2 = \{\langle p \wedge q, 2 \rangle, \langle \neg s, 1 \rangle, \}$.

	$+q, +s$	$+q, -s$	$-q, +s$	$-q, -s$
$+p$	(3, 2)	(1, 3)	(2, 0)	(0, 1)
$-p$	(2, 0)	(0, 1)	(3, 0)	(1, 1)

1 Introduction

2 Background

- Boolean game
- **Deontic logic**
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

What is deontic logic?

Deontic logic is the logical study of deontic modality

obligation, permission, forbidden, must, ought, might

and norms

- You **should** drive on the right side.
- Murdering is **forbidden**.
- A PhD student in China is **permitted** to take no holiday.
- A PhD student in Luxembourg is **obliged** to take all his holidays before the end of his contract.

Approaches to deontic logic

Different approaches of deontic logics have been studied: standard deontic logic (SDL) (Georg von Wright), dyadic standard deontic logic (Lennart Åqvist), dynamic deontic logic (John-Jule Meyer, Ron van der Meyden), deontic action logic (Krister Segerberg), deontic STIT logic (John Horty), input/output logic (David Makinson, Leendert van der Torre), imperative logic (Jörg Hansen), deontic default logic (John Horty) and deontic defeasible logic (Guido Governatori).

In input/output logic, imperative logic, deontic default logic and deontic defeasible logic, norms are **explicitly** represented whereas in other approaches norms are represented implicitly.

We use deontic norm logic to refer input/output logic, imperative logic, deontic default logic and deontic defeasible logic.

Advantages of deontic norm logic 1: solving Jorgesen's dilemma

- Philosophically, it is widely acknowledged that norms have no truth value. They may be complied or violated. But it makes no sense to describe norms as true or as false.

“Mary, you may enter now!”: they do **not describe**, but **demand** a behavior of Mary. Being non-descriptive, they cannot meaningfully be termed true or false.

- Hence, though there certainly exists a logical study of normative expressions and concepts, it seems there cannot be a logic of norms: this is the well-known Jorgensen's dilemma.

In deontic norm logic, norms do not bear truth values. Norms are not even treated as formulas, but simply as ordered pairs (a, x) of formulas or a conditional imperative $a \Rightarrow !x$.

Deontic norm logic solves Jorgensen's dilemma at its starting line.

Advantages of deontic norm logic 2: solving contrary-to-duty paradox

Contrary-to-duty paradox

The original phrasing of the contrary-to-duty paradox requires a formalisation of the following scenario in which the sentences are mutually consistent and logically independent.

- 1 It ought to be that John goes to help his neighbours.
- 2 It ought to be that if John goes to help his neighbours, then he tells them he is coming.
- 3 If John doesn't go to help his neighbours, then he ought not to tell them he is coming.
- 4 John does not go to help.

But formalisations using SDL is either inconsistent or not logically independent.

Advantages of deontic norm logic 3: characterization various permission

- Philosophically, it is common to distinguish between two kinds of permission: negative permission and positive permission.
- Negative permission is straightforward: something is negatively permitted iff it is not forbidden.
- Positive permission is more elusive. Intuitively, something is positively permitted according to certain norms iff it can be derived from the norms. But what exactly does “derive” mean? In mathematics we can derive theorems in a “straight” way or by contradiction. These two methods of derivation give two different notions of positive permission.
- In legal philosophy, exception is considered as another notion of permission.

All these kinds of permission can be captured by deontic norm logic.

Advantages of deontic norm logic 4: dealing with moral conflicts

Before you go to a party, you become the recipient of various imperative sentences:

- ① Your mother says: if you drink anything, then don't drive.
- ② Your best friend says: if you go to the party, then you drive.
- ③ Some acquaintance says: if you go to the party, then have a drink with me.

Assume mother is more important than best friend, which is more important than an acquaintance.

What will you do?

Advantages of deontic norm logic 4: dealing with moral conflicts

Before you go to a party, you become the recipient of various imperative sentences:

- 1 Your mother says: if you drink anything, then don't drive.
- 2 Your best friend says: if you go to the party, then you drive.
- 3 Some acquaintance says: if you go to the party, then have a drink with me.

Assume mother is more important than best friend, which is more important than an acquaintance.

What will you do?

Intuitively, you should obey your mother and your best friend, and hence do the driving and not accept your acquaintance's invitation.

