Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Applied Logic 3 (2005) 396–420
JOURNAL OF APPLIED LOGIC
www.elsevier.com/locate/jal

On obligations and normative ability: Towards a logical analysis of the social contract ☆

Michael Wooldridge *, Wiebe van der Hoek

Department of Computer Science, University of Liverpool, Liverpool L69 7ZF, UK

Available online 23 May 2005

Editors: A. Lomuscio and D. Nute

Abstract

We develop a logic of *normative ability*, as an extension to the Alternating-time Temporal Logic (ATL) of Alur, Henzinger, and Kupferman. While conventional ATL contains cooperation modalities of the form $\langle\langle C\rangle\rangle\varphi$, intended to express the fact that coalition C have the capability to bring about φ, in Normative ATL* (NATL*), these expressions are replaced with constructs of the form $\langle\langle \eta : C\rangle\rangle\varphi$, with the intended interpretation that C have the ability to achieve φ *within the context of the normative system* η. A normative system is a set of constraints on the actions that may be performed in any give state. We show how these normative ability constructs can be used to define obligations and permissions: φ is said to be obligatory within the context of the normative system η if φ is a necessary consequence of every agent in the system behaving according to the conventions of η. After introducing NATL*, we investigate some of its axiomatic properties. To demonstrate its value as a logic for reasoning about multi-agent systems, we show how NATL* can be used to formalise a version of the *social contract*.

Keywords: Obligation; Normative ability; Agency; Social contract; ATL

☆ We gratefully acknowledge the support of the ESPRC under research grant GR/S62727/01 ("Virtual Organisations for e-Science").
* Corresponding author.
E-mail addresses: mjw@csc.liv.ac.uk (M. Wooldridge), wiebe@csc.liv.ac.uk (W. van der Hoek).

1570-8683/$ – see front matter
doi:10.1016/j.jal.2005.04.006

1. Introduction

Over the past decade, there has been steadily increasing interest in the logical foundations of multi-agent systems [17,37–39]. While the literature on this subject encompasses a wide range of ideas and logical techniques, the *Alternating-time Temporal Logic* (ATL) proposed by Alur, Henzinger, and Kupferman appears to be gaining popularity as a key system in the area [1]. ATL is a logic of *cooperative ability*: it is intended to support reasoning about the powers of agents and coalitions of agents in game-like multi-agent systems. Thus, for example, in ATL it is possible to express properties of a system such as "the coalition C_1 can guarantee that the system will never enter an invalid state", and "the coalition C_2 can ensure that, eventually, the message will be received".

From a language point of view, ATL represents an elegant generalisation of the well-known branching time temporal logic CTL [11], while at the same time containing an explicit notion of *agency*, which gives it the flavour of an action logic, in the sense of dynamic logic and its relatives [14,29]; from a semantic point of view, ATL is based on models that combine ideas from distributed computing systems and game theory, thus reflecting current thinking about the semantics of concurrent computation [27]; and from a computational point of view, model checking and theorem proving in ATL appear to have the same complexity as their counterpart problems in CTL [1,10], and in particular, ATL has a tractable (deterministic polynomial time) model checking problem, for which efficient software tools have been implemented [2].

The fact that ATL bears a close family resemblance to logics of action has prompted several researchers to investigate this relationship in more detail. One obvious issue is the link between ATL and *deontic logic*: the logic of obligation and permission [8,24]. Our primary aim in this paper is to investigate the relationship between ability and obligations in detail. More specifically, the paper makes three main contributions to this understanding.

First, we introduce a variant of ATL called *Normative* ATL* (NATL*).[1] The logic NATL* is based on cooperation modalities of the form $\langle\langle \eta : C \rangle\rangle \varphi$, where η is a *normative system*, C is a coalition, and φ is a sentence of the logic. The intended interpretation of $\langle\langle \eta : C \rangle\rangle \varphi$ is that operating within the context of the normative system η, coalition C have the ability to bring about φ; more precisely, that C have a winning strategy for φ, where this strategy conforms to the strictures of the normative system η. A normative system in our framework is a set of rules, which constrain the actions of the agents in the system in certain states. Given these cooperation modalities, we can recover the cooperation modalities of ATL by considering ability within the context of the "empty" normative system, i.e., the normative system $\eta_\perp$, which places no constraints on the actions of agents other than those imposed by the system designer.

Second, we introduce an indexed collection of indexed unary modal operators P_η and O_η, where $\mathsf{P}_\eta \varphi$ is intended to mean that φ is permissible within the context of the normative system η, and $\mathsf{O}_\eta \varphi$ is intended to mean that φ is obligatory within the context of the normative system η. Perhaps the larger contribution we make here is to show how permission

[1] Note that our logic is closer to ATL* than ATL, and hence we feel obliged to use a "*" in the name! We will sometimes informally refer to "Normative ATL", and it should be understood that when we do this, we in fact mean NATL*.

and obligation can be given a natural and compelling interpretation in terms of normative ability. Crudely, we say that φ is obligatory in the context of normative system φ if φ will necessarily result if every agent acts according to the norm η. Similarly, φ is permissible in the context of η if there is some way that φ can be brought about by a coalition acting in accordance with the conventions of η.

Third, and finally, we show how NATL* can be used for reasoning about multi-agent systems, by developing a logical model of the *social contract*. Crudely, the term "social contract" refers to the collection of norms or conventions that a society abides by. These norms serve to regulate and restrict the behaviour of citizens within a society. The benefit of a social contract is that it prevents mutually destructive behaviours. However, there are many apparent paradoxes associated with the social contract, not the least being that of why a rational, self-interested agent should choose to conform to the social contract, when choosing to do otherwise might lead to a better individual outcome; the problem being that if everyone reasons this way (and as rational agents, they should), then nobody conforms to the social contract, and its benefits are lost. There have been several game theoretic accounts of the social contract, which attempt to understand how a social contract can work in a society of self-interested agents [6,7,33]; our work can be understood as a preliminary, tentative attempt to give a logical account. Note that our focus on the social contract is not prompted by a desire to shed light on issues of political or economic philosophy, but by a desire to better understand how to engineer societies of self-interested autonomous software agents [38]; we, along with other researchers [9], believe that the concept of the social contract is potentially a useful one for understanding and engineering such artificial societies.

This article is structured as follows. We shortly introduce *Action-based Alternating Transition Systems* (AATSs), the structures used to give a semantics to NATL*. In Section 3, we introduce our model of normative systems; we discuss the operations that may be performed on them, as well as the possible relationships that exist between them. In Section 4, we introduce the logic of Normative ATL itself, and briefly discuss some of its properties. We define our deontic modalities for NATL* in Section 5, and briefly consider some of their properties. In Section 6, we show how NATL* may be applied to an understanding of the social contract, and we conclude in Sections 7 and 8 with a discussion of related work and some conclusions.

2. Action-based alternating transition systems

The semantic structures underpinning ATL are known as *Action-based Alternating Transition Systems* (AATSs) [34]. We need to be clear about the role that these structures are intended to play. AATSs are structures for modelling game-like, dynamic, multi-agent systems. The main characteristics of such systems are that there are multiple agents, each of which can perform actions in order to modify and attempt to control the system in some way. Our intention in this paper is that an AATS should be used to model the *physical* properties of the system at hand—the actions that agents can perform in the empty normative system, unfettered by any considerations of their legality or usefulness. However, many of the systems of interest to us are not "physical world" systems in the obvious sense of

the term, but consist of agents in virtual/software environments, and so we prefer the more neutral term "natural". We also note in passing that, inevitably, when we define an AATS, this system will represent an *abstraction* of the "actual" system that we intend to model, and interpreting an AATS as representing the *physical* characteristics may therefore not be entirely appropriate.

We first assume that the systems of interest to us may be in any of a finite set Q of possible *states*, with some $q_0 \in Q$ designated as the *initial state*. Systems are populated by a set Ag of *agents*; a *coalition* of agents is simply a set $C \subseteq Ag$, and the set of all agents is known as the *grand coalition*. Notice that this is *all* we mean by the term "coalition" in this paper: our usage here does not imply any common purpose or shared goal—a coalition in this paper is simply a set of agents.

Each agent $i \in Ag$ is associated with a set Ac_i of possible actions, and we assume that these sets of actions are pairwise disjoint (i.e., actions are unique to agents). We denote the set of actions associated with a coalition $C \subseteq Ag$ by Ac_C, so $Ac_C = \bigcup_{i \in C} Ac_i$.

A *joint action* j_C for a coalition C is a tuple $\langle \alpha_1, \ldots, \alpha_k \rangle$, where for each α_j (where $j \leqslant k$) there is some $i \in C$ such that $\alpha_j \in Ac_i$. Moreover, there are no two different actions α_j and $\alpha_{j'}$ in J_C that belong to the same Ac_i. We denote the set of all joint actions for coalition C by J_C, so $J_C = \prod_{i \in C} Ac_i$. Given an element j of J_C and agent $i \in C$, we denote i's component of j by j_i.

