ISSN 1862 – 4405

Dagstuhl Seminar Proceedings 07122

Normative Multi-agent Systems

Vol. II

Guido Boella, Leendert van der Torre, Harko Verhagen (Eds.)

Dagstuhl Seminar Proceedings 07122

Normative Multi-agent Systems

G. Boella, L. v. d. Torre, H. Verhagen (Eds.)

Dagstuhl Seminar 07122, 18.03. - 23.03.2007

Vol. II

ISSN 1862 - 4405

What an Agent Ought To Do
Authors: Broersen, Jan M. ; van der Torre, Leendert

What is Input/Output Logic? Input/Output Logic, Constraints, Permissions
Authors: Makinson, David ; van der Torre, Leendert

Interaction between Normative Systems and Cognitive Agents in Temporal Modal Defeasible Logic

Regis Riveret[1], Antonino Rotolo[1], Guido Governatori[3]

[1] CIRSIFD and Law Faculty
Via Galliera 3, 40121, Bologna, Italy
rriveret@cirsfid.unibo.it
rotolo@cirsfid.unibo.it
[2] University of Queensland
Brisbane, Queensland, QLD 4072, Australia
guido@itee.uq.edu.au

Abstract. While some recent frameworks on cognitive agents addressed the combination of mental attitudes with deontic concepts, they commonly ignore the representation of time. An exception is [1], which also manages some temporal aspects with regard to both cognition and deontic provisions. We propose in this paper a variant of the logic presented in [1] to deal in particular with temporal intervals.

Keywords. Time, Norm, Temporal Modal Defeasible Logic

1 Introduction

A common approach in the agent literature for programming cognitive agents in a BDI (belief, desire, intention) framework is the use of rules to represent or manipulate the agents mental attitudes. In addition to the three mental attitudes of beliefs, desires and intentions, some works include deontic concepts to denote norms, commitments of social agents and social rationality [2,3,4,5,6]. However, these frameworks commonly ignore the representation of time. An exception is [1], which adopts the rule-based approach of [7,8,9] and extends it to accommodate temporal aspects. Time is integrated by pairing assertions with instants representing the time at which assertions hold and by descriminating transient and persistent conclusions. Persistent conclusions persists until some interrupting event occurs. Pairing assertions with instants is unsatisfactory for at least two reasons: (i) some properties may end at a certain time not associated to any explicit external event, (ii) we may like to represent rules where conditions have to hold for certain temporal intervals. To remedy these issues, in this paper, we increase the expressive power of the logic presented in [1] with temporal intervals. The framework presented is based on Temporal Defeasible Logic (TDL), an umbrella expression designating extensions of Defeasible Logic to capture time. Beside [1], TDL has proved useful in modelling temporal aspects of normative reasoning, such as temporalised normative provisions [10]; in addition, the notion of temporal viewpoints -the temporal positions from which things are viewed- allows for a logical account of retroactive norms and norm modifications [11].

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/923

The paper is organised as follows. In section 2, we introduce the general conceptual model behind the framework. Section 3 provides an outline of basic Defeasible Logic. Section 4 describes a variant of modal TDL that formalises the model of cognition.

2 Time, norms and mental attitudes

Our model aims to give an account of some temporal aspects with regard to both mental attitudes and deontic provisions. The starting point is the acknowledgement that, on the one hand, recent works shows that reasoning about agents can be embedded in frameworks based on non-monotonic logic, as the most interesting problems concern cases where the agent's mental attitudes are in conflict or when they are incompatible with deontic provisions. On the other hand, in a temporal setting, non-monotonicity can also be used to conclude that mental attitudes or deontic provisions persist up to some future time unless there is a reason for it not to persist. One can thus argue that a type of non-monotonicity concerns situations where mental attitudes are in conflict or when they are incompatible with some deontic provisions, while another type of non-monotonocity concerns temporal aspects. Our model is based on these two types of non-monotonicity.

We adopt the model of [1] that extends the works of [7,8,9] with time. These later works are themselves inspired by Bratman's analysis of so-called policy-based attitudes. In Bratman's view intentions are used to choose partial plans for realisation of a goal and have a close relation to mean-ends, whereas [7,8,9] intentions are related not only to means-ends but also to their consequences. This notion is particularly relevant with deontic and normative notions, for example if we want to say that an agent is legally for A if the A is a side effect and if the agent did A with the intention to do A. [7,8,9] extends this policy-based approach to other attitudes and motivational factors as beliefs, intentions and obligations. An agent types correspond to the different ways through which conflicts are detected and solved: a realistic agent thus corresponds to a conflict-resolution type in which beliefs override all other factors, while other agent types, such as simple minded, selfish or social ones adopt different orders of overruling.

[1] is on the same line of research of [7,8,9] and focus on some temporal aspects. [1] is based on Bratman's [12] which in his pursuit for a temporally extended rational agency exposed a principle that can be roughly stated as follows:

- At $t0$, agent A consider the policy to adopt with respect a certain range of activities. On this basis, agent A forms a general intention to φ in circumstances of type ψ.
- From $t0$ to $t1$, A retains this general intention.
- At $t1$, A notes that he/she is or will be in circumstance ψ at $t2$.
- Based on the previous steps, A forms the intention at $t1$ to φ at $t2$.

Given the temporal nature of Bratmans historical principle, and the idea that some intentions can be retained from one moment to another, [1] accounts for two types of temporal intentions: transient intentions which hold only for an instant of time, and persistent intentions which an agent is going to retain unless some interrupting event occurs that forces the agent to reconsider them. This event can be just a brute fact or it can be a modification of the policy of the agent.

The expressive power of [1] is unsatisfactory for at least two reasons. First some properties may end at a certain time not associated to any explicit external event, for example, an obligation or a norm in force may hold until some specific temporal reference. Secondly, we may like to represent rules where conditions have to hold for certain temporal intervals. To remedy these issues, in this paper, we increase the expressive power of the logic presented in [1] with temporal intervals.

Ordinarily, intervals are defined as sets of instants between two indicated instants. Doing so, some difficulties may arise when we want to express that an event (for example) occurs in an interval. This refers to the non-homogeneity or transient character of events: if an event occurs in an interval conceived as a set of instants, then it would also occur in the set of instants that defines it and this would conflict with the transient characterisation of events. Hence, we deviate somewhat to the standart definition of intervals as a set of instants, and define an interval as a pair of instants of the form $[t_i, t_f]$ and usually denote them by T (plus eventual subscript). We identify two subtsets of interval to differentiate intervals in which an associated property holds at any instant between the boundaries and intervals in which an associated property holds at least one instant between the boundaries. We shall call the firsts A-interval and the seconds B-intervals. A-intervals are represented by expressions of the form $\overline{[t_i, t_f]}$ and are usually denoted by $\overline{T}$ while B-intervals are represented by expressions of the form $\widehat{[t_f, t_f]}$ and denoted by $\widehat{T}$. If the wide hat or the line over an interval is omitted then it is either an A-interval or a B-interval.
Mental attitudes and normative provisions are related to temporal references and the passage of time allows change of these elements. This is in accordance with the commonly accepted opinion that in a static system where nothing changes, the temporal dimension does not provide more understanding. Our references are intervals and allows us to temporalise literals and rules. In its simplest form, a temporal literal is an expression of the form $l{:}T$ where l is a literal and T is either an A-interval or a B-interval. Intuitively, $l{:}\overline{T}$ means that l holds for all instants between the boundaries of $\overline{T}$ while $l{:}\widehat{T}$ means that l holds for at least an instant between the boundaries of $\widehat{T}$. For example, $adult(bob){:}\overline{[1973, max]}$ means that Bob has legally reach adulthood in 1973. Similarly, rules are temporalised by associating to it a time interval, and so a temporal rule is an expression of the form:

$$(r\!: a_1{:}T_1 \ \ldots\ a_n{:}T_n \hookrightarrow b{:}\overline{T}){:}\overline{T_r}$$

The time labels allow us to deal formally with the different temporal dimensions of a normative system. The temporal intervals labelling the antecedent of a rule, the consequent of the rule and the overall rule are interpreted respectively as the intervals of *efficacy*, *applicability* and *time of force* of the represented provision. These different temporal dimensions are in line with the legal temporal model developed in [13]. and that allows us to give an accurate account of temporal aspects of norms and therefore to be consistent with legal principles. Note that the interval $\overline{T_r}$ labelling the entire rule is an A-interval because the force of a provision is generally an homogeneous property. Similarly, we constraint for the sake of simplicity the interval labelling the literal in the head of the rule to be an A-interval. Intervals in the body can be A-intervals or B-intervals. An example of a temporal rule is:

$$(r\colon\ born(X){:}[t,t] \rightarrow major(X){:}\overline{[t+18,max]}){:}\overline{[1970,max]}$$

This rule formalises the provision in force in 1970 and later that people legally reach adulthood at 18. Consequently of the different temporal dimensions, a conclusion can be associated to two temporal intervals. The first interval is the interval with which the consequent of the rule is labelled while the second interval corresponds to the time of force interval associated to the rule. We represent such temporalisation of conclusion by concatenation of intervals by means of the symbol ':' and we call such concatenation chain of viewpoints. For example. giving the rule r and the fact that Bob was born in 1960, then one can conclude $major(bob){:}[1978,max]{:}[1970,max]$, that is, Bob is legally adult in 1978 (and later) from somebody reasoning in 1970 (and later).

Chain of viewpoints are of the upmost importance when one has to deal with the retroactivity of norms. Retroactivity usually occurs when the effects of a rule r apply to an interval $\overline{[t_i,t_f]}$ which begins before the interval $[t'_i,t'_f]$ attached to the antecedent of r, that is, $t_i < t'_i$. Another case of retroactivity is when the consequence of a rule r' in force in $\overline{[t_{ri},t_{rf}]}$ has as intervalof applicability $\overline{[t_i,t_f]}$ and $t_i < t_{ri}$. For an illustration of the utility of chain of viewpoints with respects to retroactivity, consider the following rules:

$$(r1\colon Income > 90{:}[1Mar\widehat{06,1J}un06] \Rightarrow_{\mathrm{OBL}} \neg Tax{:}\overline{[1Jan06,1Jun06]}){:}\overline{[15Jan06,1Jun06]}$$

$$(r2\colon Income > 100{:}[1Mar\widehat{06,1J}un06] \Rightarrow_{\mathrm{OBL}} Tax{:}\overline{[1Jan06,1Jun06]}){:}\overline{[1Apr06,1Jun06]}$$

Rule $r1$ states that if the income of a person is in excess of ninety thousand between the 1st March 2006 and the 1st June 2006 then she has not to pay the tax from 1st January 2006 to 1st June 2006 with the policy being in force from 15 January 2006 to 1st June 2006. This means that the norm is part of the tax regulation from 15 January 2006 to 1st June 2006. The second rule, in force from 1st April 2006, establishes a tax returns lodged after 1st April 2006. These two rules illustrate the concept of viewpoints. Consider that the conditions in the antecedent of both rules hold, then one would derive $\neg Tax{:}\overline{[1Jan06,1Jun06]}{:}\overline{[15Jan06,1Jun06]}$ but $Tax{:}\overline{[1Jan06,1Jun06]}{:}\overline{[1Apr06,1Jun06]}$, that is, if one reason from a point of view between the 15 January and the 1st April then the tax is due while if one reason from a point of view between the 1st April and he 1st June 2006 then no tax is due. Even though trivial cases of the phenomenon of retroactivity are captured by rules such as $r1$ and $r2$, we should be able to detect retroactivity also in other scenarios, where normative effects are in fact applied retroactively to some conditions as a result of complex arguments that involve many rules. This problem is of great importance not only because the designer of a normative system may have the goal to state retroactive effects in more articulated scenarios, but also because she should be able to check whether such effects are not obtained when certain regulations regard matters for which retroactivity is not in general permitted. This is the case of criminal law, where the principle -Nullum crimen, nulla poena sine praevia lege poenali- is valid.

3 Defeasible Logic

Our system is formalised in an extension of Defeasible Logic. We provide in this section a brief recall of it. Defeasible Logic [14,15,16] is based on a logic programming-like language and it is a simple, efficient but flexible non-monotonic formalism capable of dealing with many different intuitions of non-monotonic reasoning. An argumentation semantics exists [17] that makes its use possible in argumentation systems. DL has a linear complexity [18] and also has several efficient implementations [19].

A Defeasible Logic theory is a structure $D = (F, R, \prec)$ where F is a finite set of facts, R a finite set of rules, and $\prec$ a superiority relation on R. Facts are indisputable statements, for example, "Bob is a minor," formally written as $\mathit{minor}(\mathit{bob})$. Rules can be strict, defeasible, or defeaters. Strict rules are rules in the classical sense; whenever the premises are indisputable, so is the conclusion. An example of a strict rule is "Minors are persons," formally written as $r1$: $\mathit{minor}(X) \rightarrow \mathit{person}(X)$. Defeasible rules are rules that can be defeated by contrary evidence. An example of a defeasible rule is "Persons have legal capacity"; formally, $r2$: $\mathit{person}(X) \Rightarrow \mathit{hasLegalCapacity}(X)$. Defeaters are rules that cannot be used to draw any conclusion. Their only use is to prevent some conclusions by defeating some defeasible rules. An example of this kind of rule is "Minors might not have legal capacity," formally expressed as $r3$: $\mathit{minor}(X) \leadsto \neg \mathit{hasLegalCapacity}(X)$. The idea here is that even if we know that someone is a minor, this is not sufficient evidence for the conclusion that he or she does not have legal capacity. The superiority relation between rules indicates the relative strength of each rule. That is, stronger rules override the conclusions of weaker rules. For example, if $r3 \succ r2$, then the rule $r3$ overrides $r2$, and we can derive neither the conclusion that Bob has legal capacity nor the conclusion that he does have legal capacity.

Given a set R of rules, we denote the set of all strict rules in R by R_s, the set of defeasible rules in R by R_d, the set of strict and defeasible rules in R by R_{sd}, and the set of defeaters in R by R_{dft}. $R[q]$ denotes the set of rules in R with consequent q. In the following $\sim p$ denotes the complement of p, that is, $\sim p$ is $\neg p$ if p is an atom, and $\sim p$ is q if p is $\neg q$. For a rule r we will use $A(r)$ to indicate the body or antecedent of the rule and $C(r)$ for the head or consequent of the rule. A rule r consists of its antecedent $A(r)$ (written on the left; $A(r)$ may be omitted if it is the empty set), which is a finite set of literals; an arrow; and its consequent $C(r)$, which is a literal. In writing rules we omit set notation for antecedents. Conclusions are tagged according to whether they have been derived using defeasible rules or strict rules only. So, a conclusion of a theory D is a tagged literal having one of the following four forms:

$+\Delta q$ meaning that q is definitely provable in D.
$-\Delta q$ meaning that q is not definitely provable in D.
$+\partial q$ meaning that q is defeasibly provable in D.
$-\partial q$ meaning that q is not defeasibly provable in D.

These different notions of provability come of use here because they enable the system to label a suggestion as stronger or weaker depending on the kind of proof associated with it. Provability is based on the concept of a derivation (or proof) in D. A derivation is a finite sequence $P = (P(1),...,P(n))$ of tagged literals. Each tagged literal satisfies some proof conditions. A proof condition corresponds to the inference rules that refer to one of the four kinds of conclusions we have mentioned above. $P(1..n)$ denotes the initial part of the sequence P of length n. We state below the conditions for defeasibly derivable conclusions:

If $P(i+1) = +\partial q$ then
(1) $+\Delta q \in P(1..i)$ or
(2) (2.1) $\exists r \in R_{sd}[q]\ \forall a \in A(r) : +\partial a \in P(1..i)$ and
 (2.2) $-\Delta{\sim}q \in P(1..i)$ and
 (2.3) $\forall s \in R[{\sim}q]$ either
  (2.3.1) $\exists a \in A(s) : -\partial a \in P(1..i)$ or
  (2.3.2) $\exists t \in R_{sd}[q]$ such that
   $\forall a \in A(t) : +\partial a \in P(1..i)$ and $t \succ s$.

If $P(i+1) = -\partial q$ then
(1) $-\Delta q \in P(1..i)$ and
(2) (2.1) $\forall r \in R_{sd}[q]\ \exists a \in A(r) : -\partial a \in P(1..i)$ or
 (2.2) $+\Delta{\sim}q \in P(1..i)$ or
 (2.3) $\exists s \in R[{\sim}q]$ such that
  (2.3.1) $\forall a \in A(s) : +\partial a \in P(1..i)$ and
  (2.3.2) $\forall t \in R_{sd}[q]$ either
   $\exists a \in A(t) : -\partial a \in P(1..i)$ or $t \not\succ s$.

Informally, a defeasible derivation for a provable literal consists of three phases: First, we propose an argument in favour of the literal we want to prove. In the simplest case, this consists of an applicable rule for the conclusion (a rule is applicable if its antecedent has already been proved). Second, we examine all counter-arguments (rules for the opposite conclusion). Third, we rebut all the counter-arguments (the counter-argument is weaker than the pro-argument) or we undercut them (some of the premises of the counterargument are not provable).

4 Temporal Modal Defeasible Logic

Defeasible Logic allows us to deal with defeasibility but as such does not provide any mean to deal with modalities and temporal aspects. Temporal Modal Defeasible Logic is an umbrella expression to designate possible extensions of Defeasible Logic to capture modalities and time. We present in this section an extension of [1] with intervals as exposed in the model (see Section 2).

4.1 Modal Domain

The combination of mental attitudes and obligations are framed in extending Defeasible Logic following the works of [7,8,9] and capture some basic facets of the modal notions of knowledge, intentions, action and obligation.

To extend Defeasible Logic with modal operators, new types of rules relative to modal operator are introduced: arrows of the rules are labelled by the different modalities we want to deal with. This solution leads to distinguishing different modes through which the literals can be derived using rules. How such types of derivation are related to the introduction of the corresponding modalised literals can be expressed as follows: if $X \in \{\mathrm{KNOW}, \mathrm{INT}, \mathrm{ACT}, \mathrm{OBL}\}$, then

$$\frac{\Gamma \quad \Gamma \Rightarrow_X \psi}{\Gamma \mid\!\sim X\psi}\ \mathrm{MI}$$

We make an exception when rules for knowledge are concerned. The reason for this is that we assume that beliefs are conceived of as the knowledge the agent has of the environment, and so they are used by the agent to make inferences about how the world is: in this perspective, belief conclusions correspond to factual knowledge and do not need to be modalised. But besides this exception, which can be removed if required, schema MI captures the basic logical behaviour of our modal rules. Notice, also, that actions are successful and intentional and so, when $\mathrm{ACT}\psi$ is derived, this also implies that ψ and $\mathrm{INT}\psi$ are the case.

Other relations between modalities are captured by means of *rule conversions* and *conflicts*.

The notion of *rule conversion* permits to use rules for a modality X as they were for another modality Y. Suppose that a rule of a specific type is given and also suppose that all the literals in the antecedent of a rule are provable in one and the same modality, then it is arguable that the conclusion of the rule inherits the modality of the antecedent. For example, consider the following formalisation of the Yale Shooting Problem.

$$\text{load:}\overline{[t,t]},\ \text{shoot:}\overline{[t,t]} \Rightarrow_{\mathrm{KNOW}} \text{kill:}\overline{[t,t]}$$

This rule encodes the knowledge of an agent that knows that loading the gun with live ammunitions, and then shooting will kill her friend. This example clearly shows that the qualification of the conclusions depends on the modalities relative to the individual acts "load" and "shoot". In particular, if we obtain that the agent intends to load and to shoot the gun ($\mathrm{INT}(\mathit{load})$, $\mathrm{INT}(\mathit{shoot})$), then, since she knows that the consequence of these actions is the death of her friend, she intends to kill him. However, if shooting was not intended, then we have prima facie to say that killing, too, was not intentional. To define the admitted conversions we introduce a binary relation Convert over the modalities of the language. When we write $\mathrm{Convert}(\mathrm{KNOW}, \mathrm{OBL})$ this means that a knowledge rule r can be used to derive an obligation (of course, provided that all its antecedents are derived as obligations): r can thus be converted into a rule for intention.

Beside conversions, *Conflicts* play an important role in the current context and it is crucial to establish criteria for detecting and solving conflicts between the different components which characterise the cognitive profiles of agent's deliberation, and, above all between mental states and normative provisions. Conflicts are detected and solved by a similar strategy than basic Defeasible Logic, i.e, by following a pattern such that (i) in a first phase an argument supporting the conclusion is advanced (ii) in the second phase any possible attack are considers, and (iii) finally the counter-attack for each attack. Accordingly we introduce a ternary relation Attack over the set of modalities

that defines which types of rules are in conflict and which are the stronger ones. For example, if we write $\mathrm{Attack}(\mathrm{OBL},\mathrm{INT},\mathrm{ACT})$ this means that, in the reasoning pattern illustrated above, obligations in general override intentions, which in turn override actions.
The relation Attack is explicitly linked to that of agent type. Classically, agent types are characterised by stating conflict resolution types in terms of orders of overruling between rules [3,7,9,8]. In this perspective, agent types are meaningful within a non-monotonic setting and are nothing but general strategies to detect and solve conflicts between the different components of the cognitive profiles of agent's deliberation. In [3] 24 possible types are identified while, in [8], based on a different framework, 20 combinations are proposed. Typically, rational agents are assumed to be at least *realistic*: a realistic agent, in fact, is such that rules for knowledge override all other components. If the realistic condition is abandoned, we may have various forms of wishful thinking. Given the minimal assumption that a rational agent should be realistic, we may further constrain agent's deliberation in order not to violate obligations: a *social agent* type requires that obligations are stronger than the other motivational components with the exception of beliefs. Other agent types can be specified, for which see [7,8,9].

4.2 Temporal domain

Approaches in temporal reasoning are traditionally based on either instants, intervals or both by representing one through the other. We represent intervals by means of instants. Formally, we consider a totally ordered discrete set $\mathscr{T}$ of points of time termed "instants" and over it the order relation $> \subseteq \mathscr{T} \times \mathscr{T}$. We usually denotes the variables ranging over the members of $\mathscr{T}$ by t and its eventual subscripts, and the minimla unit by u.

Ordinarily, intervals are defined as sets of instants between two indicated instants. Here we deviate to this definition because of the non-homogeneity or transient character of events: if an event occurs in an interval conceived as a set of instants, then it would also occur in the set of instants that defines it and this would conflict with the transient characterisation of events. Hence, we define an interval as a pair of instants. Formally, an interval is a member of the set $\mathrm{Inter} = \{[t_1,t_2] \in \mathscr{T} \times \mathscr{T} | t_1 \leq t_2\}$. As can be noted, this definition allows "punctual intervals" , i.e. intervals of the form $[t,t]$. Among the set Inter, we identify two subtsets of interval to differentiate intervals in which an associated property holds at any instant between the boundaries and intervals in which an associated property holds at least one instant between the boundaries. We shall call the first A-interval and the seconds B-intervals. The set of A-intervals is denoted AInter while the set of B-intervals is denoted BInter. We shall usually denote intervals by T, A-intervals by $\overline{T}$ and B-intervals by $\widehat{T}$ (plus eventual subscripts). We consider the functions start() and end() that returns respectively the lower bound and upper bound of an interval.

As explained in section 2, a conclusion can be associated to two temporal intervals consequently of the different temporal dimensions. The first interval is the interval of applicability with which the consequent of the rule is labelled while the second interval corresponds to the time of force interval associated to the rule. Each interval can be assimilated to temporal Russian-dolled viewpoints from which conclusions are considered. We represent such temporalisation of conclusion by concatenation of intervals by means of the symbol ':' and we call such concatenation chain of viewpoints. Chain of viewpoints are denoted by V (plus eventual subscripts).

Temporal calculi are driven by operators over intervals. In the literature, one can find many relations that hold between intervals. For example, [20] proposes an algebra of intervals with thirteen mutually exclusive relations between two intervals. For our purpose, we consider the set of relations to "subinterval" denoted $\sqsubseteq$, "over" denoted over, "meet" denoted meet, "start in" denoted si, "start before end" denoted sbe, and "start before start" denoted sbs.

Definition 1. *Let two intervals* $T \in \text{Inter}$ *and* $T' \in \text{Inter}$,

$T \sqsubseteq T'$ *iff* $\text{start}(T') \leq \text{start}(T)$ *and* $\text{end}(T) \leq \text{end}(T')$.

$\text{over}(T, T')$ *iff* $\text{start}(T') \leq \text{start}(T) \leq \text{end}(T')$ *or*
$\text{start}(T') \leq \text{end}(T) \leq \text{end}(T')$ *or* $\text{start}(T) \leq \text{start}(T') \leq \text{end}(T)$.

$\text{meet}(T, T')$ *iff* $\text{end}(T) + u = \text{start}(T')$.

$\text{si}\ (T, T')$ *iff* $\text{start}(T') \leq \text{start}(T) \leq \text{end}(T')$.

$\text{sbe}\ (T, T')$ *iff* $\text{start}(T) \leq \text{end}(T')$.

$\text{sbs}\ (T, T')$ *iff* $\text{start}(T) \leq \text{start}(T')$.

Note that $T \sqsubseteq T'$, $\text{si}(T, T')$ or $\text{sbe}(T, T')$ implies $\text{over}(T, T')$, that $T \sqsubseteq T'$ implies $\text{si}(T, T')$ and that $\text{over}(T, T')$ implies $\text{over}(T', T)$.

In order to lighten the paper, we may use the abbreviation consisting in placing chain of viewpoints as arguments of the previous operators, such that for example,

- $T \sqsubseteq T' : T''$ stands for $T \sqsubseteq T'$ and $T \sqsubseteq T''$.
- $T : T' \sqsubseteq T'' : T'''$ stands for $T \sqsubseteq T''$ and $T' \sqsubseteq T'''$.
- $T : T' \sqsubseteq T''$ stands for $T \sqsubseteq T''$ and $T' \sqsubseteq T''$.

and similarly for other operators. Finally, we also use some abbreviations with regard to the function start() and end(), such that for example,

- $\text{start}(T) = \text{end}(T'' : T''')$ stands for $\text{start}(T) = \text{end}(T'')$ and $\text{start}(T) = \text{end}(T''')$.
- $\text{start}(T : T') = \text{end}(T'' : T''')$ stands for $\text{start}(T) = \text{end}(T'')$ and $\text{start}(T') = \text{end}(T''')$.
- $\text{start}(T : T') = \text{end}(T'')$ stands for $\text{start}(T) = \text{end}(T'')$ and $\text{start}(T') = \text{end}(T'')$.

and similarly for others combinations of relation between start() and end().

4.3 The Language

A temporal defeasible agent theory consists of a discrete totally ordered set of instants of time, a set of *facts* or indisputable statements, four sets of rules for knowledge, intentions, intentional actions, and obligations, and a *superiority relation* $>$ among rules saying when a single rule may override the conclusion of another rule. For $X \in \{\mathrm{KNOW}, \mathrm{INT}, \mathrm{ACT}, \mathrm{OBL}\}$, a temporal *strict rule* is an expression of the form $(\phi_1, \ldots, \phi_n \rightarrow_X \psi)$:$\overline{T_r}$ such that whenever the premises $\phi_1 : \widehat{T_r}, \ldots, \phi_n : \widehat{T_r}$ are indisputable so is the conclusion $\psi : \overline{T_r}$. A *defeasible rule* is an expression of the form $(\phi_1, \ldots, \phi_n \Rightarrow_X \psi)$:$\overline{T_r}$ whose conclusion can be defeated by contrary evidence. An expression $(\phi_1, \ldots, \phi_n \rightsquigarrow_X \psi)$:$\overline{T_r}$ is a *defeater* used to defeat some defeasible rules by producing evidence to the contrary. It is worth noting that modalised literals can occur only in the antecedent of rules: the reason of this is that the rules are used to derive modalised conclusions while we do not conceptually need to iterate modalities. This limitation makes the system more manageable.

Definition 2 (Language). *Let $\mathscr{T}$ a discrete totally ordered set of instants of time,* Prop *be a set of propositional atoms,* $\mathrm{Mod} = \{\mathrm{KNOW}, \mathrm{INT}, \mathrm{ACT}, \mathrm{OBL}\}$ *be the set of modal operators, and* Lab *be a set of labels. The sets below are the smallest sets closed under the following rules:*

Literals

$$\mathrm{Lit} = \mathrm{Prop} \cup \{\neg p | p \in \mathrm{Prop}\}$$

Modal Literals

$$\mathrm{ModLit} = \{Xl, \neg Xl | l \in \mathrm{Lit}, X \in \{\mathrm{INT}, \mathrm{ACT}, \mathrm{OBL}\}\};$$

Intervals

$$\mathrm{Inter} = \{T = [t1, t2] | t1, t2 \in \mathscr{T}, t1 \leq t2\};$$

A-Intervals

$$\mathrm{AInter} = \{\overline{T} = \overline{[t1, t2]} | t1, t2 \in \mathscr{T}, t1 \leq t2\};$$

B-Intervals

$$\mathrm{BInter} = \{\widehat{T} = \widehat{[t1, t2]} | t1, t2 \in \mathscr{T}, t1 \leq t2\};$$

Chain of Viewpoints

$$\mathrm{ChainView} = \{V = T1, V' = T1 : T2 | T1, T2 \in \mathrm{AInter} \cup \mathrm{BInter}\};$$

Temporal Literals

$$\mathrm{TempLit} = \{l : T | l \in \mathrm{Lit}, T \in \mathrm{AInter} \cup \mathrm{BInter}\};$$

Multi-Temporal Literals

$$\mathrm{MTempLit} = \{l : V | l \in \mathrm{Lit}, V \in \mathrm{ChainView}\};$$

Temporal Modal literals

$$\mathrm{TempModLit} = \{Xl : T | Xl \in \mathrm{ModLit}, T \in \mathrm{AInter} \cup \mathrm{BInter}\};$$

Multi-Temporal Modal literals

$$\text{MTempModLit} = \{Xl : V | Xl \in \text{ModLit}, V \in \text{ChainView}\};$$

Temporal Rules

$$\begin{array}{l}
\text{Rule}_s = \{(r : \phi_1, \ldots, \phi_n \rightarrow_X \psi) : T | \\
\qquad r \in \text{Lab}, A(r) \subseteq \text{TempLit} \cup \text{TempModLit}, X \in \text{Mod}, \psi \in \text{TempLit}, T \in \text{AInter}\} \\
\text{Rule}_d = \{(r : \phi_1, \ldots, \phi_n \Rightarrow_X \psi) : T | \\
\qquad r \in \text{Lab}, A(r) \subseteq \text{TempLit} \cup \text{TempModLit}, X \in \text{Mod}, \psi \in \text{TempLit}, T \in \text{AInter}\} \\
\text{Rule}_{dft} = \{(r : \phi_1, \ldots, \phi_n \leadsto_X \psi) : T | \\
\qquad r \in \text{Lab}, A(r) \subseteq \text{TempLit} \cup \text{TempModLit}, X \in \text{Mod}, \psi \in \text{TempLit}, T \in \text{AInter}\} \\
\text{Rule} = \text{Rule}_s \cup \text{Rule}_d \cup \text{Rule}_{dft}
\end{array}$$

We use some abbreviations: $A(r)$ denotes the set $\{\phi_1, \ldots, \phi_n\}$ of *antecedents* of the rule r, and $C(r)$ to denote the *consequent* ψ of the rule r. We use also superscript for mental attitude, subscript for type of rule, and $\text{Rule}[\phi]$ for rules whose consequent is ϕ. If one does not refer to the content of the rule, a temporal rule can be written as $r{:}\overline{T}$ where r is the label of the rule and $\overline{T}$ is a temporal interval. If q is a literal, $\sim q$ denotes the complementary literal (if q is a positive literal p then $\sim q$ is $\neg p$; and if q is $\neg p$, then $\sim q$ is p);

Definition 3 (Defeasible Agent Theory). *A defeasible agent theory is a structure*

$$D = (\mathscr{T}, F, R^{\text{KNOW}}, R^{\text{INT}}, R^{\text{ACT}}, R^{\text{OBL}}, >, \mathscr{C}, \mathscr{A})$$

where

- $\mathscr{T}$ *a discrete totally ordered set of instants of time;*
- $F \subseteq \text{TempLit} \cup TempModLit$ *is a finite set of facts;*
- $R^{\text{KNOW}} \subseteq \text{Rule}^{\text{KNOW}}$, $R^{\text{INT}} \subseteq \text{Rule}^{\text{INT}}$, $R^{\text{ACT}} \subseteq \text{Rule}^{\text{ACT}}$, $R^{\text{OBL}} \subseteq \text{Rule}^{\text{OBL}}$ *are four finite sets of rules such that each rule has a unique label;*
- $> \subseteq R^{\text{KNOW} \cup \text{INT} \cup \text{ACT} \cup \text{OBL}} \times R^{\text{KNOW} \cup \text{INT} \cup \text{ACT} \cup \text{OBL}}$ *is an acyclic superiority relation.*
- $\mathscr{C} \subseteq \{Convert(X,Y) | X,Y \in \text{Mod}\}$ *is a set of conversions.*
- $\mathscr{V} \subseteq \{Attack(X,Y,Z) | X,Y,Z \in \text{Mod}\}$ *is a set of attack relation.*

4.4 Proof Theory

The formalism we have introduced allows us to temporalise rules, thus we have to admit the possibility that rules are not only given but can be proved to hold for certain span of time. Accordingly we have to give conditions that allow us to derive rules instead of literals. A conclusion of a theory D is a tagged temporal literal or rule having one of the following forms:

$+\Delta \gamma{:}V$ meaning that $\gamma{:}V$ is definitely provable in D.
$-\Delta \gamma{:}V$ meaning that $\gamma{:}V$ is not definitely provable in D.

$+\partial\gamma{:}V$ meaning that $\gamma{:}V$ is defeasible provable in D.
$-\partial\gamma{:}V$ meaning that $\gamma{:}V$ is not defeasible provable in D.

Provability is based on the concept of a derivation (or proof) in D. A derivation is a finite sequence $P = (P(1),..,P(n))$ of tagged modal literals or rules temporalised by chain of viewpoints. Each tagged temporal modal literal or rule satisfies some proof conditions, which correspond to inference rules for the four kinds of conclusions we have mentioned above. In order to lighten the presentation of the proof conditions, we present separately the condition for applicability of rules:

If Convert(Y,X) and $r{:}\overline{T_r}$ is *Δ-applicable* in the proof condition for $\pm\Delta_X$ then
(1) $+\Delta r{:}\overline{T_r} \in P(1..i)$, and either
(2) $r{:}\overline{T_r} \in R^X$,
  (2.1) $\forall\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r})$,
    (2.1.1) $+\Delta_{\text{KNOW}}\alpha : \overline{T_\alpha} \in P(1..i)$, or $+\Delta_{\text{KNOW}}\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
    (2.1.2) $+\Delta_{\text{ACT}}\alpha : \overline{T_\alpha} \in P(1..i)$, or $+\Delta_{\text{ACT}}\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$,
  (2.2) $\forall\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r})$,
    (2.2.1) $+\Delta_{\text{KNOW}}\alpha : \widehat{T_\alpha} \in P(1..i)$, or $+\Delta_{\text{KNOW}}\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
    (2.2.2) $+\Delta_{\text{ACT}}\alpha : \widehat{T_\alpha} \in P(1..i)$, or $+\Delta_{\text{ACT}}\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, and
  (2.3) $\forall Z\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r}), +\Delta_Z\alpha : \overline{T_\alpha} \in P(1..i)$, or $+\Delta_Z\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, and
  (2.4) $\forall Z\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r}), +\Delta_Z\alpha : \widehat{T_\alpha} \in P(1..i)$, or $+\Delta_Z\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
(3) $r{:}\overline{T_r} \in R^Y$,
  (3.1) $\forall\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r})$, $+\Delta_X\alpha : \overline{T_\alpha} \in P(1..i)$, or $+\Delta_X\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, and
  (3.2) $\forall\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r}), +\Delta_X\alpha : \widehat{T_\alpha} \in P(1..i)$, or $+\Delta_X\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$.

The conditions for a rule r to be *∂-applicable* are the same as those for *Δ-applicable*, but where we replace Δ with ∂.

If Convert(Y,X) and $r{:}\overline{T_r}$ is *Δ-discarded* in the proof condition for $\pm\Delta_X$ then
(1) $-\Delta r{:}\overline{T_r} \in P(1..i)$, or either
(2) $r{:}\overline{T_r} \in R^X$,
  (2.1) $\exists\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r})$,
    (2.1.1) $-\Delta_{\text{KNOW}}\alpha : \overline{T_\alpha} \in P(1..i)$, and $-\Delta_{\text{KNOW}}\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, and
    (2.1.2) $-\Delta_{\text{ACT}}\alpha : \overline{T_\alpha} \in P(1..i)$, and $-\Delta_{\text{ACT}}\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
  (2.2) $\exists\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r})$,
    (2.2.1) $-\Delta_{\text{KNOW}}\alpha : \widehat{T_\alpha} \in P(1..i)$, and $-\Delta_{\text{KNOW}}\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, and
    (2.2.2) $-\Delta_{\text{ACT}}\alpha : \widehat{T_\alpha} \in P(1..i)$, and $-\Delta_{\text{ACT}}\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
  (2.3) $\exists Z\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r}), -\Delta_Z\alpha : \overline{T_\alpha} \in P(1..i)$, and $-\Delta_Z\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
  (2.4) $\exists Z\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r}), -\Delta_Z\alpha : \widehat{T_\alpha} \in P(1..i)$, and $-\Delta_Z\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
(3) $r{:}\overline{T_r} \in R^Y$,
  (3.1) $\exists\alpha{:}\overline{T_\alpha} \in A(r{:}\overline{T_r})$, $-\Delta_X\alpha : \overline{T_\alpha} \in P(1..i)$, and $-\Delta_X\alpha : \overline{T_\alpha}{:}\widehat{T_r} \in P(1..i)$, or
  (3.2) $\exists\alpha{:}\widehat{T_\alpha} \in A(r{:}\overline{T_r}), -\Delta_X\alpha : \widehat{T_\alpha} \in P(1..i)$, and $-\Delta_X\alpha : \widehat{T_\alpha}{:}\widehat{T_r} \in P(1..i)$.

The conditions for a rule $r{:}\overline{T_r}$ to be *∂-discarded* are the same as those for *Δ-discarded*, but where we replace Δ with ∂.

We are now ready to define the proof theory that is, the inference conditions to derive tagged conclusions from a given theory D. We begin with the proof conditions to determine whether a rule is a definite conclusion of a theory D. A temporal rule $r{:}\overline{T}$ is definitely provable ($+\Delta$) if (1) there exists a rule $r{:}\overline{T_r}$ in the set of rule such that $\overline{T} \sqsubseteq \overline{T_r}$, or (2) r is defined in two intervals $\overline{T_{r1}}$ and $\overline{T_{r2}}$ that make up $\overline{T}$. Formally:

If $P(i+1) = +\Delta r{:}\overline{T}$ then
(1) $\exists \overline{T_r}, \sqsubseteq \overline{T_r}, r{:}\overline{T_r} \in R, \overline{T}$, or
(2) $\exists \overline{T_{r1}}, \exists \overline{T_{r2}}$, meets$(\overline{T_{r1}}, \overline{T_{r2}})$, start$(\overline{T_{r1}})$ = start$(\overline{T})$, end$(\overline{T_{r2}})$ = end$(\overline{T})$,
$r{:}\overline{T_{r1}} \in R$ and $r{:}\overline{T_{r2}} \in R$.

A rule r is not definitely provable at interval $\overline{T}$ if (1) there is not such rule in the set rules defined in a larger interval (2) r is not defined in any intervals $\overline{T_{r1}}$ and $\overline{T_{r2}}$ that make up $\overline{T}$. Formally:

If $P(i+1) = -\Delta r{:}\overline{T}$ then
(1) $\forall \overline{T_r}, \sqsubseteq \overline{T_r}, r{:}\overline{T_r} \notin R, \overline{T}$, and
(2) $\forall \overline{T_{r1}}, \forall \overline{T_{r2}}$, meets$(\overline{T_{r1}}, \overline{T_{r2}})$, start$(\overline{T_{r1}})$ = start$(\overline{T})$, end$(\overline{T_{r2}})$ = end$(\overline{T})$,
$r{:}\overline{T_{r1}} \notin R$ or $r{:}\overline{T_{r2}} \notin R$.

A temporal rule $r{:}\widehat{T}$ is definitely provable ($+\Delta$) if there exists a rule $r{:}\overline{T_r}$ in the set of rule such that over$(\widehat{T}, \overline{T_r})$. Formally:

If $P(i+1) = +\Delta r{:}\widehat{T}$ then $\exists \overline{T_r}$,over$(\widehat{T}, \overline{T_r}), r{:}\overline{T_r} \in R$.

If $P(i+1) = -\Delta r{:}\widehat{T}$ then $\forall \overline{T_r}$,over$(\widehat{T}, \overline{T_r}), r{:}\overline{T_r} \notin R$.

We can now move to definite conclusion of temporal literals. We begin with literals temporalised by a chain of viewpoints $\overline{V}$, i.e. temporalised by $\overline{T}$ or $\overline{T}{:}\overline{T'}$.

If $P(i+1) = +\Delta_X \gamma{:}\overline{V}$ then
(1) $\exists \overline{T_\gamma}, \overline{V} \sqsubseteq \overline{T_\gamma}, X\gamma{:}\overline{T_\gamma} \in F$, or
(2) if $X =$ KNOW then $\exists \overline{T_\gamma}, \overline{V} \sqsubseteq \overline{T_\gamma}, \gamma{:}\overline{T_\gamma} \in F$, or
(3) if $X =$ INT then $\exists \overline{V_\gamma}, \overline{V} \sqsubseteq \overline{V_\gamma}, +\Delta_{\mathrm{ACT}}\gamma{:}\overline{V_\gamma}$, or
(4) $\exists r{:}\overline{T_r} \in R_s[\gamma : \overline{T_\gamma}], \overline{V} \sqsubseteq \overline{T_\gamma}{:}\overline{T_r}$, $r{:}\overline{T_r}$ is Δ-*applicable*, or
(5) $\exists \overline{V_{\gamma 1}}, \exists \overline{V_{\gamma 2}}$, meets$(\overline{V_{\gamma 1}}, \overline{V_{\gamma 2}})$, start$(\overline{V_{\gamma 1}})$ = start$(\overline{V})$, end$(\overline{V_{\gamma 2}})$ = end$(\widehat{V})$
$+\Delta_X \gamma{:}\overline{V_{\gamma 1}} \in P(1..i)$ and $+\Delta_X \gamma{:}\overline{V_{\gamma 2}} \in P(1..i)$.

To prove that a modal literal temporalised by a chain of viewpoints is not definitely provable we have to show that any attempt to give a definite proof fails.

If $P(i+1) = -\Delta_X \gamma{:}\overline{V}$ then
(1) $\forall \overline{T_\gamma}, \overline{V} \sqsubseteq \overline{T_\gamma}, X\gamma{:}\overline{T_\gamma} \notin F$, and
(2) if $X =$ KNOW then $\forall \overline{T_\gamma}, \overline{V} \sqsubseteq \overline{T_\gamma}, \gamma{:}\overline{T_\gamma} \notin F$, and
(3) if $X =$ INT then $\forall \overline{V_\gamma}, \overline{V} \sqsubseteq \overline{V_\gamma}, -\Delta_{\mathrm{ACT}}\gamma{:}\overline{V_\gamma}$, and
(4) $\forall r{:}\overline{T_r} \in R_s[\gamma : \overline{T_\gamma}], \overline{V} \sqsubseteq \overline{T_\gamma}{:}\overline{T_r}$, $r{:}\overline{T_r}$ is Δ-*discarded*, and
(5) $\forall \overline{V_{\gamma 1}}, \forall \overline{V_{\gamma 2}}$, meets$(\overline{V_{\gamma 1}}, \overline{V_{\gamma 2}})$, start$(\overline{V_{\gamma 1}})$ = start$(\overline{V})$, end$(\overline{V_{\gamma 2}})$ = end$(\widehat{V})$
$-\Delta_X \gamma{:}\overline{V_{\gamma 1}} \in P(1..i)$ or $-\Delta_X \gamma{:}\overline{V_{\gamma 2}} \in P(1..i)$.

The conditions for a temporal literal $\gamma{:}\widehat{V}$ (i.e. $\gamma{:}\widehat{T}$ or $\gamma{:}\widehat{T}{:}\widehat{T'}$) to be not definitely provable with modality X ($\pm\Delta_X$) are formally expressed below.

If $P(i+1) = +\Delta_X\gamma{:}\widehat{V}$ then $\exists\widehat{V_\gamma}, over(\widehat{V},\overline{V_\gamma}), +\Delta_X\gamma{:}\overline{V_\gamma}$.

If $P(i+1) = -\Delta_X\gamma{:}\widehat{V}$ then $\forall\widehat{V_\gamma}, over(\widehat{V},\overline{V_\gamma}), -\Delta_X\gamma{:}\overline{V_\gamma}$.

The definition of Δ-*applicable*, and Δ-*discarded* of rules contains the definite (un)provability of modal literals temporalised by chain of viewpoint of the tpye $\overline{T}{:}\widehat{T_r}$. We cater for such cases in the two next proof conditions.

If $P(i+1) = +\Delta_X\gamma{:}\overline{T}{:}\widehat{T_r}$ then
$\exists\overline{T_{\gamma 1}}, \exists\overline{T_{r1}}, \overline{T} \sqsubseteq \overline{T_{\gamma 1}}$, over$(\widehat{T_r}, \overline{T_{r1}}), +\Delta_X\gamma{:}\overline{T_{\gamma 1}}{:}\overline{T_{r1}}$ ∈P(1..i).

If $P(i+1) = -\Delta_X\gamma{:}\overline{T}{:}\widehat{T_r}$ then
$\forall\overline{T_{\gamma 1}}, \forall\overline{T_{r1}}, \overline{T} \sqsubseteq \overline{T_{\gamma 1}}$, over$(\widehat{T_r}, \overline{T_{r1}}), -\Delta_X\gamma{:}\overline{T_{\gamma 1}}{:}\overline{T_{r1}}$ ∈P(1..i).

We now turn our attention to defeasible derivations, that is, derivations giving a temporal assertion $\gamma{:}V$ as a defeasible conclusion of a theory D. We begin with the proof conditions to determine whether a rule is a defeasible conclusion.

If $P(i+1) = +\partial r{:}\overline{T}$ then $+\Delta r{:}\overline{T} \in P(1..i)$

If $P(i+1) = +\partial r{:}\widehat{T}$ then $+\Delta r{:}\widehat{T} \in P(1..i)$.

Defeasible provability ($+\partial$) for temporal literals consists of three phases. In the first phase, we put forward a supported reason for the temporal assertion that we want to prove. Then in the second phase, we consider all possible attacks against the desired conclusion. Finally in the last phase, we have to counter-attack the attacks considered in the second phase.

