Logic and Games for Ethical Agents in Normative Multi-agent Systems

Xin Sun, Livio Robaldo[*]

Faculty of Science, Technology and Communication, University of Luxembourg
xin.sun@uni.lu, livio.robaldo@uni.lu

Abstract. In this paper we study how to characterize ethical agents in normative multi-agent systems. We adopt a proposition control game together with input/output logic. Norms create the normative status of strategies. Agents' preference in proposition control games are changed by the normative status of strategies. We distinguish four ethical types of agents: moral, amoral, negatively impartial and positively impartial. Agents of different ethical types use different input/output systems and different procedures to change their preference. Preference changes induce normative proposition control games and notions like normative Nash equilibrium are then introduced. We study some complexity issues related to normative reasoning/status and normative Nash equilibrium.

Key words: ethical type, propositional control game, norm, input/output logic

1 Introduction

Norms prominently affected agent's behavior by creating obligations and permissions. Different agents have different reactions when there are conflicts between their obligations and preference. Intuitively, it seems acceptable that:

- A *moral agent* will consider fulfilling obligations to be more important that maximizing preference.
- An *amoral agent* will act in accordance with his preference and ignore obligations.
- An *impartial agent* will first classify his actions into legal and illegal categories according to norms, then rank his actions using preference within the two categories.

Based on such intuition, our main research concern in this paper is to answer the following question: "How to formally characterize different ethical types of agents?"

This research question is understood in the setting of normative multi-agent system. Normative multi-agent system [3] is a new interdisciplinary academic area developed in recent years bringing together researchers from multi-agent system [16], deontic logic [8] and normative system [1, 11, 2]. In this paper we adopt a proposition control game

---

together with input/output logic. Proposition control game, as a variant of Boolean game [10, 4], is a class of games based on propositional logic. Input/output logic [12] appears as one of the new achievements in deontic logic in recent years [8].

Norms are social rules regulating agents' behavior by prescribing which actions are obligatory, forbidden or permitted. [15]'s early work on behavior change under norms has considered only a relatively simple view of norms, where some actions or states are designated as violations. [2] studies how conditional norms regulate agents' behaviors, but permissive norms plays no role in their framework. In this paper, agents' behavior are regulated by conditional norms including permissive norms.

In the proposition control game theoretical setting, norms classify strategies as moral, legal or illegal. Such classification transforms the game by changing the preference relation in the proposition control game. To represent norms in proposition control games, we make use of input/output logic. The preference relation in proposition control games are changed by the normative status of strategies. Agents of different ethical types use different input/output logic for normative reasoning and have different procedures of preference change. The input/output logic and the procedure of preference change characterizes different types of ethical agents.

The structure of this paper is the following: we present some background knowledge on proposition control game and input/output logic in Section 2. Normative status and ethical agents are introduced and its complexity issues are studied in Section 3 and 4. We summarize and conclude this paper with future work in Section 5.

2 Proposition control game and input/output logic

2.1 Proposition control game

Proposition control game is a variant of Boolean game. Boolean game is super succinct in the sense that agents' strategy and utility function are represented implicitly. Such succinctness is reached with a cost: many decision problems in Boolean games are intractable. For example deciding whether there is a pure strategy Nash equilibrium in a given Boolean game is Σ_2^P hard [5]. To find a balance between succinctness and tractability, we introduce proposition control game.

In a proposition control game, the strategies available to each agent consist in assigning a truth value to each variable he can control. The goal of each agent is represented by a set of weighted formulas. Formally, let $\mathbb{P} = \{p_0, p_1, \ldots\}$ be a finite set of propositional variables and let $L_{\mathbb{P}}$ be the propositional language built from $\mathbb{P}$. $2^{\mathbb{P}}$ is the set of all valuations for $\mathbb{P}$, with the usual convention that for $V \in 2^{\mathbb{P}}$ and $p \in V$, V gives the value true to p if $p \in V$ and false otherwise. Let $X \subseteq \mathbb{P}$, 2^X is the set of X-valuations. A partial valuation (for $\mathbb{P}$) is an X-valuation for some $X \subseteq \mathbb{P}$. Partial valuations are denoted by listing all variables of X, with a " $+$ " symbol when the variable is set to be true and a " $-$ " symbol when the variable is set to be false: for instance, let $X = \{p, q, r\}$, then the X-valuation $V = \{p, r\}$ is denoted $\{+p, -q, +r\}$. If $\{\mathbb{P}_1, \ldots, \mathbb{P}_n\}$ is a partition of $\mathbb{P}$ and $V_1, \ldots, V_n$ are partial valuations, where $V_i \in 2^{\mathbb{P}_i}$, $(V_1, \ldots, V_n)$ denotes the valuation $V_1 \cup \ldots \cup V_n$.