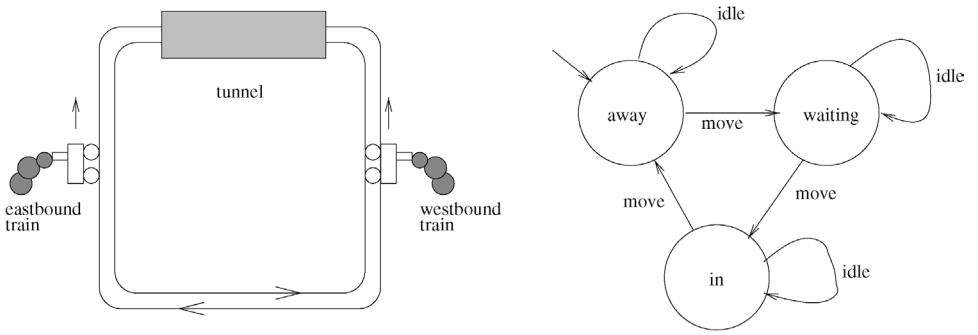
An *Action-based Alternating Transition System*—hereafter referred to simply as an AATS—is an $(n + 7)$-tuple $S = \langle Q, q_0, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Phi, \pi \rangle$, where:

- Q is a finite, non-empty set of *states*;
- $q_0 \in Q$ is the *initial state*;
- $Ag = \{1, \ldots, n\}$ is a finite, non-empty set of *agents*;
- Ac_i is a finite, non-empty set of *actions*, for each $i \in Ag$, where $Ac_i \cap Ac_j = \emptyset$ for all $i \neq j \in Ag$;
- $\rho : Ac_{Ag} \to 2^Q$ is an *action precondition function*, which for each action $\alpha \in Ac_{Ag}$ defines the set of states $\rho(\alpha)$ from which α may be executed;
- $\tau : Q \times J_{Ag} \to Q$ is a partial *system transition function*, which defines the state $\tau(q, j)$ that would result by the performance of j from state q—note that, as this function is partial, not all joint actions are possible in all states (cf. the pre-condition function above);
- Φ is a finite, non-empty set of *atomic propositions*; and
- $\pi : Q \to 2^\Phi$ is an interpretation function, which gives the set of primitive propositions satisfied in each state: if $p \in \pi(q)$, then this means that the propositional variable p is satisfied (equivalently, true) in state q.

We require that AATSs satisfy the following two coherence constraints:

(1) *Non-triviality* [26]. Agents always have at least one action available:

$$\forall q \in Q, \forall i \in Ag, \exists \alpha \in Ac_i \text{ s.t. } q \in \rho(\alpha)$$



(a) Overall structure of the train sytem (b) Train states and transitions

Fig. 1. The train system S_1.

(2) *Consistency*. The ρ and τ functions agree on actions that may be performed:

$$\forall q, \forall j \in J_{Ag}, (q, j) \in \operatorname{dom} \tau \quad \text{iff} \quad \forall i \in Ag, q \in \rho(j_i)$$

We denote the set of sequences over Q by Q^*, and the set of non-empty sequences over Q by Q^+.

Example 1. There are two trains, one of which (E) is Eastbound, the other of which (W) is Westbound, each occupying their own circular track. At one point, both tracks pass through a narrow tunnel—a crash will occur if both trains are in the tunnel at the same time. Unlike the original versions of this scenario [1], we do *not* assume that there is a "controller" agent, whose purpose is to ensure that collisions do not occur. Instead, we will be concerned with social laws that achieve this end.

We model each train $i \in Ag = \{E, W\}$ as an automaton that can be in one of three states (see Fig. 1(b)): "$away_i$" (the initial state of the train); "$waiting_i$" (waiting to enter the tunnel); and "in_i" (the train is in the tunnel). Each train $i \in \{E, W\}$ has two actions available: $Ac_i = \{move_i, idle_i\}$. The $idle_i$ action is the identity, which causes no change in the train's state (i.e., it stays where it is). If a train i executes a $move_i$ action while it is $away_i$, then it goes to a $waiting_i$ state; executing a $move_i$ while $waiting_i$ causes a transition to an in_i state; and finally, executing a $move_i$ while in_i causes a transition to $away_i$ as long as the other train was not in the tunnel, while if both trains are in the tunnel, then they have crashed, and are forced to *idle* indefinitely. Initially, both trains are *away*.

The overall state of the system at any given time can be characterised in terms of the propositional variables $\{away_E, away_W, waiting_E, waiting_W, in_E, in_W\}$, where these variables have the obvious interpretation. The overall structure of the train system, and the model of trains is illustrated in Fig. 1; a formal definition of the train system AATS is given in Fig. 2 (the function ρ is left implicit, but can be read off from τ: e.g., $\rho(move_W) = Q \setminus \{q8\}$, etc.).

Of course, not all combinations of propositional variables correspond to reachable system states (i.e., states that the system could possibly enter). For example, an agent i cannot

States and Initial States:

$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8\}$, initial state q_0

Agents, Actions, and Joint Actions:

$Ag = \{E, W\}$ $\quad Ac_E = \{idle_E, move_E\}$ $\quad Ac_W = \{idle_W, move_W\}$

$J_{Ag} = \{\underbrace{\langle idle_E, idle_W\rangle}_{j_0}, \underbrace{\langle idle_E, move_W\rangle}_{j_1}, \underbrace{\langle move_E, idle_W\rangle}_{j_2}, \underbrace{\langle move_E, move_W\rangle}_{j_3}\}$

Propositional Variables:

$\Phi = \{away_E, away_W, waiting_E, waiting_W, in_E, in_W\}$

Transitions/Pre-conditions/Interpretation:

$q \backslash j$	j_0	j_1	j_2	j_3	$\pi(q)$
q_0	q_0	q_1	q_3	q_5	$\{away_E, away_W\}$
q_1	q_1	q_2	q_5	q_6	$\{away_E, waiting_W\}$
q_2	q_2	q_0	q_6	q_3	$\{away_E, in_W\}$
q_3	q_3	q_5	q_4	q_7	$\{waiting_E, away_W\}$
q_4	q_4	q_7	q_0	q_1	$\{in_E, away_W\}$
q_5	q_5	q_6	q_7	q_8	$\{waiting_E, waiting_W\}$
q_6	q_6	q_3	q_8	q_4	$\{waiting_E, in_W\}$
q_7	q_7	q_8	q_1	q_2	$\{in_E, waiting_W\}$
q_8	q_8	—	—	—	$\{in_E, in_W\}$

Fig. 2. The AATS for the trains scenario.

be both $waiting_i$ and in_i simultaneously. There are in fact just nine reachable states of the system; see Fig. 2.

2.1. Strategies

Given an agent $i \in Ag$ and a state $q \in Q$, we denote the *options* available to i in q—the actions that i may perform in q—by $options(i, q)$:

$$options(i, q) = \{\alpha \mid \alpha \in Ac_i \text{ and } q \in \rho(\alpha)\}$$

We then say that a *strategy* for an agent $i \in Ag$ is a function:

$$\sigma_i : Q \to Ac_i$$

which must satisfy the *legality* constraint that $\sigma_i(q) \in options(i, q)$ for all $q \in Q$.

A *strategy profile* for a coalition $C = \{a_1, \ldots, a_k\} \subseteq Ag$ is a tuple of strategies $\langle \sigma_1, \ldots, \sigma_k\rangle$, one for each agent $a_i \in C$. We denote by Σ_C the set of all strategy profiles for coalition $C \subseteq Ag$; if $\sigma_C \in \Sigma_C$ and $i \in C$, then we denote i's component of σ_C by σ_C^i. Given a strategy profile $\sigma_C \in \Sigma_C$ and state $q \in Q$, let $out(\sigma_C, q)$ denote the set of possible states that may result by the members of the coalition C acting as defined by their components of σ_C for one step from q:

$$out(\sigma_C, q) = \{q' \mid \tau(q, j) = q' \text{ where } (q, j) \in \operatorname{dom}\tau \text{ and } \sigma_C^i(q) = j_i \text{ for } i \in C\}$$

Notice that, for any grand coalition strategy profile σ_{Ag} and state q, the set $out(\sigma_{Ag}, q)$ will be a singleton.

2.2. Computations

A *computation* is an infinite sequence of states $\lambda = q_0, q_1, \ldots$. A computation $\lambda \in Q^+$ starting in state q is referred to as a *q-computation*; if $u \in \mathbb{N}$, then we denote by $\lambda[u]$ the component indexed by u in λ (thus $\lambda[0]$ denotes the first element, $\lambda[1]$ the second, and so on). We denote by $\lambda[0, u]$ and $\lambda[u, \infty]$ the finite prefix $q_0, \ldots, q_u$ and the infinite suffix $q_u, q_{u+1}, \ldots$ of λ respectively.

Given a strategy profile σ_C for some coalition C, and a state $q \in Q$, we define $comp(\sigma_C, q)$ to be the set of possible runs that may occur if every agent $a_i \in C$ follows the corresponding strategy σ_i, starting when the system is in state $q \in Q$. That is, the set $comp(\sigma_C, q)$ will contain all possible q-computations that the coalition C can "enforce" by cooperating and following the strategies in σ_C.

$$comp(\sigma_C, q) = \{\lambda \mid \lambda[0] = q \text{ and } \forall u \in \mathbb{N}\colon\ \lambda[u+1] \in out(\sigma_C, \lambda[u])\}$$

Again, note that for any state $q \in Q$ and any grand coalition strategy σ_{Ag}, the set $comp(\sigma_{Ag}, q)$ will be a singleton, consisting of exactly one infinite computation.

3. Normative systems

In this section, we introduce our model of normative systems, and briefly investigate some of its properties. When we use the term "normative system" in this paper, it has a *technical* meaning: we are much less concerned with the philosophical issues surrounding normative systems and their role in human societies—although this will, of course, not prevent us from borrowing ideas and terminology from the literature of norms, conventions, and normative systems [20]. For us, a normative system is simply a set of constraints on the behaviour of agents in a system. More precisely, a normative system defines, for every possible system state and action, whether or not that action is considered to be legal or not, in the context of the normative system. Different normative systems, of course, may differ on whether or not a particular action is considered legal in a particular state.

We model a normative system, η, as a function

$$\eta : Ac_{Ag} \to 2^Q$$

with the intended interpretation that $q \in \eta(\alpha)$ means the normative system η forbids action α from being performed when the system is in state q.