If $P(i+1) = +\partial_X\gamma{:}\overline{V}$ and Convert(Y,X) and Attack(W,Z,X) then
(1) $+\Delta_X\gamma{:}\overline{V} \in P(1..i)$, or
(2) $-\Delta_X{\sim}\gamma{:}\widehat{V} \in P(1..i)$, and
  (2.1) if $X = \text{INT}$ then $\exists\overline{V_\gamma}, \overline{V} \sqsubseteq \overline{V_\gamma}$, $+\partial_{\text{ACT}}\gamma{:}\overline{V_\gamma}$, or
  (2.2) $\exists r{:}\overline{T_r} \in R^{X\cup Y}_{sd}[\gamma{:}\overline{T_\gamma}]$, $\overline{V} \sqsubseteq \overline{T_\gamma}{:}\overline{T_r}$, $r{:}\overline{T_r}$ is ∂-applicable,
  (2.3) $\forall s{:}\overline{T_s} \in R^{W\cup Z\cup X\cup Y}[{\sim}\gamma{:}\overline{T_{\sim\gamma}}]$, si$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{T_\gamma}{:}\overline{T_r})$, sbe$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{V})$,
    (2.3.1) $s{:}\overline{T_s}$ is ∂-discarded, or
    (2.3.2) $\exists w{:}\overline{T_w} \in R^K[\gamma{:}\overline{T_{w\gamma}}]$, $\overline{V} \sqsubseteq \overline{T_{w\gamma}}{:}\overline{T_w}$,
      (2.3.2.1) $w{:}\overline{T_w}$ is $+\partial$-applicable, and either
      (2.3.2.2) $s{:}\overline{T_s} \in R^{X\cup Y}$,
        (2.3.2.2.1) $w{:}\overline{T_w} \in R^{W\cup Z}$, or
        (2.3.2.2.2) $w{:}\overline{T_w} \in R^{X\cup Y}$, $w{:}\overline{T_w} \succ s{:}\overline{T_s}$, or
      (2.3.2.3) $s{:}\overline{T_s} \in R^Z$,
        (2.3.2.3.1) $w{:}\overline{T_w} \in R^W$, or
        (2.3.2.3.2) $w{:}\overline{T_w} \in R^Z$, $w{:}\overline{T_w} \succ s{:}\overline{T_s}$, or
      (2.3.2.4) $s{:}\overline{T_s} \in R^W$, $w{:}\overline{T_w} \in R^W$, $w{:}\overline{T_w} \succ s{:}\overline{T_s}$, or
(3) $\exists\overline{V_{\gamma 1}}, \exists\overline{V_{\gamma 2}}$, meets$(\overline{V_{\gamma 1}}, \overline{V_{\gamma 2}})$, start$(\overline{V_{\gamma 1}})$ = start$(\overline{V})$, end$(\overline{V_{\gamma 2}})$ = end$(\widehat{V})$
$+\partial_X\gamma{:}\overline{V_{\gamma 1}} \in P(1..i)$ and $+\partial_X\gamma{:}\overline{V_{\gamma 2}} \in P(1..i)$.

If $P(i+1) = -\partial_X \gamma{:}\overline{V}$ and Convert(Y,X) and Attack(W,Z,X) then

(1) $-\Delta_X \gamma{:}\overline{V} \in P(1..i)$, and

(2) $+\Delta_X {\sim}\gamma{:}\widehat{V} \in P(1..i)$, or

(2.1) if $X = \mathrm{INT}$ then $\forall \overline{V_\gamma}, \overline{V} \sqsubseteq \overline{V_\gamma}$, $-\partial_{\mathrm{ACT}} \gamma{:}\overline{V_\gamma}$, and

(2.2) $\forall r{:}\overline{T_r} \in R_{sd}^{X \cup Y}[\gamma{:}\overline{T_\gamma}]$, $\overline{V} \sqsubseteq \overline{T_\gamma}{:}\overline{T_r}$, $r{:}\overline{T_r}$ is ∂-applicable,

(2.3) $\exists s{:}\overline{T_s} \in R^{W \cup Z \cup X \cup Y}[{\sim}\gamma{:}\overline{T_{\sim\gamma}}]$, si$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{T_\gamma}{:}\overline{T_r})$, sbe$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{V})$,

(2.3.1) $s{:}\overline{T_s}$ is ∂-applicable, and

(2.3.2) $\forall w{:}\overline{T_w} \in R[\gamma{:}\overline{T_{w\gamma}}]$, $\overline{T} \sqsubseteq \overline{T_{w\gamma}}{:}\overline{T_w}$, either

(2.3.2.1) $w{:}\overline{T_w}$ is ∂-discarded, or

(2.3.2.2) $s{:}\overline{T_s} \in R^{X \cup Y}$,

(2.3.2.2.1) $w{:}\overline{T_w} \notin R^{W \cup Z}$, and

(2.3.2.2.2) $w{:}\overline{T_w} \in R^{X \cup Y}$, $w{:}\overline{T_w} \not> s{:}\overline{T_s}$, and

(2.3.2.3) $s{:}\overline{T_s} \in R^Z$

(2.3.2.3.1) $w{:}\overline{T_w} \notin R^W$, and

(2.3.2.3.2) $w{:}\overline{T_w} \in R^Z$, $w{:}\overline{T_w} \not> s{:}\overline{T_s}$, and

(2.3.2.4) $s{:}\overline{T_s} \in R^W$, $w{:}\overline{T_w} \in R^W$, $w{:}\overline{T_w} \not> s{:}\overline{T_s}$, and

(3) $\forall \overline{V_{\gamma 1}}, \forall \overline{V_{\gamma 2}}$, meets$(\overline{V_{\gamma 1}}, \overline{V_{\gamma 2}})$, start$(\overline{V_{\gamma 1}})$ = start$(\overline{V})$, end$(\overline{V_{\gamma 2}})$ = end$(\widehat{V})$
$-\partial_X \gamma{:}\overline{V_{\gamma 1}} \in P(1..i)$ and $-\partial_X \gamma{:}\overline{V_{\gamma 2}} \in P(1..i)$

Let us illustrate the proof condition of the defeasible provability of X$\gamma{:}\overline{V}$. We have two cases: 1) We show that X$\gamma{:}\overline{V}$ is already definitely provable; or 2) we need to argue using the defeasible part of D. In this second case, to prove X$\gamma{:}\overline{V}$ defeasibly we must show that X${\sim}\gamma{:}\widehat{V}$ is not definitely provable (2). We require then there must be a strict or defeasible rule $r{:}\overline{T_r} \in R^{X \cup Y}$ which can be applied and with head $\gamma{:}\overline{T_\gamma}$ such that $\overline{V} \sqsubseteq \overline{T_\gamma}{:}\overline{T_r}$ (2.1). But now we need to consider possible attacks, i.e., reasoning chains in support of ${\sim}\gamma{:}\overline{V}$, that is, any rule $s{:}\overline{T_s} \in R^{W \cup Z \cup X \cup Y}$ which has head ${\sim}\gamma{:}\overline{T_{\sim\gamma}}$ such that si$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{T_\gamma}{:}\overline{T_r})$, and sbe$(\overline{T_{\sim\gamma}}{:}\overline{T_s}, \overline{V})$. Note that here we consider defeaters, too, whereas they could not be used to support the conclusion X$\gamma{:}\overline{V}$; this is in line with the motivation of defeaters given earlier. These attacking rules $s{:}T_s$ have to be discarded (2.3.1), or must be counterattacked by a stronger rule $w{:}T_w$ which has a head $\gamma{:}T_{w\gamma}$ such that $\overline{V}$ is contained in $\overline{T_{w\gamma}}{:}\overline{T_w}$ (2.3.2). Finally, we have to cater for the case where Xγ is defeasible provable on $\overline{V_{\gamma 1}}$ and $\overline{V_{\gamma 2}}$ that make up $\overline{V}$ (3).

The defeasible proof for a temporal literal to hold in some instants of a chain of viewpoints $\widehat{V}$ is less demanding since it is sufficient to prove it for at least an instant in $\overline{V}$.

If $P(i+1) = +\partial_X \gamma{:}\widehat{V}$ and Convert(Y,X) and Attack(W,Z,X) then
$\exists \overline{V_\gamma}$,over$(\widehat{V}, \overline{V_\gamma}), +\partial_X \gamma{:}\overline{V_\gamma}$ ∈P(1..i).

If $P(i+1) = -\partial_X \gamma{:}\widehat{V}$ and Convert(Y,X) and Attack(W,Z,X) then
$\forall \overline{V_\gamma}$,over$(\widehat{V}, \overline{V_\gamma}), -\partial_X \gamma{:}\overline{V_\gamma}$ ∈P(1..i).

Similarly to definite conclusions, the definition of *∂-applicable*, and *∂-discarded* of rules contains the defeasible (un)provability of modal literals temporalised by a chain of viewpoints of the type $\overline{T}{:}\widehat{T_r}$. We cater for such cases by these two finals proof conditions.

If $P(i+1) = +\partial_X\gamma{:}\overline{T}{:}\widehat{T}_r$ then
$\exists\overline{T_{\gamma 1}}, \exists\overline{T_{r1}}, \overline{T} \sqsubseteq \overline{T_{\gamma 1}}$, over$(\widehat{T}_r, \overline{T_{r1}})$, $+\partial_X\gamma{:}\overline{T_{\gamma 1}}{:}\overline{T_{r1}}$ $\in$P(1..i).

If $P(i+1) = -\partial_X\gamma{:}\overline{T}{:}\widehat{T}_r$ then
$\forall\overline{T_{\gamma 1}}, \forall\overline{T_{r1}}, \overline{T} \sqsubseteq \overline{T_{\gamma 1}}$, over$(\widehat{T}_r, \overline{T_{r1}})$, $-\partial_X\gamma{:}\overline{T_{\gamma 1}}{:}\overline{T_{r1}}$ $\in$P(1..i).

Proof conditions for modal literals temporalised by chain of viewpoints of the type $\widehat{T}{:}\overline{T_r}$ are nor presented here but follows similar schema.

5 Conclusions

In this paper we extended the logic presented in [1] with temporal intervals in order to express its expressive power. Doing so, we have extended the programming cognitive agents approach with modal literals and rules temporalised with intervals. This makes the resulting logic more expressive and more suitable for the task at hand. In addition we have considered the notion of viewpoint. The deliberation of an agent based on a policy depends not only on the environment but also on the rules in force in the policy at the time of deliberation and at the time when the plan resulting from the deliberation will be executed. These two aspects are neglected in the literature on agent planning. An aspect we did not consider here is how revise theories in the same way as complex modification of normative codes [11].

References

1. Governatori, G., Padmanabhan, V., Rotolo, A.: Rule-based agents in temporalised defeasible logic. In Yang, Q., Webb, G., eds.: Ninth Pacific Rim International Conference on Artificial Intelligence. Number 4099 in LNAI, Berlin, Springer (2006) 31–40
2. Dastani, M., van der Torre, L.W.N.: Programming boid-plan agents: Deliberating about conflicts among defeasible mental attitudes and plans. In: AAMAS. (2004) 706–713
3. Broersen, J., Dastani, M., Hulstijn, J., van der Torre, L.: Goal generation in the BOID architecture. Cognitive Science Quarterly **2** (2002) 428–447
4. Dastani, M., de Boer, F., Dignum, F., Meyer, J.J.: Programming agent deliberation: an approach illustrated using the 3apl language. In: AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems, New York, NY, USA, ACM Press (2003) 97–104
5. Dastani, M., van Riemsdijk, B., Dignum, F., Meyer, J.: A programming language for cognitive agents: Goal-directed 3apl. In: PROMAS. (2003) 111–130
6. van Riemsdijk, M.B., Dastani, M., Meyer, J.J.C.: Semantics of declarative goals in agent programming. In: AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems, New York, NY, USA, ACM Press (2005) 133–140
7. Governatori, G., Rotolo, A.: Defeasible logic: Agency, intention and obligation. In Lomuscio, A., Nute, D., eds.: Deontic Logic in Computer Science. Number 3065 in LNAI, Berlin, Springer-Verlag (2004) 114–128
8. Dastani, M., Governatori, G., Rotolo, A., van der Torre, L.: Programming cognitive agents in defeasible logic. In Sutcliffe, G., Voronkov, A., eds.: Proc. LPAR 2005. Volume 3835 of LNAI., Springer (2005) 621–636

9. Dastani, M., Governatori, G., Rotolo, A., van der Torre, L.: Preferences of agents in defeasible logic. In Zhang, S., Jarvis, R., eds.: Proc. Australian AI05. Volume 3809 of LNAI., Springer (2005) 695–704
10. Governatori, G., Rotolo, A., Sartor, G.: Temporalised normative positions in defeasible logic. In Gardner, A., ed.: 10th International Conference on Artificial Intelligence and Law (ICAIL05), ACM Press (2005) 25–34
11. Governatori, G., Palmirani, M., Riveret, R., Rotolo, A., Sartor, G.: Norm modifications in defeasible logic. In Moens, M.F., Spyns, P., eds.: Legal Knowledge and Information Systems. Number 134 in Frontieres in Artificial Intelligence and Applications. IOS Press, Amsterdam (2005) 13–22
12. Bratman, M.E.: Intentions, Plans and Practical Reason. Harvard University Press, Cambridge, MA (1987)
13. Palmirani, M., Brighi, R.: Time model for managing the dynamic of normative system. In: Proceedings of EGOV 2006, Berlin, Springer (2006) 207–218
14. Nute, D.: Defeasible reasoning. In: Proceedings of 20th Hawaii International Conference on System Science, IEEE press (1987)
15. Nute, D.: Defeasible logic. In Gabbay, D., Hogger, C., Robinson, J., eds.: Handbook of Logic in Artificial Intelligence and Logic Programming. Volume 3. Oxford University Press (1993) 353–395
16. Antoniou, G., Billington, D., Governatori, G., Maher, M.J.: Representation results for defeasible logic. ACM Transactions on Computational Logic **2** (2001) 255–287
17. Governatori, G., Maher, M.J., Billington, D., Antoniou, G.: Argumentation semantics for defeasible logics. Journal of Logic and Computation **14** (2004) 675–702
18. Maher, M.J.: Propositional defeasible logic has linear complexity. Theory and Practice of Logic Programming **1** (2001) 691–711
19. Bassiliades, N., Antoniou, G., Vlahavas, I.: DR-DEVICE: A defeasible logic system for the Semantic Web. In Ohlbach, H.J., Schaffert, S., eds.: 2nd Workshop on Principles and Practice of Semantic Web Reasoning. Number 3208 in LNCS, Springer (2004) 134–148
20. Allen, J.F.: Towards a general theory of action and time. Artif. Intell. **23** (1984) 123–154

Normative Multi-Agent Organizations: Modeling, Support and Control. Draft Version

Benjamin Gâteau[1],[2] and Olivier Boissier[2]

[1] CITI/CRP Henri Tudor
29 Av. John F. Kennedy L-1855 Luxembourg – G.-D. of LUXEMBOURG
{benjamin.gateau}@tudor.lu
[2] SMA/G2I/ENSM Saint-Etienne
158, Cours Fauriel F-42023 Saint-Etienne Cedex 02 – FRANCE
olivier.boissier@emse.fr

Abstract. In the last years, social and organizational aspects of agency have become a major issue in multi-agent systems' research. Recent applications of MAS enforce the need of using these aspects in order to ensure some social order within these systems. Tools to control and regulate the overall functioning of the system are needed in order to enforce global laws on the autonomous agents operating in it. This paper presents a normative organization system composed of a normative organization modeling language $\mathcal{M}\text{OISE}^{Inst}$ used to define the normative organization of a MAS, accompanied with $\mathcal{S}\text{YNAI}$, a normative organization implementation architecture which is itself regulated with an explicit normative organization specification.

1 Introduction

Nowadays, current IT applications show the large scale interweaving of human and technological communities (e.g. Web Intelligence, Ambient Intelligence, Interactive TV). Using Multi-Agent System (MAS) technology introduces software entities that act on behalf of users and cooperate with those infohabitants. The complex system engineering's approach needed to build such applications highlights and stresses requirements on *openness* in terms of ability to take into account several kinds of changes and to adapt the system configuration while it keeps running [1]. As stated in [2], "Openness without control may lead to chaotic behavior". Being composed of heterogeneous and autonomous agents, tools to control and regulate the overall functioning of the system are required in order to enforce global laws on the autonomous agents operating in the system.

In this paper we present a multi-agent normative organization environment composed of $\mathcal{S}\text{YNAI}$, multiagent organization infrastructure, interpreting normative declarative organizations programmed with $\mathcal{M}\text{OISE}^{Inst}$, a normative organization modeling language. $\mathcal{M}\text{OISE}^{Inst}$ is an extension of the $\mathcal{M}\text{OISE}^{+}$ developped by [3]. $\mathcal{S}\text{YNAI}$ is composed of generic *supervisor* agents, aiming at controlling and enforcing the rights and duties of autonomous "domain" agents operating in an normative organization expressed with $\mathcal{M}\text{OISE}^{Inst}$ ($\mathcal{M}\text{OISE}^{Inst}$ extending $\mathcal{M}\text{OISE}^{+}$, $\mathcal{S}\text{YNAI}$ extends

$\mathcal{S}$-$\mathcal{M}$OISE$^+$ [4]). Whereas supervisor agents are dedicated to the control of the system, the domain agents implement the functionalities of the application. $\mathcal{M}$OISE^{Inst} is also used at a meta-level since the supervisor agents themselves operate under the control of a normative organization that structures and constrains their control behaviours on the domain agents. All along the paper, we illustrate the use of this environment with an iTV game issued from the European ITEA Jules Verne Project.

This paper is organised as follows: section 2 gives a global overview of our framework for defining normative multi-agent organizations. Its use is illustrated with an iTV application. The succeeding sections presents the organization modeling language $\mathcal{M}$OISE^{Inst} and then the infrastructure supporting it, $\mathcal{S}$YNAI. Finally, before concluding, section 5 positions our work with respect to other approaches.

2 Global view

In the recent past, multiagent technologies have been developed and deployed in different applications. Most of these efforts have been largely supported by the existence of multiagent platforms like JADE [5] or FIPA-OS [6]. These platforms have demonstrated the needs and utility of generic services for supporting the execution of multi-agent applications like for instance Agent Management System, Directory Facilitator. The recent developments in the domain (e.g. electronic commerce [7]) have shown the requirement to enrich those services to provide multiagent applications with what we call organization oriented programming [8]. Such an approach provides the possibility to express and make explicit one or more patterns of cooperation installed in a top-down approach on the agents, that constrains and drive their actions and interactions towards some purpose. Current multiagent approaches on normative organizations [9] propose to enrich those patterns of cooperation with the explicit modelling of rules stating the norms directing the functioning of the system. Agents interpret these norms and are enforced to comply with their specified behaviours. However, agents can still practise organizational autonomy, in the sense that they are able to read, to represent, and to reason about the organization and may decide whether to follow the constraints stated by the organization or not. They may also decide to adapt and change the organization in a bottom-up process, installing a new pattern/structure. Such a functioning corresponds to the combination of agent-centred organized MAS and organization oriented MAS approaches [8]

Considering the programming of normative organizations has led us to introduce norms in the $\mathcal{M}$OISE^{Inst} *organization modeling language* (OML) used to *define* the organization(s) of an MAS. It is used to collect and express specific constraints and cooperation patterns that the designer (or the agents) have in mind, resulting in an explicit representation that we call *organization specification* (OS). Finally the OS is executed and interpreted on an *Organization Implementation Architecture* (OIA) to install a collective entity in the MAS that we call *Organization Entity* (OE): a set of agents building the organization specified with an OS. Once created, the OE's history starts and runs by events like other agents entering and/or leaving it, group creation, role adoption, goal commitment, etc. The OIA may be further split into an agent part (such as, for instance, in [10]) and into an organization infrastructure part, the $\mathcal{S}$YNAI system. Implied by the introduction of the normative dimension in the OML, $\mathcal{S}$YNAI has been enriched with different mechanisms to deal with it.

Let's illustrate this sketch of a normative system with our Interactive Games application (see Fig. 1): a "questions – answers" TV game show opposing a real players' team present on the TV scene, to a televiewers' team interacting from home into the game with the help of avatars, i.e. the domain agents. Each avatar is under the control of its respective televiewer. The quizmaster is also supported by a virtual assistant, aiming at regulating the game. As in all collective games, the aim is to promote a collective behaviour among the players of a same team. The OS defined with the $\mathcal{M}$OISE^{Inst} organization modeling language states the structure and functioning of the game, with a set of *norms* defining the game rules, the sanctions and rewards in use during the game. However, since avatars are autonomous agents, they can be autonomous with respect to these constraints, e.g. a televiewer is able to decide to answer whereas it is not his turn and to take the risk to be punished. The OIA has been defined with $\mathcal{S}$YNAI as a *normative system* in order to control, regulate and reward/punish avatars when they respect or not the OS. Supervisor agents of the OIA are dedicated to the management of the organization and to the enforcement of the game rules on the avatars. Both kinds of agents (supervisor and domain) are organised and constrained according to the OS defined with the $\mathcal{M}$OISE^{Inst} normative OML [11]. Agents are thus able to reason on the organization and constraints. They have the possibility to decide to take it into account or not. The OIA reads this specification in order to supervise and control the agents as well as to be informed about its own organization specification.

3 Normative Organization Modeling Language

$\mathcal{M}$OISE^{Inst} [11] is used to define what we call an organization specification (OS) with the help of four dimensions[1]: structural specification (SS), functional specification (FS), contextual specification (CS) and normative specification (NS).

3.1 Structural specification

The *structural specification* (SS) defines the MAS structure with the notions of *roles*, *groups* and *links*. A *role* consists in a label to which constraints on the playing agents' behavior. Roles are also used as anchors to the links. A *group specification* consists in a set of links and roles. The Fig. 2 shows the structural specification of the iTV application: a "Team" group is composed of the roles corresponding to the expertises mobilized for the game ("History", "Geo", "Sport", "Science") with a special role "Chief". These roles are specialization – inheritance link – of "BasicPlayer" or "Player" abstract roles, i.e. roles which cannot be played by agents. All roles inherit of the abstract root role "Soc". Well formed attributes may be ascribed to groups. They concern intra/extra group compatibility of roles among them, minimum and maximum number of role players inside a group, minimum and maximum number of subgroups. *Cardinality* and *compatibility links* express constraints on the way agents play roles in groups. For

[1] Formal definitions of SS and FS are available in [12], CS and NS in [11]

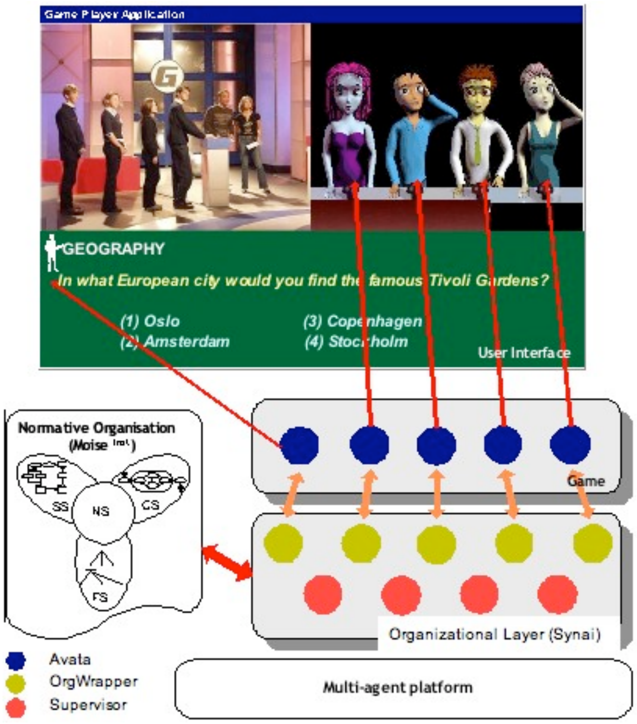


Fig. 1. Global view of normative organization environment for iTV application.

instance, cardinality ‘1..1’ on the composition link ensures that, in an group instance of a “Team”, roles can be adopted by only one agent at the same time. A compatibility link between “BasicPlayer” and “Chief”, allows the same agent to play those two roles or specialization of those roles. Thus, according to this specification, one agent may have the possibility to play at most two of those five roles. *Links* have direct effect on the agents’ behavior. They can be: *acquaintance* links (i.e. agents playing the source role are allowed to have a representation of the agents playing the destination role), *communication* links (i.e. agents are allowed to communicate with the target agents), *authority* links (i.e. source agents are allowed to control target agents). For instance, all roles inheriting from “Player” can communicate between them, and the “Chief” has the authority on all “BasicPlayer”. It means that all roles inheriting from this role are under the authority of the “Chief”.

3.2 Functional specification

In the *functional specification* (FS), goals that are to be achieved by the organization are structured according to different *social schemes*. A social scheme is a goal decomposition tree where the root is the Scheme’s goal. The operators that may appear in these

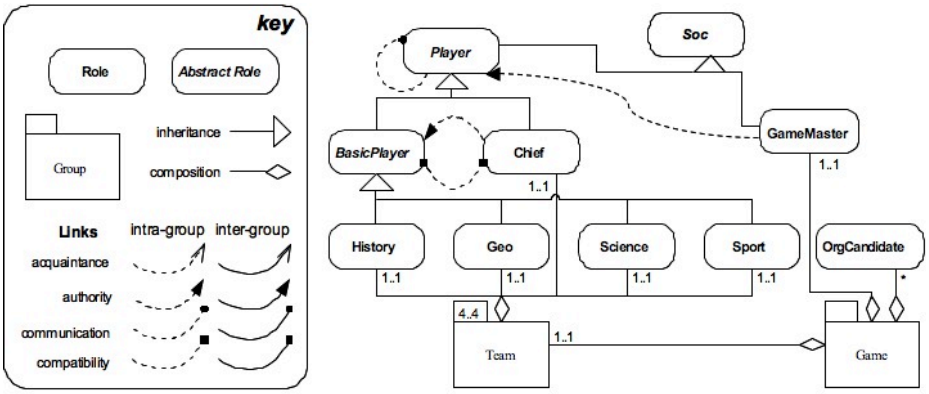


Fig. 2. Structural specification for the domain agents of the iTV application.

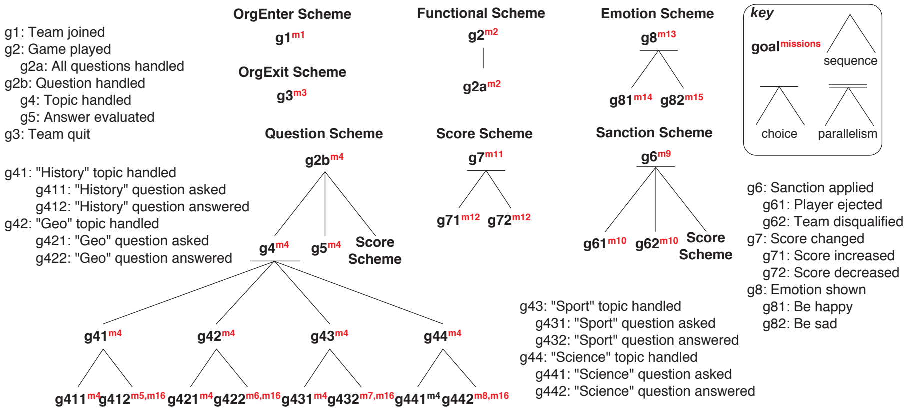


Fig. 3. Functional Specification of the organization for iTV application

plans express the execution in sequence/parallelism and the possibility of choice. All the goals of a social scheme (root goal and subgoals) are structured into *missions*: set of coherent goals that are to be assigned to roles and that an agent can commit to. More precisely, if an agent accepts a mission m_i, it commits to all goals of m_i ($g_j \in m_i$) and the agent will try to achieve a g_j goal only when the goal precondition for g_j is satisfied. In Fig. 3, the main social scheme has a goal "X pts scored" that can be satisfied by the achievement in sequence of "g4", "g5" and of the goals of the "Score Scheme". The "Emotion Scheme" deals with the specification of the emotional behaviour of avatars as: to show either an happy face or a sad one. The "OrgEnter Scheme" (resp. "OrgExit Scheme") defines the principal behaviours for entering (resp. leaving) an organization. We also define a scheme dedicated to sanctions which has to be considered by the supervisor agents (see below). For instance, a sanction consists in a choice between the ejection of a player, the disqualification of the team or the modification of the score.

3.3 Contextual specification

To tackle with the situatedness of applications in evolving environment, a *contextual specification* (CS) captures design-time a priori constraints on the evolution of the organization as a set of contexts and transitions between them (cf. Figure 4).

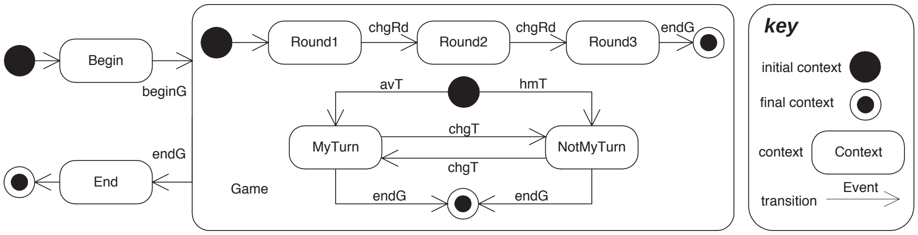


Fig. 4. Contextual Specification for the organization of the iTV application.

A context expresses a state in which agents playing role have to respect specific norms. Transitions define change from one context to an other context given the occurrence of different events. For instance, in the iTV application, the CS is used to express the different rounds of the game that impose changes to the enacted rules. Here the CS starts with a synchronous state "Begin" which allows the televiewer to connect to the system. A macro-context "Game" is decomposed into three rounds sub-contexts. This global context will be used to define the basic rules of the game while the three round sub-contexts will be used to define the corresponding specific rules. The "Game" context is also decomposed into two sub-contexts defining the turn of the players. A round sub-context and a turn sub-context can be active at the same time. Let's notice that the macro-context is active in all of its sub-contexts. The rules defined in the "Game" context are thus inherited in sub-contexts where they keep their status. Finally the last state is the context in which Avatars quit their team.

3.4 Normative specification

Finally, the *normative specification* (NS) glues all specifications (SS, FS and CS) in a coherent and normative organization with the help of norms (see Fig. 5). In $\mathcal{M}\text{OISE}^{Inst}$, norms define rights (i.e. *permission*), duties (i.e. *obligation*, *prohibition*) for agents while playing a role or being member of a group, to execute a *mission* in a particular *context* and during a given *time*. The fulfillment of a norm is supervised by an *issuer* which can apply a *sanction* on the *bearer* of the norm. Norms are represented as the following expression (φ, $context$, $sanction$, $weight$ and time constraint tc are optional):

$$norm : \varphi \rightarrow op(context, issuer, bearer, mission, sanction, weight, tc)$$

φ and $context$ refer respectively to the validity and activation conditions (see below), considered as true if not specified in the expression. *op* is a deontic operator

($op \in \{O, P, F\}$) which defines a norm as an obligation (O), a permission (P) or a prohibition (F). These operators concern the $mission$ expressed in the FS. Missions that are not prohibited or obliged are considered as permitted. The normative expressions don't refer directly to agents but to groups or roles of the organization in which agents are situated (fields $issuer$, $bearer$). This way, expressions are independent of the kinds of agents that could populate the system at one time. The $issuer$ refers to the role or group that check the status of the norm (fulfilled, violated), whereas the $bearer$ refers to the structural entities on which the norm is applied. Users who specify their own application modelling don't know how the supervision of the normative organization works. That's why they have to set the issuer up to "Supervisor" role (root role of the supervision SS, see below). The $\mathcal{S}$YNAI layer decides automatically what agents supervise what norms. Composition and inheritance that are defined in the SS among groups and roles have consequence on norms:

- When the $bearer$ (resp. $issuer$) is a group, all roles contained in this group, are considered as $bearer$ (resp. $issuer$). of this norm. For instance, the prohibition for the "Team" group to answer a question when it is not its turn, is applied on all the roles being part of this group ("History", "Science", "Geo", "Sport", "Chief").
- If the $bearer/issuer$ of a norm is a role r all roles inheriting from r are also concerned by the norm as $beared/issuer$. For instance, if a norm obliges the role "Player" to answer a question, all the inheriting roles are obliged to answer a question ("BasicPlayer", "Chief", ...).
- If the $bearer/issuer$ of the norm is a group gt then all sub-groups composing gt are concerned by the norm. For instance, if a norm concerns the "Game" group, the norm concerns also the "Team" group. As a consequence, if a norm concerns "Game" and "Team" groups, it concerns also roles belonging to both groups i.e. "History", "Science", "Geo", "Sport", "Chief", "GameMaster" and "OrgCandidate".

A $sanction$ is another norm appearing in the NS that is considered as a "sanction" to apply in case of norm violation[2]. The $weight$ defines a priority used for solving conflicts between norms in case of incoherence, when for instance an agent could be constrained by two contradictory norms[3] (e.g. $N9$ and $N14$ in Fig. 5). 1 is the highest priority.

A norm is *active* when the $context$ referred in the norm equals the current state specified in the CS. As a context can be composed of sub-contexts, if a norm is active in a context then it is also active in sub-contexts. For instance, if a norm is considered active as soon as the OE's state is equal to "Game", the norm will be considered active when the state of the organization will be either in the "Round1", "Round2", "Round3", "MyTurn" or "NotMyTurn" contexts. A norm is *valid* as long as its condition φ is satisfied. φ is the condition that defines the particular state of the OE in which the norm may be valid. As long as φ is satisfied, the norm stays valid. A norm condition could

[2] If a norm id specifies a $sanction$, then the condition of the $sanction$ contains the predicate $violated(id)$.

[3] Even, if this field is not satisfactory in case of two norms having the same weight, it was sufficient in our application. Future works will have to consider this issue.

be a conjunction/disjunction of sub-conditions. A primitive condition consists in one of the following expression:

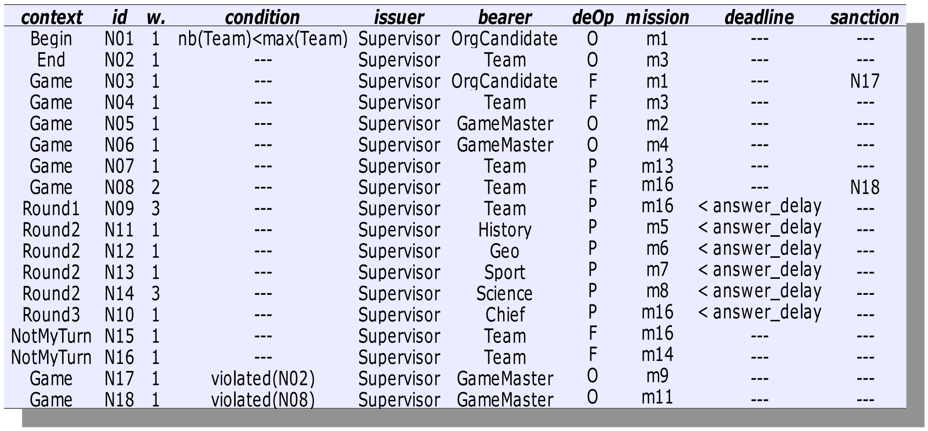
- an application-dependant predicate (e.g. *sad* or *happy* which test if an avatar shows a sad or happy face);
- a predicate related to the life cycle of the organization such as *number* or *cardinalityMax* which respectively access the number of agents being part of a group and the maximum number of agents that a group may accept;
- a predicate related to the functioning of the normative organization itself such as *violated* which tests if the norm is violated.

A norm can be *fulfilled* or *violated* from the moment it is active *and* valid. The violation detection depends on the deontic operator op and on deadline tc -date or period- appearing in the expression of the norm. If no tc is specified, the end of the $context$ is considered by default.

- An obligation states that the $mission$ ought to be accomplished by the $bearer$ before, after or during a deadline expressed in tc - date or period -. The norm is considered fulfilled if the mission is accomplished by the bearer in term else violated.
- A prohibition hinders the norm's $bearer$ to accomplish the $mission$ in a tc*. deadline*. Contrary to an obligation, a prohibition is considered fulfilled as long as the mission is not accomplished by the bearer until the deadline is over, violated in the other case.
- A permission authorizes the $bearer$ of the norm to accomplish the $mission$ in a tc. In an organization specified with $\mathcal{M}$OISE^{Inst}, agents don't restrict their action to what is authorized or obliged but all that they are able to do and that is not prohibited.

In the iTV application, norms are used to define game rules as well as what happens before and after the game. For instance, norms $N01$ to $N04$ are related to the management of the organization: constraints on when it's possible to join/quit the team. $N01$ states that *any agent playing the "OrgCandidate" role is obliged to join a team (instance of "Team" group) in case there is still a role to play in this team* (condition $nb(Team) < max(Team)$ composed of two functions representing the number of agents already in the Team group and the maximum of agents allowed in the Team). According to the $context$ field, this norm is active as long as the OE is in the "Begin" context. The norm $N02$ manages the end of the game: *any agent playing a role in the "Team" group is obliged to quit the team (instance of "Team" group) when the organization is in the "End" context*. Moreover (see $NO3$ and $NO4$) in the "Game" context, agents playing the "OrgCandidate" role are forbidden to join a team and agents playing a role in the "Team" group are forbidden to quit the team. $N03$ has a sanction which is expressed as the norm $N17$: *in case of violation of N03, any agent playing the "GameMaster" role has to eject the agent playing the "OrgCandidate" role*. Let us notice that the mission expressed in this normative expression refers to a mission expressed in the "Sanction" scheme of the FS.

Other norms define the rules of the game and constrain its performance. For instance, according to $N05$ and $N06$, as long as the OE is in the "Game" context: *any*



context	id	w.	condition	issuer	bearer	deOp	mission	deadline	sanction
Begin	N01	1	nb(Team)<max(Team)	Supervisor	OrgCandidate	O	m1	---	---
End	N02	1	---	Supervisor	Team	O	m3	---	---
Game	N03	1	---	Supervisor	OrgCandidate	F	m1	---	N17
Game	N04	1	---	Supervisor	Team	F	m3	---	---
Game	N05	1	---	Supervisor	GameMaster	O	m2	---	---
Game	N06	1	---	Supervisor	GameMaster	O	m4	---	---
Game	N07	1	---	Supervisor	Team	P	m13	---	---
Game	N08	2	---	Supervisor	Team	F	m16	---	N18
Round1	N09	3	---	Supervisor	Team	P	m16	< answer_delay	---
Round2	N11	1	---	Supervisor	History	P	m5	< answer_delay	---
Round2	N12	1	---	Supervisor	Geo	P	m6	< answer_delay	---
Round2	N13	1	---	Supervisor	Sport	P	m7	< answer_delay	---
Round2	N14	3	---	Supervisor	Science	P	m8	< answer_delay	---
Round3	N10	1	---	Supervisor	Chief	P	m16	< answer_delay	---
NotMyTurn	N15	1	---	Supervisor	Team	F	m16	---	---
NotMyTurn	N16	1	---	Supervisor	Team	F	m14	---	---
Game	N17	1	violated(N02)	Supervisor	GameMaster	O	m9	---	---
Game	N18	1	violated(N08)	Supervisor	GameMaster	O	m11	---	---

Fig. 5. Normative Specification of the organization of the iTV application. Column "context" refers to the states defined in CS, column "w" contains the weight of the norms, columns "issuer" and "bearer" refer to roles and groups defined in SS, column "deOp" contains op, column "mission" contains the missions id specified in FS, column "sanction" refers to the id of norms.

agent playing the "GameMaster" role is obliged to ask question and to evaluate the answer (see missions $m2$ and $m4$ in Functional Scheme). According to $N07$, *any agent playing a role belonging to the "Team" group is forbidden to answer a question during the game*. Exceptions to this prohibition are set by defining specific norms in the context of the different rounds occurring during the game: when OE is in the first and third rounds, $N09$ and $N10$ permit any agent playing respectively a role belonging to the "Team" group and the role "Chief" to answer all questions during the answer delay. When the Organization is in the second round, norms $N11$, $N12$, $N13$, $N14$ allow concerned roles to answer question. Exceptions are expressed by defining for same context, role and mission a different priority in the $weight$.

Finally, norms $N15$ and $N16$ forbid the team to answer a question or to show an happy face when the OE is in the "NotMyTurn" context (i.e. the question is asked to the opponent team).

4 Normative Organizational Layer

While the previous section was concerned with the presentation of $\mathcal{M}\text{OISE}^{Inst}$, normative OML, illustrated with the OS installed on the domain agents of the application, this section deals with the issues related to their support into $\mathcal{S}\text{YNAI}$, normative Organization Implementation Architecture. As it happens with organizational models [8], implementations can also take either an agent centred or an system centred point of view [4] (in [13] these points of view are called agent and institutional perspectives). In the former point of view, the focus is on how to develop agent reasoning mechanisms to interpret and reason on the OS and OE. In the latter, the main concern is how to develop an Organization Infrastructure (OI) that ensures the satisfaction of the organizational constraints and norms (e.g. agents playing the right roles, committing to the allowed missions). This point of view is important in heterogeneous and open systems where the agents that enter into the system can have unknown architectures. Of course, to develop the overall MAS, the former point of view is necessary since the agents probably need to have access to an organizational representation that enable them to reason about it. However, the agents should follow the OS despite their organizational reasoning abilities.

Many implementations of the OI follow the general architecture depicted in Fig. 6. Domain agents are responsible to achieve organizational goals and use an *organizational proxy* component to interact with the organization (OS and OE). The *organizational layer* is responsible to bind all agents in a coherent system and provides some services for them. The communication layer is responsible for connecting all components of the infrastructure in a distributed and heterogeneous applications.

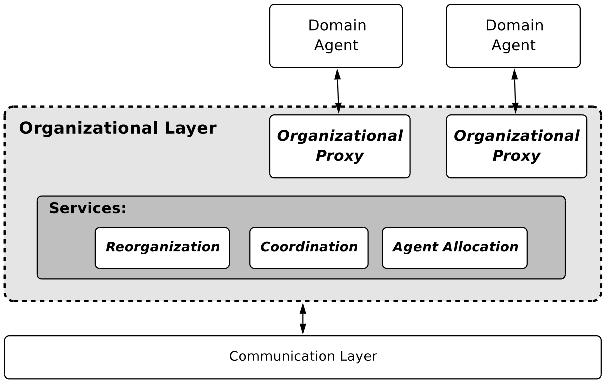


Fig. 6. Common Organization Implementation Architecture for open MAS

4.1 $\mathcal{S}$YNAI

Domain agents playing the game evolve in the OE resulting of the OS specified by the designer with the OML $\mathcal{M}$OISE^{Inst}. The OE consists in the current states of SS (roles played by agents, existing instances of groups), FS (current committed / executed / waiting missions and goals), CS (current executed/active states) and NS (current active / valid / fulfilled / violated norms). Being autonomous (under the control of a user),

[4] We prefer here system-centred to organization-centred in order to avoid confusion even if, as we have seen, the organization is reified in OE

avatars can decide to not respect the constraints stated in the OS: adopting a role in the OE which is not authorized in the OS, violating a norm, ...).

$\mathcal{S}$YNAI aims at managing and controlling the functioning of this OE by the way of different events corresponding to the entry/exit of agents of the OE, adoption/leaving roles, change of context, commitment to missions, achievement of goals, etc. Receiving requests from agents, it detects if they violate or not constraints stated in the SS and the NS (cf. Fig. 7). For instance it verifies that an agent plays compatible roles or that it is authorized to commit on mission according to the role it is playing and to the current active and valid norms.

$\mathcal{S}$YNAI is composed of a set of different supervisor agents for the management of each entity deriving from the specification of the OS: *StructManagerAg* for the SS entity, *FunctManagerAg* for the FS entity, *ContextManagerAg* for the CS entity and *NormManagerAg* for the NS entity. The *OrgManagerAg* is able to manage the OE and to coordinate the other agents. Each domain agents is supported by an *OrgWrapperAg* which is a kind of facilitator for the domain agent to access and interact with the supervisor agents.

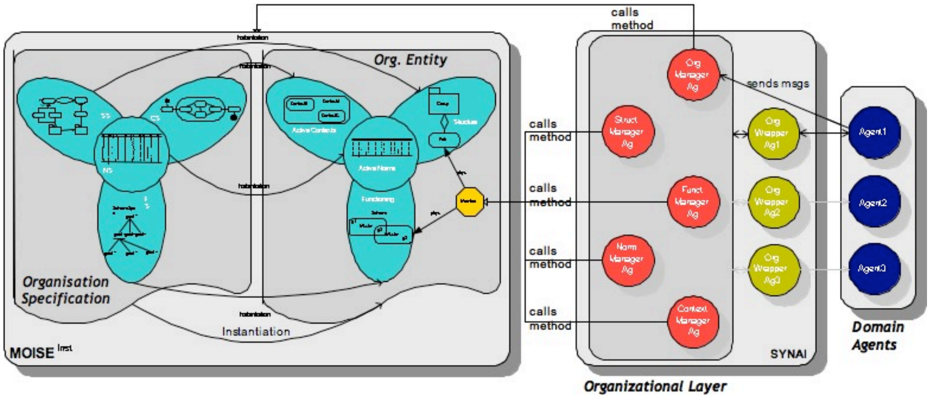


Fig. 7. Supervisor agents of the $\mathcal{S}$YNAI Organizational Layer.

4.2 Normative organization of the Organizational Layer

In order to supervise the enactment of the organization on the agents and to insure that the norms are fulfilled, the supervisor agents have to understand the $\mathcal{M}$OISE^{Inst} model. In order to make the implementation of the organizational layer independant of the structure of the supervisor agents, we chose to make explicit its organization, using the $\mathcal{M}$OISE^{Inst} OML. Supervisor agents are thus organised the same way as domain agents are, i.e. according to the OS defined with $\mathcal{M}$OISE^{Inst} in order to structure and to define their rights and duties (see Fig. 8).

The OS governing the supervisor agents is thus defined with a SS, FS, CS and NS as follows. The SS is composed of the only group “Supervision” containing the roles that supervisor agents would play in order to manage the domain agents OE. Since, all roles inherit of the “Supervisor” role, they can communicate with each other (communication link from “Supervisor” to itself). The cardinality ‘1..1’ except for “OrgWrapper” ensures that only and only one supervisor agent will play a role in this group.

The FS defines the main goals of the supervision system which is to keep the organization in a coherent state: choice between correcting the violation (*gOC* goal) or blocking the violation intention (*gOB* goal) (see supervision scheme in Fig. 8). As expressed in the scheme, the steps of supervision -expressed as social schemas- are: violation detection, correction or not of the violation (according to the choice) and sanction of the culprit. Constraints to be checked come from the SS (cardinalities, links, etc.), from the FS (mission cardinalities) and from the NS (norms). Thus a violation detection is either a NS violation, a FS violation or a NS violation.