Definition 1 (proposition control game). *A proposition control game is a tuple* $(Agent, \mathbb{P}, \pi, S_1, \ldots, S_n, Goal)$, *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 $\mathbb{P}$.
4. For each agent i, $S_i \subseteq 2^{\pi(i)}$ is his strategy set.
5. $Goal = \{Goal_1, \dots, Goal_n\}$ is a set of weighted formulas of $L_{\mathbb{P}}$. 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 representing the weight of x_j.

A strategy for agent i is a $\pi(i)-$valuation. Note that since $\{\pi(1), \dots, \pi(n)\}$ forms a partition of $\mathbb{P}$, a strategy profile $s = (s_1, \dots, s_n)$ is a valuation for $\mathbb{P}$. Agents' utilities are induced by their goals. For every agent i and every strategy profile s, $u_i(s) = \Sigma\{m_j : \langle \phi_j, m_j \rangle \in Goal_i, s \vDash \phi_j\}$. Agent's preference over strategy profiles is induced by his utility function: $s \leq_i s'$ iff $u_i(s) \leq u_i(s')$. Let $s = (s_1, \dots, s_n)$ be a strategy profile, we use s_{-i} to denote the projection of s on $Agent - \{i\}$: $s_{-i} = (s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_n)$ and s_i to denote the projection of s on i's strategy.

In a proposition control game, an agent's strategy set is a **subset** of the power set of the propositional variables he can control. This is why proposition control game are computational easier than Boolean game. For the sake of tractability, we sacrifice the super-succinctness of Boolean game and use proposition control game instead.

Example 1 Let $G = (Agent, \mathbb{P}, \pi, S_1, S_2, Goal)$ where $Agent = \{1, 2\}$, $\mathbb{P} = \{p, q, r, s\}$, $\pi(1) = \{p, r\}$, $\pi(2) = \{q, s\}$, $S_1 = \{\{p, r\}, \{p\}, \{r\}\}$, $S_2 = \{\{q, s\}, \{q\}, \{s\}\}$, $Goal_1 = \{\langle p \leftrightarrow q, 1 \rangle, \langle s, 2 \rangle\}$, $Goal_2 = \{\langle p \wedge q, 2 \rangle, \langle \neg s, 1 \rangle\}$. This is depicted as:

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

2.2 Input/output logic

In I/O logic, a norm is an ordered pair of formulas $(a, x) \in L_{\mathbb{P}} \times L_{\mathbb{P}}$. Two types of norms are used in I/O logic, obligatory norms and permissive norms. An obligatory norm $(a, x) \in O$ is read as "given a, x is obligatory". A permissive norm $(a, x) \in P$ is read as "given a, x is permitted". We further assume obligatory norms are attached with a priority relation $\geq$, which is reflexive, transitive and total. $(a, x) \geq (a', x')$ is understood as (a, x) has higher priority than (a', x'). We further extend the priority relation to permissive norms: every permissive norm has the same priority and it is strictly lower than any obligatory norm. We call $N = (O, P, \geq)$ a normative system.

Obligatory norms O can be viewed as a function from $2^{L_{\mathbb{P}}}$ to $2^{L_{\mathbb{P}}}$ such that for a set A of formulas, $O(A) = \{x \in L_{\mathbb{P}} : (a, x) \in O \text{ for some } a \in A\}$. [12] define the semantics of I/O logic from out_1 to out_4 for obligatory norms as follows:

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

Cn is the classical consequence operator of propositional logic, and a set of formulas is complete if it is either maximal consistent or equal to $L_{\mathbb{P}}$. I/O logic is given a proof theoretic characterization. An ordered pair of formulas is derivable from a set O iff (a, x) is in the least set that extends $O \cup \{(\top, \top)\}$ and is closed under a number of derivation rules. The following are the rules used by [12] to define out_1 to out_4:

- 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 the 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).