However, it is not so clear what formal mechanisms could explain this reasoning.

Deontic norm logic develops several mechanisms to deal with such

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

- Input/output logic is one of the new achievements in deontic logic in recent years.
- Operational semantics: norms are like a deductive machine.



Figure: input/output logic

Let $L_{\mathbb{P}}$ be the propositional language built from variables \mathbb{P} .

A norm is an ordered pair of formulas $(a, x) \in L_{\mathbb{P}} \times L_{\mathbb{P}}$.

- mandatory norm: “given a , x is obligatory”.
- permissive norms: “given a , x is permitted”.

Let O be a set of obligatory norms and P a set of permissive norms.

$N = (O, P)$ is called a normative system.

Mandatory norms O are viewed as a function, $O : 2^{L_{\mathbb{P}}} \mapsto 2^{L_{\mathbb{P}}}$:

for $A \subseteq L_{\mathbb{P}}$, $O(A) = \{x \in L_{\mathbb{P}} : (a, x) \in O \text{ for some } a \in A\}$.

Definition

Given O and A ,

- $out_1(O, A) = Cn(O(Cn(A)))$.
- $out_2(O, A) = \bigcap \{Cn(O(V)) : A \subseteq V, V \text{ is complete}\}$.
- $out_3(O, A) = \bigcap \{Cn(O(B)) : A \subseteq B = Cn(B) \supseteq O(B)\}$.
- $out_4(O, A) = \bigcap \{Cn(O(V)) : A \subseteq V \supseteq O(V), V \text{ is complete}\}$.

Cn is the consequence operator of propositional logic,

$Cn(A) = \{a \in L_{\mathbb{P}} : A \models a\}$.

V is complete if it is either maximal consistent or equals to $L_{\mathbb{P}}$.

(a, x) is derivable from O iff (a, x) is in the smallest set that extends $O \cup \{(T, T)\}$ and is closed under certain derivation rules.

Derivation rules

- SI (strengthening the input): from (a, x) to (b, x) whenever $b \vDash a$.
 - WO (weakening the output): from (a, x) to (a, y) whenever $x \vDash y$.
 - AND (conjunction of output): from (a, x) and (a, y) to $(a, x \wedge y)$.
 - OR (disjunction of input): from (a, x) and (b, x) to $(a \vee b, x)$.
 - CT (cumulative transitivity): from (a, x) and $(a \wedge x, y)$ to (a, y) .
-
- $deriv_1$: SI + WO + AND
 - $deriv_2$: $deriv_1$ + OR; $deriv_3$: $deriv_1$ + CT
 - $deriv_4$: $deriv_1$ + OR + CT

Theorem ([7])

$x \in out_i(O, \{a\})$ iff $(a, x) \in deriv_i(O)$ for $i \in \{1, 2, 3, 4\}$.

Constrained input/output logic

To solve the contrary-to-duty paradox, constrained input/output logic is developed.

Definition (constrained input/output logic)

Given a set of mandatory norms O , a set of input $A \subseteq L_{\mathbb{P}}$ and a set of constrains $C \subseteq L_{\mathbb{P}}$, for $i \in \{1, \dots, 4\}$:

- $maxfamily_i(O, A, C) = \{O' \subseteq O : out_i(O', A) \cup C \text{ is consistent, and } out_i(O'', A) \cup C \text{ is not consistent, for every } O' \subsetneq O''\}$.
- $x \in out_i^{\cap}(O, A, C)$ iff $x \in \bigcap \{out_i(O', A) : O' \in maxfamily_i(O, A, C)\}$.

Permission in input/output logic

Given a normative system $N = (O, P)$ and a set of formulas A ,

Definition (negative permission)

$$\text{NegPerm}_i(N, A) = \{x \in L_{\mathbb{P}} : \neg x \notin \text{out}_i(O, A)\}.$$

Definition (positive permission)

- If $P = \emptyset$, then $\text{PosPerm}_i(N, A) = \text{out}_i(O, A)$.
- If $P \neq \emptyset$, then $\text{PosPerm}_i(N, A) = \{x \in L_{\mathbb{P}} : x \in \text{out}_i(O \cup \{(a', x')\}, A)$
for some $(a', x') \in P\}$.

Intuitively, permissive norms are treated like weak mandatory norms, while mandatory norms may be used jointly, permissive norms can only be applied one by one.