The CS defines the contexts that are used for the choice of the arbitration strategies in relation to the achievement of goals *gOC* or *gOB*. During the activity of the OE, an event can be created which causes the change of state implying, according to the norms, a change in the arbitration strategy: correct violations or block violations.

The norms of the NS (cf. the table of the Fig 8) express that the organization must be kept in a coherent state by correcting violations in the “CorrArb” context ($NA1$ to $NA5$) and by blocking actions with violation intention in the “BlocArb” context ($NA6$). They express that the detection must be done in whatever context ($NA7$ to $NA10$).

The supervision OS is integrated into the domain OS to compose the global normative organization as follows. The supervision SS is integrated into the domain SS by including the “Supervision” group into the “Game” group and by installing an authority link from the “Supervisor” role on the “Soc” role. As a consequence, agents from $\mathcal{S}$YNAI playing one role of the supervision SS have authority and can control activities of all domain agents playing roles of the domain SS belonging to group “Game”. The supervision FS is just added to the domain FS. The “Sanction Scheme” defined for the domain is available and usable by the “Supervision Scheme” (see for instance the call to this scheme). This inclusion of the “Sanction Scheme” of the domain OS into the “Supervision Scheme” allows the use of domain specific sanction strategies into a generic supervision scheme that can be the same for all applications. The supervision CS is added to the domain CS as parallel transition state diagram. Both CS form a global CS. The supervision NS is added to the domain NS, composing the global NS of the normative organization.

Enacting this OS on the supervisor agents leads to an OE where *OrgManagerAg* plays “InstManager” and “Arbitrator”, and each supervisor agent plays the role corresponding to its capabilities: *StructManagerAg* plays “StructManager”, *FunctManagerAg* plays “FunctionalManager” and so on.

5 Related Works

Different organization modeling languages exist in the multi-agent domain. They use different modeling dimensions to cope with the complexity of the definition of multi-agent organizations. As in $\mathcal{M}$oise^{Inst}, they exhibit either a *structural* dimension talking about the structure of the collective level of an MAS (e.g. AGR [14], Islander [15], $\mathcal{M}$oise^{+}), generally in terms of roles/groups/links or a *functional* one talking about the global functioning of the system (e.g. Tms [16], TeamCore [17], $\mathcal{M}$oise^{+}). Some models as in Islander, add a *dialogical* dimension talking about the interaction in terms of communications between the agents. Others introduce an environmental dimension allowing to constrain the anchoring of organization in an environment such as in AGRE [18]. Inspired of Islander, $\mathcal{M}$oise^{Inst} has introduced a contextual specification to define a-priori the transition between different configurations of norms, structures and plans. We won't compare all these OML, here, in terms of the primitives or modeling power each one can offer (refer for instance to [19] for a systematic comparison of these models). Depending on these different dimensions, their influence on the agents' behavior may be quite different. In models such as Tms where only the functional dimension is specified, the organization has nothing to "tell" to the agents when no plan or task can be performed. Otherwise, if only the structural dimension is specified as in AGR, the agents have to reason for a global plan every time they want to work together. Even with a smaller search space of possible plans, since the structure constrains the agents options, this may be a hard problem. Furthermore, the plans developed for a problem are lost, since there is no organizational memory to store these plans. Thus, in the context of open systems, we hypothesize that if the organization model specifies both dimensions as in $\mathcal{M}$oise^{Inst} or TeamCore or a third one as in Islander then the MAS that follows such a model can be more effective in leading the group behavior to its purpose. On the agents' side, they can develop richer reasoning abilities about the others and their organization. Agents may gain more information on the possible cooperation (in terms of roles, groups, but also on the possible goals under achievement or on the performative structures that can be used) that may be conducted with the other agents.

Besides those dimensions, the *deontic* and *normative* dimensions used respectively in $\mathcal{M}$oise^{+} and Islander or $\mathcal{M}$oise^{Inst} address the agents autonomy problematic and consider organizations as normative constructs aiming at controling in an explicit manner the multi-agent system. While in other OML the agents are supposed to be benevolent and compliant (de-facto) to the OS, these two models add the possibility for agents to develop explicit reasoning on their autonomy with respect to the organizational constraints.

Turning now to the Organization Implementation Architecture that supports such normative OML, a few takes the same point of view of the normative organization layer developed in $\mathcal{S}$ynai. AMELI [20] is the organization layer for Islander. It provides a social layer which controls and helps the agents to participate in an e-institution with specialized governors. $\mathcal{S}$-$\mathcal{M}$oise^{+} [21] is the organizational layer for managing $\mathcal{M}$oise^{+} organizations. It provides the agents evolving in the organization with personal "OrgBoxes" giving a partial view of the organization. OrgBoxes serve as interface between heterogeneous agents and the organization. There is just one "OrgManager" for controlling the access of the agents to the organization. The deontic expressions are enforced but not controlled. For instance, violation of an obligation is hardly detectable.

6 Conclusion and Perspectives

We have presented in this paper an ongoing work for the definition of normative organization environment. It is composed of an normative organization modeling language $\mathcal{M}$OISE^{Inst} with its accompanying organizational layer $\mathcal{S}$YNAI. Different modeling dimensions are mobilised to program rich organizational patterns to control or to help the cooperation of the agents in the system: structural, functional, contextual and normative. As noticed, these dimensions are not exclusive and some dimensions are still proposed in related works (e.g. environment, dialogical). In $\mathcal{M}$OISE^{Inst}, the agents' autonomy concern is considered with the explicit definition of norms that bind all the dimensions together. The agents' autonomy is also taken into account in the organizational layer that support $\mathcal{M}$OISE^{Inst} with the definition of supervisor agents aiming at controling and enforcing norms into the system. Two kinds of agents evolve in such organization: the domain agents and the supervisor agents. With $\mathcal{M}$OISE^{Inst} we expressed at a "meta-level" the supervision organization that aims at controlling the supervisor agents by defining roles that they will play, as well as the missions related to their ability to detect norms violations and to punish culprit domain agents.

However some challenges still need to be considered and solved: decentralization of the organization infrastructure to address the scaling problem, developping reasoning abilities in order to integrate top-down predefined organizations (organization-centred) with bottom-up emergent organizations (agent-centred), with eventually solving conflicts (e.g. what if some agent playing a role must interact with another agent X playing its role, but the agent knows that X can not perform some intended task and it even prefer to interact with agent Y?), to undertsand in more depth every dimension, leading to an organization ontology to enable interoperation, reorganization issues in general (how to evaluate? how to change?).

ACKNOWLEDGEMENT

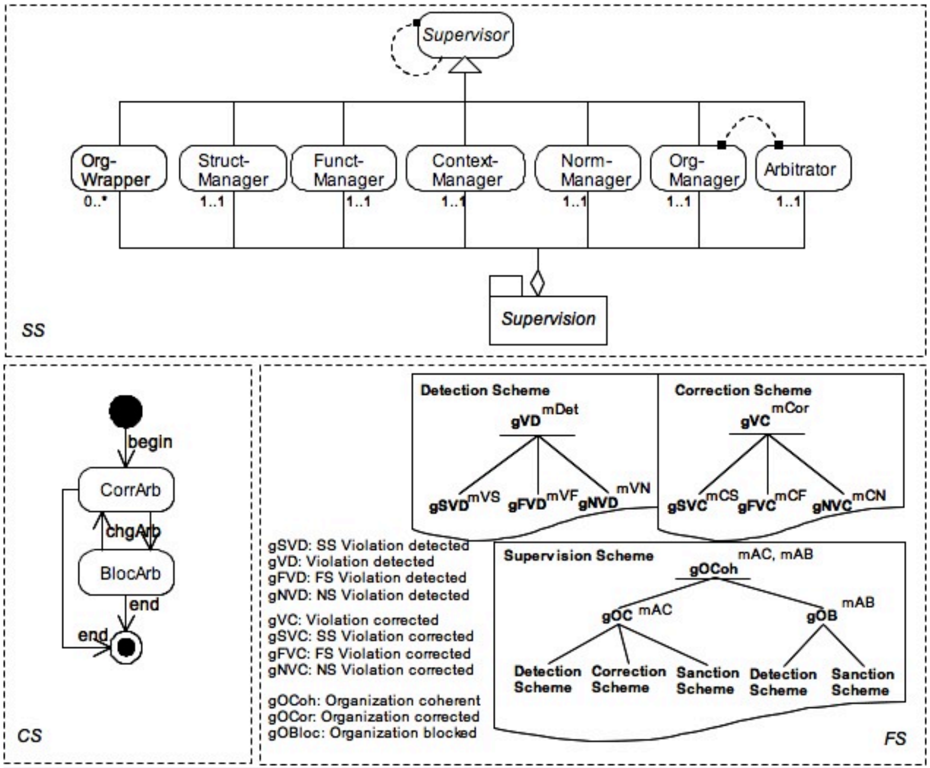
Part of this research has been funded by the University of Luxembourg through the LIASIT project.

References

1. Omicini, A., Ricci, A., Goldin, D.: Introduction to the workshop. In: Second International-Workshop on Theory and Practice of Open Computational Systems (TAPOCS 2004). (2004)
2. Esteva, M., Rodríguez-Aguilar, J.A., Rosell, B., L., J.: AMELI: An agent-based middleware for electronic institutions. In Jennings, N.R., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings of the Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'2004), New York, ACM (2004) 236–243

3. Hubner, J.F., Sichman, J.S., Boissier, O.: A model for the structural, functional, and deontic specification of organizations in multiagent systems. In Bittencourt, G., Ramalho, G.L., eds.: 16th Brazilian Symposium on Artificial Intelligence (SBIA'02). Number 2507 in LNAI, Springer (2002) 118–128
4. Hübner, J.F., Sichman, J.S., Boissier, O.: $\mathcal{S}$-$\mathcal{M}$OISE^{+}: A middleware for developing organised multi-agent systems. In Boissier, O., Dignum, V., Matson, E., Sichman, J.S., eds.: Proceedings of the International Workshop on Organizations in Multi-Agent Systems, from Organizations to Organization Oriented Programming in MAS (OOOP'2005). Volume 3913 of LNCS., Springer (2006) 64–78
5. Bellifemine, F., Poggi, A., Rimassa, G.: Jade - a fipa-compliant agent framework. In: 4th International Conference and Exhibition on The Practical Application of Intelligent Agents and Multi-Agents, London (1999)
6. Poslad, S., Buckle, P., Hadingham, R.: The fipa-os agent platform: Open source for open standards. In: PAAM. (2000)
7. Dignum, F.: Agents, markets, institutions, and protocols. In Dignum, F., Sierra, C., eds.: Agent-Mediated Electronic Commerce - The European AgentLink Perspective. Volume 1991 of Lecture Notes in Artificial Intelligence., Springer Verlag (2001) ISBN 3-540-41671-4.
8. Boissier, O., Hbner, J.F., Sichman, J.S.: Organization oriented programming, from closed to open organizations. In: Engineering Societies in the Agents World VI, Sixth International Workshop, ESAW 2006, Dublin, Ireland, September, 2006, Revised Papers. Lecture Notes in Computer Science, Springer Verlag (2007)
9. Jones, A., Carmo, J.: Deontic logic and contrary-to-duties. In: Handbook of Philosophical Logic. Kluwer (2001) 203–279
10. Castelfranchi, C., Dignum, F., Jonker, C.M., Treur, J.: Deliberate normative agents: Principles and architecture. In: Proceedings of The Sixth International Workshop on Agent Theories, Architectures, and Languages (ATAL-99). (1999)
11. Gâteau, B., Boissier, O., Khadraoui, D., Dubois, E.: $\mathcal{M}$OISE^{Inst}: An organizational model for specifying rights and duties of autonomous agents. In van der Torre, L., Boella, G., eds.: 1st International Workshop on Coordination and Organisation (CoOrg 2005) affiliated with the 7th International Conference on Coordination Models and Languages, Namur - Belgium (2005)
12. Gâteau, B., Khadraoui, D., Dubois, E.: Architecture e-business sécurisée pour la gestion des contrats. In: 3ème Conférence sur la Sécurité et Architectures Réseaux (SAR), La Londe, Cote d'Azur - France (2004)
13. Vázquez-Salceda, J., Aldewereld, H., Dignum, F.: Norms in multiagent systems: some implementation guidelines. In: Proceedings of the Second European Workshop on Multi-Agent Systems (EUMAS 2004). (2004)
14. Ferber, J., Gutknecht, O.: A meta-model for the analysis and design of organizations in multiagent systems. In: Third International Conference on Multi-Agent Systems (ICMAS98), Paris, France (1998) 128–135
15. Esteva, M., Padget, J., Sierra, C.: Formalizing a language for institutions and norms. In: Intelligent Agents VIII: 8th Intern. Workshop. Volume 2333 of LNAI. (2002) 348–366
16. Lesser, V., Decker, K., Wagner, T., Carver, N., Garvey, A., Horling, B., Neiman, D., Podorozhny, R., NagendraPrasad, M., Raja, A., Vincent, R., Xuan, P., Zhang, X.: Evolution of the gpgp/taems domain-independent coordination framework. Autonomous Agents and Multi-Agent Systems **9** (2004) 87–143 Kluwer Academic Publishers.
17. Pynadath, D.V., Tambe, M.: An automated teamwork infrastructure for heterogeneous software agents and humans. Autonomous Agents and Multi-Agent Systems **7** (2003) 71–100
18. Ferber, J., Michel, F., Baez, J.: AGRE: Integrating environments with organizations. In: E4MAS'04, 1st Int. Workshop. Volume 3374 of LNCS. (2005) 48–56

19. L. Coutinho, J.S. Sichman, O.B.: Modeling dimensions for multi-agent systems organizations. In Dignum, V., Dignum, F., Edmonds, B., Matson, E., eds.: Agent Organizations: Models and Simulations (AOMS), Workshop held at IJCAI 07. (2007)
20. Esteva, M., Rosell, B., Rodriguez-Aguilar, J.A., Arcos, J.L.: Ameli: An agent-based middleware for electronic institutions. In Jennings, N.R., Sierra, C., Sonenberg, L., Tambe, M., eds.: 3rd international joint conference on Autonomous Agents & Multi-Agent Systems (AAMAS). Volume 1., Columbia University, New York City - USA, ACM Press (2004) 236–243 ISBN 1-58113-864-4.
21. Hubner, J.F., Sichman, J.S., Boissier, O.: S-moise+: A middleware for developing organized multi-agent systems. In: International Workshop on Organizations in Multi-Agent Systems: From Organizations to Organization Oriented Programming (OOOP). (2005)



context	id	w.	condition	issuer	bearer	deOp	m ission	deadline	sanction
CorrArb	NA01	1	---	OrgManager	Arbitrator	O	mAC	---	---
CorrArb	NA02	1	---	Supervisor	Arbitrator	O	mCor	---	---
CorrArb	NA03	1	---	Supervisor	StructManager	O	mCS	---	---
CorrArb	NA04	1	---	Supervisor	FunctManager	O	mCF	---	---
CorrArb	NA05	1	---	Supervisor	NormManager	O	mCN	---	---
BlocArb	NA06	1	---	Supervisor	Arbitrator	O	mAB	---	---
---	NA07	1	---	Supervisor	Arbitrator	O	mDet	---	---
---	NA08	1	---	Supervisor	StructManager	O	mVS	---	---
---	NA09	1	---	Supervisor	FunctManager	O	mVF	---	---
---	NA10	1	---	Supervisor	NormManager	O	mVN	---	---

Fig. 8. Organization Specification of $\mathcal{S}$YNAI

Norms and accountability in multi-agent societies (extended abstract)

Rodger Kibble[1]

Goldsmiths, University of London, Department of Computing
Lewisham Way, London SE14 6NW, UK
r.kibble@gold.ac.uk

Abstract. It is argued that *norms* are best understood as classes of constraints on practical reasoning, which an agent may consult either to select appropriate goals or commitments according to the circumstances, or to construct a discursive justification for a course of action after the event. We also discuss the question of how norm-conformance can be enforced in an open agent society, arguing that some form of peer pressure is needed in open agent societies lacking universally-recognised rules or any accepted authority structure. The paper includes formal specifications of some data structures that may be employed in reasoning about normative agents.

Keywords. Norms, agents, social commitments, reasoning

1 Introduction

Researchers in multi-agent systems have often looked to analytic philosophy for suitable concepts and theoretical frameworks; indeed it could be said that philosophers since Aristotle have been engaged in writing specifications for rational agents. Some influential approaches have included Bratman's work on practical reasoning [1] and Austin and Searle's speech act theories [2,3]. Kibble [4] offered a critique of approaches to ACL semantics based on Speech Act theory such as FIPA's ACL [5] and outlined an alternative commitment-based approach drawing on more recent philosophical studies by Robert Brandom [6,7] and Joseph Heath [8]. Brandom's work presents an *inferentialist* account of theoretical and practical reasoning and communication, arguing that mentalist notions such as belief can be dispensed with in favour of more precise notions of observable practical and propositional *commitments* (though it turns out that this term does not seem to have a uniform interpretation among analysts; see [9] for discussion). Heath works at the frontiers of social theory and analytic philosophy, developing an account of the interaction of instrumental rational choice and social norms via a critical engagement with the work of Habermas [10,11]. Kibble [4] drew on this work with the aim of extending and elaborating the non-mentalist *social commitment* model of agency of [12]. Kibble proposed

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/916

an account of agent communication as *norm-governed action* for an agent to produce a dialogue act is to take on certain *commitments*, such as to defend the content of an assertion if challenged, and other agents are bound to concede that the agent is *entitled* to the propositional content of dialogue acts if those commitments are fulfilled. The present paper continues to build on this work, attempting to reconstruct and/or extend some of Brandom's and Heath's proposals concerning norms, sanctions and commitments in a form that can be applied to interactions between software agents. The paper is structured as follows:

- Section 2 considers how norms can be maintained by *peer pressure* rather than authoritarian structures of command and control, adopting elements of Heath's account of social norms;
- Section 3 argues that social norms can be represented as *constraints* on practical reasoning, rather than more primitive entities such as *goals* or *commitments*;
- Section 4 specifies some data structures to support reasoning about normative agents, adopting the notation of d'Inverno and Luck's SMART framework [13].

2 Norms, commitments and sanctions

As with multi-agent research in general, the study of normative agents suffers from inconsistent use of terminology and lack of consensus on the meaning of some fundamental terms: what exactly are *norms*? Assuming agreement can be reached on some working definitions, one of the questions that then has to be addressed is: why do (or should) agents conform to norms? There have been suggestions that failure to honour normative commitments should be subject to *sanctions*, but there have been few concrete proposals as to what form these sanctions might take or who is to be responsible for administering them.

At a certain level of abstraction we can consider norms as solutions to coordination games [14] that cannot be accounted for simply in terms of maximising utility. Classical game theory models agent interactions as a matrix of the payoffs for each agent according to the actions independently chosen by all players (*strategies*), and assumes both that the potential payoffs are known in advance to all players and that they will converge on a "Nash equilibrium" such that neither player could increase their payoff by changing strategy [15]. Non-trivial interactions tend to have many such equilibria however, yet the smooth functioning of society relies on some particular solution being commonly accepted, and thus on agents having some mechanism at their disposal for persuading or encouraging other agents to stick to the shared rules.

Moving towards the concrete: Brandom [7, 84ff] discusses three classes of norm involved in practical reasoning: the *prudential* or instrumental, *institutional* and *unconditional*, illustrated in the following examples:

α. Only opening my umbrella will keep me dry, so I will open my umbrella.

β. I am a bank employee going to work, so I shall wear a necktie.
γ. Repeating the gossip would harm someone, to no purpose, so I shall not repeat the gossip.

How would violations of these norms be sanctioned, if at all? In example (α), the "prudential" norm is a personal *preference* to stay dry rather than a social norm, so the only likely "punisher" is Nature rather than any human agent - unless perhaps the speaker is en route for some event where it would be inappropriate to turn up with wet hair and soaked clothes. The institutional norm, of wearing a necktie and being otherwise soberly dressed while working at a bank, is most likely reinforced by the threat of disciplinary action or even dismissal; minor violations might only be subject to disapproving comments from co-workers or clients. Finally, what Brandom refers to as "unconditional" rules about avoiding unnecessary harm and generally behaving in an ethical and considerate manner are not subject to institutional sanction: violations might be punished by the "voice of conscience" or if they became widely known, by expressions of reproach from friends, family etc. As Heath [8, p. 154] notes, sanctions against violations of social codes tend to be *symbolic*, intended to produce feelings of shame and regret rather than to physically harm or hinder the offender, and to articulate and re-affirm the norm.

Carrying across these distinctions into normative multi-agent systems will not be straightforward: for one thing we can safely assume that software agents are not subject to feelings of shame. Lopez y Lopez et al [16] for example use "norm" as an umbrella term encompassing "obligations, prohibitions, commitments and social codes", which would appear to fall under Brandom's headings of "institutional" and "unconditional". My approach proceeds from rather different assumptions:

Norms vs goals: The core of the definition of norms offered by [16] is a set of *normative goals* which specify "something that ought to be done". I will argue in the next section for a clear distinction between goals and norms, the latter being concerned with how goals are to be achieved and how the actions taken to achieve them can be justified.

Institutional vs bottom-up norms: The emphasis in [16] is on institutional norms, namely obligations and permissions, which are taken by [13] to be the only species of norm whose violation is punishable. Social commitments are stipulated to have rewards for compliance but no punishments for violation, while social codes have neither. Of the four different categories of artificial society described by [17], institutional norms are appropriate for closed, semi-closed and semi-open societies, where it is feasible to have commonly accepted rules and "enforcer" agents whose authority is universally recognised. With the growing potential for agent applications in open environments such as the Semantic Web, I suggest that this approach needs to be supplemented by considering whether and how norms can be sustained "horizontally" without assuming the existence of legislators, enforcers and so on.

A particular issue for MAS is indeed how norms can be enforced in open environments where norm-conformant agents interact with instrumentally rational agents. The main lacuna is probably in the area of *sanctions*: for an agent to be socially committed entails that failure to redeem the commitment will be subject to sanction, but the literature contains few concrete proposals on the precise nature of the appropriate penalties (though Walton and Krabbe [18, pp. 20, 184] offer some tentative suggestions). Brandom proposes sanctions for nonfulfilment of a commitment [6, p. 163] though apparently not for arbitrarily withdrawing commitments [7, p. 93], and no specific sanctions are specified for failure to honour propositional commitments. A more recent proposal [19] defines violation criteria for specified types of commitment and assigns fixed numerical penalties for violations. However, the authors are silent on how these numbers might translate into effective punishments that could hinder the offending agents, and on what protocols or structures of authority could be involved in the application of sanctions.

The approach taken in this paper is influenced by Heath's discussion of social norms [8, pp. 150-161], which itself draws on the work of Durkheim and Talcott Parsons (see Heath op. cit. for references). The key ideas are:

- sanctions serve to penalise *deviance* in the sense of prioritising instrumental considerations over norms;
- norm-conformant agents are characterised not only by being disposed to follow norms themselves but by "*the disposition to punish those who do not*" [8, p. 155, emphasis in original];
- agents which are not norm-conformant by design will thus have instrumental reasons to follow norms;
- prior to being sanctioned, agents may receive the opportunity to "give an account" of their reasons for action, as it may not always be evident whether an aberrant action results from deliberate deviance, *dissent* (adherence to a different set of norms from the majority) or misunderstanding (op. cit., p. 160).

Kibble [4] proposed that agents have the following options for penalising breaches of communicative norms:

- *ostracism*: the offending agent is notified that its messages will not be accepted for a specified time period, or until it performs the requested justificatory speech act;
- *blacklisting*: the complaining agent may broadcast details of the offence to trusted agents, which may decide to implement sanctions themselves.

These penalties could in principle be generalised to other instances of norm violations. For example in an e-commerce environment, temporary exclusion from the market-place would be a highly effective sanction against dubious business practices. Furthermore, if the penalties are imposed for a fixed time period, the duration of the time period could be determined according to the numerical calculations described by [19]. For an agent to choose to conform to a norm assumes either that they are designed with the capacity to recognise and reason about normative behaviour, or that their human principals may realise that something is going wrong and decide to re-engineer or replace their agent software.

3 Norms as constraints on reasoning

The principal claim I want to argue for in this section is that a norm is not simply a goal or commitment (including negative goals, i.e. prohibitions) but a set of criteria to enable agents to select an appropriate goal or adopt an appropriate commitment according to the circumstances. To take a fairly stark example, most religious and ethical systems include a precept against killing, yet also tolerate the taking of life in certain defined situations: self-defence, as a soldier in a "just war", as a policeman dealing with a life-threatening situation and so on. So the applicable norm in such systems is not simply a prohibition, *Do not kill*, it is a class of licit inference patterns leading to a conclusion *Do not kill* or *You may kill* according to the circumstances[1].

Another way of looking at things is to adopt a *discursive* account of goals and norms: instead of considering their role in *determining* an agent's actions, we may consider how they can be invoked after the event to *account* for the actions. From this point of view we can make quite a clean separation:

- *explanations* of an action or course of actions will make reference to the agent's *goals*, perhaps supplemented by a sequence of means-end reasoning.
- *justifications* of an action additionally need to refer to *norms*: goals alone cannot justify an action, since the legitimacy of the goals themselves as well as the means employed to achieve them may be at issue.

For example, there is an apocryphal tale of a career criminal who was asked why he kept robbing banks and replied, "That's where the money is". This may count as an explanation of goal-directed action, but not as a justification. If he had said something like, "The banks destroyed my livelihood by foreclosing the mortgage on the family farm", this could be understood as an appeal to an intelligible normative framework.

We could also express this distinction by saying that goals give rise to *commitments* to actions, while norms give rise to (claimed) *entitlements* to those commitments. In fact an underlying theme of this section is the relation between norms and *responsibility*, in the sense of the following statements:

> judgement and action . . . are in a distinctive sense what we are *responsible* for. They express *commitments* of ours . . . [7, p. 80]

> Accountability . . . captures two related aspects of the structure of norm-governed action, namely, that agents can be called upon to justify their actions vis-a-vis the relevant norms. . . and that they can be sanctioned for failure to comply with the prevailing normative expectations. . . [8, pp. 151-2]

[1] Of course, there are certain communities such as Quakers or Jains for whom the inference would be somewhat vacuous, in that the conclusion would invariably forbid killing.

The previous section discussed the second sense of accountability; here we are concerned with the first. Before proceeding further I wish to depart from Brandom's terminology in one respect. He uses the term "prudential norm" where I prefer to speak of instrumental preferences, reserving the term norm for *social* norms, where other agents' expected behaviour plays a part in deciding whether or not to conform. With reference to example α above, the speaker doubtless prefers to avoid getting wet whether or not anyone else will ever know about it. Having established this distinction, I propose that an action can be considered as norm-conformant if an agent called to account for the action is barred from offering a purely instrumental explanation. This ties in with the observation in section 2 that a norm is a solution to a coordination game which cannot be specified purely in terms of maximising payoffs.

The notion of accountability establishes a link between action and communication: various researchers in MAS have followed [20,18] in treating agent communications as actions that express or give rise to *commitments*: an agent can be said to be privately committed to the truth of a proposition, or publicly committed to producing an argument supporting the proposition if challenged. (A strictly non-mentalist account would only admit the second of these senses.) Likewise, we can say that an agent who has adopted a goal is privately committed to a course of action, and executing the action creates a public commitment on the agent to justify it if challenged, or to demonstrate *entitlement* to a set of goals and actions.

Brandom [7] stresses the inescapably non-monotonic nature of practical reasoning: in examples α - γ, the conclusion could be invalidated by an additional premise. For instance if we accept β as a good inference, the following variant β' may still be classed as bad:

> β' I am a bank employee going to work, and today is Dress-down Friday,
> so I shall wear a necktie.

The implication is that in the above examples of practical reasoning, the particular norm being invoked cannot simply be filled in as a missing premise: *dress soberly when working at a bank*, *do not cause undeserved harm* etc, but rather defines a particular class of inference patterns. Using explicitly normative terminology such as "employees *should* wear neckties" serves to express endorsement of particular patterns of inference:

> Different patterns of inferences should be understood as corresponding to different sorts of norms or pro-attitudes. [7, p.90]

The most general or abstract way to define a norm is thus as a *subset* of the set of inferences available to an agent according to its propositional vocabulary and reasoning capabilities.

4 Data structures for reasoning about normative agents

This section outlines some data structures, at a fairly high level of abstraction, which could be employed in reasoning about the actions and commitments of a normative agent, either by observing its behaviour and utterances or by directly querying it about the reasons for its actions. We begin by adopting some notation from d'Inverno and Luck's SMART framework [13] in the hope that this will facilitate comparison with more established approaches.

4.1 Basic definitions

I will follow d'Inverno and Luck up to the definition of an Agent, after which there will be some divergence.

The framework includes the following primitive types, where *Attribute* is the type of basic facts about the world:

$$[\mathit{Attribute}, \mathit{Action}]$$

Entities are not taken as primitives, but as bundles of attributes. What follows is a simple example of a Z schema [21], comprising a name (*Entity*), a section where variables are declared (the *signature*) and a *property* section.

$$\begin{array}{|l}\hline \mathit{Entity} \\ \mathit{attributes} : \mathbb{P}\,\mathit{Attribute} \\ \hline \mathit{attributes} \neq \varnothing \\ \hline \end{array}$$

The Environment, *Env* is defined as some non-empty set of Attributes:

$$\mathit{Env} == \mathbb{P}_1\,\mathit{Attribute}$$

An Object an Entity capable of Actions. The notation here says that the *Object* schema includes the specifications of the *Entity* schema and extends it with additional statements.

$$\begin{array}{|l}\hline \mathit{Object} \\ \mathit{Entity} \\ \mathit{capabilities} : \mathit{Actions} \\ \hline \mathit{capabilities} \neq \varnothing \\ \hline \end{array}$$

Objects do not have their own goals, so only act in furtherance of goals imposed from outside. If an Object is endowed with goals, it becomes or instantiates an Agent:

$$\mathit{Goal} == \mathbb{P}_1\,\mathit{Attribute}$$

$$\begin{array}{|l}\hline \mathit{Agent} \\ \mathit{Objects} \\ \mathit{goals} : \mathbb{P}\,\mathit{Goal} \\ \hline \mathit{goals} \neq \varnothing \\ \hline \end{array}$$

4.2 Definitions for norm-conformant agents

In order to be able to talk about the reasoning capacities of normative agents, I define a few new types.
A Proposition is a bundle of attributes which may or may not be true in a given situation. It appears that a Proposition is denotationally equivalent to an Entity, but they will be differentiated by their different roles in agent schemas.

$$\mathit{Proposition} == \mathbb{P}_1\ \mathit{Attribute}$$

An Inference is an ordered triple involving an Environment, a set of Propositions constituting the premises of an argument, and a Proposition as the conclusion.

$$\begin{array}{l} \mathit{Inference} == (\mathit{Env} \times \mathbb{P}\ \mathit{Proposition} \times \mathit{Proposition}) \\ \mathit{Inferences} == \mathbb{P}\ \mathit{Inference} \end{array}$$

If we observe an Agent carrying out Inferences, we may call it a RationalAgent.

$$\begin{array}{|l} \multicolumn{1}{l}{\mathit{RationalAgent}} \\ \hline \mathit{Agent} \\ \mathit{inferences} : \mathit{Inferences} \\ \hline \mathit{inferences} \neq \varnothing \\ \hline \end{array}$$

As argued in the previous section, norms are essentially constraints on inferences, thus they delimit the class of logically possible inferences an agent may carry out. A simple way to represent this is to define the *norms* which an agent adheres to as a subset of the *inferences* of which it is capable.

$$\begin{array}{|l} \multicolumn{1}{l}{\mathit{NormativeAgent}} \\ \hline \mathit{RationalAgent} \\ \mathit{norms} : \mathit{Inferences} \\ \hline \mathit{norms} \neq \varnothing \\ \mathit{norms} \subseteq \mathit{inferences} \\ \hline \end{array}$$

This basic framework will of course need to be extended in various ways, in particular to model the *sharing* of norms within an agent society.

5 Conclusions

This extended abstract has considered normative agency in terms of the accountability of agents, where (following [8]) agents can be held accountable for their actions by being sanctioned for deviant behaviour, or by being required to give an account of the reasons and justifications for the action. The account has drawn on recent work in linguistic and social philosophy by Brandom and

Heath. I have proposed a clear distinction between norms and goals, characterising norms as constraints on reasoning which govern both the selection of appropriate goals and their justification after the event. The discussion has been conducted in rather general terms, though I have tried to indicate how it could be made more precise with the aid of the SMART framework. Future work will aim to extend this formalisation, in the hope that this will not only inform the design of normative software agents but can feed back into the philosophical arena by sharpening up the conceptual framework.

References

1. Bratman, M.: Intentions, Plans and Practical Reasoning. Harvard University Press (1987)
2. Austin, J.L.: How to do things with words. Oxford University Press, Oxford (1962)
3. Searle, J.: Speech Acts. Cambridge University Press, Cambridge, UK (1969)
4. Kibble, R.: Speech acts, commitment and multi-agent communication. Computational and Mathematical Organisation Theory **12** (2006) 127 – 145
5. FIPA: FIPA communicative act library specification. Technical Report SC00037J, Foundation for Intelligent Physical Agents, Geneva, Switzerland (2002) Specification dated 2002/12/03.
6. Brandom, R.: Making it Explicit. Harvard University Press, Cambridge, Massachusetts and London (1994)
7. Brandom, R.: Articulating Reasons: An Introduction to Inferentialism. Harvard University Press, Cambridge, Massachusetts and London (2000)
8. Heath, J.: Communicative Action and Rational Choice. MIT Press, Cambridge, Massachusetts and London (2001)
9. Kibble, R.: Reasoning about propositional commitments in dialogue. Research on Language and Communication (2006)
10. Habermas, J.: Theory of Communicative Action, vols 1 and 2. Polity Press, Cambridge, UK (1984)
11. Habermas, J.: On the Pragmatics of Communication. Polity, Cambridge, UK (1998) Edited by Maeve Cooke.
12. Singh, M.: A social semantics for agent communication languages. In: Proc. IJCAI'99 Workshop on Agent Communication Languages. (1999) 75–88
13. d'Inverno, M., Luck, M.: Understanding Agent Systems. Springer Verlag, Berlin Heidelberg New York (2004)
14. Lewis, D.: Convention. Blackwell, Oxford (1969)
15. Osborne, M., Rubinstein, A.: A Course in Game Theory. MIT Press (1994)
16. Lopez y Lopez, F., Luck, M., d'Inverno, M.: A normative framework for agent-based systems. In Boella, G., van der Torre, L., Verhagen, H., eds.: AISB'05: Symposium on Normative Multi-Agent Systems, University of Hertfordshire (2005)
17. Davidsson, P., Johansson, S.: On the potential of norm-governed behaviour in different categories of artificial societies. In Boella, G., van der Torre, L., Verhagen, H., eds.: AISB'05: Symposium on Normative Multi-Agent Systems, University of Hertfordshire (2005)
18. Walton, D., Krabbe, E.: Commitment in dialogue. State University of New York Press, Albany (1995)
19. Amgoud, L., de Saint-Cyr, F.D.: Towards ACL semantics based on commitments and penalties. In: Proceedings of ECAI 2006, Riva del Garda, Italy (2006)

20. Hamblin, C.: Fallacies. Methuen, London (1970)
21. Spivey, J.M.: The Z notation: a reference manual. 2nd edition. Prentice Hall (1994)

Norms and plans as unification criteria for social collectives

Aldo Gangemi, Jos Lehmann, Carola Catenacci

Laboratory for Applied Ontology
Institute of Cognitive Science and Technology
Italian National Research Council, Rome, Italy
{aldo.gangemi,jos.lehmann,carola.catenacci}@istc.cnr.it

Abstract. Based on the formal-ontological paradigm of *Constructive Descriptions and Situations*, we propose a definition of social collectives that includes social agents, plans, norms, and the conceptual relations between them. We also propose a typology of social collectives, including *collection of agents*, *knowledge community*, *intentional collective*, and *intentional normative collective*. Our ontology, represented as a first-order theory, provides the expressivity to talk about the contexts (social, informational, circumstantial, and epistemic), in which collectives *make* and *produce* sense.

Keywords. Formal Ontology, Constructivism, Social Entities, Semantic Web

1 Introduction

In this article we lay down the basis for an integrated ontology of the mutual dependencies between agents, collectives, concepts, information, plans, and norms. The ontology has a constructive approach, and is represented as a first-order theory, as well as an OWL(DL) ([1]) ontology, for applications in the semantic web[1],and semantic web services domains (cf. [2]).
In previous work [3], we have treated some problems of *collective intentionality* by introducing a formal-ontological definition of the notion of *intentional collective*. Our approach pivoted on two general ideas. On the one hand, we investigated and formalized the grounds based on which we define a set of items as a *collection*, and collected items as *members* of a collection[2]. On the other hand, we proposed a way of relating collections and their members to intentional and agentive notions[3]. According to our reconstruction, collections can be seen as social objects (as defined in [20]) that (generically) depend on their members at a certain time. This entails, for instance, that a collection of books in a library remains the same entity even if some books are lost and others acquired over time. Collections depend also (specifically) on the *role(s)* played by their members. Consider, for example, the constellation of Orion. Should the role 'being a member of Orion'

[1] See e.g. the EU NeOn project site: http://www.neon-project.org

[2] There is a large and heterogeneous literature on collections and plural entities; in our work, we considered in particular [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14]

[3] On this topic, we made reference to classical works such as [15], [16], [17], [18], [19]

cease to exist, the relative constellation would disappear too. Collections can also be (and usually are) characterized by further roles; consider, for instance, the collection of different (cutting, pasting, etc.) machines in a factory. Collections, finally, are unified by 'theory-like' entities that we call *descriptions*, which contain and specify the covering or characterizing roles of the collection.

Following this notion of collection, collectives in our proposal are collections of agents which are unified by the kind of descriptions that we call *plans*. The members of a collective are 'held together' by one plan, which specifies a goal and (one or more) covering or characterizing role(s). In order for a collective to be intentional, there must be a plan, and the agentive members of a collection must play the covering or characterizing roles of that plan. For instance, in our view, both a group of people running towards a common shelter because of a sudden storm [21], and a pack of hunting wolves are to be considered as examples of intentional collectives.

In this article, the proposal presented in [3] is updated and enriched under two respects. Firstly, the very foundations of the original proposal are profoundly restructured by a new paradigm, called Constructive Descriptions and Situations. Secondly, our definition of intentional collectives is exteded with *normative elements*, as well as with the conceptual relations between such new normative elements and the orginal planning elements. This move provides us with the conceptual means to define collective entities like *collection of agents*, *knowledge community*, *intentional collective*, and *intentional normative collective*. Within this framework, an issue we address is how to *represent* in our formal ontological framework the influence that norms may have on plans. We are not concerned here with providing a full-fledged theory of these interactions. We rather want to set a formal-ontological basis for modeling such theories. In other words, we are aware of the fact that relevant work on norms, and possibly on their interactions with plans, may be found both in the legal-philosophical literature (e.g. [22], [23]), in the sociological literature (e.g. []), as well as in the multi-agent systems literature (e.g. [24]). Here though we introduce a minimal setup of formal distinctions between the types of interactions that norms may have with plans. Future work will refine such distictions, possibly modelling existing proposals on that basis.

Section 2 provides a brief informal overview on our previous work, including Constructive Descriptions and Situations, and the ontologies of Plans, Norms, and Collectives. Section 3 presents a formalization of our proposal. Finally, section 4 draws some conclusions.

2 Constructive DnS and its extensions at a glance

In this section we informally introduce the ontological apparatus, on which our treatment of collectives is based. We start with a brief presentation of Descriptions and Situations (DnS), an ontology developed in [25], [26], [3]. We then present Constructive DnS, a restructured version of DnS proposed in [27] and in this paper. Finally, we provide a schematic introduction to our ontologies of Plans, Norms and Collectives.

2.1 Relations to previous work

In [3], we have provided a formal-ontological definition of the notion of *intentional collective*. Our approach there pivoted on two general ideas: on the one hand, we investigated and formalized the grounds based on which we define a set of items as a *collection* and collected items as *members* of a collection; on the other hand, we proposed a way of relating collections and their members to intentional notions. The work presented in [3] was based on three ontologies: the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [28], the ontology of Descriptions and Situations (DnS) [25] and the ontology of Plans defined in [26].

In this article, the work described above is updated and enriched under two respects. Firstly, the very foundations of our original proposal are profoundly restructured. The definition of collectives given here is not based anymore on a combination of DOLCE and DnS, but on a brand-new version of DnS, called Constructive DnS (hereafter, *c.DnS*). Secondly, our definition of intentional collectives is exteded with *normative elements*. This provides us with the conceptual means to define typologies of normed intentional collectives.

In its original version, DnS is a formal tool that allows to extend other (possibly, but not exclusively, foundational) ontologies with a number of reified concepts and relations, thus making the extended ontology more expressive without making it computationally more complex. Suppose, for instance, that you want to use DnS to extend DOLCE[4]. The final result of this extension would be DOLCE+. DOLCE+ would consist, on the one hand, of DOLCE – which would play the role of ground ontology, i.e. the ontology that specifies the entities of a given domain irrespective of any possible epistemological status or concern. On the other hand, DOLCE+ would also consist of the DnS extension, that provides the formal means to specify the epistemological perspective from which the entities of the domain are considered. By way of example, suppose that such perspective is of legal nature. The DnS extension makes it possible to express the legal constraints imposed by norms and regulations on the domain of the ground ontology, i.e. to describe the entities of DOLCE (in particular, entities pertaining to social reality) under a legal perspective. In other words, in DOLCE+ it would become possible to describe a legal view on the behaviour of DOLCE's (social) entities according to a given legal system.

Two are the key-elements of DnS:

The distinction between descriptions and situations which allows to separate, within the same model of the legal domain, relations between 'conceptual elements' like laws, norms, regulations, crime types, etc. (which are all descriptions) from relations between 'observable elements' like legal facts, cases, states of affairs, etc. (which are all situations).

A reification mechanism which allows to have descriptions and situations in the same domain of quantification (i.e. at the same logical level) and to relate them by means of a reified relation of satisfaction. For instance, according to a DnS-based legal extension of a ground ontology like DOLCE, a case (a situation) satisfies a norm (a description). This means that the norm (i.e. the description and the concepts devised in it) classifies the entities of DOLCE,

[4] The OWL-DL version of DnS combined with DOLCE can be loaded from http://www.loa-cnr.it/ontologies/ExtendedDnS.owl.

the ground ontology. This very classification gives rise to the case (i.e. the situation), which is a setting for the entities of the ground ontology that satisfy (i.e. are individually classified by) the concepts devised in the description. Both the case and the norm are part of the same domain of quatification. In other words, the DnS extension makes it possible to enhance the expressivity of the language of the ground ontology while keeping its complexity under control – the usual advantage of reification.

While DOLCE+ is being effectively used in several projects, specially in its version encoded in the Web Ontology Language (OWL), we have realized that the expressive power of the reification vocabulary in DnS can be axiomatized and reused beyond the scope of a specific foundational ontology (DOLCE or any other). Based on this working hypothesis, we have created a new ontology that applies the constructivist paradigm, and remains agnostic with respect to which foundational, core, or domain ontology should be used as a primary modelling framework for a knowledge base. We have named this ontology *Constructive Descriptions and Situations* (*c.DnS*).

2.2 The constructive stance

Constructivism is the epistemological stance according to which reality and its structure are not given 'as such' for our minds to passively discover, but are rather actively constructed by cognitive agents in specific contexts and for specific purposes. This implies a rejection of naive interpretations of Aristotelian notion of 'truth as correspondence' (between constructs and chunks of 'reality'), and a deep awareness of the historical and social nature of all kinds of knowledge. It does not imply, however, that we have to reject the idea that there are (physical, biological and cultural) constraints on the way the mind builds and manages its constructs, nor that the whole scientific enterprise is devoid of meaning. Rather, constructivism promotes a view according to which every scientific theory or model should be seen as a 'tool', which is the product of a specific 'knowledge collective'[5] and whose adequacy in representing and handling specific aspects of our interaction with the world has to be tested against actual usage and effectiveness, and always be open to revision (cf. [30]).

In cognitive sciences, in particular, this has led to see our mental representations as context-dependent (or 'situated') and action-oriented views on the world, relating only to those aspects of the environment which are salient for the perceiver/cognizer [31]. Moreover, focusing on the non-abstract nature of cognition has lead to put a new emphasis on the 'gestaltic' aspects of representations, i.e. the need of taking into account "the interconnected whole that gives meaning to the parts" [32].

In current ontology research and engineering, however, epistemology is usually left out of the picture. Viewpoints on, and theories of, the represented entities are assumed not to play any relevant role inside an ontology, since the latter reflects a static, 'frozen', and widely shared portion of knowledge in a given field. So, although even a common-sense concept, like *sun*, refers to an aspect of reality that is 'seen' and understood in the terms set by a culturally determined conceptualization, there seems to be little or no point in introducing this whole

[5] The term is borrowed from Ludwik Fleck's epistemological observations on 'thought-collectives' ($Denkkollektiv$) and 'thought-styles' ($Denkstil$); cf. [29].

conceptualization explicitly in, e.g., an ontology of weather conditions.
The intuition underlying this practice, however, comes to odds when an ontology needs to be extended with social entities, such as social institutions, organizations, plans, regulations, narratives, schedules, parameters, diagnoses, etc. Important fields of investigation have denied an ontological primitiveness to social objects, since the latters are taken to have meaning only in combination with some other entity, i.e. it is assumed that their intended meaning results from a statement (see e.g. [33]).
In that view, for example, a norm, a plan, or a social role should be better represented as a (set of) logical statement(s), not as logical individuals. This position is documented by the almost exclusive attention dedicated by many relevant frameworks (such as BDI agent model, theory of trust, situation calculus, and formal context analysis) to states of affairs, facts, beliefs, and contexts, whose logical representation is set at the level of theories or models, not at the level of concepts or relations.
In *c.DnS*, we take seriously the attempt to build a constructive formal ontology that assumes social entities as first-class citizens in a logical theory's domain of quantification.