Of course, our normative systems cannot be considered in a vacuum. They are designed (or emerge [35], though we shall not consider this issue here), in the context of an AATS, and AATSs have their own notion of legality: whether or not an action is *naturally* possible, that is, whether or not it is "physically possible" in the context of the system. It makes no sense, (in our framework at least), to consider normative systems that permit actions that are naturally impossible to perform, and so we will place one requirement on normative systems: that they forbid anything that is forbidden by "nature". Formally, the requirement is that:

$$\forall \alpha \in Ac_{Ag}\colon\ \big(Q \setminus \rho(\alpha)\big) \subseteq \eta(\alpha)$$

As an aside, notice the duality between the pre-condition function ρ, and normative system η: if $q \in \rho(\alpha)$, then α is naturally possible in q, whereas if $q \in \eta(\alpha)$, then α is forbidden in q by the normative system η. We denote the set of all normative systems, with respect to some implicit AATS, by N.

We say a strategy $\sigma_i \in \Sigma_i$ is η-conformant if it never selects an action that is forbidden by η. We denote the fact that σ_i conforms to η by $conf(\sigma_i, \eta)$.

$$conf(\sigma_i, \eta) \quad \Leftrightarrow \quad \forall q\colon q \notin \eta\big(\sigma_i(q)\big)$$

Given a strategy profile $\sigma_C \in \Sigma_C$, we will abuse notation and write $conf(\sigma_C, \eta)$ to indicate that all the strategies in σ_C conform to η.

$$conf(\sigma_C, \eta) \quad \Leftrightarrow \quad \forall i \in C\colon conf(\sigma_C^i, \eta)$$

Finally, we denote the set of all η-conformant strategy profiles for C by Σ_C^η.

$$\Sigma_C^\eta \mathrel{\hat{=}} \big\{\sigma_C \in \Sigma_C \mid conf(\sigma_C, \eta)\big\}$$

Example 2. Recall the trains example, given earlier. We will define a normative system η_1, the primary purpose of which is to ensure that the trains never crash, i.e., that the system never enters state q_8. From examination of the state transition function τ (see Fig. 2), we can see that $\tau(q_5, j_3) = \tau(q_6, j_2) = \tau(q_7, j_1) = q_8$, and there are no other transitions leading to q_8 (apart from when the trains have already crashed, which we need not consider!). So, consider the normative system η_1, as follows.

$$\eta_1(\alpha) = \begin{cases} \emptyset & \text{if } \alpha = idle_E \\ \emptyset & \text{if } \alpha = idle_W \\ \{q_5, q_6\} & \text{if } \alpha = move_E \\ \{q_7\} & \text{if } \alpha = move_W \end{cases}$$

This normative system ensures that:

- when both agents are waiting to enter the tunnel, the eastbound train is forbidden to move;
- when the westbound train is already in the tunnel and the eastbound train is waiting to enter the tunnel, then the eastbound train is not allowed to move;
- when the eastbound train is already in the tunnel and the westbound train is waiting to enter the tunnel, then the westbound train is forbidden to move.

Notice that η_1 is, in a sense, asymmetric, as it constrains the eastbound train rather than the westbound train: we could equally well replace the first constraint with the requirement that if both trains are waiting to enter the tunnel, then the *westbound* train is prevented from moving, thus enabling the eastbound train to enter.

3.1. Operations on normative systems

We find it convenient to distinguish two particular normative systems. We denote the *empty* normative system by $\eta_\perp$. This system imposes *no* constraints on the actions that

agents may perform other than those imposed by the underlying AATS: any action that is physically/naturally feasible is legal according to $\eta_\perp$.

$$\forall \alpha \in Ac_{Ag}\colon\ \eta_\perp(\alpha) = Q \setminus \rho(\alpha)$$

In the *trivial* normative system $\eta_\top$, every action is forbidden in every state.

$$\forall \alpha \in Ac_{Ag}\colon\ \eta_\top(\alpha) = Q$$

We can develop a kind of calculus of normative systems based on the standard set theoretic operations of intersection and union, as follows.

$$\eta \sqcap \eta'(\alpha) \mathrel{\hat{=}} \eta(\alpha) \cap \eta(\alpha')$$
$$\eta \sqcup \eta'(\alpha) \mathrel{\hat{=}} \eta(\alpha) \cup \eta(\alpha')$$

Notice that the set N of normative systems over some AATS will be closed under these operators. Having a calculus as sketched would allow one to reason about the composition of normative systems, similar to the way that one constructs complex programs from simpler ones in Dynamic Logic [14]. However, N would not be closed under difference and complement operations, which is why we do not consider these.

The laws of these operators are analogous to properties of set theory: for example, $\eta_\top$ — the least liberal normative system — serves as the identity under $\sqcap$, and $\eta_\perp$ — the most liberal normative system — serves as the identity under $\sqcup$. We will not exhaustively list these laws, but simply give the following examples as a flavour.

$$\eta \sqcup \eta = \eta \sqcup \eta_\perp = \eta \text{ and } \eta \sqcup \eta_\top = \eta_\top$$
$$\eta \sqcap \eta = \eta \sqcap \eta_\top = \eta \text{ and } \eta \sqcap \eta_\perp = \eta_\perp$$

3.2. Relationships between normative systems

Let us now consider the possible relationships between normative systems. Of the relationships that we might consider, we argue that the most obvious — and the most important — is that of when one normative system is *less restrictive* than another. Let us introduce a binary relation $\preceq \subseteq N \times N$ on normative systems, with the intended interpretation that $\eta \preceq \eta'$ means that η is less restrictive (equivalently, more liberal) than η'. (To be precise, $\eta \preceq \eta'$ will mean that η is "at most as restrictive as" η', but where no confusion is possible, we will ignore this distinction in the text.) Formally, we define the relation "$\preceq$" as follows.

$$\eta \preceq \eta' \quad \Leftrightarrow \quad \forall \alpha \in Ac_{Ag}\colon\ \eta(\alpha) \subseteq \eta'(\alpha)$$

Example 3. With respect to the trains system S_1, and the normative systems η_1 and η_2 (where η_2 is defined later), we have $\eta_1 \preceq \eta_2$. In other words, η_1 is more liberal than η_2.

The $\preceq$ relation defines a partial order over N: it is reflexive, transitive, and antisymmetric. Moreover, observe that for any normative system $\eta \in N$, we have $\eta_\perp \preceq \eta$ and $\eta \preceq \eta_\top$. Recalling the definitions of $\sqcap$ and $\sqcup$ from above, we immediately obtain the following.

Proposition 1. *Let N be the set of normative systems over some* AATS*, and let $\preceq \subseteq N \times N$ be the associated "less restrictive" relation. Then the pair $(N, \preceq)$ forms a complete lattice, with least upper bound $\eta_\top$ and greatest lower bound $\eta_\bot$; the meet operation is $\sqcap$ and the join operation is $\sqcup$.*

Combining the $\preceq$ relation with the meet and join operations on normative systems, we get the following properties.

$$\eta \sqcap \eta' \preceq \eta \quad \eta \sqcap \eta' \preceq \eta' \quad \eta \preceq \eta \sqcup \eta' \quad \text{and} \quad \eta' \preceq \eta \sqcup \eta' \tag{1}$$

Thus taking the *union* of two normative systems yields a normative system that is *more restrictive* (less liberal) than either of its parent systems, while taking the *intersection* of two normative systems yields a normative system which is *less restrictive* (more liberal). Notice that the $\sqcup$ operation is intuitively the act of superposition, or composition of normative systems: imposing one law on top of another. The $\sqcup$ operation thus gives us (the beginnings of) a calculus through which to understand the composition of normative systems.

The $\preceq$ relation can also be characterised in terms of the strategies available to agents. Let us call a normative system η to be *non-trivial* if it allows, everywhere, the grand coalition Ag to perform an action: η is non-trivial iff

$$\forall q \in Q \exists j \in J_{Ag}\, \forall i \in Ag: \quad q \notin \eta(j_i)$$

Proposition 2. *Let $S = \langle Q, q_0, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Phi, \pi \rangle$ be an* AATS*, and let η and η' be non-trivial normative systems over S. Then*:

$$\eta \preceq \eta' \quad \Leftrightarrow \quad \forall C \subseteq Ag:\ \Sigma_C^{\eta'} \subseteq \Sigma_C^{\eta}$$

Proof. The $\Rightarrow$ is obvious: If η is less restrictive than η', then any η'-conformant strategy profile for C must also be η-conformant. For $\Leftarrow$, assume for purposes of contradiction that $\forall C \subseteq Ag:\ \Sigma_C^{\eta'} \subseteq \Sigma_C^{\eta}$ but $\eta \not\preceq \eta'$. Since $\eta \not\preceq \eta'$, then for some agent $i \in Ag$ and $\alpha \in Ac_i$, we have $\eta(\alpha) \not\subseteq \eta'(\alpha)$. Hence for some $q' \in Q$, we have:

$$q' \in \eta(\alpha) \quad \text{and} \quad q' \notin \eta'(\alpha)$$

Now, take any σ_i from $\Sigma_i^{\eta'}$ (since η' is non-trivial, we know that such a σ_i exists), and define a new strategy σ_i^* for i as follows:

$$\sigma_i^*(q) = \begin{cases} \alpha & \text{if } q = q' \\ \sigma_i(q) & \text{otherwise} \end{cases}$$

Now by construction, $\sigma_i^* \in \Sigma_i^{\eta'}$, but $\sigma_i^* \notin \Sigma_i^{\eta}$, and so $\Sigma_i^{\eta'} \not\subseteq \Sigma_i^{\eta}$: a contradiction. □

Although the $\Rightarrow$-direction of Proposition 2 holds for arbitrary systems η and η', the other direction does not. Suppose that η' is such that it forbids every action for every agent i in a particular state q. Then, for any C, no strategy σ_C exists that is η'-conformant, so $\Sigma_C^{\eta'} = \emptyset$. Would the $\Rightarrow$-direction of Proposition 2 be true, then we would have $\eta \preceq \eta'$ for any η, which obviously need not be the case.