A man is permitted to date either one of two girls, but not both of them.

Prioritized input/output logic

Parent [9] introduces a formation of prioritized input/output logic to dealing with moral conflicts. Let $O^{\succeq} = (O, \succeq)$ be a set of prioritized norms where $\succeq \subseteq O \times O$ is a reflexive, transitive and total relation representing the priority of norms: $(a, x) \succeq (b, y)$ means the priority of (a, x) is (weakly) higher than the priority of (b, y) .

The priority relation over norms is lifted to priority over sets of norms: $O_1 \succeq O_2$ iff for all $(a_2, x_2) \in O_2 - O_1$ there is $(a_1, x_1) \in O_1 - O_2$ such that $(a_1, x_1) \succeq (a_2, x_2)$.

$$x \in out_i^P(O^{\succeq}, A, C) \text{ iff } x \in \bigcap \{out_i(O', A) : O' \in preffamily_i(O^{\succeq}, A, C)\}.$$

Here $preffamily_i(O^{\succeq}, A, C)$ is defined via the following steps:

- 1 $maxfamily_i(O, A, C) = \dots$
- 2 $preffamily_i(O^{\succeq}, A, C)$ is the set of \succeq -maximal elements of $maxfamily_i(O, A, C)$.

1 Introduction

2 Background

- Boolean game
- **Deontic logic**
 - Input/output logic
 - **Imperative logic**
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Based on unconstrained input/output logic, Hansen [5] develops his prioritized imperative logic.

Definition (materialization)

$m(O) = \{a \rightarrow x : (a, x) \in O\}$ is the materialization of O .

Definition (preferred maximal obeyable family)

Given a set of prioritized norms O^{\geq} and a set of formulas A . Let $(a_1, x_n), \dots, (a_n, x_n)$ be a strict linear ordering of O such that $(a_i, x_i) > (a_{i+1}, x_{i+1})$.

$O' \in \text{pomfamily}(O^{\geq}, A)$ if A is consistent and

- 1 $O_0 = \emptyset$,
- 2 $O_{i+1} = O_i \cup \{(a_i, x_i)\}$ if $A \cup m(O_i \cup \{(a_i, x_i)\})$ is consistent.
Otherwise $O_{i+1} = O_i$,
- 3 $O' = \bigcup_{i=0}^n O_i$

$x \in \text{out}_i^h(O^{\geq}, A)$ iff $x \in \bigcap \{\text{out}_i(O', A) : O' \in \text{pomfamily}_i(O^{\geq}, A)\}$.

1 Introduction

2 Background

- Boolean game
- **Deontic logic**
 - Input/output logic
 - Imperative logic
 - **Deontic default logic**
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

A prioritized default theory is notated (O, A, \geq) .

Given a scenario $S \subseteq O$, $Conclusion(S) = \{x : (a, x) \in S\}$.

$$Triggered_{(O, A, \geq)}(S) = \{(a, x) \in O : A \cup Conclusion(S) \vdash a\}.$$

$$Conflicted_{(O, A, \geq)}(S) = \{(a, x) \in O : A \cup Conclusion(S) \vdash \neg x\}.$$

For $O_1, O_2 \subseteq O$, $O_1 \succ O_2$ if for all $(a_1, x_1) \in O_1$, $(a_2, x_2) \in O_2$, $(a_1, x_2) > (a_2, x_1)$.

Let $S^{S'/O'} = (S - S') \cup O'$.

$Defeated_{(O, A, \geq)}(S) = \{(a, x) \in O : \exists O' \subseteq Triggered_{(O, A, >)}(S) \text{ s.t.}$

- ① $O' > \{(a, x)\}$
- ② $\exists S' \subseteq S$ with $O' > S'$ such that
 - ① $A \cup Conclusion(S^{S'/O'})$ is consistent
 - ② $A \cup Conclusion(S^{S'/O'}) \vdash \neg x$

Definition (Proper scenario)

Let S be a scenario based on the prioritized default theory $(O, A, >)$. Then S is a proper scenario based on (O, A, \geq) just in case $S = \bigcup_{i \geq 0} S_i$ where

- $S_0 = \emptyset$
- $S_{i+1} = \{(a, x) \in O : (a, x) \in \text{Triggered}_{(O, A, \geq)}(S_i), (a, x) \notin \text{Conflicted}_{(A, O, >)}(S), (a, x) \notin \text{Defeated}_{(O, A, \geq)}(S)\}$

$x \in \text{out}^d(O^{\geq}, A)$ if for all proper scenario S , $\text{Conclusion}(S) \vdash a$

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Given $\mathcal{P} = \{p_0, p_1, \dots\}$ a finite set of propositional variables, let $Lit_{\mathcal{P}} = \mathcal{P} \cup \{\neg p : p \in \mathcal{P}\}$ be the set of all literals build from \mathcal{P} . Let O and P be two modality representing obligation and permission receptively.