2.3 Informal description of c.DnS

The core structure of *c.DnS* [27] is the following:

$$\langle D, S, C, E, A, K, I, T \rangle$$

where D is the class of *Descriptions*, S is the class of *Situations*, C is the class of *Concepts*, E is the class of *Entities*, A is the class of (social) *Agents*, K is the class of *Collections*, I is the class of *Information Objects*, and T is the class of *Time intervals*.
In intuitive terms, these classes allow to model how a social agent, as a member of a certain community, singles out a situation at a certain time, by using a descriptive relation that assigns concepts to entities within that situation. In other words, these classes are meant to formalize the constructivist assumption according to which, in order to contextualize a concept, we need to take into account the viewpoint/s or description/s inside which the concept is defined or used, the situation/s this viewpoint 'carves out' the perceived environment, the entities which are in the setting of said situation/s, the social agents who share the viewpoint, the collective, or community, of which these agents are members, the information object/s by which the viewpoint is expressed, and, finally, the time or time-span characterizing the viewpoint.
c.DnS' classes are substrucured as follows: E is the class of everything that is assumed to exist in some domain of interest, for any possible world. E is partitioned in the class of 'schematic entities', i.e. entities which are axiomatized in *c.DnS* (D, S, C, A, K, I), and the class of 'non-schematic entities', which are not characterized in c.DnS (T). Other non-primitive social enties may be added as subtypes of E, depending on the needs of the modeled domain. For instance in the application of *c.DnS* presented throughout section 3 the following additional entities are also considered: *physical agents* (PA) , *internal representations* (R), *physical realizations* (PR), *objects* ($Object$), *actions* ($Action$), and *regions* ($Region$).

The main purpose of c.DnS is to *redescribe* entities that are (or are assumed to be) existing. For example, an existing situation including humans, cars, roads and signs can be redescribed as a driving situation, as a racing situation, as well as a speed-limit-violation situation, depending on the circumstances and on the intention of the interpreter of that situation.

In the field of developmental psychology, this ability has been described in terms of *Representational Redescription*, “a process by which (implicit) information that is in a cognitive system becomes progressively explicit to that system” [34], allowing for greater flexibility.

This ‘redescription game’ is played in terms of a number of projections of the general c.DnS relation, which allows to relate schematic and/or non-schematic entities to one another. We provide here a brief overview of such projections per class.

Descriptions are entities which represents a conceptualization, it is generically dependent on some (physical) agent and communicable ([20]). Examples of descriptions are regulations, plans, laws, diagnoses, projects, plots, techniques, etc. Descriptions have typical components, called concepts, and are related to other entities in *c.DnS* by means of the following projections: $defines$, $uses$ (which hold between descriptions and concepts); $involves$, *individuallyConstructedAs* (compositions of relations holding between descriptions and entities); $unifies$ (holding between a descriptions and a collections).

Situations are entities which represents a state of affairs, under the assumption that its components ‘carve up’ a view (a setting) on the domain of an ontology by virtue of a description. Examples of situations (corresponding to the examples of descriptions above) are: facts, plan executions, legal cases, diagnostic cases, attempted projects, performances, technical actions, etc. Situations are related to other entities in *c.DnS* by means of the following projections: $satisfies$ (holding between situations and descriptions); $hasInScope$ (holding between situations); $settingFor$ (holding between situations and entities).

Concepts are defined by a description and can be used in other descriptions. Concepts are related to other entities in *c.DnS* by means of the following projections: $classifies$ (holding between concepts and entities); $covers$, $characterizes$ (holding between concepts and collections).

Entities are anything that is assumed to exist in some domain of interest, for any possible world. Main subtypes of entities are ‘schematic’ and ‘non-schematic’. Both subtypes may have a $memberOf$ relation with collections, while non-schematic entities are related to other entities in *c.DnS* by means of the following projections: $constructs$ (holding between non-schematic entities); $actsFor$ (holding between non-schematic entities and agents; $realizes$ (holding between non-schematic entities and information objects).

Social agents may be a person or an organization, but never a bio-physical system that plays an agentive role[6]. Social agents are related to other entities in *c.DnS* by means of the following projections: $shares$ (holding between social agents and descriptions); $redescribes$ (holding between social agents and situations); $deputes$ (holding between social agents and concepts).

[6] Agents of the latter kind are introduced as non-schematic entities.

Collections are a naturalization in space-time of non-empty proper classes with (at least one) basic properties for membership. This seems to capture the common sense intuition underlying groups, teams, collections, collectives, associations, etc.. For an extensive treatment of similarities and dissimilarities between this notion of collection and the notions of (natutalized) set or class refer to [3].

Information Objects are units of information which are related to other entities in *c.DnS* by means of the following projections: $expresses$ (between information objects and descriptions) and $about$ (between information objects and entities). Collection are inversely related to other entities in *c.DnS* by means of the following projections: $unifies$ (holding between a descriptions and a collections); $covers$, $characterizes$ (holding between concepts and collections); $memberOf$ (holding between entities and collections).

Time intervals, which are not characterized in DnS (i.e. they are non-schematic entities), are used for tagging descriptions, situations and projections. Time intervals should be added to the ontologies, that do not include them in their domain, when aligned to Constructive DnS.

c.DnS	Description (D)	Situation (S)	Concept (C)	Entity (E)	Social agent (A)	Collection (K)	Information object (I)
Description (D)	na	-satifies	defines [1a], uses [1a]	involves [1c], individuallyConstructedAs [2b]	na	unifies [1h]	-expresses
Situation (S)	satifies [1b]	hasInScope [1f]	na	settingFor [1d]	na	na	na
Concept (C)	-defines, -uses	na	na	classifies [1c]	na	covers [1h], characterizes [1h]	na
Entity (E)	-individuallyConstructedAs	-settingFor	-classifies	constructs [2b], -involves	acts for [2a]	memberOf [1h]	realizes [2c], -about
Social agent (A)	shares [1e]	redescribes [1f]	deputes [1e]	-acts for		na	na
Collection (K)	-unifies	na	-covers, -characterizes	-memberOf	na	na	na
Infomation object (I)	expresses [1g]	na	na	about [1g] -realizes	na	na	na

Table 1: c.DnS's classes, projections and principles, inverse projections

All projections mentioned above are based on two main constructive principles: the social construction principle and the grounded construction principle. On their turn, these principles are based on a larger set of other principles, listed below. Moreover, Table 1 shows the classes of *c.DnS* and their projections with a reference to the corresponding principles.

1. The *social construction principle* is based on:
   (a) Relationality principle: concepts are always defined in a relational context (i.e. a $description$ or a gestalt).
   (b) Interpretability principle: situations are always emerging/interpreted in a relational context according to some expected configuration.
   (c) Classification principle: entities are always internally represented with reference to a concept.

   (d) Situatedness principle: entities are always internally represented within a context according to some expected or reconstructed configuration.
   (e) Sharing principle: descriptions are always dependent on social agents.
   (f) Epistemological layering principle: given a description d_1 that involves another description d_2 , a situation s_1 that satisfies d_1 has in its scope a situation s_2 that satisfies d_2.
   (g) Formedness principle: descriptions are always expressed by information objects that provide a form to them.
   (h) Containment principle: there exists a collection for all entities classified by a concept.
   (i) Interaction principle: any social agent must be member of a knowledge collective in order to share a description.
2. The *grounded construction principle* is based on:
   (a) Agent efficacy principle: social agents should always be acted by some entity.
   (b) Cognitive counterpart principle: for any description there is a social agent who shares it and who is acted by a physical agent that constructs an internal representation which is the individual construction of that description.
   (c) Information grounding principle: any information object must have a physical support.

2.4 Plans, Norms and Collectives

Based on *c.DnS*, we define the notions of Plan, Norm and Collective and exploit their relations for defining typologies of normed collective entities.

For what concerns the notion of plan, we stick here to [26] where a plan is a description that represents an *action schema* that is *shared* by a social agent but *constructed* by a physical agent. In addition, a plan $defines$ or $uses$ or has as proper parts tasks, roles, goals, where:

Task is a concept that classifies *actions* (or similar non-schematic action-like entities)

Role is a concept that classifies *objects* (or similar non-schematic object-like entities)

Goal is the proper part of the plan that is desired by an agent

For what concerns the notion of norm, we take here the stance of [35] where a norm is a description, i.e. norms are treated there in their social sense (which includes but is not limited to the legal sense). This view on norms takes also into account Searle's distinction [21] between regulative and constitutive norms: regulative norms provide codes of conduct (i.e. regulations), while constitutive norms create social individuals and possibly contain no regulations at all. Here we mainly concentrate on regulative norms, on their social *usage* and on their influence on agents' plans and collectives. If the entities of a situation are classified by the concepts (roles, tasks, parameters) used in the norm, then the situation *falls under* the norm. This is an important difference from plans, which are *executed* in novel situations: norms are satisfied in a more complex and indirect way, because a situation that falls under a norm does not necessarily satisfy it.

Based on this simple model of norms, we address in this paper the issue of how to represent in *c.DnS* the influence that norms may have on plans. We are not concerned here with providing a full-fledged theory of this interactions. We rather want to set a formal-ontological basis for modeling such theories. In other words, we are aware of the fact that relevant work on norms, and possibly on their interactions with plans, may be found both in the legal-philosophical literature [22], [23] and in the multi-agent systems literature [24]. Here though we base our analysis on an a set of intuitive distinctions between the types of interactions that norms may have with plans. Future will refine such distictions, possibly based on other existing proposals. So, for the moment, in *c.DnS* norms may be seen as interacting with plans in one of three ways.

Norms as conventions that emerge from existing practices or plans. A norm, either social or legal, usually reflects an existing practice within a community. Typically, social and legal systems are the main way to maintain a stability among the members and the resources of a community, population, or country.

Norms as compliance checking protocols over social behavior or legal cases. Once a norm lifecycle is established, norms can be enforced by using them as filters for social behavior. Typically, the initiative for compliance checking is limited to the interest of other parties with respect to the behavior of one party.

Norms as constraints within plans. Once a norm lifecycle is established, and appropriate enforcing and compliance checking practices emerge, they can be used by social agents as constraints within their own plans. In this sense, norms are akin to *behavioral principles*. If taken as principles for social behaviour, norms can be executed, similarly to plans, and in fact they are executed as subplans.

Finally, *c.DnS*, together with the theories of plans and norms described above, provide us with the formal means to define the notion of collective and a typology for these:

Collection of agents is a collection unified by some rationale that is extrinsic with respect to the knowledge shared by the member agents.

Knowledge community is a collection of agents unified by descriptions that are shared by the member agents.

Intentional collective is a knowledge community that is unified by a plan shared by member agents.

Intentional normative collective is a knowledge community that is unified by a plan that, in turn, is entrenched with norms according to of the possible interaction between norms and plans described above.

3 Formal apparatus

3.1 The *c.DnS* relation

The *c.DnS* relation is given in (1). Each element of the core structure is encoded as a domain in a relation with arity=8:

$$c.DnS(d, s, c^*, e^*, a^*, k^*, i^*, t^*) \rightarrow \\ D(d) \wedge S(s) \wedge C(c^*) \wedge E(e^*) \wedge A(a^*) \wedge K(k^*) \wedge I(i^*) \wedge T(t^*) \qquad (1)$$

D can be read as *Description*, S as *Situation*, C as *Concept*, E as *Entity*, A as *social Agent*, K as *Collection*, I as *Information object*, and T as *Time interval*.

Intuitively, the *c.DnS* relation says that *a social agent, as a member of a given knowledge community, singles out a situation at a certain time, by using a descriptive relation that assigns concepts to entities within that situation*.
The '*' variables are *ordered-list variables*, i.e. they can occur more than one time in an orderly way (ordered lists are paired, based on the projections described in section 3.3). Without list variables, *c.DnS* relation would formalize only 'atomic' situations, based e.g. on only one concept, one entity, one time interval, etc.). In real modelling (see 3.3), several occurrences of argument types are possible, for example admitting different agent's and situation's times, and several entities within a same situation, as when a detective singles out an event occurred days before, for the sake of interpreting a killer's *modus operandi*. Such a case is exemplified in the statement 2, which contains four entities, four concepts, and two time intervals.

$$\begin{aligned}&c.DnS(KnowledgeOfPreviousCases\#1, KillingSituation\#1,\\ &\qquad \{Precedent, Killer, Tool, HypotheticalIntention\},\\ &\qquad \{Event\#1, PhysicalAgent\#1, PhysicalTool\#1, Plan\#1\},\\ &Detective\#1, InvestigationTeam\#1, PreviousCaseReport\#1,\\ &\qquad \{TimeOfEvent\#1, TimeOfInterpretation\#1\})\end{aligned} \tag{2}$$

In the following, list variables are not used, because *c.DnS* relation is best explained through its projections, which make it useful in most real world projects, where computational languages do not allow (or make it too complex) representing list variables and >2-ary relations.

3.2 Characterization of classes

E is the class of everything that is assumed to exist in some domain of interest for any possible world. (3):

$$\Box\forall x(E(x)) \tag{3}$$

D, S, C, A, K, I, and T are subclasses of E (1):

$$(D(x) \vee S(x) \vee C(x) \vee A(x) \vee K(x) \vee I(x) \vee T(x)) \rightarrow E(x) \tag{4}$$

D, S, C, A, K, and I are all mutually disjoint, and constitute the class SE of *schematic entities* (5).

$$SE(x) =_{df} D(x) \vee S(x) \vee C(x) \vee A(x) \vee K(x) \vee I(x) \tag{5}$$

All instances of E that are not instances of SE are *non-schematic* entities. SE and non-schematic entities cover the class E.

$$E(e) \equiv SE(e) \vee \neg SE(e) \tag{6}$$

Since time intervals are not in SE, they are non-schematic entities. Time intervals are important in *c.DnS* because we need to add a temporal indexing to some constructivist relations. In practice, when using *c.DnS* jointly with an existing ontology that does not reflect any commitment to time intervals, we need adding time intervals to it.

$$T(t) \rightarrow \neg SE(t) \tag{7}$$

The main application of *c.DnS* is *redescribing* existing entities, independent of such existence being derived from other (formal or informal) ontologies or assumed. For example, an existing situation including humans, cars, roads and signs can be redescribed as a driving situation, as a racing situation, as well as a speed-limit-violation situation, depending on other circumstances and on the intention of the interpreter of that situation. We define a $redescription$ relation as a partial projection of *c.DnS* as follows:

$$redescription(a, e, s, t) \rightarrow A(a) \wedge E(e) \wedge S(s) \wedge T(t) \tag{8}$$

Axiom 8 states that redescription holds for agents and entities within a situation, at some time. For example, the sentence *the Italian road police has fined Manuel Fangio for a speed-limit violation on Thursday, January 18th 2007* can be modelled by using the redescription relation (9, 10).

$$\begin{gathered} redescription(ItalianRoadPolice, ManuelFangioDriving, \\ SpeedLimitViolation18010732, ThursdayJanuary18th2007) \end{gathered} \tag{9}$$

$$\begin{gathered} redescription(ItalianRoadPolice, ManuelFangio, \\ SpeedLimitViolation18010732, ThursdayJanuary18th2007) \end{gathered} \tag{10}$$

Each redescription concerns one of the entities that get redescribed in that situation. Statements 9 and 10 exemplify two such entities: Manuel Fangio, and his driving.

Based on the redescription relation, we define a class of $GroundEntities$ as those entities that get redescribed:

$$G(e) =_{df} E(e) \wedge \exists a, s, t(redescription(a, e, s, t)) \tag{11}$$

Definition in 11 introduces ground entities as entities that are redescribed by an agent that 'frames' them within a situation at some time.

For the sake of intuition, the generic relation of *interpretation* (12) between an agent and an entity whatsoever at some time can be considered as a poorer projection of *c.DnS*:

$$interprets(a, e, t) \rightarrow A(a) \wedge E(e) \wedge T(t) \tag{12}$$

In practice, the constructive assumption in our ontology makes interpretation of entities by some agent at some time logically dependent on descriptions, situations, concepts, collectives, and information objects. How this assumption is unfolded, both formally and intuitively, is the theme of the next subsections. In the remainder of *c.DnS* presentation, we show how the *c.DnS* relation is projected and axiomatized so that the $redescription$ relation can be actually used in as many domains as possible.

Additional entity types While a purely constructivist theory can live without postulating types of entities (besides time intervals), some common-sense type distinctions are of obvious practical advantage in the management of many social domains. In particular, we will present both a purely *constructivist* and a *grounded* (or 'common-sense') versions of the construction principle. In the grounded version, we make use of some types of non-schematic entities: PA (*physical agents*), R (internal representations), and PR (*physical realizations*). In the plan ontology (see below), we also use the classes $Object$, $Action$, and $Region$.

3.3 Projections of *c.DnS*: the *social* construction principle

Several projections of the *c.DnS* relation can be defined by means of binary or ternary relations, and axioms. Most projections we consider are irreflexive, asymmetric, and intransitive. Besides the specific projection signature, their axiomatization requires a temporalized *properPartOf* relation.
Here we list, mostly in an informal way, the projections that we deem necessary in order to lay a foundation for *c.DnS*. Each projection implies the full *c.DnS* relation according to the axiom schema in 13; these implication axioms should be assumed, and are not included in the axiomatization.

$$[projection](x_1...x_n, x_i \in \{d,s,c,e,a,k,i,t\}) \rightarrow \\ c.DnS(d,s,c,e,a,k,i,t) \qquad (13)$$

Each projection is introduced as a *principle* for the logical representation of the construction of the social realm. The principles proposed will be finally composed into a *social construction principle*. Altogether, they constitute an extended account of the social constraints acting when an agent's ontological commitment is formed. On the other hand, one or more principles could be dropped if considered unnecessary or too strong for a particular ontology project (this implying that the construction principle is lost).
For the complete axiomatization, and for technical details on how *c.DnS* is applied in domain ontology projects, we refer to the technical reports from our ontology portal [7].
The following is the signature of the (basic) *c.DnS* projections:

$$\langle specializes, defines, uses, satisfies, classifies, involves, settingFor, \\ shares, deputes, hasInScope, redescribes, expresses, about, \\ memberOf, covers, characterizes, unifies, gUnifies\rangle$$

The relationality principle The $defines$ relation (14) is the projection of *c.DnS* over descriptions and concepts (cf. [20]).

$$defines(d,c) \rightarrow D(d) \wedge C(c) \qquad (14)$$

$Defines$ formalizes the intuition of a *gestalt* [36], or 'context' [32], that gives meaning to the parts. Some examples iare modelled in 15, 16, 17.

$$defines(ItalianConstitution, Minister) \qquad (15)$$

$$defines(LinneanTaxonomy, Species) \qquad (16)$$

$$defines(CNRRegulation, SeniorResearcher) \qquad (17)$$

If we assume that a $defines$ relation is required for concepts, i.e. that concepts are always defined in a relational context - a $description$ -, that assumption can be called the $relationality$ principle (18).

$$C(c) \rightarrow \exists d(D(d) \wedge defines(d,c)) \qquad (18)$$

[7] http://www.loa-cnr.it/ontologies/index.html

The $uses$ relation (axiom 19, exemplified in Statement 20) reflects the fact that, besides defining concepts, descriptions can also use concepts defined by some other description.

$$uses(d,c) \rightarrow D(d) \wedge C(c) \wedge \exists d'(d \neq d' \wedge defines(d',c)) \tag{19}$$

$$uses(ChiefOfStateVisitEtiquette, MasterOfCeremonies) \tag{20}$$

Descriptions can also *introduce* social agents, which are here entities such as persons, organizations, institutional figures, etc. (see 21, with some examples in 22, 23, 24)).

$$introduces(d,a) \rightarrow D(d) \wedge A(a) \tag{21}$$

$$introduces(ItalianConstitution, ItalianGovernment) \tag{22}$$

$$introduces(FIATLegalConstitution, FIAT\_SpA) \tag{23}$$

$$introduces(ItalianLawBirthDeclaration, PhysicalLegalPerson) \tag{24}$$

Although introduction of agents falls under the relationality principle, like definition and usage, it has a different intuition from definition and usage, because concepts and agents are disjoint classes, where the differences are:

- agents can *share* descriptions (see section 3.3)
- agents (specially *organizations*) typically *depute* concepts (see axiom 41)
- in the grounded version of the construction principle (see section 3.4), social agents are *acted by* (axiom 81) some physical agent (axiom 84) that is classified by (axiom 29) some concept deputed by (axiom 41) a social agent

The interpretability principle The $satisfies$ relation is the projection of *c.DnS* over situations and descriptions.

$$satisfies(s,d) \rightarrow S(s) \wedge D(d) \tag{25}$$

It formalizes the intuition of an *instantiation* of a gestalt, i.e. the application of gestalts to actually occurring contexts in the life of a cognitive agent. For example:

$$\begin{aligned} &satisfies(MandateForGovernmentToProdi,\\ &\qquad LawForGovernmentFormation) \end{aligned} \tag{26}$$

If we assume that a $satisfies$ relation is required for situations, i.e. that situations are always emerging/interpreted in a relational context according to some expected configuration, that assumption can be called the $interpretability$ principle.

$$S(s) \rightarrow \exists d(D(d) \wedge satisfies(s,d)) \tag{27}$$

Each description generates a *situation class*, which contains all the situations that satisfy that description. For example,

$$\begin{aligned} &satisfies(x, LawForGovernmentFormation) \rightarrow\\ &\qquad LegalGovernmentFormation(x) \end{aligned} \tag{28}$$

A situation class can be empty however, since there may be descriptions that are never satisfied by any situation.

The classification principle The $classifies$ relation is the projection of *c.DnS* over concepts and entities.

$$classifies(c, e, t) \rightarrow C(c) \wedge E(e) \wedge T(t) \tag{29}$$

It formalizes the intuition of a *redescription* of an entity, i.e. the application of a (new) gestaltic concept to something which is already provided with an available identity in actually occurring contexts in the life of a cognitive agent. For example, the statement 30 has the consequence that the social agent Napolitano is provided with the additional identity of ItalianPresidentRole for 2007:

$$classifies(ItalianPresidentRole, Napolitano, 2007) \tag{30}$$

Note that *ItalianPresident* is a social agent, since it has the properties given in section 3.3; anyway, that social agent also needs to *depute* (41) the concept *ItalianPresidentRole*[8]that can classify different entities at different times, but only one at a time, while other concepts admit to classify different entities at the same time, e.g. the concept *Senator* (31). The different ways of classifying entities usually depend on the type of agent that deputes the concept, see section 3.4.

$$classifies(Senator, Napolitano, 2005) \tag{31}$$

$$classifies(Senator, LeviMontalcini, 2005) \tag{32}$$

If we assume that a $classifies$ relation is required for ground entities to be considered in *c.DnS*, i.e. ground entities are always internally represented with reference to a concept, that assumption can be called the $classification$ principle.

$$G(x) \rightarrow \exists c(C(c) \wedge classifies(c, x, t)) \tag{33}$$

Compositional projections can be defined from primitive ones. The projection *involves* is compositionally defined, and states that a description involves a ground entity when the latter is classified by a concept defined or used by the description.

$$\begin{aligned} involves(d, e, t) =_{df} D(d) \wedge E(e) \wedge T(t) \wedge \\ \exists c((defines(d, c) \vee uses(d, c)) \wedge classifies(c, e, t)) \end{aligned} \tag{34}$$

The situatedness principle The $settingFor$ relation is the projection of *c.DnS* over situations and entities.

$$settingFor(s, e) \rightarrow S(s) \wedge E(e) \tag{35}$$

It formalizes the intuition of *contextualization* of an entity, i.e. the application of gestalts to actually occurring contexts in the life of a cognitive agent. For example:

$$settingFor(MandateForGovernmentToProdi, Napolitano, 2007) \tag{36}$$

If we assume that a $settingFor$ relation is required for ground entities, i.e. ground entities are always internally represented within a context according to some expected or reconstructed configuration, such assumption can be called the $situatedness$ principle (37).

$$G(x) \rightarrow \exists s(S(s) \wedge settingFor(s, x)) \tag{37}$$

[8] Differently from the legal one, the common sense notion of Italian president is usually that of a concept, not of a social agent

The sharing principle The $shares$ relation is the projection of *c.DnS* over social agents and descriptions.

$$shares(a,d,t) \rightarrow A(a) \wedge D(d) \wedge T(t) \tag{38}$$

It formalizes the intuition of the *social nature* of a description, i.e. the mapping of descriptions on social agents that are acted by one or more physical agents. Note that by ‘social nature’ we do not mean that a description should actually be shared by a community (although this is typically what happens), but that a description must be communicable among social agents. For example:

$$shares(Napolitano, LawForGovernmentFormation, 2006) \tag{39}$$

If we assume that a $shares$ relation is required for descriptions, i.e. descriptions are always dependent on social agents, that assumption can be called the $sharing$ principle.

Notice that social agents include both persons and organizations, but not physical systems that can play agentive roles (these are introduced in the grounded version of DnS, see section 3.4).

The sharing principle states that descriptions must be shared by at least one agent (40).

$$D(x) \rightarrow \exists(a,t)(A(a) \wedge T(t) \wedge shares(a,x,t)) \tag{40}$$

Besides sharing descriptions, social agents can *depute* (41) concepts (e.g. *roles*) that are supposed to enact the agent’s actions (see section 3.4).

$$deputes(a,c,t) \rightarrow A(s) \wedge C(c) \wedge T(t) \tag{41}$$

For example, a telecom company can depute the role ’engineer’ that can classify certain entities (typically, persons with appropriate curricula) to act for the company. Back to our legal example:

$$deputes(ItalianState, ItalianPresident, 2006) \tag{42}$$

The epistemological layering principle The $hasInScope$ relation reflects the intuition that situations can be *epistemologically layered*, when a description d_1 involves another description d_2, and a situation s_1 that *satisfies* d_1 has another situation s_2 that *satisfies* d_2 in its scope. (43).

$$\begin{aligned} hasInScope(s_1,s_2) =_{df}\ & S(s_1) \wedge S(s_2) \wedge s_1 \neq s_2 \wedge \\ & \exists(d_1,d_2,a,t)(D(d_1) \wedge D(d_2) \wedge A(a) \wedge T(t) \wedge \\ & d_1 \neq d_2 \wedge satisfies(s_1,d_1) \wedge satisfies(s_2,d_2) \wedge \\ & involves(d_1,d_2,t) \wedge shares(a,d_1,t) \wedge shares(a,d_2,t)) \end{aligned} \tag{43}$$

An example is in 44.

$$hasInScope(MurderCase_{128}, CaesarStabbedByBrutus) \tag{44}$$

Epistemological layering is a principle in the *c.DnS* core, corresponding to the *figure-ground shifting* cognitive schema from Gestalt psychology and, more recently, cognitive linguistics [36], [37].

The application of epistemological layering is fundamental in *c.DnS*, since it accounts for the role of agents in the application of a description to some situation, i.e., in order to include the ontological commitment within an ontology's domain of discourse. In practice, ontological commitment [38] postulates the action of some agent that has the capability and the intention to *(re)describe* (=interpret) a situation.
This is formalized by means of the relation $redescribes$ (45), which is the projection of *c.DnS* over social agents and situations.

$$\begin{aligned} redescribes(a, s_2, t) =_{df}\ & A(a) \wedge S(s_2) \wedge T(t) \wedge \\ & \exists s_1(S(s_1) \wedge s_1 \neq s_2 \wedge shares(a, d_1, t) \wedge \\ & satisfies(s_1, d_1) \wedge hasInScope(s_1, s_2)) \end{aligned} \tag{45}$$

An example of application of redescribes is in 46.

$$redescribes(SherlockHolmes, HoundOfBaskervilleFact, 1890) \tag{46}$$

The formedness principle The $expresses$ relation is the projection of *c.DnS* over information objects and descriptions. It formalizes the intuition of the intrinsic *communicability* of every description (cf. [34]).

$$expresses(i, d, t) \rightarrow I(i) \wedge D(d) \wedge T(t) \tag{47}$$

For example:

$$expresses(ItalianConstitutionText, ItalianConstitution, 1946) \tag{48}$$

We call $formedness$ principle (49) the assumption that an $expresses$ relation is required for descriptions, i.e. descriptions are always expressed by information objects that provide a form to them.

$$D(x) \rightarrow \exists(i, t)(I(i) \wedge (T(t) \wedge expresses(i, x, t)) \tag{49}$$

Information objects appear in other projections of *c.DnS*, which can be defined compositionally. For example, the *aboutness* of information objects can be defined through composing the $expresses$, $satisfies$, and $settingFor$ relations (50).

$$\begin{aligned} about(i, e, t) =_{df}\ & I(i) \wedge E(e) \wedge T(t) \wedge \exists(d, s)(D(d) \wedge S(s) \wedge \\ & expresses(i, d, t) \wedge satisfies(s, d) \wedge settingFor(s, e)) \end{aligned} \tag{50}$$

Aboutness states that, if the description expressed by an information object is satisfied by a situation, the information object can be *about* any entity that is in the setting of said situation. For example, in 51, the Italian Constitution is (also) about Italy.

$$about(ItalianConstitutionText, Italy, 2006) \tag{51}$$

The containment principle The entities that are classified by a same concept or a same set of concepts, either defined by the same description or not, are easier to compare, and can be put in a same collection (K). An appropriate $memberOf$ relation (52) holds for sets of said entities[9]

$$memberOf(e,k,t) \rightarrow E(e) \wedge K(k) \wedge T(t) \tag{52}$$

For example, in 54, D'Alema is a member of the Italian Government collective in 2007. Italian Government collective is intended here as the collection of all members from any particular Italian Government.

$$memberOf(D'Alema, ItalianGovernmentCollective, 2007) \tag{53}$$

Note that *ItalianGovernmentCollective* is not the same entity as *ItalianGovernment*, which is a social agent. The difference is not purely academic, because Italian Government identity depends on the current Italian Constitution, and Italian Government collective is bound to it; but there are collectives of Italian governments that have different identities based on the way they have been elected, nominated, or behaved.
For example, from the legal viewpoint, *Prodi2Collective* has a different identity from the general Italian Government collective, although the two collectives are co-extensional (have the same members) for a limited time period (see example **??**). Of course, also Prodi2 is a social agent, which *specializes* ItalianGovernment (cf. 55).

$$memberOf(D'Alema, Prodi2Collective, 2007) \tag{54}$$

$$specializes(Prodi2, ItalianGovernment) \tag{55}$$

If we postulate a collection comprising all entities classified by a concept, for each concept, the resulting axiom represents the *containment* principle (56).

$$\begin{aligned} C(c) \rightarrow \forall(e,t)((E(e) \wedge T(t) \wedge classifies(c,e,t)) \rightarrow \\ \exists x(K(x) \wedge memberOf(e,x,t)) \end{aligned} \tag{56}$$

The concept(s) that classify all the members of a collection are said to *cover* a collection (57):

$$\begin{aligned} covers(c,k) =_{df} C(c) \wedge K(k) \wedge \\ \forall(e,t)((E(e) \wedge memberOf(e,k,t)) \rightarrow classifies(c,e,t)) \end{aligned} \tag{57}$$

Statement 58 is about the fact that the collective *ItalianMinisterCouncil* has all members that are classified by the concept *Minister*.

$$covers(Minister, ItalianMinisterCouncil) \tag{58}$$

Many collections can have subcollections covered by different concepts. In that case, we say that those concepts *characterize* the collection (59). Since subcollections can change without affecting the identity of a collection, *characterizes* is temporalized.

$$\begin{aligned} characterizes(c,k,t) =_{df} C(c) \wedge K(k) \wedge T(t) \\ \exists(k_1)(covers(c,k_1) \wedge properPartOf(k_1,k,t)) \end{aligned} \tag{59}$$

[9] Cf. [26] and [3] for a different axiomatization that also assumes DOLCE).

Statement 60 is about the fact that (from a socio-political viewpoint), the collective *ItalianMinisterCouncil* has some members that are classified by the concept *Reformer*.

$$characterizes(Reformer, ItalianMinisterCouncil, 2006) \tag{60}$$

A complex concept, whose component concepts collectively characterize all members of a collection, results to *cover* it (cf. 61).

$$\begin{aligned}\forall(c,k)(C(c) \wedge K(k) \wedge \exists(c_1,c_2,t)(c_1 \neq c_2 \wedge\\ characterizes(c_1,k,t) \wedge characterizes(c_2,k,t) \wedge\\ properPartOf(c_1,c,t) \wedge properPartOf(c_2,c,t) \wedge\\ \neg\exists(c_3)(characterizes(c_3,k,t) \wedge c_1 \neq c_3 \wedge c_2 \neq c_3)) \rightarrow\\ covers(c,k,t))\end{aligned} \tag{61}$$

The descriptions that define the concept(s) or concept collections that *cover* a collection are said to *unify* it (62).

$$\begin{aligned}unifies(d,k) =_{df} D(d) \wedge K(k) \wedge\\ \exists(c)(defines(d,c) \wedge covers(c,k))\end{aligned} \tag{62}$$

Statement 63 is inferred from 62, 15, and 58: since *unifies* composes the relations *defines* and *covers*, the description *ItalianConstitution* defines the concept *Minister*, and *Minister* covers the collection *ItalianMinisterCouncil*, then *ItalianConstitution* unifies *ItalianMinisterCouncil*.

$$unifies(ItalianConstitution, ItalianMinisterCouncil) \tag{63}$$

When unification is applied to the parts of an entity, so that the unifying description defines concepts that characterize the configuration aspects of that entity, the collection is called *configuration* (Cfg, definition 64).

$$\begin{aligned}Cfg(k) =_{df} K(k) \wedge\\ \forall(e,t)(memberOf(e,k,t) \rightarrow\\ \exists(e_1,d,t)(properPartOf(e,e_1,t) \wedge\\ unifies(d,k) \wedge involves(d,e_1,t))\end{aligned} \tag{64}$$

For example, the collection of all parts of a car, when the unifying description is its functional design, is a configuration.

In case a collection is covered or characterized by more than one concept defined in different descriptions, so that all entities in the collection are classified by characterizing concepts, then the collection results to be unified by a *bundle* of descriptions, in which the characterizing concepts are defined (definition 65).

$$\begin{aligned}Bundle(d) =_{df} D(d) \wedge \exists(d_1,d_2,t)(properPartOf(d_1,d,t) \wedge\\ properPartOf(d_2,d,t) \wedge (\exists(k,c_1,c_2)(defines(d_1,c_1) \wedge\\ defines(d_2,c_2) \wedge characterizes(c_1,k) \wedge\\ characterizes(c_2,k) \wedge unifies(d,k)) \vee\\ (\exists(s)(satisfies(s,d_1) \wedge satisfies(s,d_2) \wedge satisfies(s,d)))))\end{aligned} \tag{65}$$

The notion of bundle of descriptions is defined in 65: a bundle is a (mereological) sum of (at least two) descriptions that are either all satisfied by a situation, or all define concepts that characterize a same collection.

The taxonomy principle The $specializes$ relation (66) is the projection of *c.DnS* between schematic entities (in [20] this relation is limited to concepts only). It conveys the intuition of a taxonomic schema across schematic entities, for example in 67, the social agent *Prodi2 Government* specializes *Italian Government*.

$$specializes(se_1, se) \rightarrow SE(se_1) \wedge SE(se) \tag{66}$$

$$specializes(Prodi2Government, ItalianGovernment) \tag{67}$$

The difference between *specializes* and the traditional *subClassOf* and *instanceOf* relations is subtle. Firstly, specializes can be considered as a *reification* of subClassOf, since the latter holds for logical classes, while specializes holds for schematic entities.[10]
Secondly, since we are using first-order logic with a model-theoretic semantics, the subClassOf and instanceOf relations can also be used with schematic entities, and the choice between specializes and instanceOf often results to be a matter of good practice. For example, we may want to consider *Government* as a class instead of a social agent, if there is no given description that *introduces* (cf. axiom 21) government as a social agent. On the contrary, *ItalianGovernment* is introduced by the description *ItalianConstitution*, therefore it can be suitably modeled as a social agent. As a consequence, Government is *subClassOf* A (Social Agent), Prodi2Government *specializes* ItalianGovernment, and both Prodi2Government and ItalianGovernment are *instanceOf* Government.

The interaction principle A constructivist ontology should be able to contextualize agents' knowledge within their communities, called *thought collectives* in (cf. [39]). To this purpose, we introduce a class KC of knowledge communities, as a collection whose members share at least one description (68).

$$KC(k) =_{df} K(k) \wedge \exists(d)(D(d) \wedge \forall(a,t)(memberOf(a,k,t) \rightarrow shares(a,d,t))) \tag{68}$$

A knowledge community is different from a simple agent collection, as in cases like biological species, epidemiological groups, etc., since the members of a simple agent collection do not necessarily share any description, and it is even doubtful if agents like members of a biological or clinical group could be considered as agents at all, in the sense proposed here.
When the *membership* relation is considered necessary for descriptions to exist, the resulting axiom corresponds to the $interaction$ principle (69): any social agent must be member of a knowledge community in order to share a description.

$$shares(a,d,t) \rightarrow \exists kc(memberOf(a,kc,t) \wedge KC(kc) \wedge unifies(d,kc) \wedge \exists d_1(\forall a_1(memberOf(a_1,kc,t) \rightarrow shares(a_1,d_1,t))) \tag{69}$$

[10] In a similar vein, descriptions can be considered as reifications of intensional relations, concepts as reifications of intensional classes, situations as reifications of extensional relations, and collections as reifications of extensional classes.

The statements in 70 through 75 exemplify how sharing is independent from *assuming* a description: *FlogistonTheory* was shared by both Stahl and Lavoisier, but only Stahl assumed it; on the other hand, it is not sure that Stahl actually shared *OxygenTheory*, because in the original debate there is no proof that he understood it, therefore, we are only allowed to state that Lavoisier shared (and assumed) it.

$$shares(Stahl, FlogistonTheory) \tag{70}$$

$$shares(Lavoisier, FlogistonTheory) \tag{71}$$

$$shares(Lavoisier, OxygenTheory) \tag{72}$$

Assumes is here proposed (73) as a more specific way of sharing a description, but without defining it. Defining assumption would require much more, e.g., we should axiomatize the relation between assumptions of descriptions, and beliefs about situations: while sharing a description is certainly required to an agent in order to believe a situation that satisfies that description, it is not sufficient to conclude that sharing is sufficient to that agent to actually believe it. The issue is even subtler, because we cannot either conclude that assuming that description is sufficient to believe that situation, since there can be additional constraints that make a situation unbelievable. Conversely, there can be cases in which a situation is believed without assuming the description it satisfies. We do not attempt an axiomatization of these epistemological issues here.

$$assumes(a, d, t) \rightarrow shares(a, d, t) \tag{73}$$

$$assumes(Stahl, FlogistonTheory) \tag{74}$$

$$assumes(Lavoisier, OxygenTheory) \tag{75}$$

Although we stay neutral with reference to how assumptions and beliefs are intertwined, we can use *assumes* as a primitive to introduce the notion of *paradigm* (76), which is important for a constructive ontology and to characterize collectives. Paradigms are defined here as bundle-based configurations of descriptions that are assumed by the members of a knowledge community. Those knowledge communities (common either in the commonsense or the scientific domains) result to be unified by paradigms.

$$\begin{aligned} Paradigm(p) =_{df} Bundle(p) \wedge \forall(d, t)(properPartOf(d, p, t) \rightarrow \\ D(d) \wedge \exists(kc)(KC(kc) \wedge \forall(a)(memberOf(a, kc, t) \rightarrow \\ assumes(a, d, t)))) \end{aligned} \tag{76}$$

The social construction principle The unification relation holding for collections can also be used for ground entities, and we generalize a temporal version of it for all ground entities (*gUnifies*, or generalized unification, 80).

A composition of *c.DnS* projections leads to the *social construction principle* (77):

$$\begin{aligned} G(x) \leftrightarrow \\ \exists(d,s,c,a,i,kc,t,c_1,d_1,s_1,t_1)(D(d) \wedge S(s) \wedge C(c) \wedge A(a) \wedge \\ I(i) \wedge KC(kc) \wedge T(t) \wedge C(c_1) \wedge D(d_1) \wedge S(s_1) \wedge \\ T(t_1) \wedge classifies(c,x,t) \wedge settingFor(s,x) \wedge defines(d,c) \wedge \\ satisfies(s,d) \wedge shares(a,d,t) \wedge unifies(d,kc) \wedge \\ memberOf(a,kc,t) \wedge deputes(a,c_1,t) \wedge expresses(i,d,t) \wedge \\ settingFor(s,t) \wedge redescribes(a,s,t_1) \wedge shares(a,d_1,t_1) \wedge \\ gUnifies(d_1,d,t_1) \wedge satisfies(s_1,d_1) \wedge hasInScope(s_1,s)) \end{aligned} \quad (77)$$

According to the social construction principle, when redescribed by *c.DnS*, a ground entity x gets characterized as follows:

- x is always *classified* at some time by at least one concept that is *defined* in a description that is *satisfied* by a situation that is a *setting* for x
- x description has to be *shared* by a social agent that is a *member* of a knowledge community
- the description has to be *expressed* by an information object
- the social agent has to *depute* concepts that classify entities from a situation
- the social agent that shares x's description *redescribes* x's situation by means of *involving* x's description into another description. This is equivalent to having x's situation *in the scope of* the redescription situation. [11]

The social construction principle can be interpreted as a *unity criterion* (cf. also [40] for a non-reified account of unity criteria), which becomes available to x. In other words, its redescription allows x to be unified (80). The unification relation holding for collections (62) can be used for any ground entity, and we generalize a temporal version of it for all ground entities (*gUnifies*, or generalized unification, 78).

$$\begin{aligned} gUnifies(d,g,t) =_{df} D(d) \wedge G(g) \wedge T(t) \wedge \\ \exists(c)(C(c) \wedge defines(d,c) \wedge classifies(c,g,t) \end{aligned} \quad (78)$$

The intuition of generalized unification is that we can imply a "singleton" collection that is covered by the concept c (axiom 79), and whose unique member is the ground entity g. Since membership is temporalized, we need a temporal index in 78.

$$\begin{aligned} gUnifies(d,g,t) \rightarrow \exists(k,c)(K(k) \wedge memberOf(g,k,t) \wedge \\ unifies(d,k) \wedge defines(d,c) \wedge covers(c,k) \wedge \\ \neg\exists(e)(g \neq e \wedge memberOf(e,k,t)) \end{aligned} \quad (79)$$

Another simpler way to explain the notion of ground entity is therefore based on $gUnifies$ (axiom 80).

$$G(x) \leftrightarrow \exists(d,t)(gUnifies(d,x,t)) \quad (80)$$

[11] Note that there is no room for infinite regression, because the construction principle does not apply to ground entities that are also SE, therefore the redescription situation is not itself in the scope of a further redescription situation, unless requested by the case.

3.4 Projections of *c.DnS*: the *grounded* construction principle

So far, we have only concentrated on the core DnS relation, which focuses of schematic entities and how they can be used to provide ground entities with a unity criterion. On the other hand, we have also anticipated that a more practical version of DnS can include a mild commitment to certain types of (non-schematic) entities. In the following, we introduce new relations that will eventually allow the specification of a grounded construction principle.
The following is the signature of the additional projections for a grounded *c.DnS*:

$$\langle actsFor, constructs, individuallyConstructedAs, realizes\rangle$$

The agentEfficacy principle In 3.3 we have assumed that social agents can $depute$ (41) concepts (e.g. *roles*) that are supposed to classify the entities that can act for the agent. For example, a telecom company can depute the role engineer that can classify certain entities (typically, natural persons with appropriate curricula) to act for the company.
The $actsFor$ (81) relation holds for entities and social agents. It formalizes the intuition of *acting for* a social agent, i.e. the mapping of entities as actors that are classified by concepts that are *deputed by* a social agent.

$$actsFor(e,a,t) =_{df} E(e) \wedge A(a) \wedge T(t) \wedge \exists c(classifies(c,e,t) \wedge deputes(a,c,t)) \quad (81)$$

An example is provided in 42.

$$actsFor(Napolitano, ItalianState, 2007) \quad (82)$$

If we assume that an $actsFor$ relation is required for social agents, i.e. social agents should always be acted by some entity, that assumption can be called the $agentEfficacy$ principle (83).

$$A(x) \rightarrow \exists(e,t)(E(e) \wedge actsFor(e,x,t)) \quad (83)$$

Typically, social agents are acted by physical organisms, but actors can also be natural and legal persons, animals, robots, or even viruses. However, agent efficacy could be supported with a stronger claim, i.e. the $rationalAgentEfficacy$ principle, stating that social agents must be acted by entities that can have internal meta-representations, hence only by (a subclass of) cognitive systems.

The cognitiveCounterpart principle We introduce a class for entities that can ground the action of social agents, and call them *physical agents*, or PA (84).

$$PA(pa) \rightarrow E(pa) \wedge \neg SE(pa) \quad (84)$$

Similarly, we ground descriptions in entities that can be localized into individual physical agents, and call them *internal representations* (85).

$$R(r) \rightarrow E(r) \wedge \neg SE(r) \quad (85)$$

The $constructs$ relation holds for physical agents (or cognitive systems) and internal representations. Physical agents are considered non-schematic entities, so we have to identify them in the domain of existing ontologies, and possibly add them to an ontology when missing (similarly to time intervals).

$$constructs(pa, r, t) \rightarrow PA(pa) \wedge R(r) \wedge T(t) \tag{86}$$

Statement 87 exemplifies grounded construction.

$$\begin{aligned} & constructs(NapolitanoAsOrganism, \\ & N.'sRepresentationOfItalianConstitution, 2007) \end{aligned} \tag{87}$$

The $individuallyConstructedAs$ relation is the projection of *c.DnS* over descriptions and internal representations. It formalizes the correlate of descriptions as internal representations in a physical agent (intended as a cognitive system) that $actsFor$ a social agent that $shares$ the description. A *cognitiveCounterpart* principle states that for any description there is a social agent that shares it, and which is acted by a physical agent that constructs an internal representation that is the individual construction of that description.

$$\begin{aligned} & individuallyConstructedAs(d, r, t) \rightarrow \exists(pa)(PA(pa) \wedge \\ & shares(a, d, t) \wedge constructs(pa, r, t) \wedge actsFor(pa, a, t)) \end{aligned} \tag{88}$$

Statement 89 exemplifies individual grounded construction.

$$\begin{aligned} & individuallyConstructedAs(ItalianConstitution, \\ & N.'sRepresentationOfItalianConstitution, 2007) \end{aligned} \tag{89}$$

The agentEfficacy and the cognitiveCounterpart principles can be composed, in order to create a dependency of schematic entities on non-schematic ones, i.e. assuming that sharing descriptions requires constructing internal representations.

$$\begin{aligned} shares(a, d, t) \rightarrow \exists(pa, r)(PA(pa) \wedge constructs(pa, r, t) \\ \wedge actsFor(pa, a, t)) \end{aligned} \tag{90}$$

The informationGrounding and groundedConstruction principles An important projection concerning the way descriptions are substantially shaped and communicated is the *realizes* relation (93), holding between information objects and (physical) ground entities, which we call *physical realizations*, or PR (91).

$$PR(pr) \rightarrow E(pr) \wedge \neg SE(pr) \tag{91}$$

$$realizes(pr, i, t) \rightarrow PR(pr) \wedge I(i) \wedge T(t) \tag{92}$$

For example, the original paper document of the Italian Constitution realizes the Italian Constitution text in 1946 (93)

$$\begin{aligned} realizes(OriginalPaperDocumentOfItalianConstitution, \\ ItalianConstitutionText, 1946) \end{aligned} \tag{93}$$

This is related to information *grounding*, which is an obvious precondition for communication to happen: any information object must have a physical ‘support’.