4. Normative ATL

Alternating-time Temporal Logic (ATL), can be understood as a generalisation of the well-known branching time temporal logic CTL [11], in which path quantifiers are replaced by *cooperation modalities*. In NATL*, we contextualise cooperation modalities with normative systems. Thus cooperation modalities in NATL* have the general form $\langle\langle \eta : C \rangle\rangle \varphi$, which is intended to be read "the coalition C can achieve φ, even when it abides by the rules of normative system η". More precisely, $\langle\langle \eta : C \rangle\rangle \varphi$ means that there is a η-conformant strategy profile σ_C for C such that, if the members of C follow their components of σ_C, then φ will result. Note that this assumes that, when reasoning about what a coalition C can bring about within normative system η, we only assume that the agents from C, *but not necessarily the others*, will conform to η. Variations on this assumptions are of course possible and certainly interesting.

The *syntax* of NATL* closely resembles that of ATL*, the logic that bears the same relationship to ATL as CTL* does to CTL [1,11,12]. Thus, we make the same distinction between *state formulae* and *path formulae* that is made in branching time temporal logics such as CTL* [12]. A state formula is interpreted with respect to an individual state within an AATS, while a path formula is interpreted with respect to a path, or computation, within an AATS. In the text, when we refer to "a formula of NATL*", it should be understood that we mean a *state* formula. The main difference between ATL* and NATL* is that NATL* includes normative systems in the object language. The alphabet from which we construct formulae of NATL*, with respect to AATS S thus contains a set of symbols corresponding to normative systems over S. For convenience, we will use the same symbol to denote a normative system in the object language and in the semantics. The formal syntax of NATL* is given by the BNF grammar in Fig. 3.

We now define the *semantics* of NATL*. These are given with respect to two satisfaction relations: "$\models$" (for state formulae), and "$|\!\models$" (for path formulae). The state formula satisfaction relation "$\models$" holds between pairs of the form S, q (where S is an AATS and q is a state in S), and formulae of NATL*, while the path formula satisfaction relation "$|\!\models$" holds between pairs of the form S, λ (where λ is a computation in S):

⟨*state-fmla*⟩	::= **true**	(truth constant)
	| p	(primitive propositions)
	| $\neg$⟨*state-fmla*⟩	(primitive propositions)
	| ⟨*state-fmla*⟩ $\vee$ ⟨*state-fmla*⟩	(disjunction)
	| $\langle\langle \eta : C \rangle\rangle$ ⟨*path-fmla*⟩	(cooperative ability)
⟨*path-fmla*⟩	::= ⟨*state-fmla*⟩	(state formulae are path formula)
	| $\neg$⟨*path-fmla*⟩	(negation)
	| ⟨*path-fmla*⟩ $\vee$ ⟨*path-fmla*⟩	(disjunction)
	| $\Diamond$⟨*path-fmla*⟩	(eventually)
	| $\Box$⟨*path-fmla*⟩	(always)
	| ⟨*path-fmla*⟩ $\mathcal{U}$ ⟨*path-fmla*⟩	(until)

Fig. 3. The Syntax of NATL*: $p \in \Phi$ is a propositional variable, η is a symbol denoting a normative system, and $C \subseteq Ag$ is a set of agents.

$S, q \models \mathbf{true}$;
$S, q \models p$ iff $p \in \pi(q)$ (where $p \in \Phi$);
$S, q \models \neg\varphi$ iff $S, q \not\models \varphi$;
$S, q \models \varphi \vee \psi$ iff $S, q \models \varphi$ or $S, q \models \psi$; and
$S, q \models \langle\langle \eta : C \rangle\rangle \varphi$ iff $\exists \sigma_C \in \Sigma_C^\eta$, such that $\forall \lambda \in comp(\sigma_C, q)$, we have $S, \lambda \Vdash \varphi$.

The rules defining the path formula satisfaction relation "$\Vdash$" are as follows.

$S, \lambda \Vdash \varphi$ iff $S, \lambda[0] \models \varphi$ (where φ is a state formula);
$S, \lambda \Vdash \neg\varphi$ iff $S, \lambda \not\Vdash \varphi$;
$S, \lambda \Vdash \varphi \vee \psi$ iff $S, \lambda \Vdash \varphi$ or $S, \lambda \Vdash \psi$;
$S, \lambda \Vdash \bigcirc\varphi$ iff $S, \lambda[1, \infty] \Vdash \varphi$;
$S, \lambda \Vdash \Diamond\varphi$ iff $\exists u \in \mathbb{N}$, we have $S, \lambda[u, \infty] \Vdash \varphi$;
$S, \lambda \Vdash \Box\varphi$ iff $\forall u \in \mathbb{N}$ we have $S, \lambda[u, \infty] \Vdash \varphi$; and
$S, \lambda \Vdash \varphi \mathcal{U} \psi$ iff $\exists u \in \mathbb{N}$ s.t. $S, \lambda[u, \infty] \Vdash \psi$, and $\forall v$ s.t. $0 \leqslant v < u$: $S, \lambda[v, \infty] \Vdash \varphi$.

The remaining classical logic connectives ("$\wedge$", "$\rightarrow$", "$\leftrightarrow$") are assumed to be defined as abbreviations in terms of $\neg$, $\vee$, in the conventional manner. Note that we use the classical connectives for both path and state formulae.

We omit set brackets in cooperation modalities for singleton coalitions, writing $\langle\langle \eta : 1 \rangle\rangle$ instead of $\langle\langle \eta : \{1\} \rangle\rangle$. Note that we can recover the cooperation modality of ATL as follows: $\langle\langle C \rangle\rangle \varphi \mathrel{\hat{=}} \langle\langle \eta_\perp : C \rangle\rangle \varphi$. Given these definitions, it is useful to define the universal and existential path quantifiers of CTL [11].

$$\mathsf{A}\varphi \mathrel{\hat{=}} \langle\langle \emptyset \rangle\rangle \varphi \qquad \mathsf{E}\varphi \mathrel{\hat{=}} \langle\langle Ag \rangle\rangle \varphi$$

Note that these are indeed two extreme cases: $\mathsf{A}\varphi$ saying that even if no agents make a choice, the system will evolve such that φ, whereas $\mathsf{E}\varphi$ denotes that when all agents make up their mind and perform an action, the system will evolve such that φ. Whereas these are the only two modalities in CTL, the expressive power of ATL, with $2^{|Ag|}$ coalitional modalities gives it a real notion of agency, and, indeed, coalition. Here, the latter refers just to any subset of the grand coalition Ag: ATL does not assume any pre-defined structure between certain agents, which would qualify them to more likely form a "team" than others.

With $\langle\langle C \rangle\rangle \varphi$ meaning that "coalition C has a strategy to enforce that, no matter what the agents not in C will do, φ holds", ATL enables to reason about *powers* or *abilities* of coalitions: one can reason about *who* can bring it about, in the same way as Dynamic Logic (with its basic modality $[\alpha]\varphi$) is meant to reason about *how* to achieve it [14].

5. Obligations and permissions

Our aim in this section is to show how we can use NATL* to give what we believe is a natural, compelling, and—we hope—useful interpretation to the deontic notions of obligation and permission. We define a derived set of indexed unary modal operators P_η and O_η, where $\mathsf{P}_\eta\varphi$ is intended to mean "φ is permitted within the context of the normative system η", and $\mathsf{O}_\eta\varphi$ is intended to mean "φ is obligatory in the context of the normative

system η". To better understand our approach, recall the "standard" approach to a modal interpretation of deontic logic (see [24, p. 6]). In this approach, we give the semantics of permission and obligation modalities via a standard Kripke semantics, with permission interpreted as a modal diamond, and obligation as a modal box. The deontic accessibility relation is generally accepted to be serial, yielding the modal system KD. The possible worlds in the Kripke structures of standard deontic logic are interpreted as "perfect alternative worlds":

> The idea behind this formal set-up is that being in some possible world (the current world) one may associate a set of perfect alternative worlds, in which all norms are fulfilled. [24, p. 6]

The main distinction between our system and this standard view is that we will consider deontic notions not relative to some absolute standard of norms, but *with respect to a particular normative system*. Thus one cannot simply ask whether φ is "obligatory"; one must ask whether φ is obligatory *in the context of some normative system*. So, how should one interpret the notion of a world in which "all norms are fulfilled" in the context of a normative system η? The natural answer to this question is to interpret a perfect world as a *computation of the system in which every agent acts in respect of the normative system* η. So, our definition of obligations and permissions is as follows:

- φ is said to be permissible within the context of normative system η iff the grand coalition of agents can cooperate to achieve φ within the context of η—that is, if there is some way that the grand coalition of all agents can cause φ by behaving legally, according to the normative system η; and
- φ is said to be obligatory within the context of normative system η iff φ is *inevitable* if the grand coalition conforms to η.