The derivation system based on the rules SI, WO and AND is called $deriv_1$. Adding OR to $deriv_1$ gives $deriv_2$. Adding CT to $deriv_1$ gives $deriv_3$. The five rules together give $deriv_4$. In [12], $x \in out_i(O, a)$ iff $(a, x) \in deriv_i(O)$, for $i \in \{1, 2, 3, 4\}$ is proven.

[14] introduces a formation of prioritized I/O logic. In [14]'s, the priority relation over norms is lifted to priority over sets of norms. [14] uses the lifting originally introduced by [6]: $O_1 \succeq O_2$ 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 $(a_1, x_1) \geq (a_2, x_2)$. Let $N = (O, P, \geq)$ be a normative system and A, C be two sets of formulas. [14] define prioritized I/O logic as follows: for $i \in \{1, 2, 3, 4\}$,

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

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

1. $maxfamily_i(O, A, C) = \{O' \subseteq O : out_i(O', A) \cup C$ is consistent, and $out_i(O'', A) \cup C$ is not consistent, for every $O' \subsetneq O''\}$.
2. $filterfamily_i(O^{\geq}, A, C)$ is the set of norms $O' \in maxfamily_i(O, A, C)$ that maximize the output, i.e., that are such that $out_i(O', A) \subsetneq out_i(O'', A)$ for no $O'' \in maxfamily_i(O, A, C)$.
3. $preffamily_i(O^{\geq}, A, C)$ is the set of $\succeq$-maximal elements of $filterfamily_i(O, A, C)$.
4. $preffamily_i^d(O^{\geq}, A, C)$ is the set of elements O' of $preffamily_i(O^{\geq}, A, C)$ stripped of all the pairs (a, x) such that $out_i(O', A) = out_i(O' - \{(a, x)\}, A)$.

We simplify [14]'s prioritized I/O logic as follows:

Definition 2.

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

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

1. *$maxfamily_i(O, A, C)$ is the same as in [14]'s definition.*
2. *$preffamily_i(O^{\geq}, A, C)$ is the set of $\succeq$-maximal elements of $maxfamily_i(O^{\geq}, A, C)$.*

We drop $preffamily_i^d$ because our main concern is whether a formula x is in $out_i^p(O^{\geq}, A, C)$, $preffamily_i^d$ has no effect on whether $x \in out_i^p(O^{\geq}, A, C)$. We use the following example to illustrate why we delete $filterfamily_i$.

Example 2 *Let $O = \{(f, d), (d, a)\}$ where f means I have fever, d denotes that I go to my doctor, and a means I make an appointment with him. Let $(d, a) > (f, d)$. Put $A = \{f \wedge \neg a\}$ and $C = A$. Intuitively, I should go to an hospital and not to my doctor. Using [14]'s original definition, we have $d \in out_3^p(O^\geq, A, C)$, which prescribes me to go to my doctor without an appointment. Such behavior fulfills a lower obligation (f, d) meanwhile creates a violation of a higher obligation (d, a). Using our simplified definition, we have $d \notin out_3^p(O^\geq, A, C)$, which gives more socially acceptable prescription.*

In the setting of prioritized normative system, we choose negative and static positive permission from [13] and reformulate them as follows:

Definition 3. *Given a normative system $N = (O, P, \geq)$ and a set of input A,*

1. $NegPerm_i(N, A) = \{x \in L_\mathbb{P} : \neg x \notin out_i^p(O^\geq, A, \emptyset)\}$.
2. – *If $P \neq \emptyset$, then $StaPerm_i(N, A) = \{x \in L_\mathbb{P} : x \in out_i^p((O \cup \{(a', x')\})^\geq, A, \emptyset)$, for some $(a', x') \in P\}$.*
   – *If $P = \emptyset$, then $StaPerm_i(N, A) = out_i^p(O^\geq, A, \emptyset)$.*

We consider amoral agents as willing to commit as less obligations as possible. We choose the weakest out_1 to be the logic for amoral agents. Moral agents tends to accepts those slightly debatable rules of normative reasoning. We choose out_4 to be the logic for moral agents. Negatively/positively impartial agents classify actions according to whether they are negatively/positively permitted. Since the rule OR involves uncertainty and vagueness, out_2 seems to be not suitable for positively impartial agents. Thus, we choose out_3 for positively impartial agents and out_2 for negatively impartial agents.

3 Normative status

We use a proposition control game to represent a multi-agent system.