Definition

Let Lab be a set of arbitrary labels. Every rule is of the type

$$r : A(r) \hookrightarrow C(r)$$

where

- 1 $r \in Lab$ is the name of the rule;
- 2 $A(r) = \{a_1, \dots, a_n\}$, the *antecedent* of the rule is the set of the premises of the rule. Each a_i is a literal;
- 3 $\hookrightarrow \in \{\Rightarrow_O, \Rightarrow_P, \rightsquigarrow\}$ denotes the type of the rule. If \hookrightarrow is \Rightarrow_O then the rule is an obligatory norm. If \hookrightarrow is \Rightarrow_P then the rule is a permissive norm. If \hookrightarrow is \rightsquigarrow , then the rule is a defeater.
- 4 $C(r)$ is the *consequent* of the rule, which is a literal.

Definition

A defeasible theory is a tuple $D = (F, R, >)$ where $F \subseteq Lit_{\mathcal{P}}$ is the set of facts, R is a set of rules and $>$ the superiority relation, is a binary relation over R .

Definition

A *proof* in a defeasible theory D is a sequence $Z(1), \dots, Z(n)$ of tagged literals in the form of $+\partial_O l$, $+\partial_P l$, $-\partial_O l$ and $-\partial_P l$ where $Z(1), \dots, Z(n)$ satisfy the proof conditions given below.

Given a defeasible theory $D = (F, R, >)$, a rule r is applicable iff for all $a_i \in A(r)$, $a_i \in F$. r is discarded if it is not applicable.

Definition

The proof condition of provability for obligation is : If $Z(n+1) = +\partial_O I$ then

- ① there is $r \in R^O[I]$ such that r is applicable, and
- ② for all $s \in R[\sim q]$, either
 - ① s is discarded, or
 - ② $s \in R^O$ and there is $t \in R[I]$ such that t is applicable and $t > s$, or
 - ③ $s \in R^P \cup R_{def}$ and there is $t \in R^O[I]$ such that t is applicable and $t > s$.

Definition

The proof condition of refutability for obligation is : If $Z(n + 1) = -\partial_O I$ then

- 1 for all $r \in R^O[I]$ either r is discarded, or
- 2 there is $s \in R[\sim q]$, such that
 - 1 s is applicable, and
 - 2 if $s \in R^O$ then for all $t \in R[I]$ either t is discarded or $t \not\geq s$, and
 - 3 if $s \in R^P \cup R_{def}$ then for all $t \in R^O[I]$ either t is discarded and $t \not\geq s$.

Definition

The proof condition of provability for permission is : If $Z(n + 1) = +\partial_O I$ then

- 1 there is $r \in R^P[I]$ such that r is applicable, and
- 2 for all $s \in R^O[\sim q]$, either
 - 1 s is discarded, or
 - 2 there is $t \in R[I]$ such that t is applicable and $t > s$

- Deontic norm logics are convenient tools for the construction of ethical agents.
- All these deontic norm logics are capable of solving the contrary-to-duty paradox and reasoning on moral conflicts.
- For the ease of exposition, we choose prioritized simple-minded input/output logic (out_1^P) to illustrate how we construct ethical agents.
- Ethical agents can be constructed similarly using other deontic norm logics.

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Normative multi-agent system

A normative multiagent system contains a multiagent system, a prioritized normative system and a collection of facts which we call it environment.

Definition (normative multiagent system)

A normative multiagent system is a tuple (G, N, E) where

- $G = (Agent, \mathbb{P}, \pi, Goal)$ is a weighted Boolean game.
- $N = (O, P, \geq)$ is a finite prioritized normative system. Here every permissive norm has the same priority and it is strictly lower than any mandatory norm.
- $E \subseteq L_{\mathbb{P}}$ is the environment, which is a finite set of formulas representing facts.