This leads immediately to the following definition of permission and obligation.

$$\mathsf{P}_\eta\varphi \mathrel{\hat{=}} \langle\!\langle \eta : Ag \rangle\!\rangle \varphi \qquad \mathsf{O}_\eta\varphi \mathrel{\hat{=}} \neg\mathsf{P}_\eta\neg\varphi$$

Example 4. Recall the trains scenario S_1 introduced earlier, and the associated normative system η_1 introduced in Example 2. We have:

$$S_1, q_0 \models \mathsf{O}_{\eta_1}\Box\neg(in_E \wedge in_W)$$

To see this, simply observe that a crash state is not permissible within the context of normative system η_1: if both trains conform to η_1, then the trains can never crash. Similarly, we can capture some permissible properties of η_1: in particular, it is permissible for the trains to progress.

$$S_1, q_0 \models \bigwedge_{i\in\{E,W\}} \bigwedge_{X\in\{away,waiting,in\}} \mathsf{P}_{\eta_1}\Diamond X_i$$

Now, consider normative system η_2, which prevents the trains from moving [34].

$$\eta_2(\alpha) = \begin{cases} Q & \text{if } \alpha = move_E \text{ or } \alpha = move_W \\ \emptyset & \text{otherwise} \end{cases}$$

We have the following.

$$S_1, q_0 \models \mathsf{O}_{\eta_2}\Box\neg(in_E \wedge in_W)$$

But of course, we also have the following—undesirable—side effect.

$$S_1, q_0 \models \mathsf{O}_{\eta_2}\Box(away_E \wedge away_W)$$

Note that only the grand coalition Ag plays a role in our definition of permission P_η, implying that there is space for a much more refined analysis of permissions and obligations in the coalitional framework of ATL. The most obvious way to do this, for any coalition C, seems to be to define $\mathsf{P}_{C,\eta}\varphi \hat{=} \langle\!\langle \eta : C \rangle\!\rangle \varphi$, saying that C has a η-conformant strategy to guarantee φ. This does not put any constraint on the agents outside coalition C, they don't necessarily need to behave according to η. In the train example, we would for instance have $S_1, q_0 \models \mathsf{P}_{W,\eta_1}\Diamond in_W \wedge \neg\mathsf{P}_{E,\eta_1}\Diamond in_E$ (train W does not need social behaviour of E to go into the tunnel, whereas train E does need W's social behaviour!). We would also have $S_1, q_4 \models \neg\mathsf{P}_{W,\eta_1}\bigcirc(in_W \wedge \neg in_E)$ (when both trains are waiting, W cannot, by executing only socially acceptable strategies, enforce that in the next state he is in the tunnel without being in a crash situation).

However, we could also define a notion of permission, say $\mathsf{p}_{C,\eta}$, where $\mathsf{p}_{C,\eta}\varphi$ means that, if it is given that *all* agents behave in accordance to η (i.e., both those in C and the others), C can enforce φ. In the trains example we *would* then have $S_1, q_4 \models \mathsf{p}_{W,\eta_1}\bigcirc(in_W \wedge \neg in_E)$ (if *both* trains behave according to η_1, W is permitted to enter the tunnel safely, when both are waiting). We leave a thorough investigation addressing these options for later work, in this paper restricting ourselves to the definition of permissions that refer to the grand coalition only.

We leave a full axiomatization of NATL* and even NATL for future work, but mention here some validities. Note that Example 4 deals with NATL-formulas, in which the object of obligation and permission are *temporal*. Indeed, these seem natural candidates for specific study, since it makes little sense to reason about an agent's deontic status if the world is not subject to change anymore. Indeed, for any objective formula σ, we have:

$$\models (\sigma \leftrightarrow \mathsf{P}_\eta\sigma) \wedge (\sigma \leftrightarrow \mathsf{O}_\eta\sigma) \tag{2}$$

This is not to say, of course, that agents do not have responsibilities to change states of affairs (the following claim is also true for objective formulas):

$$\not\models \mathsf{O}_\eta\varphi \rightarrow \mathsf{O}_\eta\Box\varphi \quad \text{and} \quad \not\models \mathsf{P}_\eta\Diamond\varphi \rightarrow \mathsf{O}_\eta\varphi \tag{3}$$

From a (modal) logic point of view, P_η can be conceived of as a diamond, and O_η as a box-like operator. Thus we have:

$$\models \mathsf{P}_\eta(\varphi \vee \psi) \leftrightarrow (\mathsf{P}_\eta\varphi \vee \mathsf{P}_\eta\psi) \quad \text{and} \quad \models \mathsf{O}_\eta(\varphi \wedge \psi) \leftrightarrow (\mathsf{O}_\eta\varphi \wedge \mathsf{O}_\eta\psi)$$

If something is naturally, or physically inevitable, then it is obligatory in any normative system; if something is an obligation within a given normative system η, then it is permissible in η; and if something is permissible in a given normative system, then it is naturally (physically) possible. Thus we have the following chain of implications (where η is an

arbitrary *non-trivial* normative system).

$$\models (\mathsf{A}\varphi \rightarrow \mathsf{O}_\eta\varphi) \quad \models (\mathsf{O}_\eta\varphi \rightarrow \mathsf{P}_\eta\varphi) \quad \models (\mathsf{P}_\eta\varphi \rightarrow \mathsf{E}\varphi)$$

Note that the second of these properties does not hold for arbitrary normative systems η: If $\eta = \eta_\top$ for example, we have $\mathsf{O}_\eta\varphi$ $(= \neg\mathsf{P}_\eta\neg\varphi)$ for any φ, but at the same time, $\mathsf{P}_\eta\psi$ for no ψ.

Notice that we would not expect physical ability to imply ability within a normative system, and indeed in NATL*, it does not.

$$\not\models \langle\langle C\rangle\rangle\varphi \rightarrow \langle\langle \eta : C\rangle\rangle\varphi$$

If the LHS of this implication were true, then the witness to its truth would be a strategy profile for C; but this strategy profile would not necessarily be η-conformant.

Moreover, the fact that something is obligatory does not imply that any individual coalition can achieve it, either within or outside the context of a normative system.

$$\not\models \mathsf{O}_\eta\varphi \rightarrow \langle\langle \eta : C\rangle\rangle\varphi \quad \text{and} \quad \not\models \mathsf{O}_\eta\varphi \rightarrow \langle\langle C\rangle\rangle\varphi$$

Considering the distinguished normative systems $\eta_\bot$ and $\eta_\top$, we get the following.

$$\models \mathsf{O}_{\eta_\bot}\bigcirc \mathbf{true}$$
$$\models \mathsf{O}_{\eta_\top}\bigcirc \mathbf{false}$$

If we look at properties with respect to the $\preceq$ relation over normative systems, we obtain:

Proposition 3. *Let S be an* AATS*, and let η, η' be arbitrary non-trivial normative systems over S such that $\eta \preceq \eta'$ (i.e., η is less restrictive than η'). Then*:

(1) $S \models \langle\langle \eta' : C\rangle\rangle\varphi \rightarrow \langle\langle \eta : C\rangle\rangle\varphi$
(2) $S \models \mathsf{P}_{\eta'}\varphi \rightarrow \mathsf{P}_\eta\varphi$
(3) $S \models \mathsf{O}_\eta\varphi \rightarrow \mathsf{O}_{\eta'}\varphi$

Proof. For (1), assume $S, q \models \langle\langle \eta' : C\rangle\rangle\varphi$. Then $\exists\sigma_C \in \Sigma_C^{\eta'}$ such that $\forall\lambda \in out(\sigma_C, q)$, we have $S, \lambda|\models \varphi$. But since by Proposition 2, we have that $\forall C \subseteq Ag : \Sigma_C^{\eta'} \subseteq \Sigma_C^{\eta}$, then $\sigma_C \in \Sigma_C^{\eta}$, and hence $S, q \models \langle\langle \eta : C\rangle\rangle\varphi$. Part (2) is the special case of (1) where C is the grand coalition; part (3) is the contrapositive of (2). □

We combine this result with (1), as follows.

Proposition 4. *Let S be an* AATS*, and let η, η' be arbitrary non-trivial normative systems over S. Then*:

(1) $S \models \langle\langle \eta \sqcup \eta' : C\rangle\rangle\varphi \rightarrow \langle\langle \eta : C\rangle\rangle\varphi$
(2) $S \models \langle\langle \eta : C\rangle\rangle\varphi \rightarrow \langle\langle \eta \sqcap \eta' : C\rangle\rangle\varphi$
(3) $S \models \mathsf{P}_{\eta\sqcup\eta'}\varphi \rightarrow \mathsf{P}_\eta\varphi$
(4) $S \models \mathsf{P}_\eta\varphi \rightarrow \mathsf{P}_{\eta\sqcap\eta'}\varphi$

(5) $S \models \mathsf{O}_{\eta}\varphi \rightarrow \mathsf{O}_{\eta \sqcup \eta'}\varphi$
(6) $S \models \mathsf{O}_{\eta \sqcap \eta}\varphi \rightarrow \mathsf{O}_{\eta}\varphi$

Proof. Part (1) follows from Proposition 3 and the fact that $\eta \preceq \eta \sqcup \eta'$. The remaining cases are identical to these. □

Finally, we briefly look at *iterative* behaviour of obligations and permissions: it is interesting to note that imposing several norms does *not* have a *cumulative* effect

$$\not\models \mathsf{P}_{\eta_1}\mathsf{P}_{\eta_2}\varphi \rightarrow \mathsf{P}_{\eta_1}\varphi \quad \text{and} \quad \not\models \mathsf{O}_{\eta_1}\varphi \rightarrow \mathsf{O}_{\eta_1}\mathsf{O}_{\eta_2}\varphi \tag{4}$$

(4) should be clear from the truth definition: when unfolding the definition for $\langle\langle \eta_1 : Ag \rangle\rangle \langle\langle \eta_2 : Ag \rangle\rangle \varphi$ for instance, the, the search for η_1-conformant strategies is completely 'overruled' by a search for strategies that are η_2-conformant. This even holds when we insert a number of temporal operators, i.e., we also have $\not\models \mathsf{O}_{\eta_1}\varphi \rightarrow \mathsf{O}_{\eta_1}\bigcirc\mathsf{O}_{\eta_2}\varphi$. If one wants to impose a normative system η_2 on top of another η_1, the union operator $\sqcup$ seems to be the most appropriate way to do it.