Similarly to time intervals and physical agents, physical realizations of information must be present in the ground domain, or need to be added to it.
The agentEfficacy and the cognitiveCounterpart principles can be composed with the informationGrounding principle, in order to create a dependency of description sharing (and their internal construction) on realizing information objects that, as physical realizations, express the descriptions (94).

$$\begin{aligned} shares(a,d,t) \rightarrow \exists(pa,r,i,pr)(PA(pa) \wedge constructs(pa,r,t) \wedge \\ actsFor(pa,a,t) \wedge expresses(i,d) \wedge realizes(pr,i)) \end{aligned} \quad (94)$$

Axiom 94 enables a stronger interpretation of the *redescription relation* (8), since it now requires a social agent to be grounded in a physical agent, a description to be grounded in an internal representation constructed by the physical agent, and an information object to be grounded in a physical realization that realizes it. Stronger redescription results into *grounded scoping* (96), and the *grounded construction principle* (97). Grounded scoping allows us to distinguish two different times for a situation s: the time of its *setting*, and the time of its *redescription*. Only the first is the ‘real’ time of s, while the second one is actually the time of the redescription situation $s' \neq s$, so that $hasGroundedScope(s', s)$.
Notice that not all applications of *c.DnS* need the specification of a redescription situation. Such a situation is postulated by the theory, but its explicit naming and specification are useful only when the epistemological decision concerning interpretation is of some concern. E.g., when assessing witnesses in a legal case, or when selecting between judgments that may have different authoritativeness, trust, or contextual bindings.

$$\begin{aligned} hasGroundedScope(s_1,s_2) =_{df} hasInScope(s_1,s_2) \wedge \\ \exists(a,t,pa,r)(A(a) \wedge T(t) \wedge PA(pa) \wedge R(r) \wedge \\ redescribes(a,s_2,t) \wedge actsFor(pa,a) \wedge constructs(pa,r)) \end{aligned} \quad (95)$$

An example of grounded scoping is 96.

$$\begin{aligned} hasGroundedScope(SherlockHInterpretation, \\ HoundOfBaskervilleFact) \end{aligned} \quad (96)$$

Finally, we present the enriched construction principle with grounding in the axiom 97.

$$\begin{aligned} G(x) \leftrightarrow \\ \exists(d,s,c,a,r,i,kc,t,c_1,pa,pr,d_1,s_1,t_1)(D(d) \wedge S(s) \wedge C(c) \wedge \\ A(a) \wedge R(r) \wedge I(i) \wedge KC(kc) \wedge T(t) \wedge C(c_1) \wedge PA(pa) \wedge \\ PR(pr) \wedge D(d_1) \wedge S(s_1) \wedge T(t_1) \wedge classifies(c,x,t) \wedge \\ settingFor(s,x) \wedge defines(d,c) \wedge satisfies(s,d) \wedge \\ shares(a,d,t) \wedge unifies(d,kc) \wedge memberOf(a,kc,t) \wedge \\ constructs(pa,r,t) \wedge actsFor(pa,a,t) \wedge deputes(a,c_1,t) \wedge \\ classifies(c_1,pa,t) \wedge expresses(i,d,t) \wedge realizes(pr,i,t) \wedge \\ settingFor(s,t) \wedge settingFor(s,pa) \wedge redescribes(a,s,t_1) \wedge \\ settingFor(s_1,pa) \wedge settingFor(s_1,pr) \wedge settingFor(s_1,r) \wedge \\ shares(a,d_1,t_1) \wedge gUnifies(d_1,d,t_1) \wedge satisfies(s_1,d_1) \wedge \\ hasInScope(s_1,s)) \end{aligned} \quad (97)$$

According to the construction principle, when redescribed by *c.DnS*, a ground entity x gets now an additional characterization:

- the description of x has to be shared by a social agent that is acted by at least one physical agent capable of constructing an internal representation
- the description has to be expressed by an information object that is realized by a physical realization
- the social agent has to depute a concept that classifies the physical agent acting for it.

3.5 Plans

Before discussing our typology of collectives, we introduce here some axioms for *plans* [26], which have the following properties (103):

- A **plan** is a description that represents an *action schema*
- Coherently with *c.DnS*, we assume that a plan is *shared* by a social agent, provided that it is *constructed* by a physical agent (90).
- A plan $defines$ or $uses$ at least one *task* (101) and one *role* (100), which are two kinds of concepts.
- A plan has at least one *goal* (108 below) as a proper part (*properPart* is assumed with its usual mereological semantics).

Tasks are concepts that classify action-like entities, which we assume here as **Actions** (98) without a specific characterization, while roles are concepts that classify object-like entities, which are also assumed generically as **Objects** (99)[12]. Finally, roles can have tasks as *targets* (102).

$$Action(e) \rightarrow E(e) \tag{98}$$

$$Object(e) \rightarrow E(e) \tag{99}$$

$$Role(c) =_{df} C(c) \wedge \forall(e,t)(classifies(c,e,t) \rightarrow Object(e)) \tag{100}$$

$$Task(c) =_{df} C(c) \wedge \forall(e,t)(classifies(c,e,t) \rightarrow Action(e)) \wedge \exists(r)(Role(r) \wedge targets(r,c)) \tag{101}$$

$$targets(x,y) \rightarrow Role(x) \wedge Task(y) \tag{102}$$

In [26], roles are explicitly defined as concepts that classify DOLCE objects, while tasks are defined as concepts that classify DOLCE actions, but here we do not make any commitment on how action-like or object-like entities should

[12] The choice of introducing actions and objects as pure primitives follows our practice of avoiding overcommitment, i.e. the attempt to provide axiomatic constraints without a specific need coming from a domain or problem to be represented or solved. We have followed the same practice in general *c.DnS* when introducing time intervals, physical agents and physical realizations with no characterization, except being $entities$

be represented in a (legacy) ontology. Based on previous axioms, a *Plan* class is characterized in 103.

$$\begin{aligned} Plan(d) \rightarrow Description(d) \wedge \exists(a, t, c_1, c_2, g)(shares(a, d, t) \wedge \\ A(y) \wedge T(t) \wedge Task(c_1) \wedge uses(d, c_1) \wedge Role(c_2) \wedge \\ uses(d, c_2) \wedge Goal(g) \wedge properPartOf(g, d, t)) \end{aligned} \tag{103}$$

Examples of plans include: a *way to prepare an espresso in the next five minutes*, a *company's business plan*, a *military air campaign*, a *car maintenance routine*, a *plan to start a relationship*, etc.

Plans can have a rich internal structure, because they can have *subplans*, main and intermediate goals, roles that target more than one task, tasks that are targeted by more than one role, hierarchical tasks and roles, parameters on attributes of entities classified by tasks or roles, etc. A rich axiomatization of plan structures and task types is provided in [26]; here we only concentrate on goals.

Parts of plans A plan can have several *proper parts* (regulations, goals, norms), including other plans. For example, social agents are introduced by *constitutive descriptions* (104); if a plan *introduces* (21) a social agent, the related *constitutive description* is a *proper part* of the plan (105):

$$ConstitutiveDescription(x) =_{df} D(x) \wedge \exists(a)(introduces(x, a)) \tag{104}$$

$$\begin{aligned} Plan(x) \rightarrow introduces(x, a) \leftrightarrow \\ \exists(y)(ConstitutiveDescription(y) \wedge defines(y, a) \wedge \\ properPartOf(y, x, t)) \end{aligned} \tag{105}$$

For example, some plans introduce *temporary* agents, such as *teams* or *task forces*, whose lifecycle starts and ends within the plan lifecycle.

Plans can have subplans (106).

$$\begin{aligned} subPlan(x, y, t) \rightarrow properPartOf(y, x, t) \wedge Plan(x) \wedge \\ Plan(y) \wedge T(t) \end{aligned} \tag{106}$$

Goals are necessary proper parts of plans, and are considered here as desires (another kind of description) that are proper parts of a plan.

$$Desire(x) \rightarrow Description(x) \tag{107}$$

For example, a *desire to start a relationship* can become a *goal to start a relationship* if someone assumes a plan in order to *take action* - or to let someone else take action on her behalf - with the purpose of starting that relationship. We propose a restrictive notion of **goal** that relies upon its desirability by some agent, which does not necessarily play a role in the execution of the plan the goal is part of. For example, an agent can have an attitude towards some task defined in a plan, e.g. *duty towards*, which is different from desiring it (*desire towards*). We might say that a goal is usually desired by the creator or beneficiary of a plan. The minimal constraint for a goal is anyway to be a proper part of a plan:

$$\begin{aligned} Goal(x) =_{df} Desire(x) \wedge \exists(p, t)(Plan(p) \wedge \\ properPartOf(x, p, t)) \end{aligned} \tag{108}$$

Goal dependencies A *main goal* (109) is defined as a goal that is part of a plan but not of one of its subplans (i.e. it is a goal, but not a subgoal in that plan):

$$\begin{aligned} mainGoal(p_1, x, t) =_{df} properPartOf(x, p_1, t) \wedge \\ Plan(p_1) \wedge Goal(x) \wedge T(t) \wedge \\ \neg\exists(p_2)(Plan(p_2) \wedge properPartOf(p_2, p_1, t) \\ \wedge properPartOf(x, p_2, t)) \end{aligned} \tag{109}$$

$$\begin{aligned} subGoal(p_1, x, t) =_{df} properPartOf(x, p_1, t) \wedge \\ Plan(p_1) \wedge Goal(x) \wedge T(t) \wedge \\ \exists(p_2)(Plan(p_2) \wedge properPartOf(p_2, p_1, t) \\ \wedge properPartOf(x, p_2, t)) \end{aligned} \tag{110}$$

It is not necessarily for a subgoal of a plan to be a part of the main goal of that plan. E.g. consider the main goal: *being satiated*; *eating food* can be a subgoal, but it is not a part of *being satiated*. Nonetheless, we can also conceive of an *influence* relation between a subgoal and the main goal of the plan the first goal is a subgoal of (111).

$$\begin{aligned} influenceOn(x, y) =_{df} Goal(x) \wedge Goal(y) \wedge \exists(z, t) \\ (Plan(z) \wedge subGoal(z, x, t) \wedge mainGoal(z, y, t)) \end{aligned} \tag{111}$$

$InfluenceOn$ can be used to talk of expected causal dependencies between goals, either within a same or different plans.

By using the previous definitions, we can also define a *disposition* relation (112) between the roles used in a plan having a main goal, and the influenced goal.

$$\begin{aligned} dispositionTo(x, y) =_{df} (Role(x) \wedge Goal(y) \wedge \\ \exists(p, g, t)(Plan(p) \wedge Goal(g) \wedge mainGoal(p, g, t) \wedge \\ uses(p, x) \wedge influenceOn(g, y)) \end{aligned} \tag{112}$$

For example, the role *eater* can have a disposition to *being satiated*, meaning that a person playing the role of *eater* that adopts that plan can act in order to be satiated.

In interesting cases, supergoals can be created in order to support the adoption of a subgoal. In order to describe these cases, we need an adoption relation for either plans (114) or goals (113).

$$\begin{aligned} adoptsGoal(a, g, t) =_{df} shares(a, g, t) \wedge A(a) \wedge Goal(g) \\ \wedge\forall(p, z)(Plan(p) \wedge Task(z) \wedge uses(p, z) \wedge \\ properPartOf(g, p)) \rightarrow adoptsPlan(a, z, t)) \end{aligned} \tag{113}$$

$$adoptsPlan(a, p, t) \rightarrow shares(a, p, t) \wedge A(x) \wedge Plan(p) \tag{114}$$

Adoption is a kind of sharing, but not a kind of assuming: assuming concerns beliefs, and not executions. From that viewpoint, the BDI (Belief-Desire-Intention) paradigm is not distinctive enough: when some agent adopts a plan, that agent might believe the (meta-fact) that execution will be appropriate if complying to

the adopted plan.
But even in that case, adoption is different from assumption, because in the latter case the assumed description is supposed to be directly satisfied by the believed situation. This is not the case in adoption: an agent does not 'believe' an executed situation or its resulting goal situation, but (maybe) the possibility of its execution or outcome.
In interesting cases, given a plan and its *main* goal, e.g. some service to be delivered, it is a common practice to envisage the *super*goals of the main goal that can be more clearly desirable from e.g. prospective users of a service (for example, a claim like the following generates a supergoal for the service's goal: *our service will improve your life*). These cases can be represented by interlacing adopted goals with influences between them.

Executions Plan executions (115) are situations that proactively satisfy a plan, meaning that plan sharing time precedes (anticipates) its execution time:

$$PlanExecution(s) =_{df} S(s) \wedge \exists(p)(Plan(p) \wedge satisfies(s,p) \wedge \exists(t_1,t_2)(successor(t_1,t_2) \wedge shares(a,p,t_1) \wedge settingFor(s,t_2)) \quad (115)$$

Axiom 116 formalizes that subplan executions are parts of the whole plan execution.

$$\forall(p_1,p_2,s_1,s_2)((Plan(p_1) \wedge Plan(p_2) \wedge properPartOf(p_2,p_1) \wedge PlanExecution(s_1) \wedge PlanExecution(s_2) \wedge satisfies(s_1,p_1) \wedge satisfies(s_2,p_2)) \rightarrow properPartOf(s_2,s_1)) \quad (116)$$

A goal situation is a situation that satisfies a goal:

$$GoalSituation(x) =_{df} S(x) \wedge \exists(y)(Goal(y) \wedge satisfies(x,y)) \quad (117)$$

Contrary to the case of subplan executions, which are part of the overall plan execution, a goal situation is not part of a plan execution:

$$GoalSituation(x) \rightarrow \neg\forall(y,p,s,t)((Goal(y) \wedge Plan(p) \wedge PlanExecution(s) \wedge satisfies(x,y) \wedge properPartOf(y,p) \wedge satisfies(s,p)) \rightarrow properPartOf(x,s,t)) \quad (118)$$

In other words, it is not true in general that any situation satisfying a goal is also part of the situation that satisfies the overall plan. This can account for the following cases:

- Execution of plans containing *abort* or *suspension* conditions (the plan would be satisfied even if the goal has not been achieved)
- *Incidental* satisfaction, as when a situation satisfies a goal without being intentionally planned (but anyway desired).

3.6 Norms

Norms are treated here in their social (including legal as a special case) sense, as some specification of a conceptualization whose objective is regulatory. On the other hand, the very idea of a regulation is far from clearcut, and we prefer to delimit our area of interest to the relations between aspects of social norm *usage*, agents' plans, and agent collectives.

We follow the major distinction proposed by Searle [21] between regulative and constitutive norms. Constitutive norms create social individuals, and can contain very few or no regulation. Here we deal only with regulative norms.

In [35], a norm is defined as a description which a case can fall under, if the entities in that case are properly classified by the concepts (roles, tasks, parameters) used in the norm. Differently from plans, which are *executed* in novel situations, norms are satisfied in a more complex and indirect way, which we formalize in the *fallsUnder* relation (129). Norm execution is limited to the cases represented in 130, and is anyway dependent on plan execution.

Norm aspects Firstly, we must consider at least three different aspects, in which norms relate to plans:

1. Norms as conventions out of existing practices or plans (see axiom 121). A norm, either social or legal, usually reflects an existing good practice within a community. Typically, social and legal systems are the main way to maintain a stability within the members and resources of a community, population, or country, and that stability is dynamically addressed by evolving practices (shared plans), either in a positive form (norm creation), or in a negative one (norm deletion or update). This aspect of norms makes them contributions to social engineering, i.e. to the creation of social reality as reflecting either ideology or existing practices.

   For example, a legal speed limit is supposed to encode the social practice of avoiding excessive speed on vehicles that can be dangerous for people. Therefore, a legal speed limit provides constraints to any plan execution that requires driving at a certain speed. A plan, on its turn, is supposed to encode the way a desire can be realized. Such ways can be limited by existing norms. For example, the execution of my intention to drive fast for my pleasure, or to arrive somewhere in a short time, has constraints coming from a legal speed limit.

   Norms as conventionalized practices depart from plans because they are not executed, and do not have an inherent goal. When (specially in legal domains) there is a talk about the goal or objective of this aspect of a norm, we assume that that goal is actually the goal of social regulation politics, aimed at enforcing established practices that are negotiated appropriately to a community dynamics.

   Alternatively, politics or authoritative social ruling may enforce norms that are not established practices: in that case, the norm is imposed, and the relationships with the other aspects are affected (see section).
2. Norms as compliance checking protocols over social behavior or legal cases (see axiom 123). Once a norm lifecycle is established, norms can be enforced by using them as filters for social behavior. Typically, the initiative for compliance checking is limited to the interest of other parties with respect to the behavior of one party.

   Norms as case descriptors depart from plans because they are not executed, and do not have an inherent goal. On the contrary, the goal involved in this aspect of norms is the goal of an *interpretation* plan that is aimed at finding a social (or specifically legal) framework to a case or behavior. Typical examples of this aspect of norms include e.g. investigations about the (social or legal) responsibility for a certain event that caused damage to someone else. When checking the compliance of visible behavior, in many cases it is necessary to attribute plans to agents' behavior. For this reason, also the plans that can be assumed as being executed in those cases are also involved in this aspect of norms. Moreover, sometimes we must attribute norms as parts of agents' plans, and therefore to assess if and how agents' plans differ from the norm they were expected to follow within those plans.

3. Norms as behavioral rules (constraints) within plans (see axiom 124). Once a norm lifecycle is established, and appropriate enforcing and compliance checking practices emerge, they can be used by social agents as constraints within their own plans. In this sense, norms are akin to *behavioral principles*. If taken as principles for social behaviour, norms can be executed, similarly to plans, and in fact they are executed as subplans.
   Norms as constraints within plans depart from plans only because their goal is dependent on the main goal of an agent that shares the plan they are part of. Typical examples of this aspect of norms include e.g. the assumption of agent's knowledgeability of a norm within a certain community.

These aspects of norms evidence the mutual dependency between plans and norms. Plans are constrained by norms, norms encode conventional plans, and are supposed to constrain agents' plans.

Basic axiomatization of norms Aspects of norms are axiomatized here as additional axioms to the class of norms, because their complementarity makes them parts of a unique ontology design pattern for normative descriptions. A norm is assumed as a description, similarly to plans (119).

$$Norm(x) \rightarrow D(x) \tag{119}$$

Norms are disjoint from plans (120), following the rationale we have given in the previous section.

$$Norm(x) \rightarrow \neg Plan(x) \tag{120}$$

The three aspects of norms are axiomatized as follows: conventionalized practices (121, 122), case descriptors (123), constraints within plans (124). From the viewpoint of a conventionalized practice, a norm uses concepts that are defined by a (usually precedent) social practice (121). In addition, norms are parts of plans that address a community, and use the norms to maintain some kind of social equilibrium (122).

$$\begin{aligned} Norm(x) \rightarrow \exists(p)(Plan(p) \wedge \forall(c)((C(c) \wedge uses(x,c)) \rightarrow \\ defines(p,c)) \end{aligned} \tag{121}$$

$$\begin{aligned} Norm(x) \rightarrow \exists(p,kc)(Plan(p) \wedge KC(kc) \wedge \\ properPartOf(x,p,t) \wedge involves(p,kc,t)) \end{aligned} \tag{122}$$

From the viewpoint of case descriptors, a norm is used by an agent to interpret an agent's behavior (as a plan execution), so providing a redescription for it.

$$\begin{aligned} Norm(x) \rightarrow \exists(pe,a_1,a_2,t_1,t_2) \\ (PlanExecution(pe) \wedge A(a_1) \wedge A(a_2) \wedge setting(a_1,pe,t_1) \wedge \\ shares(a_2,x,t_2) \wedge redescribes(a_2,pe,t_2) \wedge involves(x,pe,t_2)) \end{aligned} \tag{123}$$

From the viewpoint of plan constraints, a norm is a proper part of agents' plans.

$$Norm(x) \rightarrow \exists(p,a,t)(Plan(p) \wedge A(a) \wedge shares(a,p,t) \wedge properPartOf(x,p,t)) \quad (124)$$

Norm application and falling under Having clarified the multiple aspects of social norms, now we can distinguish norm satisfaction, called *appliedIn* (126), obtained by a class of situations called *norm applications* (125), from what we call *falling under* (129). A social situation (case, behavior, etc.) falls under a norm when an agent applies that norm in order to redescribe the situation, either in positive (compliance) or negative (non-compliance) terms.

Formally, a plan execution can: a) be outside the scope of a norm application (127); b) fall under a norm (129); c) execute a norm (130). In the first case, the plan is not constrained by the norm. In the second case, the plan execution is a case for the norm. In the third case, the plan explicitly includes the norm as a proper part.

The *fallsUnder* relation holds between plan executions and norms, and is defined by composing the *hasInScope* relation, and other relations, including *appliedIn*.

$$NormApplication(x) =_{df} Situation(x) \wedge \exists(y)(Norm(y) \wedge satisfies(x,y)) \quad (125)$$

$$appliedIn(x,y) =_{df} Norm(x) \wedge NormApplication(y) \wedge satisfies(y,x) \quad (126)$$

$$outsideTheScopeOf(x,y) =_{df} \neg hasInScope(y,x) \quad (127)$$

$$outsideTheScopeOfNorm(x,y) =_{df} outsideTheScopeOf(x,y) \wedge PlanExecution(x) \wedge NormApplication(y) \quad (128)$$

$$fallsUnder(x,y) =_{df} PlanExecution(x) \wedge Norm(y) \wedge \exists(z)(hasInScope(z,x) \wedge satisfies(z,y)) \quad (129)$$

$$executesNorm(x,y) =_{df} PlanExecution(x) \wedge Norm(y) \wedge \exists(p,pe,t)(Plan(p) \wedge properPartOf(y,p,t) \wedge executes(pe,p) \wedge properPartOf(x,pe,t)) \quad (130)$$

Since norm aspects are complementary and interrelated, also norm axioms and relations are entrenched in interesting ways. For example, norm applications may include different situations, e.g. applying a legal norm to a case requiring an up-to-date interpretation based on current social practices; applying a social norm to overstate the inappropriate behavior of an agent; applying a norm as a principle in regulating an agent's friendship relations, etc.

These different examples of norm application can be made complementary in some cases; e.g. applying an ethical principle with friends may be adopted or not depending on the source of the norm, and if that source is a conventionalized practice in a community whose members share similar descriptions of the social world, that adoption can be easier for, or considered advisable by, a larger number of agents.

3.7 Collectives

We have introduced, so far, two notions of *agent collection*. The first is a simple *collection of agents* (either social or physical), which are unified by some rationale that is extrinsic with respect to the knowledge shared by the agents, e.g. the collection of all agents that use to drink beer, or have green eyes, or the collection of all mosquitoes. The second notion, more relevant for this work, is *knowledge community* (KC, cf. 68), which is a collection of social agents that is unified by descriptions that are shared by the members, e.g., the community of semantic web researchers.

We have also suggested that knowledge communities can be based on *paradigms* (76) whose descriptions are *assumed*, not only shared, specially in scientific communities and communities of practice. The notion of community of practice leads us to a more complex notion of collective, which is based on sharing (or assuming) ways of doing things, i.e. plans and workflows.

Therefore, in this section we augment the notion of knowledge community with more types, based on more specific descriptions that agents can share: plans and norms. We call *intentional collectives* (see definition 131) those knowledge communities that are unified by a plan; while we call *intentional normative collectives* (see definition 134) those knowledge communities which are unified by plans that, in turn, are entrenched with norms, according to the aspects described and formalized in section 3.6. Finally, we introduce *knowledge collectives* (see definition 138) as those intentional normative collectives that also share an *epistemic workflow* (see axiom 137), in order to exchange or modify knowledge.

Social and physical agents in collectives In knowledge communities, covering and characterizing concepts classify social agents. How to talk of knowledge communities and collectives whose members are physical agents? From a constructivist's viewpoint, this move is not necessary; in fact, since social agents need to be acted by physical agents (or cognitive systems), every non-empty knowledge community will be eventually enacted by physical agents. From the grounded construction principle (97), we can infer that, whenever we talk of a knowledge community, the possibility is created of having physical agents that construct internal representations of descriptions shared by social agents. Therefore, in the following we do not make any attempt to distinguish knowledge communities whose members are social from those whose members are physical, since the dual nature of the construction principle (both in the social and grounded version) guarantees expressive means as well as correct inferences.

Intentional collectives As proposed in [3], collective action can only originate from the adoption of a common action schema, i.e. from the unification of a collective by means of a plan. Intentional collectives are defined in a straightforward way in (131) as knowledge communities unified by plans.

$$IntentionalCollective(x) =_{df} KC(x) \wedge \exists(p)(Plan(p) \wedge unifies(p, x)) \tag{131}$$

Organizations, teams, task forces, governments, committees, etc. can be modeled as social agents that are acted by intentional collectives unified by shared plans.

For example, we can generalize over *organizations* as in axiom 132.

$$Organization(x) \rightarrow A(x) \wedge \forall(y,t)(actsFor(y,x,t) \rightarrow \\ \exists(k)(memberOf(y,k,t) \wedge IntentionalCollective(k))) \quad (132)$$

On the other hand, organizations and most free associations are also based on rules. In the next section, we show how this aspect of complex social agents can be modeled.

Intentional normative collectives Plans can be framed in a wider descriptive context, including regulations, normative constraints, social relationship types, etc. In that case, collective action results to emerge from the 'bundle' of descriptions that unifies the collective. Our notion of *Bundle*, introduced in 3.3, helps us in creating another type of knowledge communities, which are defined as unified by a bundle including entrenched plans and norms.
Firstly, we introduce the notion of *NormPlanBundle*.

$$NormPlanBundle(b) =_{df} Bundle(b) \wedge \exists(p,n,t)(Plan(p) \wedge \\ Norm(n) \wedge properPartOf(p,b,t) \wedge properPartOf(n,b,t) \wedge \\ (involves(n,p,t) \vee involves(p,n,t) \vee properPartOf(n,p,t))) \quad (133)$$

Definition 133 refers to entrenchment of norms and plans as three possible cases:

- A norm involves a plan, i.e. when a norm, which is supposed to rule the behavior of a community, considers some typical plans of the agents specifically. E.g. *a speed limit regulation that promotes specific counteractions for the drivers that attempt to escape the enforcement of the norm*
- A plan involves a norm, e.g. *a driving plan is built in order to avoid the consequences of not complying to a speed limit (as when decreasing speed when seeing a police vehicle)*
- A norm is a part of a plan, e.g. *a speed limit is straightforwardly considered as a parameter in a driving plan shared by an agent*

Now, an intentional normative collective (134) is a knowledge community unified by a NormPlanBundle (133):

$$IntentionalNormativeCollective(x) =_{df} KC(x) \wedge \\ \exists(y)(NormPlanBundle(y) \wedge unifies(d,x)) \quad (134)$$

Whereas a NormPlanBundle is explicitly stated ('anticipated'), like in a closed set of tasks that describe, for instance, the possible actions for a social agent, there exists a unique, communicable motivation (the plan defining the tasks) for the collective action.
On the contrary, whereas a bundle is not anticipated, collective action is an *epiphenomenon*, or something that dynamically appears out of local conditions. Here we do not attempt a formalization of epiphenomenic bundles, leaving it to further research.

Knowledge collectives Having introduced norms into the identity criteria for complex agents and their collective action, we are still left with the problem of defining the *knowledge-level* structure of those agents, and how the collectives that act for them can be characterized at that level. In order to do that, we merge the notion of *Paradigm* (76), with that of *NormPlanBundle*. *Epistemic influences* (definition 137) are *NormPlanBundles* that govern the influence between the agents from a community, with respect to their **core** knowledge, i.e. the collection of their assumed descriptions, or *paradigm* (76)[13].In order to define it, we need some specific concept types.

A *knowledge role* (axiom 135) is a concept that classifies only information objects that have an epistemic relevance, i.e. they express descriptions that help maintaining the identity of a community by being exchanged, enriched, or revised according to appropriate plans and norms. E.g. the term "force" plays a knowledge role in contemporary physics because the relations and axioms (the *descriptions*) that are assumed when using that term contribute to the stability of contemporary physics' paradigm (the notion of *paradigm* has been formalized here in 76).

$$\begin{aligned} &KnowledgeRole(x) \rightarrow C(x) \wedge \\ &\forall(i,t)(classifies(x,i,t) \rightarrow I(i) \wedge \exists(d,p)(expresses(i,d,t) \wedge \\ &Paradigm(p) \wedge properPartOf(d,p,t))) \end{aligned} \quad (135)$$

An *agent role* (definition 136) is a concept that can only classify social agents.

$$AgentRole(x) =_{df} C(x) \wedge \forall(a,t)(classifies(x,a,t) \rightarrow A(a)) \quad (136)$$

Epistemic influence (definition 137) is now formalized as a NormPlanBundle (definition 133) that necessarily *uses* (axiom 19) at least one knowledge role (axiom 135) and at least one agent role (axiom 136).

$$\begin{aligned} &EpistemicInfluence(x) =_{df} NormPlanBundle(x) \wedge \exists(y,z) \\ &(KnowledgeRole(y) \wedge AgentRole(z) \wedge uses(x,y) \wedge uses(x,z)) \end{aligned} \quad (137)$$

The notion of epistemic influence is very flexible, since it can be used to talk about one agent that is influenced by some knowledge, as well as about two or more agents that mutually influence each other through their individual knowledge.

Based on it, we define *knowledge collectives* are intentional normative collectives whose unifying bundle is an epistemic influence that has a paradigm (76) as part.

$$\begin{aligned} &KnowledgeCollective(x) =_{df} \\ &IntentionalNormativeCollective(x) \wedge \\ &\exists(y,p,t)(EpistemicInfluence(y) \wedge unifies(d,x) \wedge \\ &Paradigm(p) \wedge properPartOf(p,y,t)) \end{aligned} \quad (138)$$

Epistemic influences found most agent interactions. In particular, the involvement in a *social relation* (definition 139) depends on the fact that involved agents are members of a same knowledge collective, i.e. on agreeing on a shared epistemic influence bundle, towards which all agents are accountable. Accountability is not treated here, but it will be axiomatized on the more basic notion of *assumption* (axiom 73), which is used here for axiom 139.

$$\begin{aligned} &SocialRelation(x) =_{df} EpistemicInfluence(x) \wedge \exists(kc) \\ &(KnowledgeCollective(kc) \wedge unifies(x,kc) \wedge \forall(a,t) \\ &(memberOf(a,kc,t) \rightarrow assumes(a,x,t))) \end{aligned} \quad (139)$$

[13] We remark that we are not interested here in how descriptions are grounded in physical agents, but only in the fact that social agents share those descriptions

Intuitively, if the agents that participate in a social relationship do not comply to the plans and norms that they are expected to assume (according to a given epistemic influence bundle that unifies the relationship), the underlying collective cannot be "brought about" by them.

Brings about (axioms 140 and 141) is a specialized projection of *c.DnS* maximal relation, holding for social agents and knowledge collectives at a certain time, and requires agents to assume an epistemic influence bundle that has to be adopted by the member agents in the collective.

$$\begin{aligned} bringsAbout(x,k,t) \rightarrow actsFor(x,k,t) \wedge \\ A(x) \wedge KnowledgeCollective(k) \wedge T(t) \end{aligned} \tag{140}$$

$$\begin{aligned} bringsAbout(x,k,t) \rightarrow \exists(y)(assumes(x,y,t) \wedge \\ EpistemicInfluence(y) \wedge unifies(y,k) \wedge \\ \forall(a,t)(memberOf(a,k,t) \rightarrow adoptsPlan(a,y,t))) \end{aligned} \tag{141}$$

When applied to social relations in general, *brings about* requires participating agents to assume it, not just to adopt it as a plan. In other words, social relations are maintained by knowledge collectives that are brought about by their members.[14]

With *c.DnS*, norms, plans, epistemic influence bundles, knowledge collectives, and social relationships, we have got a rich ontology to describe the nature and the behavior of complex social agents like organizations, institutions, corporations, teams, lobbies, movements, etc.

A recap of the main classes of the schematic entities introduced here, with their taxonomy and disjointness axioms, is depicted in Figure 1.

4 Conclusions

In this article, we have presented a formal framework to represent social agents, collectives, plans, norms, and their dependencies. The framework is based on the constructive version of the ontology of Descriptions and Situations (*c.DnS*), which has been applied to the modelling of social reality and information objects in several applications for the semantic web, business interaction, healthcare informatics, digital libraries, etc.

$c.DnS$ provides a complex pattern for social entities, axiomatized as the so-called *construction principle*, and can be extended to represent a *grounded* version of the principle. Plans and norms are represented here as extensions of *c.DnS*, and a typology of collectives is defined on top of these extensions.

Social agents are taken as primitives, independently from their embodiments as cognitive systems, organisms, robots, etc. Embodiments can be represented in

[14] The notion of social relation proposed here may be perceived as having a 'scientific' flavor, because we are proposing that agents in a social relationship assume a common paradigm, i.e. a bundle of plans and norms. But this common paradigm should be taken as minimal as possible, and subject to continuous revision, on the basis of the dynamics that operate on the agents. Therefore, no special claim on the stability or scientific foundedness of a social relationship is made here.

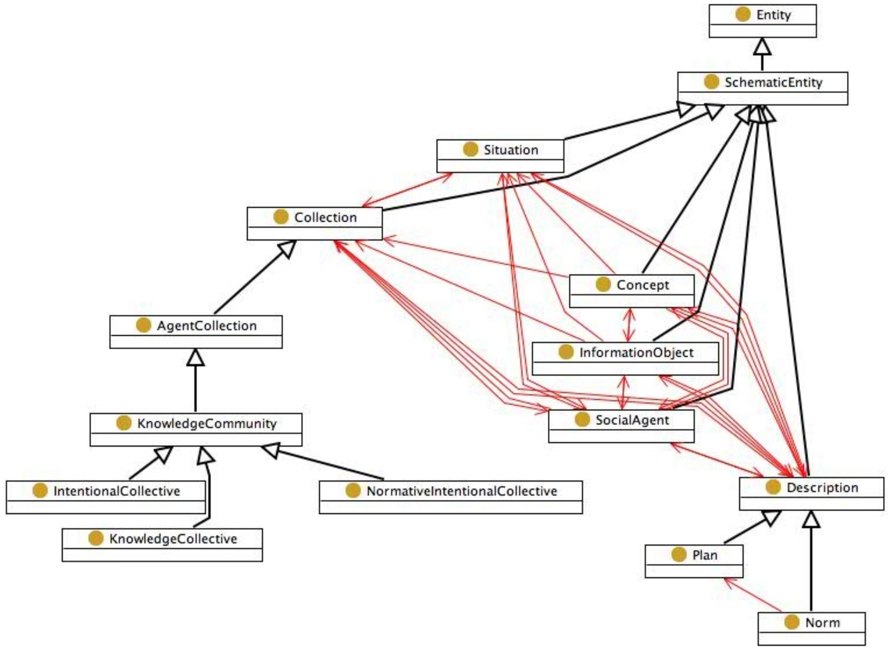


Fig. 1: The main classes of schematic entities, with their taxonomy (sub-class of, bold arrows) and disjointness axioms (red light arrows)

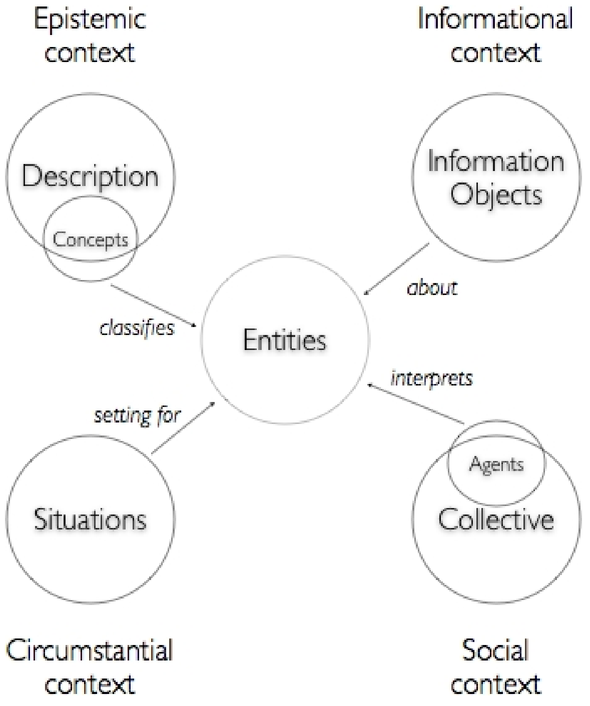


Fig. 2: The contextual bindings for the representation of a conceptualization

the grounded version of *c.DnS*. This move avoids the typical multi-hierarchies generated by classes that can be agentive or non-agentive, depending on local commitments.

Information and knowledge play a major role in *c.DnS*, and have constructive counterparts, i.e. so-called *information objects*, e.g. texts, images, etc., and *descriptions*, i.e. reified relations between entities. Constructivism (knowledge is inherent in *thought collectives* that create *knowledge paradigms*) is represented by positing collectives of agents that share descriptions expressed by information objects. These descriptions use concepts to classify entities within situations. All these notions: collective, agent, description, information object, concept, situation, are first-class citizens in our ontology.

In other words, *c.DnS* allows to represent the contextual binding of conceptualizations on the circumstantial level (via situations), on the cognitive level (via descriptions and concepts), on the social level (via collectives and agents), and on the informational level (via information objects) (see Figure 2).

Plans and norms have been introduced on top of *c.DnS*, as description types. Plans and norms are shown to be disjoint classes, but entrenched within ‘bundles’ of descriptions that provide formal unification criteria for intentional normative collectives.

Finally, we have used our formal apparatus in order to model the interdependencies of social agents, collectives, plans, and norms against the background of *c.DnS*. The main outcome can be considered a novel ability to talk -in a first-order theory- about complex agents like organizations and institutions, together with their knowledge-level structure (knowledge collectives, unifying plans and norms, paradigms), as well as arriving at a more sophisticated formal notion of social relationships as immersed into (and therefore dependent on) knowledge collectives.

Future plans include a finer-grained classification of collectives, based on more common sense distinctions, like:

- the type of their members (e.g. physical persons, boys, cows, left-handers);
- their knowledge domains (e.g. genetic, taxonomic, epidemiological);
- their related social practices (e.g. neighborhood, geographic, ethnic, linguistic, commercial, industrial, scientific, political, religious, institutional, administrative, professional, sportive, interest-based, stylistic, devotional);
- the ways members of collectives explicitly interact with the description bundles that are expected to unify their collectives (e.g. complete or partial adoption, external control and distribution of accountability, emergence, negotiation, trustfulness);
- the causal relations the characterize them, according to the set-up of the *DnS*-based treament of causal relations presented in [41].

Another item for our research agenda is investigating the assumptions inherent in relevant theories of action, collective intentionality, plural reference, etc., and describing them according to the ontology introduced here.

Acknowledgments

Part of the research presented in this paper (in particular, the notions of epistemic influence, knowledge communities, and related ones) was developed under the auspices of the NeOn project[15] funded by the European Commission within the 6th IST Framework Programme.

References

1. : OWL Web Ontology Language Family of Specifications. http://www.w3.org/2004/OWL (2004)
2. Oberle, D., Mika, P., Gangemi, A., Sabou, M.: Foundations for service ontologies: Aligning OWL-S to DOLCE. In: Proceedings of the World Wide Web Conference (WWW2004). Volume Semantic Web Track. (2004)
3. Bottazzi, E., Catenacci, C., Gangemi, A., Lehmann, J.: From collective intentionality to intentional collectives: an ontological perspective. Cognitive Systems Research - Special Issue on Cognition and Collective Intentionality **7** (2006) 2–3
4. Cocchiarella, N.: Denoting concepts. Reference and the logic of names, classes as many, groups and plurals. http://www.formalontology.it/essays/plurals.pdf (2004)
5. King, P.: The Metaphysics of Peter Abelard. In K., G., ed.: The Cambridge companion to Abelard. Cambridge University Press, Cambridge, U.K.; New York (2004)
6. Russell, B., Whitehead, A.N.: Principia Mathematica. Cambridge University Press, Cambridge, UK (1910)
7. Zeman, J.J.: Peirce on Abstraction. The Monist **65** (1982) 211–222
8. Dauben, J.W.: Georg Cantor: His Mathematics and Philosophy of the Infinite. Princeton University Press, Princeton (1979)
9. Dugac, P.: Richard Dedekind et les fondements des mathematiques. J. Vrin, Paris (1976)
10. Link, G.: The Logical Analysis of Plurals and Mass Terms: A Lattice-theoretical Approach. In von Stechow, A., ed.: Meaning, Use and Interpretation of Language. Walter de Gruyter, Berlin (1983) 302–323
11. Marcus, R.B.: Classes, Collections, Assortments, and Individuals. In Marcus, R.B., ed.: Modalities: Philosophical Essays. Oxford University Press, New York (1993) 90–100
12. Simons, P.: Parts: a Study in Ontology. Clarendon Press, Oxford (1987)
13. Devlin, K.: The Joy of Sets: Fundamentals of Contemporary Set Theory. Springer-Verlag, New York (1993)
14. von Neumann, J.: An Axiomatization of Set Theory. In Van Heijenoort, J., ed.: From Frege to Gödel; a source book in mathematical logic, 1879-1931. Harvard University Press, Cambridge, MA (1967)
15. Searle, J.R.: Intentionality. An Essay in Philosophy of Mind. Cambridge University Press, Cambridge, UK (1983)
16. Gilbert, M.: Social Facts. Princeton University Press, Princeton, New Jersey (1992)
17. Bratman, M.E.: Shared Cooperative Activity. The Philosophical Review **101** (1992) 327–341
18. Tuomela, R.: Collective Acceptance, Social Institutions, and Social Reality. The American Journal of Economics and Sociology **62** (2003) 123–165

[15] http://www.neon-project.org

19. Wooldridge, M.J.: Reasoning about rational agents. MIT Press, Cambridge, Mass.; London (2000)
20. Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N.: Social roles and their descriptions. In: KR. (2004) 267–277
21. Searle, J.R.: The construction of social reality. Free Press, New York (1995)
22. Hart, H.: The Concept of Law. Oxford: Clarendon Press (1961)
23. Kelsen, H.: General Theory of Norms. Oxford Univesity Press (1991)
24. Boella, G., van der Torre, L., Verhagen, H.: Introduction to normative multiagent systems. Computational and Mathematical Organization Theory **12** (2006) 71–79
25. Gangemi, A., Mika, P.: Understanding the semantic web through descriptions and situations. In: CoopIS/DOA/ODBASE. (2003) 689–706
26. Gangemi, A., Borgo, S., Catenacci, C., Lehmann, J.: Task Taxonomies for Knowledge Content. Deliverable D07 of the Metokis Project. http://www.loa-cnr.it/Papers/D07_v21a.pdf (2005)
27. Gangemi, A., Catenacci, C.: A constructive ontology of descriptions and situations. Technical report, ISTC-CNR (2006) http://www.loa-cnr.it/TR/ConstructiveDnS.pdf.
28. Masolo, C., Gangemi, A., Guarino, N., Oltramari, A., Schneider, L.: WonderWeb EU Project Deliverable D18: The WonderWeb Library of Foundational Ontologies. http://wonderweb.semanticweb.org/deliverables/documents/D18.pdf (2004)
29. Cohen, R., Schnelle, T.E.: Cognition and Fact - Materials on Ludwik Fleck. Reidel, Dordrecht (1986)
30. Stanzione, M.: Epistemologie Naturalizzate. Bagatto, Roma (1990)
31. Churchland, P.S., Ramachandran, V.S., Sejnowski, T.J.: A critique of pure vision. In Koch, C., Davis, J., eds.: Large scale neuronal theories of the brain. MIT Press, Cambridge, Mass. (1994)
32. Light, P., Butterworth, G.e.: Context and cognition: Ways of learning and knowing. Lawrence Erlbaum Associates, Hillsdale, NJ (1992)
33. Moore, M.S.: Legal Reality: A Naturalist Approach to Legal Ontology. Law and Philosophy **21** (2002)
34. Karmiloff-Smith, A.: Précis of 'Beyond modularity: A developmental perspective on cognitive science'. Behavioral and Brain Science **17** (1994)
35. Gangemi, A., Sagri, M.T., Tiscornia, D.: A constructive framework for legal ontologies. In: Law and the Semantic Web. Volume LNCS 3369. Springer (2005) 97–124
36. Köhler, W.: Gestalt Psychology. Liveright, New York (1947)
37. Talmy, L.: Toward a Cognitive Semantics. MIT Press, Cambridge, Mass. (2003)
38. Quine, W.: On what there is. In: From a Logical Point of View. 2nd edn. Harvard University Press, Cambridge, MA (1980)
39. Fleck, L.: The Problem of Epistemology. In Cohen, R., Schnelle, T., eds.: Cognition and Fact - Materials on Ludwik Fleck. Reidel, Dordrecht (1936/86) 79–112
40. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Understanding top-level ontological distinctions. In Stuckenschmidt, H., ed.: Proceedings of the IJCAI Workshop on Ontologies and Information Sharing, 2001. (2001)
41. Lehmann, J., Gangemi, A.: An ontology of physical causation as a basis for assessing causation in fact and attributing legal responsibility. Artificial Intelligence and Law **DOI 10.1007/s10506-007-9035-3** (2007)

Norms of Conversation in a Framework for Agent Communication Languages

Rodrigo Agerri

School of Computer Science, Univ. of Birmingham
B15 2TT, Birmingham, UK
r.agerri@cs.bham.ac.uk

Abstract. In open and heterogeneous environments offered by the Internet, where agents are designed by different vendors, the development of standards for agent communication needs to keep abreast of new dynamic interaction modalities. The objective of this paper is to contribute to FIPA's standardization effort by proposing a pragmatic approach to the design of agent communication languages (ACLs) in which the meaning of messages is the combination of its semantics and pragmatics. First, we present a reformulation of FIPA's communicative acts (ACL semantics) using a grounded specification language which overcomes some of the usual problems attributed to FIPA's ACL semantics. Then the ACL pragmatics aims to account for the contextual factors that enriches the semantics, such agents' roles, turn-taking, and the satisfiability of messages' perlocutionary effects. We claim that the ACL pragmatics is best specified by means of norms related to agents' obligations, permissions and rights.

Keywords. Agent Communication Languages, Normative Pragmatics, Multi-Agent Systems.

1 Introduction

Enabling communication between heterogeneous agents a is crucial issue in the developing of open environments. Ideally, languages for agent communication should facilitate effective interaction without violating the autonomy and heterogeneity of the agents. This is particularly true in open environments, such as electronic commerce applications based on the Internet, where agents are designed by different constructors and work following their individual interests.