Definition 4 (normative multi-agent system). *A normative multiagent system is a tuple (G, N, E, ρ) where*

– $G = (Agent, \mathbb{P}, \pi, S_1, \ldots, S_n, Goal)$ *is a proposition control game.*
– $N = (O, P, \geq)$ *is a finite normative system.*
– $E \subseteq L_\mathbb{P}$ *is the environment, which is a finite set of formulas representing facts.*
– $\rho : Agent \mapsto \{1, 2, 3, 4\}$ *is an agent type assignment function which assigns each agent a unique ethical type.*

Strategies are classified as moral, positively legal, negatively legal or illegal.

Definition 5 (moral, legal and illegal strategy). *Given a normative multi-agent system (G, N, E, ρ), a strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is:*

– *moral: if $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in out_k^p(O^\geq, E, \emptyset)$.*
– *positively legal: if $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in StaPerm_k(N, E)$.*
– *negatively legal: if $p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n \in NegPerm_k(N, E)$.*
– *illegal: if $\neg(p_1 \wedge \ldots \wedge p_m \wedge \neg q_1 \wedge \ldots \wedge \neg q_n) \in out_k^p(O^\geq, E, \emptyset)$.*

Example 3 *Let (G, N, E, ρ) be a normative multi-agent system as follows:*

- *$G = (Agent, \mathbb{P}, S_1, S_2, \pi, Goal)$ is a proposition control game with*
  - *$Agent = \{1, 2\}$, $\mathbb{P} = \{p, q\}$,*
  - *$\pi(1) = \{p\}$, $\pi(2) = \{q\}$, $S_1 = \{\{p\}, \emptyset\}$, $S_2 = \{\{q\}, \emptyset\}$,*
  - *$Goal_1 = \{\langle p \wedge q, 1\rangle\}$, $Goal_2 = \{\langle p \vee q, 1\rangle\}$,*
- *$N = (O, P, \geq)$ where $O = \{(\top, p)\}$, $P = \{(\top, q)\}$, $\geq = \emptyset$.*
- *$E = \emptyset$, and both 1 and 2 are type-1 agents.*

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

$out_1(O, E) = Cn(\{p\}) = out_1^p(O^{\geq}, E, \emptyset)$, $StaPerm_1(N, E) = Cn(\{p, q\})$. *The norma-tive status of $+p, +q, -q, -p$ is respectively moral, positively/negatively legal, illegal.*

Having defined the normative status of strategies, we now study the complexity of some decision problems related to normative reasoning and normative status.

Theorem 1. *Given a normative multi-agent system (G, N, E, ρ), deciding whether a type-k agent strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, is moral is Π_2^p complete.*

Corollary 1. *Given a normative multi-agent system (G, N, E, ρ), a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$,*

1. *deciding whether this strategy is illegal is Π_2^p complete.*
2. *deciding whether this strategy is negatively legal is Σ_2^p complete.*

Theorem 2. *Given a normative multi-agent system (G, N, E, ρ), a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ deciding whether this strategy is positively legal is Π_2^p complete.*

Corollary 2. *Given a normative multi-agent system (G, N, E, ρ), a type-k agent and his strategy $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$, deciding the normative status of $(+p_1, \ldots, +p_m, -q_1, \ldots, -q_n)$ is Σ_2^p hard and in $\Delta_3^p = P^{\Sigma_2^p}$.*

4 Ethical agents

Different types of agents change their preference in different ways. Informally:

1. Amoral agents prefer strategy profiles with higher utility; for two profiles of the same utility, the one containing the strategy of higher normative status is preferred.
2. Moral agents prefer strategy profiles containing the strategy of higher normative status; for two profiles of the same status, the with higher utility is preferred.
3. Negatively impartial agents classify strategies into negatively legal category and illegal category; then they rank the strategies using utility within the two categories.
4. Positively impartial agents classify strategies into positively and not-positevely legal category; then they rank the strategies using utility within the two categories.

Given a normative multi-agent system, it induces a normative proposition control game by changing the preference of agents.