6. Multi-agent systems, social laws, and social contracts

In this section, we demonstrate how the apparatus of NATL* may be used to analyse the properties of multi-agent systems. We present a formal analysis of the *social contract*, a well-known concept from political and economic philosophy. The idea of the social contract is generally attributed in its original form to Thomas Hobbes (1588–1679) and his concept of a society as a "Leviathan", with substantial subsequent refinements and contributions to the theory by John Locke (1632–1704), Jean-Jacques Rousseau (1712–1778), and most notably in the present era, John Rawls. The term "social contract" is usually understood as referring to the set of rules, norms, or conventions that a society implicitly accepts in order to coordinate and manage its behaviour. Ken Binmore, a game theorist and recent commentator on the social contract [6,7], understands the term as follows:

> We are all players in the game of life, with divergent aims and aspirations that make conflict inevitable. In a healthy society, a balance between these differing aims and aspirations is achieved so that the benefits of cooperation are not entirely lost in internecine strife. Game theorists call such a balance an *equilibrium*. Sustaining such equilibria requires the existence of commonly understood conventions about how behaviour is to be coordinated. It is such a system of coordinating conventions that I shall identify with a social contract. [5, p. 6]

Notice that the term social contract does not only refer to the *formalised* laws that a society imposes upon itself, but also to the informally accepted norms and conventions that are part and parcel of everyday life.

Understanding how a social contract works is of great interest to political philosophers and economists. Apart from anything else, much of the function of government can be understood as attempting to *engineer* a society's social contract. And yet there are many

paradoxes associated with the social contract, not the least of which is why a rational agent should comply with such a contract. A rational agent, seeking to maximise its own welfare, may well observe that the best outcome would be if it chose to ignore the rules, while allowing other agents to conform to them; and yet if every agent reasons the same way, then everyone ignores the social contract, and its benefits are lost. A game theoretic analysis of such a scenario leads to a model resembling the well known "prisoner's dilemma" scenario (or perhaps more accurately, the *iterated* prisoner's dilemma), which accounts in no small measure for the interest this scenario has attracted [3,6,7].

In this section, we will use NATL*, our logic of normative ability, to formalise a model of the social contract, and begin a preliminary analysis of its properties. To do this, we will work with three different types of structure, as follows:

A *Multi-Agent System* consists of an AATS (which specifies the underlying behaviour of the system, and the effect that agent's actions have on the system), together with a set of goals, one for each agent. Thus a multi-agent system determines what agents want to achieve (their aspirations), and the fundamental—physical or natural—rules within which they must operate.

A *Social Law* is a structure that is developed and manipulated by the overseer, designer, or manager of a system. Following [34], we define a social law to consist of a normative system (i.e., a set of rules) together with some objective, or goal. The idea is that the designer, overseer, or manager of the system will develop the normative system so that, if the norms are followed, then the objective will be achieved (in which case we say the normative system is *globally effective*). If one thinks about the social contract in the conventional sense, then we can think of the "designer" as the politician, who is trying to modify or replace an existing social contract. The designer will try to construct a set of rules so that, if they are followed, the objective will inevitably follow. Of course, the designer cannot ignore the agents within the system, which will typically be self-interested, with their own goals and objectives to achieve.

A *Social Contract* consists of a multi-agent system together with a social law. That is, a social contract defines (i) the natural or physical properties of a system; (ii) the aspirations of the agents within the system; (iii) a set of normative rules, in addition to those inherent within the physical structure of the system, which are intended to restrict the behaviour of the agents in the system in certain desirable ways; and (iv) a system-level objective, or goal, which it is hoped will be achieved if the agents within the system conform to the normative rules.

The main relationships between the concepts in our structures are described by the entity-relationship diagram in Fig. 4. In what follows, we will formalise each type of structure, give examples to illustrate them, and investigate some of their properties. Note that, throughout this section, when we refer to a normative system, it should be understood that we mean a *non-trivial* normative system.

A *multi-agent system* $\mathcal{M}$ is an $(n+1)$-tuple: $\mathcal{M} = \langle S, \gamma_1, \ldots, \gamma_n \rangle$ where:

- $S = \langle Q, q_0, Ag, Ac_1, \ldots, Ac_n, \rho, \tau, \Phi, \pi \rangle$ is an AATS, intended to represent the physical properties of the system in question; and
- for all $i \in Ag$, γ_i is a path formula of NATL*, intended to represent the *goal* of agent i.

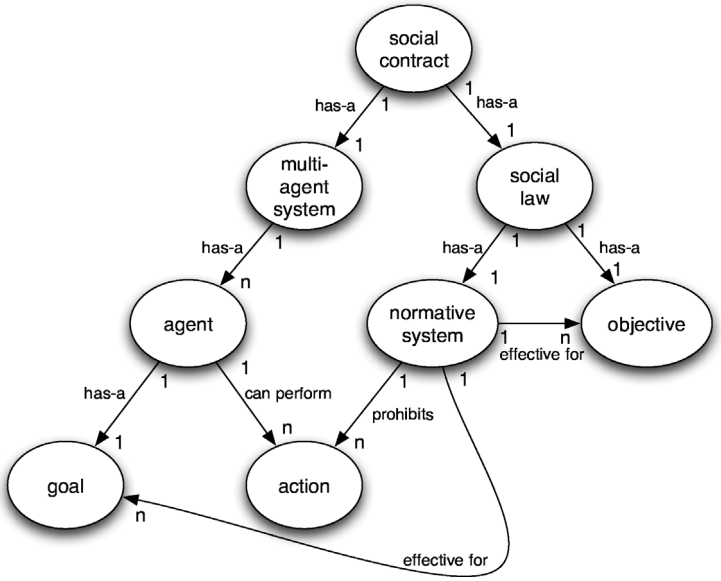


Fig. 4. The main structures and concepts in our logic-based model of social contracts.

Example 5. In the trains scenario, consider the following goals for the two agents:

$$\gamma_E = \Box\Diamond in_E \qquad \gamma_W = \Box\Diamond in_W$$

That is, the goal of each train is to enter the tunnel *infinitely often*. (In the terminology of the reactive systems literature, these are *liveness properties* [11,22].)

Following [34], we take a *teleological* view of social laws. That is, we consider social laws with respect to the goal, or *objective* that they are intended to achieve. Formally, we define a *social law* over an AATS S to be a pair: $\mathcal{L} = \langle \Psi, \eta \rangle$ where:

- Ψ is a path formula of ATL representing the *objective* of the law; and
- $\eta : Ac_{Ag} \to 2^Q$ is a normative system over S.

We say a social law $\langle \Psi, \eta \rangle$ (over a MAS $\mathcal{M} = \langle S, \gamma_1, \ldots, \gamma_n \rangle$) is:

globally effective	if	$S, q_0 \models \mathsf{O}_\eta \Psi$;
weakly globally effective	if	$S, q_0 \models \mathsf{P}_\eta \Psi$; and
globally ineffective	if	$S, q_0 \models \mathsf{O}_\eta \neg\Psi$

Example 6. In the trains system, perhaps the main goal of a designer will be to prevent negative interactions between agents, and in particular, to prevent the trains from crashing in the tunnel. We denote this objective as Ψ_1:

$$\Psi_1 = \Box\neg(in_E \wedge in_W)$$

The following are globally effective social laws:

$$\langle \Psi_1, \eta_1 \rangle \qquad \langle \Psi_1, \eta_2 \rangle$$

Thus, if the designer of the system is only concerned about preventing the trains from crashing, then either η_1 or η_2 would appear to be satisfactory.

Of course, the designer of a normative system will not only be concerned about whether it will be globally effective, i.e., whether or not it would "succeed" if everyone adhered to the constraints it imposed. The designer must also consider whether or not these constraints *will be adhered to*. This question cannot be answered without reference to the goals that agents have. This motivates the introduction of the next structure: a social contract.

Formally, we model a *social contract* Ω as a pair: $\Omega = \langle \mathcal{M}, \mathcal{L} \rangle$ where:

- $\mathcal{M} = \langle S, \gamma_1, \ldots, \gamma_n \rangle$ is a multi-agent system; and
- $\mathcal{L} = \langle \Psi, \eta \rangle$ is a social law over S.

Now, the basic question we ask of a social contract is whether or not it is *successful*. The overall success of a social contract hinges on two distinct issues. The first, and in some sense easier, issue is that of whether the social law component $\mathcal{L} = \langle \Psi, \eta \rangle$ is globally effective, as defined above. Thus, this means asking whether or not it is the case that, *if every agent in the system adhered to the normative system* η, the corresponding objective Ψ would be achieved.

However, there is a second issue in determining whether a social contract is successful, which is arguably more troublesome. We must determine whether or not the agents in the system will actually conform to the rules of the normative system η. We assume agents are autonomous (we cannot impose decisions upon them), and self-interested (they will choose to perform a particular action if they believe it is in their best interests, and not otherwise). So, let us say that a social contract is

locally effective for agent i	if	$S, q_0 \models \mathsf{O}_\eta \gamma_i$;
partially locally effective for agent i	if	$S, q_0 \models \langle\langle \eta : i \rangle\rangle \gamma_i$;
weakly locally effective for agent i	if	$S, q_0 \models \mathsf{P}_\eta \gamma_i$; and
locally ineffective for agent i	if	$S, q_0 \models \mathsf{O}_\eta \neg \gamma_i$

We will say a social contract is simply "locally effective" if it is locally effective for all agents in the system, and similarly for weakly locally effective and locally ineffective.

Example 7. Let us return to the train system S_1: consider the system with goals γ_E and γ_W, as defined earlier, and normative system η_3 defined as follows.

$$\eta_3(\alpha) = \begin{cases} Q & \text{if } \alpha = idle_E \\ Q \setminus \{q_0\} & \text{if } \alpha = idle_W \\ \emptyset & \text{if } \alpha = move_E \\ \{q_0\} & \text{if } \alpha = move_W \end{cases}$$

Thus on the first time step, this social law forces the east bound train to move while forcing the westbound train to stay still, and thereafter prevents either train from idling: they move in lock-step. It is not hard to see that, thus defined, the trains do not crash (and hence $\langle \Psi_1, \eta_3 \rangle$ is a globally effective social law), and moreover, the trains are in the tunnel infinitely often, hence η_3 is locally effective.