Most of the approaches to ACL semantics [1,2,3,4] are based on speech acts theory [5]. According to this theory, linguistic communication is just a special type of action that can be analyzed from three different points of view. An *illocution* is the central component of a communicative act and it corresponds to what the act intends to achieve. The illocution should be distinguished from the effect that the communicative action is meant to produce on the receiver (*perlocution*), as well as from how the actual communication is physically carried out

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/920

(*locution*). Agents communicate sending speech acts (also called *communicative acts* or *performatives* [1,2]).

Generally speaking, each speech act consists of a set of preconditions that need to hold for an agent to perform the speech act, a propositional content, and a set of perlocutionary effects (also called rational effects and post-conditions [1,2]) that encodes the effect that the speech act causes in the receiver. FIPA ACL [1] nowadays remains of the main efforts to standardization of ACLs. The definition of the speech acts is based on a *mentalistic* approach, that is, speech acts are defined in terms of agents' mental states, and the definitions of mental operators for beliefs, intentions, etc., are given in multimodal logics based on possible world semantics (SL). The main criticism to *mentalistic semantics* is that its specification language is defined using a multimodal logic which cannot be related to a computational model and therefore, it does not facilitate its pre-runtime *verification*. In relation to this, mental states in SL are not *public*, meaning that they are not verifiable by looking at the history of agents' behaviour [6,7,3]. Besides, some assumptions such as *sincerity* and *co-operativity* are rather problematic to maintain in open environments [3,8].

An answer to FIPA's shortcomings came by rethinking the general principles in agent communication and taking a social approach as opposed to a mentalistic one. From this point of view, performing a speech act produces a number of social consequences, for example agents acquiring a commitment by sending a particular message. Several authors put *commitment* as the core social notion for the specification of speech acts [3,9,10,8]. The result was the specification of *public* ACLs which, combined with the use of temporal logics [3], were a huge step forward towards the verification of ACLs.

To abandon the mentalistic concepts of goal and intention in favour of the notion of *commitment* means that the *illocutionary* aspect of communication is missing [11]. A typical case is the *request* speech act, whose illocutionary point consists of the sender having the goal or intending that the receiver execute a certain task on its behalf. The perlocutionary effect would state that the result of performing a *request* be the receiver executing the content of the speech act. Note that from a goal-based approach to communication, the consequences of the performance of a *request* affect the receiver which in this case has to either accept it or reject it. However, if we are primarily focused on the social aspects, these intuitions are not very easy to express. For example, a *request* in a commitment-based ACL would have as preconditions that "the sender commits that the receiver has committed to accepting a request from him" [3], which shows that agents' intuitive motivations when performing a request are rather odd. Furthermore, it is not easy to see how the social semantics would account for the fulfilment of the *perlocutionary effects* of performing a speech act. Dealing with autonomous agents, it is not possible to guarantee that the perlocutionary effects are satisfied in the ACL semantics, because its fulfilment depends upon the receiving agent.

As a matter of fact, we claim that trying to satisfy the perlocutionary effects by means of semantic specifications is the wrong strategy. In fact, we go further and say that if in order to explain the social consequences of performing a speech act, then the illocutionary aspect must be abandoned, we are going down the wrong path. However, leaving the fulfilment of the perlocutionary effects up to the receiver endanger the success of the communicative interchange. Therefore, a key question remains to be answered: How can we reach an equilibrium between (agents') autonomy and (communicative) efficiency?

A possible answer is to look at the interaction protocols proposed to specify and guide agents' conversations. Interaction protocols are generally concerned with the order in which speech acts are uttered. Thus, traditional approaches to conversation in agent communication do not consider the speech context (speaker, receiver, scenario, state of discourse, etc.) nor the content of the speech acts to propose the protocols [1,12,13,14]. Furthermore they adopt a *procedural* approach that reduces agent communication to an exchange of meaningless tokens [3].

Some authors [15,16] have argued in favour of a broader view of interaction protocols. They distinguished interaction protocols (also called conversation specifications) from conversation policies. The later restrict the interaction protocols based on contextual information (sender, receiver, roles, propositional content, etc.) and not only in virtue of the order. However, these approaches (notably [16]) do not give a formal and precise definition of the concepts they use in the protocols and policies. Moreover, they do not account for the interaction between the ACL semantics and pragmatics, which is necessary if we want to explain how the perlocutionary effects are to be achieved. In fact, they claim that the ACL pragmatics constitute an independent module from the semantics [16].

As an alternative we propose to modify the conception of *meaning* in agent communication. In particular, the view that meaning consists only on the specification of the ACL semantics and that the ACL pragmatics are simple protocols which give the order in which speech acts are to be used. Instead, we consider that the performance of an speech act in agent communication occurs always under certain conversational circumstances, in which agents play specific roles, respond to their own interests, etc, and that these issues should be taken into account by a full-fledged ACL pragmatics. In this sense the meaning of a speech act is the result of *using* it according to a set of rules of conversation. In this approach, the social perspective is included in the ACL pragmatics.

An ACL pragmatics based on *normative concepts* offers a convenient solution to the problems discussed above. Norms of conversation (protocols and policies) may restrict the use of certain speech acts, facilitate the achievement of the perlocutionary effects which are defined by the ACL semantics, provide mechanisms of turn-taking and take into account contextual factors to do so. Conversation norms are therefore dependent on the preconditions and perlocutionary effects established by the ACL semantics.

This work aims to be a contribution to FIPA's effort towards the standardization of agent communication. Therefore, we include in our framework a reformulation of some of the FIPA communicative acts in a speech acts library (SAL).

We use motivational notions in the definitions of the speech acts to preserve the illocutionary aspect of communication. A salient point of our approach is that the motivational operators (goals and intentions) will be interpreted *externally*. In other words, they will not refer to agents' internal mental states. Moreover, we include definitions for some categories absent in FIPA's specification (commissives and declaratives). Both the ACL semantics and pragmatics are built upon the same computational model. In this unified framework the pragmatic component accounts for the social effects of performing a speech act and thereby facilitating the achievement of its perlocutionary effects.

Next section introduces the motivations for an ACL normative pragmatics. Section 3 provides an overview of the communicative framework. In section 4, we introduce a specification language which is used in section 5 for the definition of a set of speech acts. After the specification of the ACL semantics, we discuss in section 6 two types of norms that structure conversation and we formalize the main deontic concepts in section 7. We apply their semantics in the specification of norms of conversation by means of automata using a declarative language. We finish the paper discussing some conclusions and further work.

2 Normative Pragmatics: Motivation

There is a at least a precedent in agent communication literature with respect to the view of *meaning* as the combination of the ACL semantics (speech acts) and pragmatics. Singh [3] argued that

> "What we usually refer to informally as *meaning* is a combination of the semantics and the pragmatics. We will treat the semantics as the part of the meaning that is relatively fixed and minimal. Pragmatics is the component of meaning that is context-sensitive and depends on both the application and the social structure within which is applied. [...] Pragmatic claims would be based on considerations such as the Gricean maxims of manner, quality and quantity."

Unfortunately, this paragraph does not refer to his completed work. Instead, it seems to be pointing out a direction of development in agent communication languages. This paper does assume that view of meaning and places it at the core of our proposal. ACL semantics does not fully determine the meaning of performing an speech act because the uttering and satisfiability of a speech act may depend on contextual aspects such as authority or trust of agents involved in the conversation. In this sense, we say that the ACL semantics is *underdetermined* and that pragmatic rules are required to fully determine the meaning of an speech act. Having an *underdetermined* semantics does not mean that the semantics is ambiguous, it only means that the semantic specification cannot take into account every possible scenario and linguistic interchange without loss of generality, and without violating agents' autonomy.

However, we will not follow Singh's suggestion that Gricean work on implicatures may be directly applicable to agent communication (see [17] for a preliminary Gricean approach to agent communication). Instead, we are more inclined towards a development of *normative pragmatics*. The introduction discussed the problems of a semantic-based approach to agent communication and how we may benefit from a more *unified* perspective where meaning is the combination of both the semantic and pragmatic levels. However, we still have not made an explicit point on the motivations in favour of a normative pragmatics.

In order to do so, let us assume that the semantic specification of an *inform* speech act states that when an agent i performs this act: (i) It believes its propositional content ϕ, (ii) it has the goal that the receiver j will eventually come to believe that ϕ holds, and (iii) the perlocution is that j comes eventually to believe that ϕ holds. The interpretation process would presumably be described as follows: When agent j receives the message, it will assume that the first two preconditions hold. As a consequence, j should believe that i believes that ϕ, if j trusts the sender's message, j will believe ϕ, which corresponds to the communicative goal i wanted to achieve. Assuming that agents shall do all this reasoning is, however, too idealistic. Moreover, it is computationally expensive to let agents do all this reasoning. While text recognition may be an interesting problem for computational linguistics, an agent communication language should allow agents to communicate with each other effectively and efficiently in open environments in order to achieve some goals.

The basic motivation to propose a normative pragmatics is based on the hypothesis that the reasoning described above could be avoided if we establish flexible norms of conversation (based on rights, obligations, permissions, etc.). The norms of conversation would take into account those factors that influence the satisfiability of perlocutionary effects (e.g., the receiver's responses) and it also may influence agents' behaviour according to specific circumstances.

In relation to the latter, the FIPA CAL specification provides another good example of why a normative pragmatics may be useful to regulate the use of the speech acts. The specification of the *agree* communicative act contains a *pragmatic note* that reads:

> "The precondition on the action being agreed to can include the perlocutionary effect of some other CA, such as an *inform* act. When the recipient of the agreement (for example, a contract manager) wants the agreed action to be performed, it should then bring about the precondition by performing the necessary CA. This mechanism can be used to ensure that the contractor defers performing the action until the manager is ready for the action to be done". [1, p.4]

There are a few other *pragmatic remarks* like this one throughout the FIPA CAL that are not part of the semantic specification itself. These *pragmatic remarks* point out to the need of regulating agents' use of speech acts, but the FIPA specification does not go further. The fact that the designer felt compelled to add such a note illustrates the valuable role that a normative pragmatics can play. First, it states that agents play a specific role in the interaction. Second, it constrains the behaviour of the agents in a specific context and even the timing of executing a particular action. Furthermore, norms of conversation for agent conversation should combine nicely upon normative multi-agent systems, where notions of violation and sanction, etc., are specified and – in the case of sanctions – enforced. Thus, if an agent violates some agents' rights by not following the pragmatics, a sanction mechanism is in place, providing that exists an effective relation to the general social structures and norms of the system.

Kagal and Finin [16] propose to use obligations and permissions to specify conversation policies. However, there are a number of important differences to what we are proposing: First, they do not provide a formalization for any of the deontic operators they use. Second, they claim that policies are independent of the ACL semantics, and that in fact policies should be specified in the general structure of the system. We claim the opposite for reasons given below. Third, the use of obligations produces policies that in our view are too restrictive for autonomous agents. Four, they do not consider how their policies relate to an general purpose ACL (such as FIPA ACL) for its use in open environments. Finally, they use an ontology language based on OWL as the policy specification language, but we believe that formal logic is a more suitable language to reason about normative multi-agent systems.

There is an enormous amount of work done on the theory and practice of normative multi-agent systems [18,19,20,21,22,23] traditionally related to specification of multi-agent systems using various types of deontic logic. Some of these approaches include a communicative module that allows only a domain-based interaction [19], while others have tried to build commitment-based ACLs within an institutional framework [24]. As far as we know, it is a contribution of this paper the specification and use of normative and organizational concepts to design an all-purpose unified ACL framework for agent communication, where the normative concepts are given a precise and formal definition. One of the basic concepts of our normative pragmatic approach is the notion of 'right'. Note that we are not trying to investigate what the nature of rights are, or how many different types of rights can be distinguished or anything of the like (as discussed by [20,22] among others). Instead, we give a formal definition of a notion of *right* which is convenient for communicative purposes. Thus, the meaning of 'right' in our system is restricted to this definition. We do not aim to elucidate in this paper the meaning of several deontic notions useful for the specification and reasoning about normative multi-agent systems, but to show how deontic notions can be used for specifying ACLs relevant to normative multi-agent systems.

Summarizing, an ACL Normative Pragmatics (NPRAG) shall address the effect that the following issues have on the sender's choice of speech act and the receiver's interpretation of a message:

1. **Context**: Conversation policies state the relation between participants' roles and any particular contextual information (politeness, etc.) specific of the scenario.
2. **Perlocutionary effects**: NPRAG specifies policies about agents' communicative behaviour for a given speech act.

3. Participants' methods of **turn-taking**, constructing sequences of messages across turns, and how conversation works in different conventional settings are mainly dealt with the constitutive rules or protocols of the theory.

Regarding our interests in contribution to FIPA's work on agent communication, we could have adapted the normative pragmatics with FIPA's communicative acts library to offer a unified framework for agent communication. However, we agree with most of the criticisms discussed in the introduction towards FIPA's mentalistic semantics. At the same time, we argued in favour of preserving the illocutionary component in the ACL semantics. With this aim in mind, we include in our ACL framework a grounded specification language $MLTL_I$ for the ACL semantics where the motivational operators for goals and intentions are interpreted from an *external* point of view. Moreover, the reformulation of FIPA's communicative acts using $MLTL_I$ result in a more simple and natural representation.

3 ACL Framework

There are a number of properties that an ACL framework should comply with if we want to develop a general purpose and efficient high-level communication language for multi-agent systems. We have already discussed several that are regularly mentioned in the literature. We echo the voices of authors such as [3,25,26,27] among others to propose a number of requirements that are desirable for ACLs to exhibit:

- **Autonomous**: Agent communication must endeavour in the development of artificial languages for autonomous agents.
- **Complete**: The semantics must include a wide range of speech act types, so that there are at least available those categories defined by Searle's taxonomy.
- **Contextual**: The context of FIPA ACL is fixed with the sender. This impedes to use the language in different contexts, which affects the heterogeneity of agents. Contextual factors such as agents' roles, propositional content of messages, etc., must be considered for the ACL to be applicable in a variety of scenarios.
- **Declarative**: The semantics should state the meaning of the messages, and not the order in which can be used. Guiding the use of ACLs should be done contextually. Thus, it would be possible to adapt the ACL by constraining the use of a subset in a specific context.
- **Formal**: ACL semantics and pragmatics must be formally defined. A clear and explicit specification would facilitate interoperability for open multi-agent systems.
- **Grounded**: The ACL presented should be grounded into a computational model. This will allow to translate the properties of the agents of the system into program properties. This also facilitates the verification of the ACL.

- **Public**: Communication must be public. ACL semantics must not depend on agents' private mental states. Social consequences of performing speech acts must be addressed by the ACL pragmatics.
- **Perlocutionary**: ACL pragmatics should aim to facilitate the achievement of the perlocutionary effects.

The unified ACL consists of a set of speech acts, the Speech Acts Library (SAL), and the ACL pragmatics consisting of norms that constrain agents' behaviour. We also define two specification languages, $MLTL_I$ and $NLTL_I$, to define the semantics of the cognitive and normative concepts used in the ACL. Besides, the two specification languages have a temporal component to take into account the evolution of the system over time. In this paper, the ACL semantics captures the illocutionary character of communication. The ACL pragmatics contextually regulates the use of the speech to facilitate the achievement of the perlocutionary effects. Thus, a unified ACL is defined as the tuple (we build on [26]):

$$\mathcal{UACL} = \langle SAL, MLTL_I, NPRAG, NLTL_I \rangle$$

Following FIPA CAL [1], messages of SAL are based on a STRIPS-like language with preconditions and effects. On the one hand, the preconditions have to be true for the agent to send a message (including the goal the sender intends to achieve by sending that message). On the other hand, the effects state the response that the sender wants to produce in the audience. This is a problematic issue because, as it has been already discussed, autonomous agents, by definition, cannot be forced to guarantee the effects. The semantics of SAL are given by a function

$$[\![-]\!]_{SAL} : \mathit{wff}(SAL) \rightarrow \mathit{wff}(MLTL_I)$$

The syntax of the communication language SAL is based on the FIPA ACL [1]. The semantics of the motivational and temporal operators is given by $MLTL_I$ in the next section. The language $MLTL_I$ is based on Linear Temporal Logic (LTL) extended with operators for beliefs, goals and intentions. We combine the cognitive notions with temporal operators á la Fagin *et al.* [28]. In doing so, we aim to ground $MLTL_I$ upon a computational model, the first stage to facilitate its verification [29].

In the interpretation for beliefs, goals and intentions proposed here, they are ascribed to agents by an external reasoner about the system. Following the Interpreted Systems approach [28] for modelling knowledge, agents in our framework do not compute their beliefs, goals and intentions, and as a consequence, the ACL defined using $MLTL_I$ as the semantic specification language does not rely on agents' internal (mental) states.

$NLTL_I$ consists of linear temporal operators combined with a deontic operator for obligations. $NLTL_I$ provides the semantics for the normative operators used in the specification of NPRAG. The conversation policies and interaction protocols of NPRAG can be specified using a logic-based declarative language.

4 $\boldsymbol{MLTL_I}$

Traditionally, the role of formal logic in artificial intelligence and distributed computing is to provide clear formal tools to specify complex systems. However, the logic-based specifications have been criticized on the grounds that they do not provide real methodologies for building distributed systems. In order to cope with the increasing complexity of the capabilities required by agents, researchers have been using complex multimodal logics for their specification which are generally ignored by programmers that do not see a clear relation between the specifications in formal logic and computational systems [30,31].

Several authors [28,31] have argued that to bridge the gap between theory and practice, the multimodal logics used in the specification of multi-agent systems must be grounded in a computational model. There are two main semantic approaches to the formalization of agent systems via modal logics. The traditional model is based on the work of [32] on possible-world semantics. The possible-world approach includes the theory of intention [33] and the BDI logic of [34]. The problem with possible world semantics is that the accessibility relations used to define mental operators are not easily related to a computational model. Appropriate grounded semantics ensures that a clear correspondence can be found between states in the computing system and configurations in the logical description (see [29] for a good discussion on these issues).

The second approach, the Interpreted Systems model, offers a natural interpretation of the notion of knowledge in terms of states of agents in distributed systems [28]. We adapt the interpreted system approach to our purposes of giving a *grounded* and *public* semantics for beliefs, goals and intentions.

4.1 Syntax

The syntax of the language $MLTL_I$ (Motivational Linear Temporal Logic on Interpreted Systems) associated to the interpreted system IS consists of the usual vocabulary of interpreted systems IS and the accessibility relations for beliefs, goals and intentions. $MLTL_I$ structures are the result of the combination of IS with the accessibility relations $\mathcal{B}_i$, $\mathcal{G}_i$ and $\mathcal{I}_i$ defined for the structure M_I.

The following symbols and abbreviations will be used: $=$ for definitions. To start to construct a formal language, a set of atomic propositions (where each proposition corresponds to a variable in the model) and the usual Boolean connectives are introduced: negation $\neg$, disjunction $\vee$, conjunction $\wedge$, conditional $\rightarrow$, and material equivalence $\leftrightarrow$. Atomic formulae will be denoted by ϕ, ϕ_0, ϕ_1, $\psi \ldots$

The operators X, F, G, U are called the temporal operators. All the temporal operators are interpreted relative to a *current global state*. There are many runs (sequences of global states) of the system starting at the current state. The temporal operators describe the ordering of events in time along a run and have the following intuitive meaning:

- $F\phi$ (reads "ϕ holds sometime in the future") is true of run if there exists a global state in the run where formula ϕ is true.

- $G\phi$ (reads "ϕ holds globally") is true of a if ϕ is true at every global state in the run.
- $X\phi$ (reads "ϕ holds in the next state") is true of a path if ϕ is true in the state reached immediately after the current state in the run.
- $\phi\ U\psi$ (reads "ϕ holds until ψ holds", is true of a run if ψ is true in some state in the run, and ϕ holds in all preceding states. In other words, ψ does eventually hold and that ϕ will hold everywhere until ψ holds.

Definition 1 **($MLTL_I$ Syntax).**
The syntax of the semantic specification language $MLTL_I$ is given by the following BNF expression (consider n agents):

$$\phi := AP|\neg\phi|\phi \wedge \psi|B_i\phi|G_i\phi|I_i\phi|X\phi|F\phi|G\phi|\phi U\psi$$

We use $True$ and $False$ as shorthands for $\phi \vee \neg\phi$ and $\neg True$ respectively. Although we have include in the syntax every temporal operator, we can define X, F and G as abbreviations:

$X\phi \equiv False\ U\ \phi$
$F\phi \equiv True\ U\ \phi$
$G\phi \equiv \neg F \neg\phi$

The next operator X is true at some state s whenever ϕ is true at some future point t and there are no other states between s and t. F holds if a formula is true at some point in the future and G is true always in the future, that is, there is not a future global state in which ϕ is not true.

We can conventionally establish several binding priorities for $MLTL_I$ connectives. The unary connectives ($\neg$, the temporal connectives G, F, X, and the mental attitudes operators B_i, G_i and I_i) bind most tightly. Next in priority are $\wedge$ and $\vee$, and finally $\rightarrow$ and U.

In this framework, "agent i believes ϕ" means that, "as far as agent beliefs are concerned, the system could be at a point in which ϕ holds". In other words, beliefs refer to the runs of the system. The notion of belief used in this paper does not require that the belief be true. Therefore, an agent holding a belief does not automatically made true the content of the belief. This property is central for open multi-agent systems, where agents have available incomplete and modifiable information.

An "agent i has the goal of bringing about ϕ" means that, "with respect to the agent's goals, the system could be at a point where ϕ holds". Goals can be seen as facts ϕ at a global state that an agent wants to bring about. "An agent i intending to bring about ϕ", means that from the point of view of the agents' intentions, there is a run in which i intends, along that run, to bring about ϕ.

To ascribe cognitive states from an external point of view we generate a structure M_I by associating an Interpreted System IS with a serial, transitive and euclidean structure M, so that beliefs, goals and intentions refer to *runs* of the multi-agent system. The fundamental notion in this approach is the one of *local state*. If we look the system at any point in time, every agent is in some unique *state*. The only assumptions we need to make about local states is that all the information that agents' possess of the system is encoded in their local state. Now, given that we are interested in having an ACL semantic specification language which can be used to describe the unique state of a multi-agent system at each point in time so we do not rely on the agents' internal states to evaluate and verify their communicative behaviour, we need not only to describe the local state of the agents but also the rest of the multi-agent system, which is called the *environment*. For example, when analyzing a system where agents send messages along some communication channel, useful to keep a record or history of the messages sent. Thus, when describing a multi-agent system as a whole (agents and environment), we use the notion of *global state*. These ideas are formalized in the following section where a semantics for $MLTL_I$ are defined.

4.2 Semantics

The key idea of interpreted systems is that agents are in some state at any point in time. This state is the agent's local state which consists of all the information about other agents and about the environment to which agents have access (we follow [28] in the definition of Interpreted System). Furthermore, we can also think of the whole system as being in some state. In this sense, the notion of *environment* refers to everything else in the system that is not an agent. Both the agent's local state and the environment's state conform the global state of a system.

Definition 2 (Global States).
A tuple $(s_e, s_1, \ldots, s_n)$ represents a global state in a multi-agent system where s_e is the environment's state and s_i is agent i's local state, for $i = 1, \ldots, n$.

A system evolves over time. Thus, a run is defined as a function from time to global states which gives a complete description of what happens over time in one possible execution of the system. Following this, a system consists of a set of runs. A system is always at a global state at some point.

Definition 3 (Runs).
A run r over nonempty sets of global states GS is a sequence of global states in GS that gives a complete description of an execution. A point consists of a tuple (r, m) where r is a run and m is the time. If $r(m) = (s_e, s_1, \ldots, s_n)$ is the global state at point (r, m), then we say that $r_e(m) = s_e$ and $r_i(m) = s_i$, for $i = 1, \ldots, n$.

A system can be seen as a Kripke structure with no labelling or interpretation function to assign truth values to the atomic propositions.

Definition 4 (Interpreted System).
A system T over a set of global states GS is a set of runs over GS. An interpreted system is a pair (T, L) where T is a system of runs over global states and L is a labelling function for the atomic propositions AP over GS, which

assigns truth values to the atomic propositions at the global states. For every $\phi \in AP$ and $g \in GS$, $L(g)(\phi) \in \{true, false\}$. A point is in the interpreted system IS if $r \in T$. Formally, an interpreted system IS is defined by the tuple (T, GS_0, L).

We extend the interpreted system models with beliefs, goals and intentions. Beliefs are given a standard $KD45$ axiomatization relative to each agent. For goals and intentions, we assume a minimal KD axiomatization to ensure consistency.

Definition 5 (M_I structure).
Given a system of runs T, a structure M_I is generated by associating the interpreted system $IS = (T, L)$ with the serial, transitive and euclidean Kripke structures $M = (S, \mathcal{B}_i, \mathcal{G}_i, \mathcal{I}_i, L)$, such that $M_I = (GS, \mathcal{B}_i, \mathcal{G}_i, \mathcal{I}_i, L)$ where:

- *GS corresponds to the sets of global states in IS.*
- *L is a labelling function $L : S \rightarrow 2^{AP}$ from global states to truth values, where AP is a set of atomic propositions. This function assign truth values to the primitive propositions AP at each global state in GS.*
- *$\mathcal{B}_i$ where $i = (1, \dots, n)$ is a set of agents, gives the accessibility relation on global states, which is serial, transitive and euclidean. Thus, we have that $(l_e, l_1, \dots, l_n)$ $\mathcal{B}_i$ $(l'_e, l'_1, \dots, l'_n)$ if $l'_i \in GS_i$. If $g = (l_e, l_1, \dots, l_n)$, $g' = (l'_e, l'_1, \dots, l'_n)$, and l_i $\mathcal{B}_i$ l'_i, then we say that g and g' are $\mathcal{B}_i$-accessible to agent i. The formula $B_i\phi$ is defined to be true at g exactly if ϕ is true at all the global states that are $\mathcal{B}_i$-accessible from g.*
- *The accessibility relations for goals $\mathcal{G}_i$ and intentions $\mathcal{I}_i$ are defined in the same manner.*

The relations for goals and intentions are serial, so we simply adopt their definition to say that the accessibility relations that characterized goals and intentions between two global states, g $\mathcal{G}_i$ g', and $g\mathcal{I}_i$ g' respectively, are serial. Given that $g = (s_e, s_1, \dots, s_n)$ is the global state, we say that $g_e = s_e$ and $g_i = s_i$ for $i = 1, \dots, n$; this means that g_i is the local state of agent i at a given time. Agents' beliefs, goals and intentions are defined with respect to their local states and can be induced to relate points. For convenience, we will sometimes use this simplified notation to refer to global states g.

We can now apply the definition of M_I to define truth of a formula ϕ at a global state $r(m)$ of the interpreted system IS.

Definition 6 (Satisfaction in IS with respect to M_I).
In this framework, to say that a formula ϕ is true at a global state $r(m)$ in an interpreted system IS if it is true in the related M_I. Formally,

$$(IS, r, m) \models \phi \text{ if } (M_I, s \models \phi).$$

We would like to remark that the semantics of the accessibility relations presented here relates global states and not points. We choose global states to stress the intuitions behind interpreted systems IS. Moreover, it allows us to give a natural definition to the time operators.

Definition 7 **($MLTL_I$ semantics).**
The semantics of $MLTL_I$ is inductively defined as follows:

$(IS, r, m) \models \phi$ *iff* $L(r, m)(\phi) = true$
$(IS, r, m) \models \phi \wedge \psi$ *iff* $(IS, r, m) \models \phi$ *and* $(IS, r, m) \models \psi$
$(IS, r, m) \models \neg\phi$ *iff it is not the case that* $(IS, r, m) \models \phi$
$(IS, r, m) \models B_i\phi$ *iff* $\forall(r', m')$ *such that* $(r, m)\ \mathcal{B}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models G_i(\phi)$ *iff for all* (r', m') *such that* $(r, m)\ \mathcal{G}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models I_i(\phi)$ *iff for all* (r', m') *such that* $(r, m)\ \mathcal{I}_i\ (r'm')$*, then* $(IS, r'm') \models \phi$
$(IS, r, m) \models X\phi$ *iff* $(IS, r, m + 1) \models \phi$
$(IS, r, m) \models F\phi$ *iff for some time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models G\phi$ *iff for all time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models \phi U \psi$ *iff there is some time* $m' \geq m$ *such that along the run such that* $(IS, r, m') \models \psi$ *and for each* $m \leq m'' < m'$ *we have* $(IS, r, m'') \models \phi$*.*

There are various issues worth to comment on the semantics of $MLTL_I$: L is a labelling function on global states, that is, the truth of a primitive proposition ϕ at a state g depends only on the global state g, since the global state encapsulates all the system information at a particular point. However, there are situations, such as "agent i receiving agent j's message", where its truth does not depend on the whole global state, but only on the agents' local state. On the other hand, there are other statements which describe situations in which their truth depends on more than the global state. An statement such as "at some point in the run, the variable x is set to 5" (example from [28]) could be true at the global state g, and false at the same global state of g at a different time. This problem is solved by introducing the temporal operators, so we can easily express the idea that something is to be true in the system at some later time, namely, $F\phi$. The formula $\phi\ U\ \psi$ holds on a run if it is the case that ϕ holds continuously until ψ holds. Moreover, $\phi\ U\ \psi$ actually requires that ψ holds in some future state.

In the interpretation for beliefs, goals and intentions proposed here, these attitudes are ascribed to agents by an external reasoner about the system. In this approach, agents do not compute their beliefs, goals and intentions in any way, and as a consequence, the communication protocol defined using $MLTL_I$ does not rely on agents' private mental states. In the case of $G_i\phi$ and $I_i\phi$ the two points (r, m) and (r', m') are related if $(r'm')$ makes possible to achieve the intention (respectively, the goal) of agent i at the point (r, m).

Agents in multi-agent systems are seen as runs. In the next section we will show how $MLTL_I$ is used to externally ascribe beliefs, goals and intentions in the definition of a set of speech acts. By combining cognitive and temporal operators, we make statements about the evolution of the agents' propositional attitudes in the system. For example, we can say that agent i believes that ϕ will eventually hold along a run: $B_iF\phi$.

It is also important to remark that the semantics of $MLTL_I$ could have been presented in a different way, closer to the possible world semantics models [32], that is, by defining the accessibility relations over points of the system [35,36]. The choice of global states stresses the intuitions related to multi-agent systems.

There has been quite a lot of work in the Computer Science community on the theoretical aspects of temporal logic. In particular, the issues of decidability, complexity and axiomatizability have been largely studied. We present in the next section an axiomatization for $MLTL_I$ and discuss some issues on the complexity of reasoning about beliefs, goals and intentions with linear time. Then, we will put $MLTL_I$ into use by defining a complete set of Speech Acts.

4.3 Axiomatics

Multi-agent systems quite often operate without complete information about their environment, which could include other agents. Thus, it is interesting to use formalisms that allow us to talk of the system's changes over time. The axiomatics of $MLTL_I$ consists of the traditional $KD45_n$ for belief and KD_n for goals and intentions. i denotes a set of agents such that $i = 1, \ldots, n$.

PC All instances of propositional tautologies.
MP If ϕ and $\phi \rightarrow \psi$, then ψ.
NEC_b If ϕ, then $B_i\phi$.
NEC_g If ϕ, then $G_i\phi$.
NEC_i If ϕ, then $I_i\phi$.
K_b $B_i(\phi \rightarrow \psi) \rightarrow (B_i\phi \rightarrow B_i\psi)$.
D_b $B_i\phi \rightarrow \neg B_i\neg\phi$.
4_b $B_i\phi \rightarrow B_iB_i\phi$.
5_b $\neg B_i\neg\phi \rightarrow B_i\neg B_i\neg\phi$.
K_g $G_i(\phi \rightarrow \psi) \rightarrow (G_i\phi \rightarrow G_i\psi)$.
D_g $G_i\phi \rightarrow \neg G_i\neg\phi$.
K_i $I_i(\phi \rightarrow \psi) \rightarrow (I_i\phi \rightarrow I\psi)$.
D_i $I_i\phi \rightarrow \neg I_i\neg\phi$.

The following axioms are known to provide a sound and complete axiomatization for LTL [37].

PC All tautologies of propositional logic.
T1 $X(\phi \rightarrow \psi) \rightarrow (X\phi \rightarrow X\psi)$.
T2 $X(\neg\phi) \equiv \neg X\phi$.
T3 $\phi\ U\ \psi \equiv \psi \vee (\phi \wedge X(\phi\ U\ \psi)$.
RT1 From ϕ infer $X\phi$.
RT2 From $\phi' \rightarrow \neg\psi \wedge X\phi'$ infer $\phi' \rightarrow \neg(\phi\ U\ \psi)$.
MP From ϕ and $\phi \rightarrow \psi$ infer ψ.

The axiomatic system is denoted by the expression $(B_{KD45}G_{KD}I_{KD})_{LTL}$, which is abbreviated by $MLTL_I - Ax$.

Theorem 1. *The system $MLTL_I - Ax$ is a sound and complete axiomatization with respect to the class of models $MLTL_I$ that are serial, transitive and euclidean.*

Completeness can be shown following the technique used in [38], who gave a sound and complete axiomatization for a logic with linear time and an operator for knowledge. Furthermore, [39] has very recently given a complete axiomatization for deontic and epistemic operators with branching time. [34] also prove completeness for BDI with branching time. The sketch of the proof is as follows: We need to show that the logic complies with the finite-model property, hence it is decidable. In order to do that, we define two structures, a Hintikka structure for a given formula φ and the quotient structure for a given model. From here we can prove that φ is satisfiable by constructing a Hintikka structure for φ and we build a pseudomodel of $MLTL_I$ structures using its quotient structure. For details, we refer to the reader to the papers cited above since the length of this proof exceeds the purpose of this paper.

Our work is obviously related and influenced by the work done on linear temporal logics [40] and the interpreted systems literature [28] about knowledge. Most of the formal apparatus defined in this section will be inherited by the ACL pragmatic specification language $NLTL_I$. The main difference (if only) is that the we combine a deontic operator with the linear time component defined here.

5 Speech Acts Library (SAL)

We had three main motivations to define a semantic specification language like $MLTL_I$:

- First, given that $MLTL_I$ is going to define the semantics of the speech acts, this logic had to allow operators for beliefs, goals and intentions to express the *illocutionary* character of communication.
- Second, $MLTL_I$ had to be grounded in a computational model, so it was interesting to find an alternative to possible world semantics to include motivational attitudes in our language.
- Finally, the temporal logic component allows us to analyze how a system evolves over time.

In this section we use $MLTL_I$ to propose a *public* and *grounded* semantics. The ACL semantics consists of a Speech Acts Library which is defined using the semantic specification language $MLTL_I$. The main purpose of this semantics is to show how the different validity claims can be understood in terms of our specification language, and formalized using the logic developed. The illocutionary point of speech act are expressed in the Feasibility Preconditions (FPs). We also specify Rational Effects to capture the perlocutionary effects that the sender intends to produce on the receiver. However, note that to provide mechanisms that allow agents to achieve the Rational Effects is a task to be performed by the ACL pragmatics.

Unlike some other alternatives to FIPA ACL discussed in the introduction, we view our Speech Acts Library as a contribution to the standardization effort lead by the FIPA project. In this sense, the definition of a *public* and *grounded* semantics aims to tackle the FIPA CAL shortcomings discussed. Furthermore, in many cases the informal description of a speech act includes references such as “at some point in the future”, “once the given precondition is true”, etc. We will see that those aspects of the specification can be naturally expressed in a simpler way using $MLTL_I$. With this point in mind, we not only define at least one speech act or communicative act for each of the categories proposed by [5], but also a version for several of the communicative acts defined in the FIPA ACL is given (see [41] for a complete reformulation of *every* FIPA communicative act using $MLTL_I$).

Following [5], we classify speech acts into assertives, commissives, directives, declarations and expressives. The last category is not relevant for the purposes of this paper, so it will not be included (we are not considering *affective agents*). The syntax of the speech acts is based on the FIPA ACL. Table 1 presents our new definitions of four speech acts plus two more expressing commissives and declaratives not present in FIPA's specification.

$\langle i, inform(j, \phi)\rangle$ $FP : B_i(\phi) \wedge G_i(B_j(\phi))$ $RE : B_j\phi$	$\langle i, request(j, \phi)\rangle$ $FP : G_i(I_j(F\phi))$ $RE : F\phi$
$\langle i, confirm(j, \phi)\rangle$ $FP : B_i(\phi) \wedge B_i(B_jF\phi \vee B_jF\neg\phi))$ $RE : B_j\phi$	$\langle i, disconfirm(j, \phi)\rangle$ $FP : B_i\neg\phi \wedge B_i(B_j\phi)$ $RE : B_j\neg\phi$
$\langle i, agree(j, \phi)\rangle$ $\langle i, inform(j, (I_i\phi\ U\ \psi))\rangle$ $FP : I_i\phi\ U\ \psi$ $RE : B_j(I_i\phi\ U\ \psi)$	$\langle i, refuse(j, \phi)\rangle$ $\langle i, inform(j, \neg(I_i\phi\ U\ \psi))\rangle$ $FP : \neg(I_i\phi\ U\ \psi)$ $RE : B_j(\neg(I_i\phi\ U\ \psi))$
$\langle i, promise(j, \phi)\rangle$ $FP : I_iF\phi$ $RE : F\phi$	$\langle i, declare(j, \phi)\rangle$ $FP : G_i(X\phi)$ $RE : X\phi$

Table 1. A complete set of speech acts.

The two performatives at the top, *inform* and *request*, represent the assertives and directives respectively. *Agree* and *refuse* are included as possible exchanges after the reception of a *request*. *Declare* is an action of the declarative class and *promise* is a commissive. These last two are our contribution to the FIPA CAL specification. Therefore, adding *promise* and *declare* to the list of primitives acts in our library (SAL) together with *inform*,*request*, *confirm* and *disconfirm* results in the total number of speech acts of SAL to be twenty four [41], although it is by no means a closed catalogue.

We use Searle's taxonomy in the knowledge that there is little agreement on the number of speech acts and types which should be covered, or whether it is possible at all to provide a complete list of speech acts. In any case, this partial list of actions cover the usual communicative requirements imposed on agents. The eight speech acts provided in table 1 are representative enough of to compare FIPA's specification with respect to our own definitions.

5.1 Assertives

Assertives perform statements about the real world. The typical assertive act is *inform.* This type of actions do not intend to modify the behaviour of the receiver, but only to affect its mental states. In particular, to modify the set of beliefs the receiver holds about a proposition ϕ. The definition of *inform* proposed by FIPA ACL indicates that the sending agent believes that some proposition ϕ is true, intends that the receiving agent also believes that ϕ is true, and does not already believe that the receiver has any knowledge of the truth of ϕ. This is regarding the Feasibility Preconditions. The Rational Effect consists of the receiver coming to believe ϕ. In FIPA's formalization of this communicative act the Feasibility Precondition consists of a conjunction: The first conjunct states quite simply that agent i has to believe the proposition ϕ, and the second one states that the sender believes that the receiver does not have any knowledge of the truth of ϕ. This provided by the form $\neg B_i(Bif_j\phi \vee Uif_j\phi)$, which it is decomposed as $\neg B_i((B_j\phi \vee B_j\neg\phi) \vee (U_j\phi \vee U_j\neg\phi))$.

It seems that this precondition is too restrictive on the sender, particularly because in open environments agents may not have any information about other agents' knowledge. When someone asserts (*inform*) that ϕ, the sender usually believes that ϕ and has the goal of affecting the receiver's mental states so that it comes to believe ϕ. Any specific constrains restricting agents to perform an *inform*, until it is completely sure that the receiver does not know that ϕ, should be formulated as a conversation policy. Therefore, we propose a new definition of *inform* in table 2.

$\langle i, inform(j, \phi)\rangle$
$FP : B_i(\phi) \wedge G_i(B_j(\phi))$
$RE : B_j\phi$

Table 2. Inform.

The first part of the Feasibility Preconditions requires the sender to believe ϕ which means that we want the sender to be sincere. This is a good assumption by default, but if we want agents to be able to negotiate in competitive scenarios this may be unrealistic. A feasible solution is to specify another speech act such as *convince* that could be used when an agent *just* aims that other agent believes a proposition ϕ, irrespective of their beliefs. This could give way to a trend of defining communicative actions to be used in argumentation and negotiation scenarios.

What about the Rational Effects? The FIPA specification says that whether or not the receiver adopts the belief in the proposition ϕ will be a function of the receiver's trust in the sincerity and reliability of the sender. FIPA does not provide a method to facilitate the achievement of the Rational Effects. Besides, it is quite clear that the nature of this observation about the receiver's trust in the sincerity of the sender, etc., points out to a number of factors that transcend the ACL semantics. It seems that we may need to model, for a specific scenario, the information relative to trust and other relations between the agents. This is the role of pragmatics in natural language communication and in our view it it is also the role that a pragmatic theory should play in agent communication.

Inform is the classic assertive speech act, but there are many others. For example, answers are generally assertives. Thus, speech acts such as *agree* and *refuse* are also assertives as are *confirm* and *disconfirm*.

According to [1], *agree* is a general-purpose agreement which answers a previously received *request*. When an agent agrees then it is informing the receiver that it intends to comply with the *request*, but not until the given precondition is true. *Agree* is not a primitive, so it is formalized in terms of an *inform*:

$$\begin{array}{l}
\hline
\langle i, agree(j, < i, act >, \phi) > \equiv \\
\langle i, inform(j, I_i Done(< i, act >, \phi)) > \\
FP : B_i \alpha \wedge \neg B_i(Bif_j \alpha \vee Uif_j \alpha) \\
RE : B_j \alpha \\
\hline
\end{array}$$

The arguments of the agree performative consist of an action to be performed, *act*, and the conditions of the agreement ϕ. The conditions are analyzed as informing of the intention to do an action *act* under the condition ϕ. The condition itself has to hold for the sender to agree with the request and to execute *act*. This particular point is not very clear in the formalization. There seems to be a mismatch between the informal description of the act and the actual formal model. In any case, this type of construction is where $MLTL_I$ proves useful, because we can naturally write $I_i\phi \; U \; \psi$ to express that the sender intends to bring about ϕ until ψ along a run. More intuitively, if ψ is true, then $I_i\phi$ is is true as long as ψ holds. The conditions of agreement are expressed in a more natural way by using the temporal operator U (until), where ψ describes the fact that constitutes the precondition of the agreement at a global state $r(m)$. The second conjunct in the Feasibility Preconditions of *agree* presents the same form as in the *inform* act, so we will not repeat the point about the operators for uncertainty, knowledge and the over-specification of agents' behaviour in the ACL semantics. The same goes for the Rational Effects.

Following the above discussion, the formalization of *agree* given in table 3 tries to capture the intuition that agent i agrees with agent j to bring about some ϕ until some precondition ψ is true. This is equal to informing j that i has the intention that ϕ will eventually hold in a run until ψ holds. The FPs state that the sender has to intend that ϕ until ψ eventually holds along a run, and the REs establish that the receiver believes that the sender possess that intention.

The dual of *agree* is *refuse*. Refuse is a negative answer to a request.

According to [1], *refuse* denotes the action of refusing to perform a given action and explaining the reason for the refusal. The arguments of the performative consist of the refused action and a proposition which provides an explanation for the refusal. Moreover, *refuse* is an abbreviation of *disconfirm*: an act is possible for the agent to be performed (and providing an explanation). An agent considers that is not possible to perform an action when the action preconditions are

$$\begin{array}{l}
\langle i, agree(j, \phi)\rangle \equiv \\
\langle i, inform(j, (I_i\phi\ U\ \psi))\rangle \\
FP : I_i\phi\ U\ \psi \\
RE : B_j(I_i\phi\ U\ \psi)
\end{array}$$

Table 3. Agree.

not satisfied. As an example, an agent may be requested to perform an action for which it has insufficient privilege (hence the explanation: I have not got enough privileges).

The definition of *refuse* given by FIPA is as follows:

$$\begin{array}{l}
\langle i, refuse(j, < i, act >, \phi)\rangle \equiv \\
\langle i, disconfirm(j, Feasible(< i, act >))\rangle; \\
\langle i, inform(j, \phi \wedge \neg Done(< i, act >) \wedge \neg I_i Done(< i, act >))\rangle \\
FP : B_i \neg Feasible(< i, act >) \wedge B_i(B_j Feasible(< i, act >) \vee \\
U_j Feasible(< i, act >)) B_i\alpha \wedge \neg B_i(Bif_j\alpha \vee Uif_j\alpha) \\
RE : B_j \neg Feasible(< i, act >) \wedge B_j\alpha
\end{array}$$

It is surprising that being *agree* and *refuse* the dual of each other their logical form does not show any similarities whatsoever. Moreover, the use of operators such as *Feasible* to provide reasons for refusing to do an action greatly complicates the complexity and decidability of the logic, as it is shown by the extremely complex definition of *refuse* given above.

Conversely, *refuse* is to be analyzed as the dual of *agree*. Following FIPA's recommendation, it is decomposed in terms of the *inform* primitive to communicate that the receiver of the request does not intend to bring about some ϕ (the object of the *request*) until ψ (the precondition of the agreement/refusal). Its definition is given by table 4.

$$\begin{array}{l}
\langle i, refuse(j, \phi)\rangle \equiv \\
\langle i, inform(j, \neg(I_i\phi\ U\ \psi))\rangle \\
FP : \neg(I_i\phi\ U\ \psi) \\
RE : B_j(\neg(I_i\phi\ U\ \psi))
\end{array}$$

Table 4. Refuse.

Formally, the precondition to send a *refuse* states that sender does not intend, along a run, to eventually bring about ϕ until ψ; the Rational Effects aims that the receiver believes that the sender does not intend to eventually bring about ϕ along a run (i.e., to fulfil the *request*) until ψ. Again, the use of temporal operators greatly simplifies the speech act definitions.

Table 1 provides a definition for two more speech acts: *confirm* and *disconfirm*. The above discussion with respect to *agree*, *refuse* and *inform* also applies to *confirm* and *disconfirm*.

In our view, the FIPA semantics are given by means of a multimodal logic with dynamic and cognitive operators (Uncertain, Feasible, Done, etc.) that greatly increases the complexity of the logic and of the speech act itself. In this sense, $MLTL_I$ greatly simplifies the speech acts definitions by using temporal operators that describe the states of the system.

With respect to the social semantics approaches, Singh [3] proposes that an *inform* means that objectively, "the sender commits that the content is true", and practically, "the sender commits that it has a reason to know the content". Singh's aim is to use commitments to make the ACL semantics public, but in doing so the idea that the sender has the specific goal that the receiver adopts a belief is missing. Another way of saying this is that the illocutionary aspect of the speech act which we defined as "what the speech act is *intended* to achieve" is lost. The analysis proposed by [10] follows similar lines to Singh, but the semantics of speech acts are not longer declarative, but they are given operationally.