Definition 6 (normative proposition control game). *Given a normative multi-agent system (G, N, E, ρ) where $G = (Agent, \mathbb{P}, \pi, S_1, \ldots, S_n, Goal)$, it induces a normative proposition control game $G^N = (Agent, \mathbb{P}, \pi, S_1, \ldots, S_n, \prec_1, \ldots \prec_n)$ where $\prec_i$ is the preference of i over strategy profiles such that*

1. *if i is type-1 (amoral), 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.*
2. *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 $u_i(s) < u_i(s')$.*
3. *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 $u_i(s) < u_i(s')$.*
4. *if i is type-4 (moral), 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 $u_i(s) < u_i(s')$.*

Theorem 3. *Given a normative multi-agent system (G, N, E, ρ), an agent i and two strategy profiles s and s', deciding whether $s \prec_i s'$ is in Δ_3^p.*

Definition 7 (normative Nash equilibrium). *Given a normative multi-agent system (G, N, E, ρ), a strategy profile s is a normative Nash equilibrium if it is a Nash equilibrium in the normative proposition control game G^N.*

Theorem 4. *Given a normative multi-agent system (G, N, E, ρ) and a strategy profile s, deciding whether s is normative Nash equilibrium is in Δ_3^p.*

5 Conclusion and future work

In this paper we adopt a proposition control game and I/O logic approach to normative multi-agent systems. We distinguish four ethical types of agents, which use different I/O logic for normative reasoning and different procedures to change their preference. Such preference change create normative proposition control games and notions like normative Nash equilibrium are then introduced. We study some complexity issues related to normative reasoning/status and normative Nash equilibrium.

The contribution of this paper is twofold: on the conceptual side, we give a formal characterization of four ethical types of agents. On the technical side, we present some complexity results of normative reasoning with respect to prioritized I/O logic. All the complexity results in this paper are intractable, we leave it as future work to find tractable fragments. We conjecture that if we restrict every formula that appears in I/O logic to be a conjunction of literals, then all decision problems studied in this paper is tractable. Such restricted prioritized I/O logic has similar expressive power to the logic of abstract normative systems [7], as well as defeasible deontic logic [9]. A detailed comparison between these logic is also left as future work.

Bibliography

[1] T. Ågotnes, W. van der Hoek, J.A. Rodríguez-Aguilar, C. Sierra, and M. Wooldridge. On the logic of normative systems. In *Proc. of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India*, 2007.

[2] N. Alechina, M. Dastani, and B. Logan. Reasoning about normative update. In *Proc. of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, 2013*, 2013.

[3] G. Boella, L. van der Torre, and H. Verhagen. Introduction to the special issue on normative multiagent systems. *Autonomous Agents and Multi-Agent Systems*, 17(1), 2008.

[4] E. Bonzon, M. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Compact preference representation and boolean games. *Autonomous Agents and Multi-Agent Systems*, 18(1), 2009.

[5] L. Bonzon, M. Lagasquie-Schiex, J. Lang, and B. Zanuttini. Boolean games revisited. In *Proc. of 17th European Conference on Artificial Intelligence, Riva del Garda, Italy*, 2006.

[6] S. Brass. *Nonmonotonic and Inductive Logic: Second International Workshop Reinhardsbrunn Castle, Germany December 2–6, 1991 Proceedings*, chapter Deduction with supernormal defaults, pages 153–174. Springer Berlin Heidelberg, Berlin, Heidelberg, 1993.

[7] S. Colombo Tosatto, G. Boella, L. van der Torre, and S. Villata. Abstract normative systems: Semantics and proof theory. In *Principles of Knowledge Representation and Reasoning*, pages 358–368, 2012.

[8] 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, 2014.

[9] G. Governatori, F. Olivieri, A. Rotolo, and S. Scannapieco. Computing strong and weak permissions in defeasible logic. *J. Philosophical Logic*, 42(6), 2013.

[10] P'. Harrenstein, W. van der Hoek, J. Meyer, and C. Witteveen. Boolean games. In *Proc. of the 8th Conference on Theoretical Aspects of Rationality and Knowledge*. Morgan Kaufmann Publishers Inc., 2001.

[11] A. Herzig, E. Lorini, F. Moisan, and N. Troquard. A dynamic logic of normative systems. In *Proc. of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Spain*.

[12] D. Makinson and L. van der Torre. Input-output logics. *Journal of Philosophical Logic*, 29:383–408, 2000.

[13] D. Makinson and L. van der Torre. Permission from an input/output perspective. *Journal of Philosophical Logic*, 32:391–416, 2003.

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

[15] Yoav Shoham and Moshe Tennenholtz. On social laws for artificial agent societies: Off-line design. *Artif. Intell.*, 73(1-2):231–252, 1996.

[16] M. Wooldridge. *An Introduction to MultiAgent Systems (2. ed.)*. Wiley, 2009.