The *disadvantage* of the normative system η_3 is that it *completely* constrains the actions of the trains. That is, neither train has any choice about what to do: they only ever have one action available to them. Thus this normative system would *not* be effective if the trains were ever to desire to stop (e.g., to pick up passengers!) So, consider the following normative system, which works by forbidding trains from lingering in the tunnel, but is otherwise the same as η_1.

$$\eta_4(\alpha) = \begin{cases} \{q_4, q_7\} & \text{if } \alpha = idle_E \\ \{q_2, q_6\} & \text{if } \alpha = idle_W \\ \{q_5, q_6\} & \text{if } \alpha = move_E \\ \{q_7\} & \text{if } \alpha = move_W \end{cases}$$

Moreover, let us weaken the goals of the agents somewhat.

$$\gamma'_E = \Box \mathsf{P}_{\eta_4} \Diamond in_E, \qquad \gamma'_W = \Box \mathsf{P}_{\eta_4} \Diamond in_W$$

The idea is that the agent's goals are not *necessarily* to enter the tunnel infinitely often, but that it is *permissible* for them to enter the tunnel infinitely often.

Then we have:

$$S, q_0 \models \mathsf{O}_{\eta_4} \Psi_1 \quad \& \quad S, q_0 \models \mathsf{O}_{\eta_4} \gamma'_E \quad \& \quad S, q_0 \models \mathsf{O}_{\eta_4} \gamma'_W$$

In sum, considering the social contract

$$\Omega' = \big\langle \langle S_1, \gamma'_E, \gamma'_W \rangle, \langle \Psi_1, \eta_4 \rangle \big\rangle$$

we see that Ω' is both globally and locally effective. It is globally effective because the objective Ψ_1 is obligatory in the context of η_4, and it is locally effective because the goals γ'_E and γ'_W of the two agents are obligatory in the context of η_4.

For completeness, let us see how some of the other social contracts that arise from our discussion stack up. Consider:

$$\Omega_1 = \big\langle \langle S_1, \gamma_E, \gamma_W \rangle, \langle \Psi_1, \eta_1 \rangle \big\rangle$$
$$\Omega_2 = \big\langle \langle S_1, \gamma_E, \gamma_W \rangle, \langle \Psi_1, \eta_2 \rangle \big\rangle$$
$$\Omega_3 = \big\langle \langle S_1, \gamma_E, \gamma_W \rangle, \langle \Psi_1, \eta_3 \rangle \big\rangle$$

The social contracts Ω_1 and Ω_2 are globally effective but not locally effective, while Ω_3 is both globally and locally effective.

Given the preceding discussion, Table 1 summarises the possible types of social contracts, with respect to their properties at the social law level (are they effective?) and the individual agents in the system (will they help or hinder agents in achieving their goals?).

Table 1
Social contract types

	Objective status	Agent status	Comment	Globally effective?	Locally effective?
1.	$\mathsf{O}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\gamma_i$	strongest	yes	yes
2.	$\mathsf{O}_\eta\Psi$	$\bigwedge_{i\in Ag}\langle\langle\eta:i\rangle\rangle\gamma_i$		yes	partially
3.	$\mathsf{O}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{P}_\eta\gamma_i$		yes	weakly
4.	$\mathsf{O}_\eta\Psi$	$\bigvee_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$		yes	no
5.	$\mathsf{O}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$		yes	no!
6.	$\mathsf{P}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\gamma_i$		weakly	yes
7.	$\mathsf{P}_\eta\Psi$	$\bigwedge_{i\in Ag}\langle\langle\eta:i\rangle\rangle\gamma_i$		weakly	partially
8.	$\mathsf{P}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{P}_\eta\gamma_i$		weakly	weakly
9.	$\mathsf{P}_\eta\Psi$	$\bigvee_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$		weakly	no
10.	$\mathsf{P}_\eta\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$		weakly	no!
11.	$\mathsf{O}_\eta\neg\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\gamma_i$		ineffective	yes
12.	$\mathsf{O}_\eta\neg\Psi$	$\bigwedge_{i\in Ag}\langle\langle\eta:i\rangle\rangle\gamma_i$		ineffective	partially
13.	$\mathsf{O}_\eta\neg\Psi$	$\bigwedge_{i\in Ag}\mathsf{P}_\eta\gamma_i$		ineffective	weakly
14.	$\mathsf{O}_\eta\neg\Psi$	$\bigvee_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$		ineffective	no
15.	$\mathsf{O}_\eta\neg\Psi$	$\bigwedge_{i\in Ag}\mathsf{O}_\eta\neg\gamma_i$	weakest	ineffective	no!

- Social contracts of type (11)–(15) will be unacceptable to the designers of a system, since these contracts ensure that the global objective will *not* be achieved. Of the remaining social contract types, (1)–(5) will be preferred over (6)–(10), since these *guarantee* the achievement of the global objective.
- Social contracts of types (5), (10), and (15) will be unacceptable to all agents within the system (the "population"), since such social contracts will prevent their goals being achieved.
- Social contracts of types (4), (9), and (14) will *disenfranchise* some agents within the system, by preventing them from achieving their goals. It is hard to see why these agents would accept such a social contract, since by definition to do so would prevent them from achieving their goals. One might comment that social contract types (4) and (9) reflect the situation in "underclass" communities, where society requires members of such a disadvantaged community to respect the laws of the society, while at the same time effectively preventing them from advancement if they do respect the laws of the society.

Since a social contract designer would reject social contract types (11)–(15), and agents within the system would reject types (5), (10), and probably (4) and (9) also, this leaves us with types (1)–(3) and (6)–(8) as potentially successful social contract types. Given that the designer would prefer all of (1)–(3) over all of (6)–(8), and an agent would prefer (1)–(2) over (3), it seems that the most viable social contracts are those of types (1) or (2).

Clearly, the "strongest" type of social contract is type (1). Here, every agent will benefit from conforming to the contract (if everyone conforms to the social contract, then everyone will have their goals achieved), while society also benefits (if everyone conforms to the

social contract then the global objective is achieved). However, from an individual agent's point of view, there is perhaps not too much to distinguish type (1) and (2), except the *guarantee* of success in (1).

Clearly, there is much more that can be done with respect to the analysis of social contracts and how they work than we have attempted to do here. For example, the issue of how a society deals with violation of a social contract by an agent is not addressed. One might therefore consider modelling sanctions within the formal framework. Similarly, one might try to develop a more fine-grained model of preferences and utilities than the simple logical specification of goals that we have adopted here.

7. Related work

With respect to the models and intuitions underpinning our framework, the closest approach in the literature to ours is the social laws framework of Moses, Shoham, and Tennenholtz. Shoham and Tennenholtz were the first to precisely articulate the notion of social laws for multiagent systems, and set up a basic formal framework within which computational questions about social laws could be formulated [30–32]. The particular application domain was that of traffic laws for robotic agents. The basic framework was extended by Fitoussi and Tennenholtz, to consider *simple* social laws—essentially, social laws that could not be any simpler without failing [13].

Moses and Tennenholtz developed a deontic epistemic logic for representing properties of multiagent systems with normative structures [26]. Although semantically similar to NATL* (and ATEL [16]), their logic was quite different to ATL in terms of the syntactic constructs it provided, and the emphasis was primarily on deriving axioms capturing static aspects of artificial social systems and social laws. The logic did contain notions of "socially reachable" states of affairs, which roughly corresponds to our normative ability operators, the normative system was fixed in the semantics of the logic—normative systems were not first-class components of the language. Nevertheless, much of the intuition underpinning this logic is similar to our own, and this system was, apart from ATL, the largest single inspiration for the present paper.

Deontic logic originally arose in the context of formal philosophy, but has recently found increasing application in computer science and multi-agent systems research [36]. Deontic logic is usually formulated as a normal modal logic with Kripke semantics, containing unary modal operators O and P, where $\mathsf{O}\varphi$ is intended to be read as "φ is obligatory" and $\mathsf{P}\varphi$ as "φ is permissible". Although there is broad agreement that the modal system KD serves as a "standard" deontic logic, there are many "paradoxes" that arise when a naive modal approach to reasoning about deontic notions is adopted, and much of the deontic logic literature is concerned both with trying to understand whether these apparent paradoxes really are problematic, and if so, how they can be fixed.

A prominent discussion in deontic logic concerns "contrary-to-duty' (CTD) obligations, in which there is a "primary" obligation together with a "secondary" one, which comes into effect when the first obligation is violated. Prakken and Sergot [28] convincingly argue that many of the paradoxes with CTD can be solved by adding a temporal component to the language. This would make NATL also an appropriate framework to at least deal with

those CTD obligations. However, although this remains for further investigation, we feel the applicability of NATL* is even broader here, since in NATL* we can explicitly refer to *different norms*. A standard example of a CTD obligation (from [28]) is obtained by the following triple: (i) there must be no fence; (ii) if there is a fence, it must be a white fence; and (iii) there is a fence. It might be well possible, in NATL, to model (i) as an obligation with respect to a system η_1, whereas (ii) then is an obligation with respect to a "fall-back" normative system η_2, which gives a recipe for the available choices if the agents cannot be obedient to the "default" system η_1.

A preliminary investigation of the relationship between ATL and deontic logic was presented in [19]: this work took the obvious route of enriching ATL directly with deontic accessibility relations and modalities, and tentatively exploring the space of possible systems that result.