Permission in prioritized normative system

Given a prioritized normative system $N = (O, P, \geq)$ and a set of formulas A ,

Definition (negative permission)

$$\text{NegPerm}_k(N, A) = \{x \in L_{\mathbb{P}} : \neg x \notin \text{out}_k^P(O, A, \emptyset)\}.$$

Definition (positive permission)

- If $P = \emptyset$, then $\text{PosPerm}_k(N, A) = \text{out}_k^P(O, A, \emptyset)$.
- If $P \neq \emptyset$, then

$$\text{PosPerm}_k(N, A) = \{x \in L_{\mathbb{P}} : x \in \text{out}_k^P(O \cup \{(a', x')\}, A, \emptyset) \text{ for some } (a', x') \in P\}.$$

In a normative multiagent system, actions are classified as moral, positively legal, negatively legal or illegal.

Definition (moral, legal and illegal action)

Given a normative multiagent system (G, N, E) , for each agent i , an action $(+p_1, \dots, +p_m, -q_1, \dots, -q_n)$ is moral if

$$p_1 \wedge \dots \wedge p_m \wedge \neg q_1 \wedge \dots \wedge \neg q_n \in \text{out}_1^P(O^\geq, E, \emptyset).$$

The action is positively legal if

$$p_1 \wedge \dots \wedge p_m \wedge \neg q_1 \wedge \dots \wedge \neg q_n \in \text{StaPerm}_1(N, E).$$

The action is negatively legal if

$$p_1 \wedge \dots \wedge p_m \wedge \neg q_1 \wedge \dots \wedge \neg q_n \in \text{NegPerm}_1(N, E).$$

The action is illegal if

$$\neg(p_1 \wedge \dots \wedge p_m \wedge \neg q_1 \wedge \dots \wedge \neg q_n) \in \text{out}_1^P(O^\geq, E, \emptyset).$$

- Moral, positively legal, negatively legal and illegal are four normative positions of actions.
- We assume the normative position degrades from moral to positively legal, then to negatively legal, and finally to illegal.
- The **normative status** of an action is the highest normative position it has.
- Normative status offers a measure of the moral value of actions.

Example

Let (G, N, E) be a normative multiagent system as follows:

- $G = (\text{Agent}, \mathbb{P}, \pi, \text{Goal})$ is a Boolean game with
 - $\text{Agent} = \{1, 2\}$,
 - $\mathbb{P} = \{p, q\}$,
 - $\pi(1) = \{p\}$, $\pi(2) = \{q\}$,
 - $\text{Goal} = \langle \{(p \wedge q, 1)\}, \{(p \vee q, 1)\} \rangle$.
- $N = (O, P, \geq)$ where $O = \{(\top, p)\}$, $P = \{(\top, q)\}$, $\geq = \emptyset$.
- $E = \emptyset$.

	$+q$	$-q$
$+p$	(1, 1)	(0, 1)
$-p$	(0, 1)	(0, 0)

Then $out_1(O, E) = Cn(\{p\}) = out_1^P(O^{\geq}, E, \emptyset)$,

$StaPerm_1(N, E) = Cn(\{p, q\})$. Therefore the normative status of

$+p, +q, -q, -p$ is respectively moral, positively legal, negatively legal and illegal.

Agent types

In a normative multiagent system, agent's preference over action profiles is changed by the normative status of strategies. Different types of ethical agents change their preference in different ways. Informally,

- ① An *amoral* agent prefers action profiles with higher utility.
- ② A *moral* agent prefers action profiles with higher normative status.
- ③
 - An *selfish* agent first prefers action profiles with higher utility.
 - For two action profiles of the same utility, the agent prefers the one of higher normative status.
- ④
 - A *social* agent first prefers action profiles with higher normative status.
 - For two action profiles of the same normative status, he prefers action profiles with higher utility.
- ⑤
 - A *negatively impartial* agent first classifies actions into negatively legal category and illegal category.
 - Then he ranks his actions using utility within these two categories.
- ⑥
 - A *positively impartial* agent first classifies actions into positively legal category and not positively legal category.
 - Then he ranks his actions using utility within these two categories.

We call amoral, selfish, negatively impartial, positively impartial, social and moral agents type-0, type-1, ..., type-5 agents respectively.