5.2 Directives

The FIPA specification of the primitive *request* consists of a sender requesting the receiver to perform some action which can also be another speech act. The argument of the performative is the action that the receiver has to perform. It seems natural to think that one precondition would be that the receiver has the goal of achieving something for the sender. However, this basic aspect is not present in the FIPA definition.

We have already made the point about the complexity of the mentalistic formalizations so we will focus on the social-based proposals: Singh [3] defines *request* to objectively mean that "the sender commits that the receiver will commit to making it true" and practically that "the sender commits that the receiver has committed to accepting a request from him". Giving this meaning to a request means that the motivation to send a *request* is not clear anymore. The motivation that the sender intends to achieve a communicative goal by means of receiver agreeing to perform the action requested cannot be expressed without using motivational operators such as goals and intentions. In this sense, the use of pre-commitments [10] to analyze *requests* fails, in our view, to express that the sender explicitly expresses its interest of having the receiver executing a particular action. In this approach, a *request* is the execution of a public method which creates an empty slot that has to be filled in.

Note, however, that we have not defined actions in $MLTL_I$. Instead, the labelling function is over atomic propositions ϕ which describe the state of affairs of the system at a global state $r(m)$. However, this reflects a simple interpretation of goals: when a *request* is made, the goal of the sender is for the system to reach a particular state of affairs, which in our case, means that we request that some proposition ϕ is true at some global state $r(m)$ of the multi-agent system. This interpretation in terms of the proposition that the sender wants the receiver to achieve fits well with the intuitions about requests. This is also very similar to the intuitive meaning of goals in [42].

In this paper, when sending a *request*, the sender holds the goal of the receiver achieving a particular proposition ϕ, that is, of making true ϕ at some global state $r(m)$. Moreover, since we want the receiver to *really* try to achieve ϕ the preconditions also require that the receiver intends along a run that ϕ be eventually true. Finally, the rational effect to be achieved is that there is a run in which ϕ eventually holds.

$\langle i, request(j, \phi)\rangle$
$FP : G_i(I_j F\phi))$
$RE : F\phi$

Table 5. Request.

5.3 Commissives

Surprisingly, FIPA does not include any commissive speech acts. The traditional example of a commitment is *promise.* The sender expresses the commitment to perform the action expressed in the content of the commissive. Commissives commit the sender to perform the action uttered by the message. That is, by performing a *promise*, the sender states its intention to bring about some ϕ at some point in the system. In our approach agents *promise* to make eventually true some ϕ along a run. When sending a *promise* the sender must hold the intention of making ϕ true. The Rational Effects must be that ϕ is made true at some later point of a run.

$\langle i, promise(j, \phi)\rangle$
$FP : I_i F\phi$
$RE : F\phi$

Table 6. Promise.

5.4 Declaratives

Declaratives are not part of the FIPA CAL either. Declarations have immediate effects in an extra-linguistic institution. They are the original *performative* verbs [43]. Declarations are particularly useful for institutional actions [24]. For example, speech acts to start or terminate an interaction (conversation) are declaratives. In that kind of situations, it is necessary to identify which agents are *allowed* to perform a specific declaration. Usually, agents have the right or the permission to perform a communicative act depending on their role in the particular scenario. In an auction, for instance, the auctioneer has the right to declare the beginning of an auction. An agent wishing to participate should be given the permission (by the auctioneer) to do so. An agent may perform an action for which it has not the right to. Again, all these points are to be included in the pragmatic component of the ACL to be presented in the next sections. In the meantime we content ourselves with defining that when an agent *declares* that ϕ, it has the goal to make ϕ true in the next step of the run. The perlocution states that ϕ holds at the next step of the run. Note the use of the temporal operator X to express that in the immediate next step, ϕ holds along the run.

$\langle i, declare(j, \phi)\rangle$
$FP : G_i(X\phi)$
$RE : X\phi$

Table 7. Declare.

Note that the ACL semantics proposed has solved some of the problems summarized in Table 8. The crucial point is that $MLTL_I$ offers a grounded semantics to beliefs, goals and intentions.

Requirements	**ACLs**	
	FIPA CAL	SAL
Autonomous	?	✓
Complete	-	✓
Contextual	-	-
Declarative	✓	✓
Formal	✓	✓
Grounded	-	✓
Public	-	✓
Perlocutionary	-	-

Table 8. Requirements for ACL semantics.

The rest of the requirements state that the semantics provided by SAL respects the *autonomy* of agents, it defines a *complete* set of speech acts, it provides a *declarative* and *formal* meaning. The requirements left, that the ACL takes into account contextual factors and facilitates the achievement of the perlocutionary effects are not meet by the ACL semantics. It is the pragmatics of the language that account for the social consequences of performing an speech act by enriching speech acts minimal meaning according to the context, scenario, agents' roles, etc.

6 Constitutive and Regulative Norms

We have argued (see 2) many of the interaction protocol approaches developed so far provide a low-level procedural characterization of interactions, or are not expressive enough to take into account the contextual factors affecting communication. Still, interaction protocols are efficient using institutional contexts to model turn-taking strategies. Interaction protocols establish which sequence of messages is appropriate in each scenario. For example, in auctions, turn-taking might underlie the specific rules to ensure that they are created only when they make sense, e.g., a bidder should not make a bid prior to the advertisement.

In our approach, institutional interactions created by a FIPA interaction protocol such as an English Auction can be seen as the *constitutive rules* according to which communication takes place. Constitutive rules only establish the allowed moves within conversation. However, interaction protocols do not *regulate* or constrain the use of the speech acts according to their content and context. In order to do so, we need *regulative rules* that specify agents' rights, obligations and permissions for specific conversational contexts. This distinction between *constitutive* and *regulative* rules in communication is due to [5].

> "Some rules regulate antecedently existing forms of behaviour. For example, the rules of polite table behaviour regulate eating, but eating exists independently of these rules. Some rules, on the other hand, do not merely regulate an antecedently existing activity called playing chess; they, as it were, create the possibility of or define the activity.[...] The institutions of marriage, money, and promising are like the institutions of baseball and chess in that they are systems of such constitutive rules or conventions" [5, p.131]

In our approach, institutional speech acts are those whose meaning depend on the institution in which they are used. Normative interaction protocols correspond to the *constitutive* rules of conversations in terms of agents' rights, obligations and permissions. Additionally, *regulative* rules in agent communication deal with context-dependent aspects: Level of trust between agents, roles, content of messages and other particularities brought about the agents involved in the exchange. For example, a *politeness* rule can be specified that states agents' obligation to send a response to a request. In our framework, regulative rules are expressed by normative *conversation policies* that facilitate the achievement of the *perlocutionary effects*. Conversation policies can also affect the meaning of speech acts in institutions because the object of the rule can refer to an institutional fact. Note that the distinction between interaction protocols and policies is not new, although their relation to constitutive and regulative rules is not explicit in other approaches [9,15].

In the next section we present the pragmatic specification language $NLTL_I$ (Normative Linear Temporal Logic on Interpreted Systems). $NLTL_I$ is defined following the same methodology used for $MLTL_I$ but instead of containing cognitive operators it includes a deontic operator. Once the syntax, semantics and axiomatics of $NLTL_I$ are presented, we define the notions of violation, right and sanction, which are also to be used in the development of the interaction protocols and conversation policies.

7 $NLTL_I$

The normative temporal logic $NLTL_I$ follows the general structure of $MLTL_I$. The main difference is that while $MLTL_I$ was designed to express agents informational and deliberative states, $NLTL_I$ includes linear temporal logic combined with a deontic operator. $NLTL_I$ structures N_I are also defined by associating structures which contain deontic accessibility relations to an interpreted system IS. The definitions of run, global state, point, and the syntax of the temporal operators defined in section 4 remain the same.

The main difference of $NLTL_I$ with other the deontic logics defined to model normative multi-agent systems [18,22,44,45,21] is the fact that their semantics are based on possible worlds. Furthermore, some of these logics are highly complex due to the combination of deontic, dynamic and temporal operators.

However, there is a recent approach to deontic logic which offers a grounded semantics [30]. They define Deontic Interpreted Systems as consisting of a static interpreted system of global states of two different types: those that are allowed and those that are disallowed states of the computation. The interpreted system presented by Lomuscio and Sergot [30] is *static* in the sense that they do not include the notion of run which provides the temporal component in standard interpreted systems [28]. In a more recent work [39], a branching temporal component and two epistemic modalities are added into Deontic Interpreted Systems. $NLTL_I$ differs from the Deontic Interpreted Systems in various ways. First, we define $NLTL_I$ with respect to a interpreted system adapted to model agent communication. Second, the global states of the system are not required to be exclusively deontic. For example, we assume that information about the history of conversation, social structure, institutional facts, etc., could be encoded in the environment's state, whereas the obligations, rights, etc. of agents are to be kept in agents' local states. Third, we include a linear time component in our logic to capture the evolution of the system over time. Linear temporal logic makes the speech act specification simpler than if we were quantifying over runs. Before we present the syntax and semantics of $NLTL_I$ a few remarks on the kind of normative notions that we are interested in is offered in the next section.

7.1 Rights in Agent Communication

The central notion in the specification of norms of conversation is the concept of *right*. *Rights* give agents enough freedom, but also constrain agents' behaviour.

Intuitively, there is a middle ground between traditional obligations and permissions as defined in standard deontic logic [46], and the concept of *right* seems to be appropriate to capture that middle ground. Other definitions of right in the agents literature largely depend on the logic used.

Norman *et al.* [22] use dynamic logic to formalize a notion of right (which resemble traditional permissions) to model agreements. Alonso [44] claims that economic-based theories of rational choice, such as game theory, cannot provide a satisfactory explanation of co-operation and collective action. The reason is that in game theory, agents calculate individually their best choice. Communication does not help either, because agents do not trust each other, and will not respect any commitments. Games with multiple equilibria or with no equilibria at all also pose problems. In particular, it is not possible to reach a rational decision about the agreements agents should make. To solve this, either *ad hoc* solutions or *local points* are proposed. Boella and van der Torre [47] describe rights as sets of strategies of agents' roles. Their proposal is interesting because they argue that rights are exercised by roles, but in our view it is not clear how their idea of right is different from the set of choices that agents have available, or the set of permissions that can be specified for a specific role.

This paper does not intend to account for any possible ambiguity found in the concept of *right*, namely, about the fact that *right* is used to refer to many different things, such as having the right to live, the right to work, a right to feel proud, a right to make pre-emptive attacks, a right to vote, etc. In this sense, rights can be classified as liberties, privileges, claims, power, etc (see [20] for a detailed discussion on these issues). Instead, we are interested in a notion of right useful to a normative approach to agent communication. These interests are based on the assumption that there is a middle ground between obligations and permissions which allows coordination through communication between autonomous agents. This idea is in some sense close to what Castelfranchi [48] calls *strong permission*. A general idea of right we are interested in is provided by the following characterization [49, p.1]:

> "Rights dominate most modern understandings of what actions are proper and which institutions are just. Rights structure the forms of our governments, the contents of our laws, and the shape of morality as we perceive it. To accept a set of rights is to approve a distribution of freedom and authority, and so to endorse a certain view of what may, must, and must not be done."

An interesting point in the etymological meaning of 'right' comes from what is *fair* or *just*. This sense is used when we say that a society is "rightly ordered". When applied to individuals, rights entitle their holders to some *freedom*. For example, an agent can be entitled with the freedom to act in certain ways. In our approach, *rights* are not merely seen as the absence of obligations.

If an agent has the right to perform a speech act, then:

- It is permitted to perform it (under certain obligations), since it does not constitute a violation.

- The rest of the agents are not allowed to perform any action that violates a right-holder's action, otherwise, they are sanctioned.
- The normative system, the group, which is represented by a special type of agent, has the obligation to sanction any violation (we follow Torre *et al.* [45] on this particular point).

The function of norms in agent communication is to stabilize social interactions by making the behaviour of agents predictable to other agents of the system. Permissions are defined as the dual of obligation. Having the right to perform an speech act means that an agent must be given permission to do so and that performing that action does not constitute a violation. Not being obliged not to bring about ϕ ($\neg O \neg \phi$) does not mean that the agent has the right to bring about ϕ ($R\phi$).

The description of agent's rights and obligations can be stored and accessed by every agent at any time, so that the ACL pragmatics is public. An agent may not know whether another agent is sincere, but it can know which rights and obligations the other agent should abide to.

Using $NLTL_I$ allows us to model the evolution of agents' obligations and rights as system changes. The need of including some sort of temporality when modelling normative systems has also been defended by other authors [18,45].

7.2 $NLTL_I$ Syntax

We need to express obligations and rights within an organizational structure in which agents have roles assigned. Rights, Violation and Sanction are not defined as primitives. The only deontic primitive operator of our framework is obligation, denoted by O_i. Following the definition of the cognitive operators in the previous chapter, we will accommodate the interpretation of the primitive deontic operator for its use with respect to runs in an interpreted system.

Regarding, roles, we use the following notation:

- $i\ rr\ j$, means that i and j are role-related by rr.
- i is a member of group c, is expressed by c_i.
- r_i denotes that i plays the role r.

A role is a set of constraints that should be satisfied when an agent plays that role. For example, the role of auctioneer constrains the obligations, permissions and rights of the agent that plays that role. The scope of the role depends on the institutional reality in which it is defined (e.g., auction). A group is a set of agents (roles) that share a specific feature (i.e., being auctioneers). Finally, role relations constrain the relations between roles (e.g., the auctioneer-bidder relation).

The syntax of $NLTL_I$ consists of the vocabulary of the interpreted system IS extended with temporal operators and the deontic accessibility relation. $NLTL_I$ structures (N_I) are actually the result of the combination of IS with an accessibility relation $\mathcal{O}_i$ of a Kripke structure M.

Definition 8 **($NLTL_I$ Syntax).**

Given a finite set of agents $i = (1, \ldots, n)$, a finite set of group names CN, a finite set RN of role names, a finite set RR of role relations, and a countable set AP of primitive propositions, the syntax is defined as follows:

$$\varphi := AP | \neg\varphi | \varphi \wedge \psi | O_i\varphi | F\varphi | G\varphi | X\varphi | \varphi U \psi$$

Regarding the deontic operator $O_i\phi$, its traditional reading is something like "agent i is obliged to bring about ϕ", or maybe "agent i ought to bring about ϕ". It is also interesting the interpretation proposed in the Deontic Interpreted Systems (DIS) framework [30]; they define a modality $O_i\phi$ to express that "if agent i is functioning correctly, then ϕ holds". Following this, and considering the fact that our system has time built in, the deontic operator for obligation $O_i\phi$ defined in $NLTL_I$ means that "the system is at a point in which ϕ holds if agent i works (acts) correctly", which shares the same spirit that the interpretation used for for the cognitive concepts defined in $MLTL_I$.

As usual, $P_i\phi$ is the dual of $O_i\phi$ such that

$$P_i\phi = \neg O_i \neg\phi$$

Which we could gloss as "the system could be at a point in which $\neg\phi$ holds if agent i is not working (acting) correctly".

7.3 $NLTL_I$ Semantics

$NLTL_I$ structures are generated by grounding a deontic Kripke structure M into the interpreted system IS. .

Definition 9 **(Deontic Structure).**

A Deontic structure $M = (S, \mathcal{O}_i, \ldots, \mathcal{O}_n, L)$ is serial if for any accessibility relation $\mathcal{O}_i$ we have that for all s there is a t such that $(s, t) \in \mathcal{O}_i$.

From the Deontic structure M and IS we generate N_I structures for $NLTL_I$:

Definition 10 **(N_I structure).**

Given a system of runs T, a structure N_I is generated by associating the interpreted system $IS = (T, L)$ with the serial Kripke structure $M = (S, \mathcal{O}_i, L)$, such that $N_I = (GS, \mathcal{O}_i, L)$ where:

- *GS corresponds to the sets of global states in IS.*
- *L is a labelling function $L : S \rightarrow 2^{AP}$ from global states to truth values, where AP is a set of atomic propositions. This function assign truth values to the primitive propositions AP at each global state in GS.*
- *$\mathcal{O}_i$ where $i = (1, \ldots, n)$ is a set of agents, gives a serial accessibility relation on global states. Thus, we have that $(l_e, l_1, \ldots, l_n)$ $\mathcal{O}_i$ $(l'_e, l'_1, \ldots, l'_n)$ if $l'_i \in GS_i$. If $g = (l_e, l_1, \ldots, l_n)$, $g' = (l'_e, l'_1, \ldots, l'_n)$, and l_i $\mathcal{O}_i$ l'_i, then we say that g and g' are $\mathcal{O}_i$-accessible to agent i. The formula $O_i\phi$ is defined to be true at g exactly if ϕ is true at all the global states are $\mathcal{O}_i$-accessible from g.*

Definition 11 **($NLTL_I$ semantics).**

The semantics of $NLTL_I$ is inductively defined as follows:

$(IS, r, m) \models \phi$ *iff* $L(r, m)(\phi) = true$
$(IS, r, m) \models \phi \wedge \psi$ *iff* $(IS, r, m) \models \phi$ *and* $(IS, r, m) \models \psi$
$(IS, r, m) \models \neg\phi$ *iff it is not the case that* $(IS, r, m) \models \phi$
$(IS, r, m) \models O_i\phi$ *iff* $\forall(r', m')$ *such that* $(r, m)\ \mathcal{O}_i\ (r', m')$*, then* $(IS, r', m') \models \phi$
$(IS, r, m) \models X\phi$ *iff* $(IS, r, m+1) \models \phi$
$(IS, r, m) \models F\phi$ *iff for some time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models G\phi$ *iff for all time* $m' \geq m$ $(IS, r, m') \models \phi$
$(IS, r, m) \models \phi U \psi$ *iff there is some time* $m' \geq m$ *such that along the run such that* $(IS, r, m') \models \psi$ *and for each* $m \leq m'' < m'$ *we have* $(IS, r, m'') \models \phi$*.*

In the interpretation for obligations proposed here, this motivational attitude is ascribed to the agents by an external reasoner. Two points (r, m) and (r', m') are $\mathcal{O}_i$-related if $(r'm')$ makes possible that agent i functions correctly at the point (r, m). The notions of violation, right and sanction are defined as non primitives.

First, we extend the language of $NLTL_I$ to include the propositional constant V as an abbreviation of the formula defined below. The meaning of the expression $V(\phi)$ states that if ϕ holds at some point (r, m) then ϕ is considered to be a violation.

Definition 12 **(Violation).**

From each literal built from a variable ϕ, $V(\neg\phi)$ *means that* $\neg\phi$ *is a violation at some point* (r, m) *in the system for some* $ns \in NS$*, such that* NS *is a set of norms, iff*

$$O_i(\phi\ U\ \psi) \rightarrow (\neg\phi\ U\ \psi)$$

If the system is at a point in which ϕ holds if agent i acts correctly until ψ holds, then $\neg\phi$ holds until ψ holds. Agent i not working correctly means that ϕ does not hold and that constitutes a violation in our system. The notion of violation is of course inspired by the work of Anderson [50].

Some authors argued that undesirable states-of-affairs do not always follow infractions, and that not all violations are sanctioned. In any case, we understand the constant V as denoting a state in which some norm is violated.

Note that we have added a new element to our framework, namely, that of the normative system $ns \in NS$ that can be seen as either a a norm of the system or as a normative agent, depending on the situation. Furthermore, we model ns as the environment's local state g_e in $NLTL_I$. Thus, the environment's local state of the system will act as a normative system that assigns agents' rights, obligations and permissions, and that it is in charge of sanctioning agents when the violate a norm. We will see that our framework allows us to model the ns as an agent in charge making agents abide by the norms quite naturally.

We can imagine a context in which if an agent i is functioning correctly then it will send an *accept* message as a response to a *request* when some agreement preconditions hold. Conversely, if agent i does not *accept* the request it violates the conversation norm that specified the correct functioning of that agent (i.e., its obligations).

In some cases, it may be interesting to specify agents' behaviour by ruling that performing some action at some point does not constitute a violation. We use the notion of *right* to express this kind of norms. Thus, by using rights we specify agents' freedom to act in some specific way without that violating a norm. In this sense, *rights* are considered here exceptions to obligations. An agent has the right to bring about ϕ under some condition ψ if bringing about ϕ is not a violation ($\neg V(\phi)$). From an external point of view, we say that "there is a point in the system where agent i is functioning rightly if the holding of ϕ does not constitute a violation". We formalize this concept as follows:

Definition 13 (Right).
Let NS be a set of norms $(ns_1, \ldots, ns_n)$ encoded in the environment's local state g_e, and let the variables of agent Ag contain a set of violation variables $V(\phi)$ such that $\phi \in AP$. Agent i's functioning is right when ϕ holds, $R_i\phi$, for some $ns \in NS$ at some global state $r(m)$, $r(m) \in GS$ iff

$$\neg V\phi \, U\psi$$

Therefore, having the right to bring about ϕ under some precondition ψ means that until ψ holds along a run, then ϕ not being a violation also holds along that run. Rights are not only permissions. When an agent is exercising a right, its freedom is specified in relation to that right.

From a linguistic point of view, we can understand right-based rules as *defaults*; if law changes and an exception to a right is made, then from that point onwards exercising that particular right is considered a violation. The linguistic interpretation is that if by default an agent has the right to *agree* or *refuse* to a *request*, then there can be a new policy that overrules the default and states that from now on exercising the right to *refuse* to a *request* sent by some agent-manager is a violation of the agent-manager's rights.

So, what happens when an agent not functioning correctly or rightly brings about some ϕ, which constitutes a violation? We stated that in these cases, there is an agent ns, called the normative agent, that, if working correctly, will sanction the offending agent. The specific nature of the sanction varies from system to system, and within the same system, from one scenario to another. The general pattern, however, is that the sanctioned agent will have the obligation to do something as a punishment for its violation. For example, agent i wants to participate in a bidding process to buy a property on behalf of some estate agents. Say that to enter the auction, you need to pay some deposit of 1,000 in advance. If the agent (its role is bidder, $bidder \in RN$) wins the auction with an agreed price of 200,000 for the property but decides to break the agreement by withdrawing the bid then it is sanctioned by having the obligation to pay a fine (given that "it is not functioning correctly", that is, following the constitutive rules that define the protocol). The fine can be the 1,000 deposit paid to enter the auction. We can formalize this notion of sanction as follows:

Definition 14 (Sanction).

Let b denote the role of bidder such that $b \in RN$, then a agent i such that $i \in Ag$ playing the role of bidder b has the obligation to pay a fine (by bringing about ϕ) iff

$$b_i \wedge (O_i\phi\ U\ \psi) \wedge (\neg F\phi\ U\ \psi) \rightarrow O_i\omega$$

Thus, if the system is at a point in which if an agent playing the role b (bidder) is acting correctly, ϕ holds until ψ holds and ϕ does not eventually happens while ψ, then i is sanctioned with the obligation of paying some fine ω.

This notion of sanction presented here can be greatly complicated by considering more complex behaviour to detect and sanction violations. However, for our purposes the relatively minimal normative structure defined in this section is sufficient to formulate a normative pragmatics for agent communication. In any case, the normative specification of multi-agent systems is a difficult problem in its own, and it exceeds the purposes of this paper.

7.4 $NLTL_I$ Axiomatics

Studying the complexity of the specification language $NLTL_I$ is interesting because we do not want that protocols defined using $NLTL_I$ that are too computationally hard.

It is well-known that the system KD_n that characterizes Standard Deontic Logic is sound and complete. In this section we give a complete and sound axiomatization of $NLTL_I$ which consists of the axioms for obligations and the linear temporal component. The following axioms provide a sound and complete axiomatization of $NLTL_I$:

PC All tautologies of propositional logic.
T1 $X(\phi \rightarrow \psi) \rightarrow (X\phi \rightarrow X\psi)$.
T2 $X(\neg\phi) \equiv \neg X\phi$.
T3 $\phi\ U\ \psi \equiv \psi \vee (\phi \wedge X(\phi\ U\ \psi)$.
RT1 From ϕ infer $X\phi$.
RT2 From $\phi' \rightarrow \neg\psi \wedge X\phi'$ infer $\phi' \rightarrow \neg(\phi\ U\ \psi)$.
MP From ϕ and $\phi \rightarrow \psi$ infer ψ.

The axiomatics for the deontic operator is as follows. i denotes a set of agents such that $i = 1, \ldots, n$.

PC All instances of propositional tautologies.
MP If ϕ and $\phi \rightarrow \psi$, then ψ.
NEC If ϕ, then $O_i\phi$.
K $O_i(\phi \rightarrow \psi) \rightarrow (O_i\phi \rightarrow O_i\psi)$.
D $O_i\phi \rightarrow \neg O_i\neg\phi$.

Theorem 2. *The system $NLTL_I - Ax$ is a sound and complete axiomatization with respect to the class of models $NLTL_I$ that are serial.*

The proof of the axiomatics of $NLTL_I$ can follow the same technique as that of $MLTL_I$ (for a proof of linear temporal logics with an *S5* axiomatics for a knowledge operator see [38]).

We had various motivations to define this logic: First, given that $NLTL_I$ is going to define the semantics of the normative operators used in the conversation norms, a deontic component was needed. We have introduced an standard operator for obligation which was then used to define several other normative concepts. Among them, the notion of *right*. Second, the semantics of $NLTL_I$ are grounded upon interpreted systems. Finally, the temporal operators provide useful tools to analyze how agents' rights and obligations change over time. This also means that coordinating communication through norms allows us to focus on the external behaviour of agents, instead of modelling their mental reasoning to interpret messages.

Next section presents the interaction protocols and conversation policies that form the ACL normative pragmatics. The set of normative operators defined by $NLTL_I$ are used in the conversation norms defined in the following sections.

8 Conversation Norms

$NLTL_I$ as a specification language provides a formal, unambiguous, and grounded meaning for the key normative concepts to be used in the specification of norms of conversation. A normative point of view to agent communication can be summarized by the following points:

- Agent conversations often occur within an institution. In fact, there are specific speech acts such as *declare* that are pure institutional facts. When the appropriate role uses the adequate speech act within an institution, the agent has performed *an action* by sending that message. The rules defining the institution are *constitutive rules* specified by means of *interaction protocols*.
- Constitutive rules specify protocols such as English Auction, whereas *regulative rules* are concerned with more context-dependent aspects in the form of *conversation policies*. Both constitutive and regulative rules are declarative and their aim is to stabilize communication by contextually constraining agents' communicative behaviour.
- Agents play roles, and their roles influence their communicative behaviour thereby facilitating the achievement of the Rational Effects.
- Right is a normative notion that rules agents' communicative behaviour by specifying their freedom instead using pure restrictions and/or obligations. Furthermore, definitions of violation and sanction are provided.

The protocols and policies that conform the norms of conversation must be declarative so that they specify *what agents can achieve* using the rules instead of *how to achieve* a particular result. In our view, formal logic constitutes a more appropriate tool reason about multi-agent systems than procedural programming languages or ontology-based languages like OWL [16]. Besides, there are a number of verification techniques for logic-based specification languages

[40] of systems that can be put to good use for the the verification of agent communication languages.

When considering which language used for the specification of the speech acts library, we conclude that although the semantics of the cognitive and temporal operators were defined by $MLTL_I$, the syntax of messages was going to follow the FIPA specification. We gave two reasons for this decision: First, most of the criticisms have been addressed to its semantics. Second, we are interested in contributing to the standardization effort of agent communication led by FIPA, so we focused on solving some of the problems of semantics of FIPA CAL.

However, we cannot use the same strategy and use UML diagrams for specifying interaction protocols and conversation policies because they merely represent the order in which messages can be uttered. This paper claims that ACL pragmatics have been largely underdeveloped and it proposes a way of providing expressive and high-level normative pragmatics.

8.1 Representation

Leaving aside the procedural and diagram-based approaches already discussed, there is a recent trend in the specification of interaction protocols based on propositional linear temporal logic (PLTL) [51] and finite-state machines [52].

Endriss [51] proposes to specify the class of all sequences of messages that are allowed by a given protocol. He uses propositional LTL (PLTL) to specify the protocols and model-checking techniques to verify the runtime conformance of conversations to the protocol. Conversation templates are defined as sequences of dialogue moves (speech acts). Those dialogues that can be captured by protocols based on finite-state machines are legal according to a protocol if and only if they are accepted by the finite-state machines that correspond to the protocols.

Standard finite-state protocols and PLTL are not suitable to interactions involving commitments, social expectations and, in our case, rights and obligations. For example, we are interested in attributing to the (role of) auctioneer the obligation to close the auction at some point, and to give the bidder the right to bid after the auctioneer *declares* the auction open. In other words, we need to consider how the system evolves as a result of agents' performing actions (speech acts in our case). It is convenient that the execution of speech acts be ruled by some protocols and policies if we want communication to be efficient.

8.2 Normative Protocols and Policies

Thus, for the formulation of a high-level norms of conversation, we need to consider taking into account the following elements:

1. A set of atomic propositions P to describe facts. They usually consist of propositional content of messages.
2. A set of **agents** that participate in the conversation.
3. A set of **speech acts** (query, request, etc.) that convey the illocutionary and perlocutionary acts of performing a communicative action.

4. A set of **normative rules** of the form $np_i(sa(i,j,P))$ which consist of a normative predicate (right, obligation), the action (a speech act) and the content of the speech act ϕ.
5. A set of **broadcasting actions**. Broadcasting actions denoting events state that a speech act sa is sent, received, answered or not-answered. This aspect refers to the history of the conversation.
6. A set of **roles** taken by the agents involved in the interaction. Roles are specified as facts about individual agents $role_i$.
7. An agent performing the role of **normative system** ns encoded in the environment's local state of the system. ns has the obligation of monitoring the conversations to detect violations, apply sanctions and making sure that messages are delivered.

In $NLTL_I$, we formalized obligations, rights and permissions as entirely dependent on agents' local states. Thus, any communicative actions they take are a function of their local state. Their local states also contain information regarding their initial state in the execution and the history of messages sent and receive (i.e., its conversational record; we build on the knowledge-based interpreted system model [28] to model the history of conversation).

Definition 15 (History).
Let us consider an agent i such that $i \in Ag$, a set of broadcasting actions BE, a set of speech acts SA, a set of initial states S_{0i} for agent i, and a set of contextual actions DO_i for i. A history for agent i is a sequence where

1. *The first element is in S_{0i},*
2. *the later elements consist of nonempty sets of broadcasting actions such as $sent_i(sa(i,j,P))$, $receive_i(sa(i,j,P))$, or $do(i,\alpha)$ such that $\alpha \in DO_i$.*

The history of conversation of an agent i at some point (r,m) of the system is composed by its initial state and the sequence of steps corresponding to i's actions up to time m. We can also say that if an agent i at a point (r,m) has only sent an *agree* speech act to agent j, $sent_i(agree(i,j,P))$, then its history at point (r,m) is the result of appending the set $\{sent(i,j,agree(P))\}$. Furthermore, a broadcasting event occurs in round $m+2$ of run r if it is contained in some agent's history of conversation in $(r,m+2)$.

We have mentioned above that our framework models the system environment as a normative agent ns whose task is to decide when performing a speech act is a violation and the sanctioning it when appropriate. In order to take these decisions the environment's local state must record the events that take place in the system, namely, the speech acts performed by the agents involved in a conversation. Furthermore, it need to keep an up to date record of the evolution of agents' rights, obligations and permissions according to the actions they have performed so far, taking into account the fact that performing speech acts' cause social expectations. Note, however, that determining and reasoning about the actions that ns can perform is part of the social structure of the system. Therefore, the ACL specification does not account for the acquisition of knowledge or beliefs by ns nor the reasoning employed to sanction violations. Doing so is not within the purposes of this paper.

Thus, we need to consider both agents' and the environment's actions to explain how their actions cause the system to change state: $(\alpha_e, \alpha_1, \ldots, \alpha_n)$ and a transition function that maps global states to global states: $\delta(\alpha_e, \alpha_1, \ldots, \alpha_n)$. We can now define a protocol as a mapping from the set L_i of agent i's local states to nonempty sets of acts in BE_i. Furthermore, a protocol P_e for the normative agent ns is a mapping from the set of the environment's local states L_e to nonempty sets of actions in DO_e.

We include normative concepts and propositional variables in our protocol rules. Furthermore, these rules must be declarative, that is, they say what the rights and permissions of the agents are, rather than a procedure to move from to one state to another. This secures the high-level character of our ACL. Interaction protocols are defined in NPRAG using if-then rules as the constitutive rules that specify the legal interactions of conversations. If agent j receives a *request* then agent j has the right to answer either by *agreeing* or by *refusing.*

We elaborate on these points in order to give specify some of the FIPA interaction protocols.

8.3 Request

Typically, protocols are described by means of programs written in some programming language. For clarity of exposition we will use in this paper $NLTL_I$ extended with parameters for agents, roles and actions. Having extended the Interpreted Systems model to express normative notions for their use in agent communication languages, we could have employed a similar strategy and adapt a simple programming language defined within the interpreted systems model [28] to express protocols that include agents' roles, rights, obligations, speech acts and broadcasting actions. After showing in this section how our approach can be used to specify an ACL pragmatics using norms, we will offer an example of a protocol using a simple programming language.

Let us consider again the FIPA Request interaction protocol. This protocol allows one agent to request to bring about some propositional content ϕ. If the receiver of the *request* speech act is functioning *rightly*, then it will send an *agree* or a *refuse* as a response to the *request.* If the answer is an *agree*, and the agent is functioning correctly at that point, then it will communicate an *inform* if the request is satisfied, or a *failure* if the object of the request is not achieved. The specification of this protocol in NPRAG looks is composed by the following norms of conversation:

1. $principal_i \wedge secretary_j \rightarrow R_i(request(i, j, \phi))$
2. $receive_j(request(i, j, \phi)) \wedge \neg sent_j(refuse(j, i, \phi)) \rightarrow R_j(refuse(j, i, \phi))$
3. $receive_j(request(i, j, \phi)) \wedge \neg sent_j(agree(j, i, \phi)) \rightarrow R_j(agree(j, i, \phi))$
4. $sent_j(agree(j, i, \phi)) \wedge F\phi \rightarrow O_j(inform(j, i, \phi))$
5. $sent_j(agree(j, i, \phi)) \wedge \neg F\phi \rightarrow O_j(failure(j, i, \phi))$

Note that the proposition of the normative predicates for rights, obligations and permissions are taken as expressing a communicative action like "agent i agrees with agent j to bring about some ϕ".

In the Request specification there are two agents i and j that take the roles of *secretary* and *principal* respectively. As a propositional content of the speech acts, we can think of a situation in which agent *principal* has the right to request to agent *secretary* to book a number of flights.

The rules state that the *principal* has the right to send any request message to the secretary, and that the secretary can answer to these messages either by agreeing or refusing if an answer has not been produced yet. The two obligation rules state that an agent has the obligation to send an *inform* having already sent an *agree* message and not having sent yet *inform* that the request has been satisfied.

As it is, the reasoning rules presented above capture the transitions that a system functioning rightly can perform under the NPRAG Request interaction protocol. However, we need something else, that is, to instantiate some of the facts of the NPRAG specification of *request.* In particular, we need to say which messages have been sent or are still pending. As discussed above, the history of conversation is part of agents' local state, whereas the status of messages and agents' rights and obligations are encoded in the environment's local state. None of these components are part of the interaction protocol specification. Indeed, for the sake of generality, it is desirable that our protocols only provide a set of norms of conversation to facilitate agents' next move *in absence of any specific circumstances.*

8.4 Query-If

In the FIPA Query-IF interaction protocol, an agent i queries agent j whether or not a proposition ϕ is true. The receiver has the right to either *agree* or *refuse* to send and *inform* message providing an answer. In the case that agent j agrees, then it has obligation to send a notification which can be an inform stating the truth of falsehood of the proposition ϕ. If agent j sends a refuse message the protocol ends there. We only show the relevant normative rules of this protocol:

1. $journalist_i \wedge policitian_j \rightarrow R_i(queryif(i,j,\phi))$
2. $receive_j(queryif(i,j,\phi)) \wedge \neg sent_j(refuse(j,i,\phi)) \rightarrow R_j(refuse(j,i,\phi))$
3. $receive_j(queryif(i,j,\phi)) \wedge \neg sent_j(agree(j,i,\phi)) \rightarrow R_j(agree(j,i,\phi))$
4. $sent_j(agree(j,i,\phi)) \wedge F\phi \rightarrow O_j(inform(j,i,\phi))$
5. $sent_j(agree(j,i,\phi)) \wedge \neg F\phi \rightarrow O_j(failure(j,i,\phi))$

We can see that its structure is almost equivalent to the Request protocol; only the use of *queryif* instead of *request* is different. This means that our proposal is high-level enough so that it is easily adaptable to represent different interaction protocols and different contexts. Only the content of the messages and the roles of the agents may change.

The specification of the constitutive rules of conversations enable us to formulate a number of policies that contextually contrain agents' communicative behaviour within the protocol in terms of their rights, obligations and permissions.

8.5 Conversation Policies

Since conversation policies usually restrict agents' behaviour within conversations, the notation of the pragmatic regulative rules that conform NPRAG conversation policies consists of the components used in the specification of interaction protocols. Moreover, we would like to stress the importance of one of the elements and propose a new one:

- A set of **contextual actions** DO_i that depend on specific scenarios, e.g., the action of *bidding* depends on the agent being in an auction.
- A **conflict resolution action** so that in case of conflict between rules of a policy, one rule has *priority* over another one.

Constructs such as the conflict resolution actions, the contextual and broadcasting actions depend on the platform in which agents run. That is, these actions are defined by the programming language in which agents are built. For example, in Java built platforms like JADE, sending messages is simply a case of creating an ACLMessage, setting the parameters (sender, receiver, reply-to, performative, etc.) and then sending it using the send() method in the agent object.

If the normative rules in the interaction protocols specify the legal structure of the conversation, conversation policies regulate agents' behaviour according to contextual information within the protocol. Roles and background knowledge provide valuable information for agents to choose the right course of action. Unlike the specification of the interaction protocols, we consider the content of the speech acts when proposing normative rules. Furthermore, note that the policies are tightly combined with the ACL semantics defined in the previous chapter. Thus, the meaning of a speech act such as *queryif* is enriched by the rights, obligations and permissions of agents to use that particular speech act.

We can imagine a situation in which an agent *paxman* has the right to *queryif* a politician agent *pm* about the truth of the "peersmoney" scandal as long as we are not in electoral campaign.

$$paxman_i \wedge pm_j \rightarrow R_i(queryif(i, j(peersmoney))U\neg(elections)$$

Another example can be of an agent j acting on behalf of an airline company serving flights to European countries, that could have a policy that states that it should agree to every request regarding flight tickets to Europe (i.e., answering about flight times and providing the best offer for a potential buyer) and another one specifying that it has the obligation to refuse every request about flights to non European countries.

- $customer_i \wedge seller_j \wedge receive_j(request(i, j, europeanFlight)) \rightarrow O_j(agree(j, i, \phi))$.
- $receive_j(request(i, j, nonEuropeanFlight)) \rightarrow O_j(refuse(j, i, \phi))$.

This issue shows how using normative conversation policies help agents to achieve the perlocutionary effects since the perlocution of *agree*, namely, that the receiver satisfies the object of the requested action, is now specified to be an obligation of the seller. This is a crucial point to help agents to achieve the rational effects of an speech act. For example, we can specify a rule to state that if an agent makes a promise to increase the taxes on air planes fuel, then it has the obligation to do so:

$$G(send_i(promise(i, public, taxairplanesFuel)) \rightarrow \\ O_i(increaseTaxes(airplanesFuel))$$

The extension of our approach to other protocols and policies in the FIPA specification is fairly straightforward. Our approach shows how a well-defined normative concepts can be used to propose a high-level ACL pragmatics that are declarative, takes into account the context and that helps agents to achieve the perlocutionary effects of the speech acts. These two properties of the normative pragmatics, *contextual* and *perlocutionary*, fill in the last gaps in the list of requirements for ACLs discussed in section 1 and table 1. Next section offers a comparison to other approaches and discusses some short term future work necessary to improve the ongoing work presented in this paper. As a final note, the simplicity of the protocols and policies specified in this section was intentional. An important point for any future application of agent communication languages remains the proposal of high-level but simple ACL semantics and pragmatics.

8.6 Programs

Fagin *et al.* introduce a simple programming language which can be easily related to an Interpreted System [28]. Although the language is designed to express agents' knowledge, it can be adapted for its use in specifying norms of conversation. The basic standard program for agent i consists of statement of the form

```
case of
        if $t_1 \wedge k_1$ do $a_1$
        if $t_2 \wedge k_2$ do $a_2$
end case
```

where the t_i's are tests about some facts, k_i are knowledge test for agent i and a_i denote agent i's actions. We modify these knowledge-based programs to express tests over obligations, rights and permissions of agents, namely, to normative-based programs. The normative component consists of a Boolean combination of the form $O_i\varphi$ where φ can be an arbitrary formula that may include other deontic and temporal operators. Using this simple language we can express high-level protocols for agent communication. We represent the Fipa Request protocol specified above in table 9.

At first glance, it may seem a bit odd to use obligations after the operator **do**. However, in the interpretation of obligations and rigths provided by $NLTL_I$, $O_i\varphi$ means that "φ holds in agent i is working correctly" whereas $R_i\varphi$ is interpreted as "φ holds at some point of the system (r, m) if agent i is acting rightly".

```
case of
        if $(principal_i \wedge secretary_j) \wedge R_i(request(i,j,\phi))$ do $send_i(request(,j,\phi))$
        if $receive_j(request(i,j,\phi)) \wedge R_j(refuse(j,i,\phi))$ do $sent_j(refuse(j,i,\phi))$
        if $receive_j(request(i,j,\phi)) \wedge R_j(agree(j,i,\phi))$ do $sent_j(agree(j,i,\phi))$
        if $sent_j(agree(j,i,\phi)) \wedge F\phi$ do $O_j(inform(j,i,\phi))$
        if $sent_j(agree(j,i,\phi)) \wedge \neg F\phi$ do $O_j(failure(j,i,\phi))$
end case
```

Table 9. Program for Request Protocol.

Therefore, the last statement of the program denotes that if agent j has *agreed* to bring about some ϕ to agent j and ϕ does not eventually happens in the run of the system then agent j does send a failure message to agent i if working correctly.

9 Concluding Remarks

The characterization of roles is inspired by the work done on organizational concepts [53,45]. Other authors [18], have also presented temporal deontic logic with dynamic operators, but the combination of deontic, dynamic and temporal notions results in a logic that is too complex for our purposes.

In a very recent paper Boella *et al.* [54] present a role-based approach to ACL semantics. They intend to make the ACL semantics public by attributing mental states to social roles instead of agents. Thus, there are two sets of beliefs, those that are public and are ascribed to roles, and those that are private and belong to the agents' private mental states. A role is constrained by a set of social rules (rights, obligations, permissions, etc.) that define the expected behaviour of any agent playing the role. These social rules may or may not conflict the private beliefs and goals of agents. In any case, even if beliefs and goals are attributed to roles, agents playing a role would still need to reason about their beliefs and goals. From a semantic point of view, defining the ACL semantics in terms of roles makes the semantics less general, since the meaning of speech acts would be affected by agents' role. For example, two roles that are considered are those of *speaker* and *receiver*.

We believe that this paper offers a new framework for agent communication where the meaning of speech acts consists of the combination of the semantic specification and the NPRAG rules that constrain their use.

First, it clearly distinguishes semantics and pragmatics of the language. Semantically, it offers a computationally grounded specification language based on $MLTL_I$. This enables to define meaningful and public communicative actions. Regarding the pragmatics, it presents a procedure using normative rules to guide agents in conversation. Unlike research in ACL semantics, there are not many works that attempt to capture both aspects of communication in the same framework.

Considering the list of requirements for ACLs discussed, the approach presented in this paper achieves a number of objectives. After the semantics of the language was specified, the aim was to produce a pragmatic theory that would consider how contextual information constrains agents' behaviour, and how proposing normative rules for the use of speech acts facilitate the achievement of the perlocutionary effects.

1. **Autonomous**: The ACL semantics (SAL) do not completely fix agents communicative behaviour because the fulfilment of the perlocutionary effects are left to the ACL pragmatics.
2. **Complete**: We have defined a complete set of speech acts, understanding "complete" as representing every category in Searle's taxonomy. Searle's taxonomy is by no means a closed list; one could imagine a more fine-grained taxonomy including more systematic distinctions between types of directives such as yes/no questions, prohibitives, etc. However, this paper completes FIPA specification by defining speech acts for commissives and declaratives.
3. **Context**: In agent communication contextual factors include the role that agents play in the application scenario, the delegated tasks agents try to achieve, the propositional content of messages, and the record of previous exchanges. The use of normative concepts to model ACL pragmatics keep to a minimum agents' reasoning about each others' mental states. In that sense, it is more *efficient*. Furthermore, by avoiding that reasoning, the specification of conversation protocols and policies is greatly *simplified*.
4. **Declarative**: By providing a declarative definition of ACL semantics and pragmatics, specifying what the meaning is instead of a follow-the-rule low-level procedure, the resultant unified ACL is a high-level language.
5. **Formal**: The unified ACL is specified using two formal logics, $MLTL_I$ and $NLTL_I$ that describe the evolution of a multi-agent systems with respect to the agents' beliefs, goals, intentions, obligations and rights. A particular care was to provide an external interpretation of beliefs, goals and intentions in a way that those attitudes would refer states of a system instead of private mental states of the agents. In doing so, we were paving the ground provide a semantics and pragmatics suitable for verification.
6. **Grounded**: The notion of interpreted system was introduced [28] upon which the two specification languages $MLTL_I$ $NLTL_I$ were grounded.
7. **Public**: We claim that the illocutive/intentional aspect of communication should be preserved in the ACL semantics. This paper proposes an external interpretation of motivational concepts by relating them to states of agents in a system.
8. **Perlocutionary**: Conversation norms in the form of protocols and policies enable agents to achieve the perlocutionary effects by specifying obligations and rights on the participants. In order to preserve agents' autonomy, we offer a notion of right which specifies agents' behaviour when acting *rightly*.

It should be made clear that complying with these requirements is not the end of the story but rather its beginning. In other words, we see these properties as the starting point for the development of agent communication languages. A number of problems are still to be solved including issues of verification, implementation and the interaction of the communicative module with the rest of the social structure of the system.