The work in this paper can perhaps be understood as developing a *computationally grounded semantics* for obligation, and in this sense, we are following Lomuscio and Sergot with their development of deontic interpreted systems [21]. The basic idea in their logic was to interpret the deontic accessibility relation as linking states where the system is correctly functioning: thus $q \models \mathsf{O}\varphi$ in their system if φ is true in all states q' that can be reached from state q such that the system is correctly functioning in q'. Lomuscio and Sergot gave an axiomatization of their logic, and also investigated the *epistemic* properties of their system—in particular, what a "correctly functioning" agent would know. Although, as we noted above, epistemic extensions to ATL have been developed [16], and Moses and Tennenholtz made use of such notions in their logic of artificial social systems, we are not aware of any attempt to analyse the knowledge implicit in normative systems, and it may be that some combination of Lomuscio and Sergot's approach with our own would yield some insights in this direction.

We should also make mention of Meyer's reduction of deontic logic to dynamic logic [23]. Meyer's insight was to see how an account of obligation could be given in dynamic logic by introducing a primitive proposition V, whose satisfaction in some state q would indicate that a violation of the normative system had occurred. We could then say that an action α was forbidden if that action lead to the V being satisfied. Building on this notion of "forbidden", Meyer went on to show how obligation and permission could be defined. Some articles examining the relationship between deontic logic and action logic, in a similar vein to that of Meyer, were presented in [25].

The abilities in ATL refer to "physical" abilities of agents, and are identified with choices; this is similar, but not the same as the notion of "responsibilities" that is ascribed to agents in the so-called STIT (Seeing To It That) theory [4]. The exact relation between ATL abilities and STIT responsibilities is an interesting issue, and deserves further analyses. This is a prerequisite to be able to compare NATL* with STIT plus obligations [18]. Although the two frameworks look semantically rather similar, there are notable difference in validities (in STIT for instance, one has that if it is obligatory that i sees to it that φ, then if follows that φ is obligatory).

Finally, we should mention our own work on social laws, which introduced the AATS structures, and investigated issues of feasibility and effectiveness of social laws in this setting [34]. The main difference is that in the present paper, we attempt to bring the reasoning

in [34] *into the object language*, rather than carrying out operations such as implementing social laws at the meta-language level.

8. Conclusions

In this paper, we have developed a logic of normative ability, as an extension to ATL, the logic of cooperative ability developed by Alur, Henzinger, and Kupferman. We have demonstrated how this logic can be used for defining deontic notions such as obligation and permission, illustrated these ideas with a running example, and applied the logic to a preliminary formal analysis of the social contract.

There are many possible routes for future investigation. One obvious question is the extent to which other notions such as knowledge can be incorporated into the framework [15,16]. Another question is the computational properties (model checking, satisfiability) of NATL*: syntactically, NATL* is closer to the "full" branching time logic CTL* than its computationally better behaved cousin CTL, and hence model checking and satisfiability are likely to be complex for NATL*. However, restricted forms of NATL* may well have more desirable computational properties, and so the extent to which such restrictions might be usable in practice is surely worth studying.

Finally, the twin issues of violation and sanction are also surely worth investigating, and a refinement of the social contract types in Table 1 would be a first step in this direction.

References

[1] R. Alur, T.A. Henzinger, O. Kupferman, Alternating-time temporal logic, J. ACM 49 (5) (2002) 672–713.
[2] R. Alur, T.A. Henzinger, F.Y.C. Mang, S. Qadeer, S.K. Rajamani, S. Taşiran, Mocha: Modularity in model checking, in: CAV 1998: Tenth International Conference on Computer-aided Verification, in: Lecture Notes Comput Sci., vol. 1, Springer-Verlag, Berlin, 1998, pp. 521–525.
[3] R. Axelrod, The Evolution of Cooperation, Basic Books, New York, 1984.
[4] N. Belnap, M. Perloff, Seeing to it that: A canonical form for agentives, Theoria 54 (1988) 175–199.
[5] K. Binmore, Fun and Games: A Text on Game Theory, D.C. Heath and Company, Lexington, MA, 1992.
[6] K. Binmore, Game Theory and the Social Contract, vol. 1: Playing Fair, MIT Press, Cambridge, MA, 1994.
[7] K. Binmore, Game Theory and the Social Contract, vol. 2: Just Playing, MIT Press, Cambridge, MA, 1998.
[8] B. Chellas, Modal Logic: An Introduction, Cambridge University Press, Cambridge, England, 1980.
[9] R. Conte, C. Castelfranchi, Cognitive and Social Action, UCL Press, London, 1995, 69–79, 1999.
[10] G. van Drimmelen, Satisfiability in alternating-time temporal logic, in: Eighteenth Annual IEEE Symposium on Logic in Computer Science (LICS 2003), Ottawa, Canada, 2003, pp. 208–217.
[11] E.A. Emerson, Temporal and modal logic, in: J. van Leeuwen (Ed.), Handbook of Theoretical Computer Science, vol. B, Formal Models and Semantics, Elsevier Science, Amsterdam, 1990, pp. 996–1072.
[12] E.A. Emerson, J.Y. Halpern, 'Sometimes' and 'not never' revisited: On branching time versus linear time temporal logic, J. ACM 33 (1) (1986) 151–178.
[13] D. Fitoussi, M. Tennenholtz, Choosing social laws for multi-agent systems: Minimality and simplicity, Artificial Intelligence 119 (1–2) (2000) 61–101.
[14] D. Harel, D. Kozen, J. Tiuryn, Dynamic Logic, MIT Press, Cambridge, MA, 2000.
[15] W. van der Hoek, M. Wooldridge, Model checking cooperation, knowledge, and time—a case study, Research in Economics 57 (3) (2003) 235–265.
[16] W. van der Hoek, M. Wooldridge, Time, knowledge, and cooperation: Alternating-time temporal epistemic logic and its applications, Studia Logica 75 (3) (2003) 125–157.

[17] W. van der Hoek, M. Wooldridge, Towards a logic of rational agency, Logic J. IGPL 11 (2) (2003) 135–159.
[18] J.F. Horty, N. Belnap, The deliberative STIT: A study of action, omission, ability, and obligation, J. Philos. Logic 24 (6) (1995) 583–644.
[19] W. Jamroga, W. van der Hoek, M. Wooldridge, On obligations and abilities, in: A. Lomuscio, D. Nute (Eds.), Deontic Logic in Computer Science, in: Lecture Notes in Artificial Intelligence, vol. 3065, Springer-Verlag, Berlin, 2004, pp. 165–181.
[20] D. Lewis, Convention—A Philosophical Study, Harvard University Press, Cambridge, MA, 1969.
[21] A. Lomuscio, M. Sergot, Deontic interpreted systems, Studia Logica 75 (1) (2003) 63–92.
[22] Z. Manna, A. Pnueli, Temporal Verification of Reactive Systems—Safety, Springer-Verlag, Berlin, 1995.
[23] J.-J.Ch. Meyer, A different approach to deontic logic: Deontic logic viewed as a variant of dynamic logic, Notre Dame J. Formal Logic 29 (1) (1988) 109–136.
[24] J.-J.Ch. Meyer, R.J. Wieringa, Deontic logic: A concise overview, in: J.-J. Ch. Meyer, R.J. Wieringa (Eds.), Deontic Logic in Computer Science—Normative System Specification, Wiley, New York, 1993, pp. 3–16.
[25] J.-J.Ch. Meyer, R.J. Wieringa (Eds.), Deontic Logic in Computer Science—Normative System Specification, Wiley, New York, 1993.
[26] Y. Moses, M. Tennenholtz, Artificial social systems, Computers and AI 14 (6) (1995) 533–562.
[27] M. Pauly, A modal logic for coalitional power in games, J. Logic Comput. 12 (1) (2002) 149–166.
[28] H. Prakken, M. Sergot, Contrary-to-duty obligations, Studia Logica 57 (1) (1996) 91–115.
[29] K. Segerberg, Getting started: Beginnings in the logic of action, Studia Logica 51 (3/4) (1992) 347–378.
[30] Y. Shoham, M. Tennenholtz, On the synthesis of useful social laws for artificial agent societies, in: Proceedings of the Tenth National Conference on Artificial Intelligence (AAAI-92), San Diego, CA, 1992.
[31] Y. Shoham, M. Tennenholtz, On social laws for artificial agent societies: Off-line design, in: P.E. Agre, S.J. Rosenschein (Eds.), Computational Theories of Interaction and Agency, MIT Press, Cambridge, MA, 1996, pp. 597–618.
[32] Y. Shoham, M. Tennenholtz, On the emergence of social conventions: Modelling, analysis, and simulations, Artificial Intelligence 94 (1–2) (1997) 139–166.
[33] B. Skyrms, Evolution of the Social Contract, Cambridge University Press, Cambridge, England, 1996.
[34] W. van der Hoek, M. Roberts, M. Wooldridge, Social laws in alternating time: Effectiveness feasibility, and synthesis, Synthese (2005).
[35] A. Walker, M. Wooldridge, Understanding the emergence of conventions in multi-agent system, in: Proceedings of the First International Conference on Multi-Agent Systems (ICMAS-95), San Francisco, CA, 1995, pp. 384–390.
[36] R.J. Wieringa, J.-J.Ch. Meyer, Deontic logic in computer science, in: J.-J.Ch. Meyer, R.J. Wieringa (Eds.), Deontic Logic in Computer Science—Normative System Specification, Wiley, New York, 1993, pp. 17–40.
[37] M. Wooldridge, Reasoning about Rational Agents, MIT Press, Cambridge, MA, 2000.
[38] M. Wooldridge, An Introduction to Multiagent Systems, Wiley, New York, 2002.
[39] M. Wooldridge, N.R. Jennings, Intelligent agents: Theory and practice, Knowledge Engrg. 10 (2) (1995) 115–152.