In Lorini [6], the *degree of moral sensitivity* is used to measure the strength of his moral value on his preference. That is, an agent is more moral if the degree of moral sensitivity is higher. Combining our terminology with Lorini's, the degree of moral sensitivity of type- i agents is higher than that of type- j agents iff $i > j$.

Given a normative multiagent system, it induces a normative Boolean game, which characterize ethical agents and their interaction.

Definition (normative Boolean game)

Given a normative multiagent system (G, N, E) where $G = (\text{Agent}, \mathbb{P}, \pi, \text{Goal})$, it induces a normative Boolean game $G^N = (\text{Agent}, \mathbb{P}, \pi, \prec_1, \dots, \prec_n)$ where \prec_i is the preference of i over action profiles such that

- if i is type-0 (amoral), then $s \prec_i s'$ if
 - $u_i(s) < u_i(s')$.
- if i is type-1 (selfish), then $s \prec_i s'$ if
 - $u_i(s) < u_i(s')$, or
 - $u_i(s) = u_i(s')$ and the normative status of s'_i is higher than that of s_i .
- if i is type-2 (negatively impartial), then $s \prec_i s'$ if
 - s_i is illegal (not negatively legal) and s'_i is negatively legal, or
 - both s_i and s'_i are illegal and $u_i(s) < u_i(s')$, or
 - both s_i and s'_i are negatively legal and $u_i(s) < u_i(s')$.

Definition (normative Boolean game)

- if i is type-3 (positively impartial), then $s \prec_i s'$ if
 - s_i is not positively legal and s'_i is positively legal, or
 - both s_i and s'_i are not positively legal and $u_i(s) < u_i(s')$, or
 - both s_i and s'_i are positively legal and $u_i(s) < u_i(s')$.
- if i is type-4 (social), then $s \prec_i s'$ if
 - the normative status of s'_i is higher than that of s_i , or
 - the normative status of s'_i is equal to s_i and $u_i(s) < u_i(s')$.
- if i is type-5 (moral), then $s \prec_i s'$ if
 - the normative status of s'_i is higher than that of s_i

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Some complexity class

$P \subseteq NP \subseteq \dots \subseteq PSPACE$

$P \subseteq coNP \subseteq \dots \subseteq PSPACE$

complete problems from logic

- 1 P: satisfiability problem of Horn formulas.
- 2 NP: satisfiability problem of propositional logic.
satisfiability problem of modal logic $S5$.
- 3 coNP: validity problem of propositional logic.
- 4 PSPACE: satisfiability problem of modal logic $K, T, S4$.

$$P = \Pi_0^P = \Delta_0^P \subseteq \text{coNP} = \Pi_1^P \subseteq \Delta_2^P \subseteq \Pi_2^P \subseteq \Delta_3^P \subseteq \dots \subseteq \text{PSPACE}$$

A Π_2^P -complete problem

The validity problem of 2-QBF[∀]: given an arbitrary 2-QBF[∀], decide whether it is valid.

A 2-QBF[∀] formula is of the form $\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ where Φ is a propositional formula with variables in $\{p_1, \dots, p_m, q_1, \dots, q_n\}$.

$\forall p_1 \dots p_m \exists q_1 \dots q_n \Phi$ is valid if for all valuations V_1 for $\{p_1, \dots, p_m\}$ there is a valuation V_2 of $\{q_1, \dots, q_n\}$ such that $V_1 \cup V_2 \models \Phi$.

More complexity class

$$P = \Pi_0^P = \Delta_0^P \subseteq \text{coNP} = \Pi_1^P \subseteq \Delta_2^P \subseteq \Pi_2^P \subseteq \Delta_3^P \subseteq \dots \subseteq \text{PSPACE}$$

A Δ_2^P -complete problem

Max SAT: given an arbitrary 1-QBF $^{\exists} \exists p_1 \dots p_m \Phi$, decide if $V_1(p_m) = 1$ where V_1 is the **lexicographically maximal** valuations for $\{p_1, \dots, p_m\}$ such that $V_1 \models \Phi$.

A Δ_3^P -complete problem

Maximum 2-QBF: given an arbitrary 2-QBF $^{\exists} \exists p_1 \dots p_m \forall q_1 \dots q_n \Phi$, decide if $V_1(p_m) = 1$ where V_1 is the **lexicographically maximal** valuations V_1 for $\{p_1, \dots, p_m\}$ such that for all valuation V_2 of $\{q_1, \dots, q_n\}$, $V_1 \cup V_2 \models \Phi$.

lexicographically larger: $1 > 0$, $11 > 10$, $1000 > 0111$, ...

Complexity of unconstrained input/output logic

Compliance problem:

Given O, A and x , decide if $x \in out_i(O, \{A\})$.

Theorem (Sun and Ambrossio [12])

- 1 For $k \in \{1, 2, 4\}$, the compliance problem is coNP complete.
- 2 For $k = 3$, the compliance problem is coNP-hard and in Δ_2^P .

Theorem (Sun [11])

Let $B_A^O = \bigcup_{i=0}^{\infty} B_{A,i}^O$, where $B_{A,0}^O = Cn(A)$, $B_{A,i+1}^O = Cn(A \cup O(B_{A,i}^O))$.

- $(a, x) \in deriv_3(O)$ iff $x \in Cn(O(B_{\{a\}}^O))$.

Here B_A^O can be interpreted as the least fixed point of function $f_A^O : 2^{L_P} \rightarrow 2^{L_P}$ such that $f_A^O(X) = Cn(A \cup O(X))$.

Complexity of constrained input/output logic

Full meet compliance problem:

Given O, A, C and x , decide if $x \in out_i^\cap(O, A, C)$.

Theorem

- 1 For $k \in \{1, 2, 3, 4\}$, the full meet compliance problem is Π_2^P complete.

Prioritized full meet compliance problem:

Given O, A, C and x , decide if $x \in out_i^P(O^\geq, A, C)$.

Theorem

- 1 For $k \in \{1, 2\}$, the full meet compliance problem is Π_2^P complete.
- 2 For $k \in \{3, 4\}$, the full meet compliance problem is Δ_3^P -hard and in Π_3^P .

Complexity of imperative logic

Prioritized full meet compliance problem for imperative logic:

Given O, A and x , decide if $x \in out_i^h(O \geq, A)$.

Prioritized full meet compliance problem for deontic default logic:

Given O, A and x , decide if $x \in out_i^d(O \geq, A)$.

Theorem

- 1 For $k \in \{1, 2, 3, 4\}$, prioritized full meet compliance problem for imperative logic is Π_2^P complete.
- 2 Prioritized full meet compliance problem for deontic default logic is Δ_3^P -hard and in Π_3^P .

Xin Sun, Xavier Parent, Livio Robaldo. 'Computational complexity of deontic logic'. In Amit Chopra, Leendert van der Torre, Harko Verhagen and Serena Villata, editors, *Handbook of normative multiagent systems*. College Publications. To appear, 2016.

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- **High complexity**
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Theorem

Given a normative multiagent system (G, N, E) , a type- k agent and his strategy $(+p_1, \dots, +p_m, -q_1, \dots, -q_n)$, deciding whether this strategy is moral is Π_2^P -complete, for $k \in \{0, 1, 2, 3, 4, 5\}$.

Theorem

Given a normative multiagent system (G, N, E) , an agent i and two action profiles s and s' , deciding whether $s \prec_i s'$ is Π_2^P -hard and in Δ_3^P .

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- **Low complexity**
- Intermediate complexity

5 Related work

6 Conclusion and future work

Let $Lit_{\mathbb{P}} = \mathbb{P} \cup \{\neg p : p \in \mathbb{P}\}$ be the set of literals build on \mathbb{P} . Let $L_{\mathbb{P}}^{cnl}$ be the conjunctions of literals (CNL) of \mathbb{P} . That is, $L_{\mathbb{P}}^{cnl}$ is the smallest set such that:

- $Lit_{\mathbb{P}} \subseteq L_{\mathbb{P}}^{cnl}$
- if $a \in L_{\mathbb{P}}^{cnl}$ and $b \in L_{\mathbb{P}}^{cnl}$ then $a \wedge b \in L_{\mathbb{P}}^{cnl}$

Theorem

Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{cnl}$, $O \subseteq L_{\mathbb{P}}^{cnl} \times Lit_{\mathbb{P}}$, an agent i and two action profiles s and s' , deciding whether $s \prec_i s'$ is in P .

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- **Intermediate complexity**

5 Related work

6 Conclusion and future work

- A strict Horn clause is a non-empty disjunction of exactly one propositional atom and zero or more negated atoms.
- A strict Horn formula is a conjunction of strict Horn clauses. Let $L_{\mathbb{P}}^{Horn}$ be the set of strict Horn formulas build from \mathbb{P} .

Theorem

Given a normative multiagent system (G, N, E) where $E \subseteq L_{\mathbb{P}}^{Horn}$, $O \subseteq L_{\mathbb{P}}^{Horn} \times L_{\mathbb{P}}^{cnf}$, an agent i and two strategy profiles s and s' , deciding whether $s \prec_i s'$ is P-complete.

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

Related work

Much research has emphasized using machine-learning techniques such as neural networks [4], case-based reasoning [8], and inductive logic programming [1] to model moral reasoning.

Bringsjord *et al* [2] propose the use of deontic logic to formalize norms. Their aim is to arrive at a methodology that allows an agent to behave ethically as much as possible.

Powers [10] considers the first formulation of Kant's categorical imperative to determine "what computational structures such a view would require and to see what challenges remain for its successful implementation." Powers proposes to use nonmonotonic logic, especially default logic, to model Kant's categorical imperatives.

- Lorini [6] develops a dynamic Logic of mental attitudes and joint actions and use it to provide a logical analysis of moral agency.
- The logic Lorini introduces is PSPACE complete.
- Instead of norms, Lorini uses an ideality function which maps every possible world a real number representing the degree of the ideality to characterized the moral aspect of an agent.
- Lorini left it as future work to investigate the relationships between an agents moral values and norms.

1 Introduction

2 Background

- Boolean game
- Deontic logic
 - Input/output logic
 - Imperative logic
 - Deontic default logic
 - Deontic defeasible logic

3 Ethical agents

4 Complexity results

- Complexity of deontic norm logic
- High complexity
- Low complexity
- Intermediate complexity

5 Related work

6 Conclusion and future work

- The aim of this work is to provide a formal analysis of ethical agents.
- We adopt a deontic logic+Boolean game approach to the construction of ethical agents.
- We characterize 6 types of ethical agents: moral, amoral, social, selfish, negatively impartial and positively impartial.
- We study some complexity issues related to agents decision making. When no restriction is imposed, those decision problems are decidable but the complexity is high. Under certain restrictions we obtained intermediate and low complexity.

Thank you!



Michael Anderson, Susan Leigh Anderson, and Chris Armen.

Medethex: A prototype medical ethics advisor.

In Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA, pages 1759–1765, 2006.



Selmer Bringsjord, Konstantine Arkoudas, and Paul Bello.

Toward a general logicist methodology for engineering ethically correct robots.

IEEE Intelligent Systems, 21(4):38–44, 2006.



Jan M. Broersen, Mehdi Dastani, and Leendert W. N. van der Torre.

Beliefs, obligations, intentions, and desires as components in an agent architecture.

Int. J. Intell. Syst., 20(9):893–919, 2005.



Marcello Guarini.

Particularism and the classification and reclassification of moral cases.

IEEE Intelligent Systems, 21(4):22–28, 2006.



Jörg Hansen.

Prioritized conditional imperatives: problems and a new proposal.
Autonomous Agents and Multi-Agent Systems, 17(1):11–35, 2008.



Emiliano Lorini.

A logic for reasoning about moral agents.
Logique et Analyse, 58(230), 2015.



David Makinson and Leendert van der Torre.

Input-output logics.
Journal of Philosophical Logic, 29:383–408, 2000.




Bruce M. McLaren.


Computational models of ethical reasoning: Challenges, initial steps,
and future directions.
IEEE Intelligent Systems, 21(4):29–37, 2006.




Xavier Parent.

Moral particularism in the light of deontic logic.
Artif. Intell. Law, 19(2-3):75–98, 2011.

 Thomas M. Powers.
Prospects for a kantian machine.
IEEE Intelligent Systems, 21(4):46–51, 2006.

 Xin Sun.
How to build input/output logic.
In Nils Bulling, Leendert W. N. van der Torre, Serena Villata, Wojtek Jamroga, and Wamberto Vasconcelos, editors, *Computational Logic in Multi-Agent Systems - 15th International Workshop, CLIMA XV, Prague, Czech Republic, August 18-19, 2014. Proceedings*, volume 8624 of *Lecture Notes in Computer Science*, pages 123–137. Springer, 2014.

 Xin Sun and Diego Ambrossio.
On the complexity of input/output logic.
In the Fifth International Conference on Logic, Rationality and Interaction, 2015.