Further work includes providing proofs for some properties of the interaction protocols with respect to interpreted systems. Furthermore, it would be interesting to provide a detail proof of the soundness and completeness of $MLTL_I$ and $NLTL_I$. We also need to verify the semantics of $NLTL_I$ in various ways. There are various methods of verification which depend on the type of ACL, on the information available, and on whether we are interested in verifying the ACL at design time or at run time [7]. Unlike other approaches, we are particularly interested in verifying the ACL pragmatics (only) because the pragmatics encodes the general communicative behaviour of agents. Following this, the type of the ACL to be verified corresponds, in our approach, to the normative component. Current work on pre-runtime verification of complex formal logics [55,56] looks very promising. Furthermore, it would be interesting to produce more sophisticated implementations of conversation protocols and policies in a manner that they could be integrated with platforms such as 3APL and BOID [57,45].

Deontic concepts are increasingly used in the specification and verification of multi-agent systems. It is unrealistic to assume that a whole open multi-agent system may be controlled by the same vendor. Thus, this makes it difficult to verify agents' conformance with the set of semantic and pragmatic specifications of ACLs. In this sense, by adopting a normative point of view, it seems more sensible to leave open the theoretical possibility of agents violating the norms. We can then use the formal language provided to reason about the consequences that result from those violations. Separating the specification language (from the implementation language) allows us to reason about external properties of the system. Further work on these issues would include the definition of more normative notions to complement *right* which may be more suitable to specific circumstances, and to embed our ACL in a normative multi-agent system.

References

1. FIPA ACL: FIPA Communicative Act Library Specification (2002) http://www.fipa.org/repository/aclspecs.html.
2. Finin, T., Weber, J.: Specification of the KQML Agent-Communication Language (1993) Finin T. and Weber J. Draft. Specification of the KQML Agent Communication Language. The DARPA Knowledge Sharing Initiative, External Interfaces Working Group.
3. Singh, M.P.: A social semantics for agent communication languages. In: Issues in Agent Communication, volume 1916 of LNAI. Berlin: Springer-Verlag (2000)
4. Colombetti, M., Formara, N., Verdicchio, M.: The role of institution in multi-agent systems. In: Atti del VII convegno dell'Associazione italiana per l'intelligenza artificiale (AI*IA 02). (2002)
5. Searle, J.R.: Speech Acts. An Essay in the Philosophy of Language. Cambridge: Cambridge University Press (1969)

6. Wooldridge, M.: Verifying that agents implement a communication language. In: Sixteenth National Conference on Artificial Intelligence (AAAI-00), Orlando, FL (1999)
7. Guerin, F., Pitt, J.: Agent Communication Frameworks and Verification. In: AAMAS 2002 Workshop on Agent Communication Languages, Bologna. (2002)
8. Chaib-Draa, B., Labrie, M.A., Bergeron, M., Pasquier, P.: Diagal: An agent communication language based on dialogue games sustained by social commiments. Automous Agents and Multi Agent Systems **13** (2006) 61–95
9. Flores, R.A., Kremer, R.C.: To commit or not to commit: Modelling agent conversations for action. Computational Intelligence **18** (2002)
10. Fornara, N., Colombetti, M.: A commitment-based approach to agent communication. Applied Artificial Intelligence an International Journal **18** (2004) 853–866
11. Agerri, R., Alonso, E.: A semantic and pragmatic framework for the specification of agent communication languages: Motivational attitudes and norms. In Kolp, M., Bresciani, P., Henderson-Sellers, B., Winikoff, M., eds.: Agent-Oriented Information Systems III (Post-Proceedings). Volume 3529., Heidelberg, Springer-Verlag (2006) 16–31
12. Cost, R.S., Chen, Y., Finin, T., Labrou, Y., Peng, Y.: Modeling agent conversations with colored petri nets. In: Workshop on Specifying and Implementing Conversation Policies, Third International Conference on Autonomous Agents (Agents '99), Seattle (1999)
13. Greaves, M., Holmback, H., Bradshaw, J.: What is a Conversation Policy? In Dignum, F., Greaves, M., eds.: Issues in Agent Communication. Heidelberg, Germany: Springer-Verlag (2000) 118–131
14. Pitt, J., Mamdani, A.: A protocol-based semantics for an agent communication language. In: Proceedings of the 16th International Joint Conference on Artificial Intelligence IJCAI'99, Stockholm, Morgan-Kaufmann Publishers (1999) 486–491
15. Phillips, L.R., Link, H.E.: The role of Conversation Policy in carrying out Agent Conversations. In: Workshop on Specifying and Implementing Conversation Policies, Autonomous Agents'99, Seattle. (1999)
16. Kagal, L., Finin, T.: Modelling conversation policies using permissions and obligations. Autonomous Agents and Multi Agent Systems (2007)
17. Holmback, H., Greaves, M., Bradshaw, J.: Agent A, Can you pass the salt? - the role of Pragmatics in Agent Communication (1999) http://citeseer.nj.nec.com/holmback98agent.html.
18. Dignum, F., Kuiper, R.: Combining dynamic deontic logic and temporal logic for the specification of deadlines. In Sprague, J.R., ed.: Proceedings of thirtieth HICSS, Hawaii (1997)
19. Esteva, M., Rodriguez, J.A., Sierra, C., Garcia, P., Arcos, J.L.: On the formal specification of electronic institutions. In Dignum, F., Sierra, C., eds.: Agent-mediated Electronic Commerce (The European AgentLink Perspective). Volume volume 1191 of LNAI. Springer, Berlin (2001) 126–147
20. Jones, A.J.I., Sergot, M.: On the characterisation of law and computer systems: The normative systems perspective. In Meyer, J.J., Wieringa, R., eds.: Deontic Logic in Computer Science: Normative System Specification. Wiley (1993)
21. Meyer, J.J., Wieringa, R., eds.: Deontic logic in computer science: normative system specification. Wiley (1993)
22. Norman, T.J., Sierra, C., Jennings, N.R.: Rights and commitment in multi-agent agreements. In: Proceedings of the Third International Conference on Multi-Agent Systems. (1998) 222–229

23. van der Torre, L.: Contextual deontic logic: Normative agents, violations and independence. Ann. Math. Artif. Intell. **37** (2003) 33–63
24. Fornara, N., Vigano, F., Colombetti, M.: Agent communication and institutional reality. In: AAMAS 2004 Workshop on Agent Communication (AC2004), New York (2004)
25. Chaib-Draa, B., Dignum, F.: Trends in agent comunication language. Computational Intelligence **18** (2002) 89–101
26. Wooldridge, M.: Semantic issues in the verification of agent communication languages. Journal of Autonomous Agents and Multi-Agent Systems **3** (2000) 9–31
27. Mayfield, J., Labrou, Y., Finin, T.: Evaluation of KQML as an Agent Communication Language. In Wooldridge, M., Muller, J.P., Tambe, M., eds.: Lecture notes in Artificial Intelligence. Berlin: Springer-Verlag (1996) 1–12
28. Fagin, R., Halpern, J.Y., Moses, Y., Vardi, M.Y.: Reasoning about Knowledge. The MIT Press, Cambridge, MA (1995)
29. van der Hoek, W., Wooldridge, M.: Towards a logic of rational agency. Logic Journal of the IGPL **11** (2003) 135–159
30. Lomuscio, A., Sergot, M.: Deontic interpreted systems. Studia Logica **75** (2003)
31. Wooldridge, M.: Computationally grounded theories of agency. In Durfee, E., ed.: Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS 2000), IEEE Press (2000)
32. Kripke, S.: Semantical considerations on modal logic. Acta Philosophical Fennica **XVI** (1963) 83–94
33. Cohen, P., Levesque, H.: Communicative actions for artificial agents. In Bradshaw, J.M., ed.: Software Agents. AAAI Press / The MIT Press, Cambridge (MA) (1997)
34. Rao, A.S., Georgeff, M.P.: Decision procedures for bdi logics. Journal of Logic and Computation **8** (1998) 293–342
35. Halpern, J., Vardi, M.: The complexity of reasoning about knowledge and time.i. lower bounds. Journal of Computer and System Sciences **38** (1989) 195–237
36. van der Meyde, R., Wong, K.: Complete axiomatizations for reasoning about knowledge and branching time. Studia Logica **75** (2003) 93–123
37. Halpern, J., van der Meyden, R., Vardi, M.: Complete axiomatizations for reasoning about knowledge and time. SIAM Journal of Computing **33** (2004) 449–478
38. Halpern, J.Y., Vardi, M.Y.: The complexity of reasoning about knowledge and time: synchronous systems. Technical Report RJ 6097, IBM (1988)
39. Lomuscio, A., Wozna, B.: A complete and decidable axiomatization for deontic interpreted systems. In: 8th International Workshop on application of Deontic Logic to Computer Science (DEON-06). LNAI, Heidelberg, Springer-Verlag (2006)
40. Manna, Z., Pnueli, A.: The Temporal Logic of Reactive and Concurrent Systems. Springer-Verlag, Berlin, Germany (1995)
41. Agerri, R.: Motivational Attitudes and Norms in a unified Agent Communication Language for open Multi-Agent Systems: A Pragmatic Approach. PhD thesis, Dept. of Computing, City University, London, UK (2006)
42. Rao, A., Georgeff, M.: Modeling rational agents within a bdi-architecture. In Allen, J., Fikes, R., Sandewall, E., eds.: 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91), Morgan Kaufmann Publishers (1991) 473–484
43. Austin, J.L.: How to do Things with Words. Oxford University Press, Oxford (1962)
44. Alonso, E.: Rights and argumentation in open multi-agent systems. Artificial Intelligence Review **21** (2004) 3–24

45. van der Torre, L., Hulstijn, J., Dastani, M., Broersen, J.: Specifying multiagent organizations. In: Proceedings of the Seventh Workshop on Deontic Logic in Computer Science (Deon'2004), LNAI 3065. Springer (2004) 243–257
46. von Wright, G.H.: Deontic logic. Mind **60** (1951) 1–15
47. Boella, G., van der Torre, L.: Role-based rights in artificial social systems. In: Proceedings of IAT05. IEEE, 2005. (2005)
48. Castelfranchi, C.: Practical permission: Dependence, power and social commitment. In: Proceedings of 2nd workshop on Practical Reasoning and Rationality, London (1997)
49. Wenar, L.: Rights. In Zalta, E.N., ed.: The Stanford Encyclopedia of Philosophy. http://plato.stanford.edu/archives/win2005/entries/rights/ (Winter 2005)
50. Anderson, A.R.: Some nasty problems in the formalization of ethics. Nous **1** (1967) 345–360
51. Endriss, U.: Temporal logic for representing agent communication protocols. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS-2005), ACM Pres (2005)
52. Endriss, U., Maudet, N., Sadri, F., Toni, F.: Logic-based communication protocols. In: Advances in Agent Communication, LNAI, volume 2922, pages 91-107, Springer-Verlag. (2004)
53. Ferber, J., Gutknecht, O.: A meta-model for the analysis of organizations in multiagent systems. In: Proceedings of the Third International Conference on Multi-Agent Systems (ICMAS'98). (1998) 128–135
54. Boella, G., Damiano, R., Hulstijn, J., van der Torre, L.: Acl semantics between social commitments and mental attitudes. In: In proceedings of the AC 2006. (2006)
55. Raimondi, F., Lomuscio, A.: Automatic verification of deontic properties of multiagent systems. In: Deontic Logic. Volume LNCS 3065., Springer Verlag (2004) 228–242
56. Baldoni, M., Baroglio, C., Martelli, A., Patti, V.: Verification of protocol conformance and agent interoperability. In Toni, F., Torroni, P., eds.: Sixth International Workshop on Computational Logic in Multi-Agent Systems. Volume 3900 of LNCS., Heidelberg, Springer (2006) 265–283
57. Dastani, M., Riemsdijk, M., Dignum, F., Meyer, J.J.: A programming language for cognitive agents: Goal-directed 3apl. In Dastani, M., Dix, J., Fallah-Seghrouchni, A.E., eds.: Programming Multi-Agent Systems (LNAI 3037). Springer-Verlag, Berlin (2004) 111–130

Normtypologies

Harko Verhagen[1]

Department of Computer and Systems Sciences, Stockholm University / KTH
SE-16440, Kista, Forum 100, Sweden
verhagen@dsv.su.se

Abstract. In this extended abstract I describe some norm typolgies developed within sociololgy and social philosophy. Using these typologies we can determine the bounderies of the different approaches to normative agent systems.

Keywords. Norms, Multiagent systems, Normative multiagent systems, Typologies

1 Extended abstract

The concept of a norm is problematic. Not only due to the different views on norms in different research areas, but also since the concept is used in everyday life in ambiguous ways. As in folkpsychology, the use of "'folksociological"' concepts in scientific research creates problems. To deal with both problems I propose to analyse available norm typologies to create a framework with which to evaluate the possiblities and impossiblities to adress different types of norms using various approaches to normative agentsystems.

Morris [1] proposes a definitional difference betweeen norms and the closely related concept of values after whcih he proceeds to present a classification scheme for different types of norms. Following [2] he proposes that values can be held individually and never include sanctions whereas norms are "'generally accepted, sanctioned prescriptions for, or prohibitions against, others' behavior, belief, or feeling, i.e. what others *ought* to do believe, feel - *or else* (original emphasis). Also, values only apply to the person having the values, while norms have subjects (who set the norms) and objects (to whom the norms are applied). Morris concludes by summing up a selection of 17 characteristics in four categories that can be used to typify norms. These are:

1. Distribution of the Norm
   (a) Extent of Knowledge of the Norm
      – By subjects (those who set the norm) - very few – almost everyone
      – By objects (those to whom the norm applies) - very few – almost everyone
   (b) Extent of Acceptance of or Agreement with the Norm
      – By subjects (those who set the norm) - very few – almost everyone

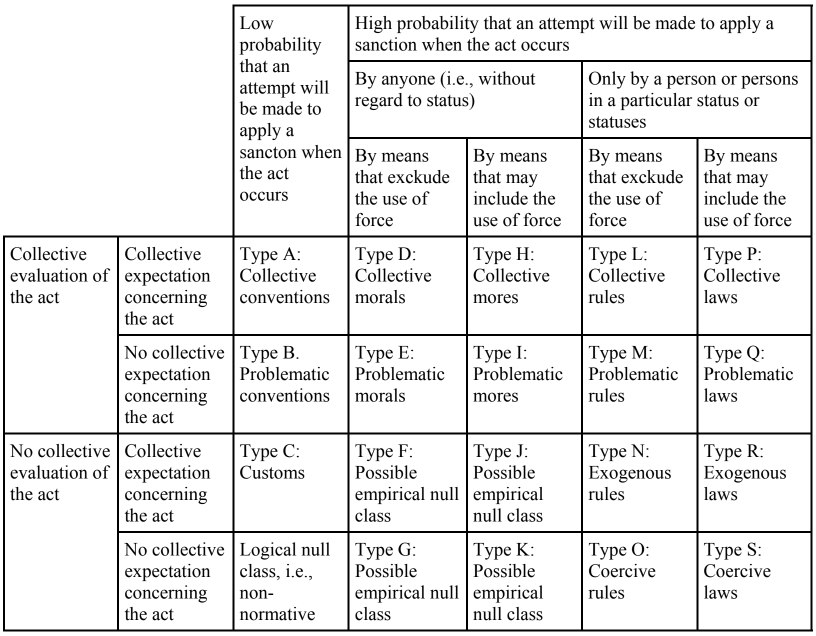
    - By objects (those to whom the norm applies) - very few – almost everyone
  - (c) Extent of Application if the Norm to Objects
    - To groups or categories - very few – almost everyone
    - To conditions - in specified few – in almost all

2. Mode of Enforcement of the Norm
   - Reward - Punishment - more reward than punishment – more punishment than reward
   - Severity of sanction - light, unimportant – heavy, important
   - Enforcing agency - specialized, designated responsibility – general, universal responsibility
   - Extent of enforcement - lax, intermittent – rigorous, uniform
   - Source of authority - rational, expedient, instrumental – divine, inherent, absolute, autonomous
   - Degree of internalization by objects - litlle, external enforcement required – great, self-enforcement sufficient
3. Transmission of the Norm
   - (a) Socialization process - late learning, from secondary relations – early learning, from primary relations
   - (b) Degree of reinforcement by subject - very little – high, persistent
4. Conformity to the Norm
   - (a) Amount of conformity attempted by objects - attempted by very few – attempted by almost everyone
   - (b) Amount of deviance by objects - very great – very little
   - (c) Kind of deviance - formation of subnorms – patterned evasion – idiosyncratic deviation

Two general types of norms that can be inferred from this classification scheme are what [1] calls an absolute norm and a conditional norm. In the first case all right hand side characteristics apply while for conditional norms all left hand extremes apply.

In the 1960's Gibbs [3] followed up on Morris's work by distinguishing definitional and contingent attributes in Morris's list of characteristics. The end product is a typology of norms encompassing conventions, morals, mores, rules and laws as depicted in figure 1.

Tuomela [4] on his turn distinguishes two kinds of social norms (meaning community norms), namely, rules (r-norms) and proper social norms (s-norms). Rules are norms created by an authority structure and are always based on agreement making. Proper social norms are based on mutual belief. Rules can be formal, in which case they are connected to formal sanctions, or informal, where the sanctions are also informal. Proper social norms consist of conventions, which apply to a large group such as a whole society or socioeconomic class, and group-specific norms. The sanctions connected to both types of proper social norms are social sanctions and may include punishment by others and expelling from the group. Aside from these norms, Tuomela also described personal norms and potential social norms (these are norms that are normally widely obeyed



		Low probability that an attempt will be made to apply a sancton when the act occurs	High probability that an attempt will be made to apply a sanction when the act occurs			
			By anyone (i.e., without regard to status)		Only by a person or persons in a particular status or statuses	
			By means that exckude the use of force	By means that may include the use of force	By means that exckude the use of force	By means that may include the use of force
Collective evaluation of the act	Collective expectation concerning the act	Type A: Collective conventions	Type D: Collective morals	Type H: Collective mores	Type L: Collective rules	Type P: Collective laws
	No collective expectation concerning the act	Type B. Problematic conventions	Type E: Problematic morals	Type I: Problematic mores	Type M: Problematic rules	Type Q: Problematic laws
No collective evaluation of the act	Collective expectation concerning the act	Type C: Customs	Type F: Possible empirical null class	Type J: Possible empirical null class	Type N: Exogenous rules	Type R: Exogenous laws
	No collective expectation concerning the act	Logical null class, i.e., non-normative	Type G: Possible empirical null class	Type K: Possible empirical null class	Type O: Coercive rules	Type S: Coercive laws

Fig. 1. Norm typology developed by Gibbs [3]

but that are not in their essence based on social responsiveness and that, in principle, could be personal only). These potential social norms contain, among others, moral and prudential norms (m-norms and p-norms, respectively). The reasons for accepting norms differ as to the kind of norms:

- Rules are obeyed because they are agreed upon.
- Proper social norms are obeyed because others expect one to obey.
- Moral norms are obeyed because of ones conscience.
- Prudential norms are obeyed because it is the rational thing to do.

The motivational power of all types of norms depends on the norm being a subjects reason for action. In other words, norms need to be internalized and accepted.

Therborn [5] distinguishes among three kinds of norms. *Constitutive norms* define a system of action and an agent's membership in it, *regulative norms* describe the expected contributions to the social system, and *distributive norms* defining how rewards, costs, and risks are allocated within a social system. Furthermore, he distinguishes between non-institutionalized normative order, made up by personal and moral norms in day-to-day social traffic, and institutions, an example of a social system defined as a closed system of norms. Institutional normative action is equaled with role plays, i.e., roles find their expressions in expectations, obligations, and rights vis-a-vis the role holder's behaviour.

References

1. Morris, R.T.: A typology of norms. American Sociological Review **21** (1956) 610 – 613
2. Kluckhohn, C.: Values and Value-Orientations in the Theory of Action. In: Towards a General Theory of Action. Cambridge: Harvard University Press (1951) 388 – 433
3. Gibbs, J.P.: Norms: The Problem of Definition and Classification. The American Journal of Sociology **70** (1965) 586 – 594
4. Tuomela, R.: The Importance of Us: A Philosophical Study of Basic Social Norms. Stanford University Press (1995)
5. Therborn, G.: Back to Norms! On the Scope and Dynamics of Norms and Normative Action. Current Sociology **50** (2002) 863 – 880

On the Logic of Constitutive Rules

Davide Grossi, John-Jules Ch. Meyer, Frank Dignum

Institute of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
{davide,jj,dignum}@cs.uu.nl

Abstract. The paper proposes a logical systematization of the notion of counts-as which is grounded on a very simple intuition about what counts-as statements actually mean, i.e., forms of classification. Moving from this analytical thesis the paper disentangles three semantically different readings of statements of the type X counts as Y in context C, from the weaker notion of contextual classification to the stronger notion of constitutive rule. These many ways in which counts-as can be said are then formally addressed by making use of modal logic techniques. The resulting framework allows for a formal characterization of all the involved notions and their reciprocal logical relationships.

Keywords. Constitutive rules, counts-as, modal logic.

1 Introduction

The term “counts-as” derives from the paradigmatic formulation that in [1] and [2] is attributed to the non-regulative component of institutions, i.e., constitutive rules:

> [...] “institutions” are systems of constitutive rules. Every institutional fact is underlain by a (system of) rule(s) of the form “X counts as Y in context C” ([1], pp.51-52).

In legal theory the non-regulative component of normative systems has been labeled in ways that emphasize a classificatory, as opposed to a normative or regulative, character: *conceptual rules* ([3]), *qualification norms* ([4]), *definitional norms* ([5]). Constitutive rules are definitional in character:

> The rules for checkmate or touchdown must ‘define’ *checkmate in chess* or *touchdown in American Football* [...] ([1], p.43).

With respect to this feature, a first reading of counts-as is thus readily available: it is plain that counts-as statements express classifications. For example, they express what *is classified* to be a checkmate in chess, or a touchdown in American Football. However, is this all that is involved in the meaning of counts-as statements?

The interpretation of counts-as in merely classificatory terms does not do justice to the notion which is stressed in the label “constitutive rule”, that is, the notion of *constitution*. Aim of the paper is to show that this notion, as it is presented in some work in legal and social theory, is amenable to formal characterization and that the theory we

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/913

developed in [6, 7] provides a ground for its understanding. The paper disentangles and analyzes three precise senses in which it can be said that “X counts as Y in context C”. For each of these different senses of counts-as a formal semantics is developed by making use of standard modal logic techniques. From a methodological point of view, we will proceed as recommended here:

> “[. . .] it seems to me obvious that the only rational approach to such problems would be the following: [1] We should reconcile ourselves with the fact that we are confronted, not with one concept, but with several different concepts which are denoted by one word; [2] we should try to make these concepts as clear as possible (by means of definition, or of an axiomatic procedure, or in some other way); [3] to avoid further confusions, we should agree to use different terms for different concepts; and then we may proceed to a quiet and systematic study of all concepts involved, which will exhibit their main properties and mutual relations” ([8], p. 355).

The structure of the paper reflects its method. Section 2 disentangles three different meanings of counts-as statements and exposes a first informal analysis. In Section 3 a modal logic of contextual classification is introduced and by means of it a formal analysis of the classificatory view of counts-as is provided. The two remaining senses of counts-as are formally analyzed in Sections 5 and 6. Finally, the relationships between the three readings are studied in Section 7. Conclusions follow in Section 8.

2 Counts-as between Classification and Constitution

Consider the following reasoning pattern.

Example 1. It is a rule of normative system Γ that conveyances transporting people or goods count as vehicles; it is always the case that bikes count as conveyances transporting people or goods but not that bikes count as vehicles; therefore, in the context of normative system Γ, bikes count as vehicles.

This is an instance of a typical reasoning pattern involving constitutive rules. The counts-as locution occurs three times. However, the second premise states a generally acknowledged classification (“bikes count as conveyances transporting people or goods”), while the conclusion states classification which is considered to hold only with respect to the normative system at issue (“according to normative system Γ, bikes count as vehicles”). The first premise expresses something yet different, a classification which is brought about —constituted— by the normative system: “conveyances transporting people or goods are classified as vehicles” is one of the rules of Γ.

2.1 The classificatory reading of counts-as

The fact that “bikes count as conveyances transporting people or goods” can be readily analyzed as a form of classification: the concept ‘bike’ is a subconcept of the concept ‘conveyance transporting people or goods’. ([6, 9, 10]).

In Example 1 one of the premises was that bikes do not always count as vehicles. In other words, there are contexts in which 'bike' is not a subconcept of 'vehicle'. This suggests that a notion of context is necessary because classifications holding for a normative system are not of a universal kind, they do not hold in general. The classificatory reading of counts-as statements of the form "X counts-as Y in context c" runs thus as follows: "X is a subconcept of Y in context c". Following much literature on context theory (see for instance [11, 12]) we conceive of a context simply as set of situations (possible worlds). What precisely these situations have to be in order to be considered a context will be clarified soon discussing the notion of constitutive rule (Section 2.3).

Classificatory counts-as will be formally studied in Section 3. A more extensive discussion of the intuitions underpinning the classificatory reading of counts-as statements can be found in [6, 7].

2.2 Counts-as statements as proper classifications

The analytic literature on constitutive norms often comes to emphasize the following characteristic feature: counts-as statements are not just classifications but "new" classifications, that is, classifications which would not hold without the normative system stating them:

> "Where the rule is purely regulative, behaviour which is in accordance with the rule could be given the same description or specification (the same answer to the question "What did he do?") whether or not the rule existed, provided the description or specification makes no explicit reference to the rule. But where the rule (or system of rules) is constitutive, behaviour which is in accordance with the rule can receive specifications or descriptions which it could not receive if the rule did not exist" ([1], p.35).

This was the case for the conclusion of the inference in Example 1: "in the context of normative system Γ, bikes count as vehicles" although this is not generally the case. In this view, counts-as statements do not only state contextual classifications, but they state new classifications which would not otherwise hold.

Observation 1 *Counts-as statements are classifications which hold with respect to a context (set of situations) but which do not hold in general (i.e., with respect to all situations).*

We call counts-as statements intended in the sense of Observation 1 *proper contextual classifications*. In other words, X counts as Y in context c because X is classified as Y in c but also because this does not hold in general, i.e., in the global context. They state that something new is brought about and in this sense the notion of proper contextual classification already captures a precise notion of constitution: the fact that X is classified as Y is constituted by context c in the sense that out of context c it might not hold. Proper contextual classifications will be formally studied in Sections 4.1 5. A more detailed exposition of the intuitions behind the proper classificatory view on counts-as can be found in [7].

2.3 Counts-as statements as constitutive rules

Example 1 sketched an inference grounded on a constitutive rule: "It is a rule of normative system Γ that conveyances transporting people or goods count as vehicles". First of all, this statement expresses a classification which is brought about by the normative system Γ ("conveyances transporting people or goods count as vehicles"), that is, what we called in the previous section a proper contextual classification. There is however something more. It explicitly states that a classification is one of the rules of Γ. This semantic ingredient is not captured by the classificatory and proper classificatory readings sketched in the previous sections and it involves two essential aspects.

The first one is that counts-as statements of the constitutive type are always part of a *set* of similar statements, the system of rules Γ.

> "Rules are constitutive if and only if they are part of a set of rules. Strictly speaking, there is no such thing as *a* rule that is constitutive in isolation" ([13], p.5).

The second aspect concerns the relation between, on the one hand, the notion of a set of rules Γ, i.e., normative system or institution, and on the other hand the notion of set of situations c, or context c. A Γ constitutes a context c by means of its rules. The set of classifications stated as constitutive rules by a normative system (for instance, "conveyances transporting people or goods count as vehicles") can be thought of as the set of situations which make that set of classifications true. Hence, the set of constitutive rules of any normative system can be seen as a set of situations. And a set of situations —we have seen— is what is called a context in much literature on context theory (see for instance [11, 12]). To put it in a nutshell, a context is a set of situations, and if the constitutive rules of a given normative system Γ are satisfied by all and only the situations in a given set, then that set of situations is *the context defined by* Γ. This simple observation allows us to think of contexts as "systems of constitutive rules" ([1], p.51). Notice that this is no exotic thought. In fact, this idea has been neatly advanced —informally— in some literature on the theory of institutions:

> "A set of constitutive rules defines a logical space" ([13], p.6).

A logical space is nothing but a set of states, i.e., a context. Getting back to Example 1, consider the statement concluding the argument: "according to Γ, bikes count as vehicles". In this light such a statement just says that "in the set of situations defined by the rules of system Γ, bike is a subconcept of vehicle".

The discussion above is distilled in the following observation.

Observation 2 *A constitutive counts-as statement is a proper contextual classification such that: (a) it is an element of the set of rules specifying a given normative system Γ; (b) the set of rules of Γ define the context (set of situations) to which the counts-as statement pertains.*

Constitutive counts-as statements will be formally studied in Sections 4.2 and 6.

To recapitulate, we distinguished between *constitutive counts-as statements*, *proper classificatory counts-as statements* and *classificatory counts-as statements*. When statements "X counts as Y in the context c of normative system Γ" are read as constitutive rules, what is meant is that the classification of X under Y is considered to be an explicit promulgation of the normative system Γ defining context c. Instead, when they are read as proper classificatory statements they are meant to denote classifications that are constituted, or brought about, by the context at issue in the sense that they might not hold if another context is considered. Finally, when they are read as mere contextual classification, they are meant to denote classificatory statements that are just the case in the given context .

3 Modal logic of Classificatory Counts-as

This section summarizes the results presented in [6]. We first introduce the languages we are going to work with: propositional n-modal languages $\mathcal{ML}_n$ ([14]). The alphabet of $\mathcal{ML}_n$ contains: a countable set $\mathbb{P}$ of propositional atoms p; the set of boolean connectives $\{\neg, \wedge, \vee, \rightarrow\}$; a finite non-empty set of n (context) indexes C, and the operator $[\ ]$. Metavariables $i, j, ...$ are used for denoting elements of C. The set of well formed formulas ϕ of $\mathcal{ML}_n$ is then defined by the following BNF:

$$\phi ::= \bot \mid p \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \phi_1 \rightarrow \phi_2 \mid [i]\phi.$$

We will refer to formulae ϕ in which at least one modal operator occurs as modalized formulae. We call instead objective formulae in which no modal operator occur and we denote them using the metavariables $\gamma_1, \gamma_2, \ldots$.

3.1 Semantics

Semantics for these languages is given via structures $\mathcal{M} = \langle \mathcal{F}, \mathcal{I} \rangle$, where:

- $\mathcal{F}$ is a CXT frame, i.e., a structure $\mathcal{F} = \langle W, \{W_i\}_{i \in C} \rangle$, where W is a finite set of states (possible worlds) and $\{W_i\}_{i \in C}$ is a family of subsets of W.
- $\mathcal{I}$ is an evaluation function $\mathcal{I} : \mathbb{P} \longrightarrow \mathcal{P}(W)$ associating to each atom the set of states which make it true.

Such frames model thus n different contexts i which might be inconsistent, if the corresponding set W_i is empty, or global if W_i coincides with W itself. This implements in a straightforward way the thesis developed in context modeling according to which contexts can be soundly represented as sets of possible worlds ([11]).

The satisfaction relation, then, results in the following.

Definition 1. (Satisfaction based on CXT frames)
Let $\mathcal{M}$ be a model built on a CXT *frame.*

$$\mathcal{M}, w \models [i]\phi \ \textit{ iff } \ \forall\, w' \in W_i : \mathcal{M}, w' \vDash \phi$$
$$\mathcal{M}, w \models \langle i \rangle\, \phi \ \textit{ iff } \ \exists w' \in W_i : \mathcal{M}, w' \vDash \phi.$$

The obvious boolean clauses are omitted. Validity in a model, in a frame and in a class of frames are defined as usual.

It is instructive to make a remark about the $[i]$-operator clause, which can be seen as the characterizing feature of the modeling of contexts as sets of worlds[1]. It states that the truth of a modalized formula abstracts from the point of evaluation of the formula. In other words, the notion of "truth in a context i" is a *global* notion: $[i]$-formulae are either true in every state in the model or in none. This reflects the idea that what is true or false in a context does not depend on the world of evaluation, and this is what we would intuitively expect especially for contexts interpreted as normative systems: what holds in the context of a given normative system is not determined by the point of evaluation but just by the system in itself, i.e., by its rules: the fact that in Γ bikes count as vehicles depends only on the rules of Γ.

3.2 Axiomatics

The multi-modal logic that corresponds, i.e., that is sound and complete with respect to the class of CXT frames, is a system we call here $\mathbf{K45^{ij}_n}$. It consists of a logic weaker than the logic $\mathbf{KD45^{ij}_n}$ investigated in [6] in that the semantic constraint has been dropped which required the sets in family $\{W_i\}_{i \in C}$ to be non-empty. As a consequence the D axiom is eliminated. To put it in a nutshell, the system is the very same logic for contextual classification developed in [6] except for the fact the we want to allow here the representation of empty contexts as well. In the knowledge representation setting we are working in, where contexts can be identified with the normative systems defining them, this amounts to accept the possibility of normative systems issuing inconsistent constitutive rules.

Logic $\mathbf{K45^{ij}_n}$ is axiomatized via the following axioms and rules schemata:

$$\begin{array}{rl}
(\texttt{P}) & \text{all tautologies of propositional calculus} \\
(\texttt{K}) & [i](\phi_1 \rightarrow \phi_2) \rightarrow ([i]\phi_1 \rightarrow [i]\phi_2) \\
(4^{ij}) & [i]\phi \rightarrow [j][i]\phi \\
(5^{ij}) & \neg[i]\phi \rightarrow [j]\neg[i]\phi \\
(\texttt{Dual}) & \langle i \rangle \, \phi \leftrightarrow \neg[i]\neg\phi \\
 & \\
(\texttt{MP}) & \phi_1, \ \phi_1 \rightarrow \phi_2 \ / \ \phi_2 \\
(\texttt{N}) & \phi \ / \ [i]\phi
\end{array}$$

where i, j denote elements of the set of indexes C. The system is a multi-modal homogeneous $\mathbf{K45}$ with the two interaction axioms 4^{ij} and 5^{ij}. Soundness and completeness are proven in Section 9.

A remark is in order especially with respect to axiomata 4^{ij} and 5^{ij}. In fact, what the two schemata do, consists in making the nesting of the operators reducible which, leaving technicalities aside, means that truth and falsehood in contexts ($[i]\phi$ and $\neg[i]\phi$) are somehow absolute because they remain invariant even if evaluated from another context ($[j][i]\phi$ and $[j]\neg[i]\phi$). In other words, they express the fact that whether something holds in a context i is not something that a context j can influence. This is indeed the kind of property to be expected given the semantics presented in the previous section.

[1] Propositional logics of context without this clause are investigated in [15, 16].

3.3 Classificatory Counts-as formalized

Using a multi-modal logic $\mathbf{K45^{ij}_n}$ on a language $\mathcal{ML}_n$, the formal characterization of the classificatory view on counts-as statements runs as follows.

Definition 2. (Classificatory counts-as: $\Rightarrow^{cl}_c$)
"γ_1 counts as γ_2 in context c" is formalized in a multi-modal language $\mathcal{ML}_n$ as the strict implication between two objective sentences γ_1 and γ_2 in logic $\mathbf{K45^{ij}_n}$:

$$\gamma_1 \Rightarrow^{cl}_c \gamma_2 \ := \ [c](\gamma_1 \rightarrow \gamma_2)$$

These properties for $\Rightarrow^{cl}_c$ follow.

Proposition 1. (Properties of $\Rightarrow^{cl}_c$)
In logic $\mathbf{K45^{ij}_n}$, the following formulas and rules are valid:

$$\gamma_2 \leftrightarrow \gamma_3 \,/\, (\gamma_1 \Rightarrow^{cl}_c \gamma_2) \leftrightarrow (\gamma_1 \Rightarrow^{cl}_c \gamma_3) \tag{1}$$
$$\gamma_1 \leftrightarrow \gamma_3 \,/\, (\gamma_1 \Rightarrow^{cl}_c \gamma_2) \leftrightarrow (\gamma_3 \Rightarrow^{cl}_c \gamma_2) \tag{2}$$
$$((\gamma_1 \Rightarrow^{cl}_c \gamma_2) \wedge (\gamma_1 \Rightarrow^{cl}_c \gamma_3)) \rightarrow (\gamma_1 \Rightarrow^{cl}_c (\gamma_2 \wedge \gamma_3)) \tag{3}$$
$$((\gamma_1 \Rightarrow^{cl}_c \gamma_2) \wedge (\gamma_3 \Rightarrow^{cl}_c \gamma_2)) \rightarrow ((\gamma_1 \vee \gamma_3) \Rightarrow^{cl}_c \gamma_2) \tag{4}$$
$$\gamma \Rightarrow^{cl}_c \gamma \tag{5}$$
$$(\gamma_1 \Rightarrow^{cl}_c \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl}_c \gamma_3) \rightarrow (\gamma_1 \Rightarrow^{cl}_c \gamma_3) \tag{6}$$
$$(\gamma_1 \Rightarrow^{cl}_c \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl}_c \gamma_1) \rightarrow [c](\gamma_1 \leftrightarrow \gamma_2) \tag{7}$$
$$(\gamma_1 \Rightarrow^{cl}_c \gamma_2) \rightarrow (\gamma_1 \wedge \gamma_3 \Rightarrow^{cl}_c \gamma_2) \tag{8}$$
$$(\gamma_1 \Rightarrow^{cl}_c \gamma_2) \rightarrow (\gamma_1 \Rightarrow^{cl}_c \gamma_2 \vee \gamma_3) \tag{9}$$

We omit the proofs, which are straightforward via application of Definition 2. This system validates all the intuitive syntactic constraints isolated in [17] (validities 1-4). In addition, this semantic-oriented approach to classificatory counts-as enables the four validities 6-9. Besides, this analysis shows that counts-as conditionals, once they are viewed as conditionals of a classificatory nature, naturally satisfy reflexivity (5), transitivity (6), and a form of "contextualized" antisymmetry (7), strengthening of the antecedent (8) and weakening of the consequent (9).

4 Beyond Classificatory Counts-as

Aim of this section is to provide formal counterparts to Observations 1 and 2 which can work as intermediate step towards the development of suitable modal logics for the analysis of proper classificatory counts-as (Section 5) and constitutive counts-as (Section 6).

4.1 From classification to proper classification

As usual, model-theoretic considerations can give us crucial hints. Let us define the set $\mathbb{T}(X)$ of all formulae which, given a model, are satisfied by all worlds in a set of worlds X:

$$\mathbb{T}(X) = \{\phi \mid \forall w \in X : \mathcal{M}, w \models \phi\}.$$

and let $\mathbb{T}^{\rightarrow}(X)$ be the set of all implications between objective formulae γ_1 and γ_2 which are satisfied by all worlds in a set of worlds X:

$$\mathbb{T}^{\rightarrow}(X) = \{\gamma_1 \rightarrow \gamma_2 \mid \forall w \in X : \mathcal{M}, w \models \gamma_1 \rightarrow \gamma_2\}.$$

Obviously, for every X: $\mathbb{T}^{\rightarrow}(X) \subseteq \mathbb{T}(X)$. In the classificatory reading, given a model $\mathcal{M}$ where the set of worlds $W_c \subseteq W$ models context c, the set of all classificatory counts-as statements holding in c, which we denote as $\mathbb{CL}(W_c)$, can be defined as the set $\mathbb{T}^{\rightarrow}(W_c)$:

$$\mathbb{CL}(W_c) := \mathbb{T}^{\rightarrow}(W_c).$$

Hence, it is easy to see that: $\mathbb{T}^{\rightarrow}(W) \subseteq \mathbb{CL}(W_c) \subseteq \mathbb{T}(W_c)$. In other words, the set of classificatory counts-as statements is:

- A subset of all the truths of W_c;
- A superset of all conditional truths of W, that is, of the “global” or “universal” context of model $\mathcal{M}$.

While the first point represents a quite banal semantic constraint to which any formal characterization of counts-as should adhere, the second one is much more questionable. Indeed, what is true anyway is not characteristic of any context (except of the global one), and it cannot be properly said to represent any new truth. In other words, interpreting counts-as statements as mere classifications, as it has been done in Section 3 make them inherit all trivial classifications which hold globally in the model. This is the reason why classificatory counts-as, as shown in Proposition 1, behaves classically enjoying antecedent strengthening as well as transitivity and reflexivity.

These considerations suggest thus a readily available strategy to specify the set of proper classificatory counts-as holding in a context c on the basis of $\mathbb{T}^{\rightarrow}(W_c)$. The problem boils down to eliminate from the set of classificatory counts-as $\mathbb{CL}$ for a context W_c those classifications which hold globally, that is, which hold with respect to the global context W. We obtain, in this way, the set of *proper classificatory counts-as* statements, or *proper contextual classifications*, holding in context c in a CXT model $\mathcal{M}$.

Definition 3. (Set of proper classificatory counts-as in c)
The set $\mathbb{CL}^{+}(W_c)$ of proper classificatory counts-as statements of a context c in a CXT *model $\mathcal{M}$ is defined as follows:*

$$\mathbb{CL}^{+}(W_c) := \mathbb{T}^{\rightarrow}(W_c) \setminus \mathbb{T}(W). \tag{10}$$

Intuitively, the set of proper classificatory count-as holding in c corresponds to the set of implications between objective formulae which hold in c, minus those implications which hold universally. Or, to put it otherwise, the set of proper classificatory count-as holding in c corresponds to the set of classificatory counts as of c, minus those implications which hold universally: $\mathbb{CL}^{+}(W_c) := \mathbb{CL}(W_c) \setminus \mathbb{T}(W)$. This is the most natural amendment of the classificatory view toward the specification of a stronger notion of contextual classification along the lines of Observation 1.

4.2 From proper classification to constitution

Let us now focus on Observation 2. What comes to play a role is the notion of a *definition* of the context of a counts-as statement. A definition of a context c, in a CXT model $\mathcal{M}$, is a set of objective formulae[2] Γ such that $\forall w \in W$:

$$\mathcal{M}, w \models \Gamma \text{ iff } w \in W_c \tag{11}$$

that is, the set of formulae Γ such that all and only the worlds in W_c satisfy Γ in $\mathcal{M}$.

Observation 2 can now get a formal formulation. Given the set of formulae Γ, we say that any formula $\gamma_1 \rightarrow \gamma_2 \in \Gamma$ is a constitutive counts-as statement w.r.t. context c iff Γ defines context c and $\gamma_1 \rightarrow \gamma_2$ belongs to the set of proper contextual classifications of c.

Definition 4. (Set of constitutive counts-as in c w.r.t. definition Γ)
The set $\mathbb{CO}(\Gamma, W_c)$ of constitutive counts-as statements of a context c defined by Γ in a CXT *model $\mathcal{M}$ is:*

$$\begin{aligned}\mathbb{CO}(\Gamma, W_c) := \{\gamma_1 \rightarrow \gamma_2 \in \Gamma \mid \gamma_1 \rightarrow \gamma_2 \in \mathbb{CL}^+(W_c)\\ \textit{and } \forall w(\mathcal{M}, w \models \Gamma \textit{ iff } w \in W_c)\}\end{aligned} \tag{12}$$

Notice that $\mathbb{CO}(\Gamma, W_c)$ is defined taking as domain the set of implicative statements of Γ. Notice also that as a result of this definition if Γ does not define context W_c then $\mathbb{CO}(\Gamma, W_c) = \emptyset$. In fact, Formula 12 can be restated as follows:

$$\mathbb{CO}(\Gamma, W_c) = \begin{cases} \mathbb{CL}^+(W_c) \cap \Gamma, & \text{if } \Gamma \text{ defines } W_c \\ \emptyset, & \text{otherwise.} \end{cases}$$

Section 6 is devoted to the development of a modal logic based on this definition. The definitions discussed are summarized in the table below.

Cxt Classification	$\mathbb{CL}(W_c) \;=\; \mathbb{T}^{\rightarrow}(W_c)$
Proper Cxt Classification	$\mathbb{CL}^+(W_c) \;=\; \mathbb{CL}(W_c) \setminus \mathbb{T}(W)$
Constitution	$\mathbb{CO}(\Gamma, W_c) \;=\; \begin{cases} \mathbb{CL}^+(W_c) \cap \Gamma, & \text{if } \Gamma \text{ defines } W_c \\ \emptyset, & \text{otherwise.} \end{cases}$

The table pinpoints the dependencies between the formal characterizations of the three different senses of counts-as which has been taken into consideration: the notion of constitution builds on the notion of proper contextual classification which in its turn builds on the notion of contextual classification. The modal logic analysis of contextual classification developed in Section 3 can thus be used as a sound starting point for the modal logic analysis of the two notions introduced in this section.

[2] This is no arbitrary choice since it can be easily seen that contextual formulae, since they denote global properties of the models, are as a matter of fact irrelevant for the definition of sets of worlds W_i such that $\emptyset \subset W_i \subset W$, that is, those sets which denote neither the empty nor the universal contexts. It is therefore natural to restrict definitions to objective formulae.

4.3 A methodological note

Before rendering the insights of Sections 4.1 and 4.2 in modal logic, it is worth making a methodological remark. We are here concerned with a term, “counts-as”, which appears to have different meanings. At this point we had two main ways to pursue the formal characterization of counts-as we were aiming at. We could proceed axiomatically by trying to single out intuitive syntactic properties of counts-as statements? Or rather semantically, by trying to enrich the semantic characterization of classificatory counts-as exposed in the previous sections in order to capture further semantic nuances? While formal approaches to counts-as ([17–19]) have been, up to now, characterized by an axiomatic perspective, we have instead chosen for a semantics-driven solution. This choice has been inspired by considering the methodological standpoint of fundamental work in philosophical logic such as [8, 20].

The same issue we are facing here in analyzing counts-as lies also at the ground of the Tarskian characterization of the notion of truth and consists in the polysemy of the to-be-analyzed term. Because of the inherent polysemy of the predicate “to be true”, Tarski found it unconvincing to proceed introducing the predicate as a primitive and then axiomatizing it:

> “[. . .] the choice of axioms always has rather accidental character, depending on inessential factors (such as e.g. the actual state of our knowledge). [. . .] a method of constructing a theory does not seem to be very natural [. . .] if in this method the role of primitive concepts —thus of concepts whose meaning should appear evident— is played by concepts which have led to various misunderstanding in the past” ([20], pag. 405-406).

Instead, he preferred to first isolate a precise sense of the predicate, i.e., truth as correspondence to reality, and then to define it in terms of a better understood notion, i.e., the notion of satisfaction of a formula by a model. An axiomatic analysis of counts-as statements runs the danger alluded to in the quote: since it is not clear what counts-as statements actually mean, an axiomatization of them could result in mixing under the the same logical representation different semantic flavors that, from an analytical point of view, should be kept separated. A systematic discussion of this issue, specifically in relation with the proposal advanced in [17], can be found in [7].

The work presented in this paper is the result of the application of this method to the notion of counts-as: in Section 2 we first disentangled different meanings of the term “counts-as” providing a first map of its polysemy; in Section 3 we formally analyzed the first and more basic of these meanings explaining it in terms of a better-understood notion (strict implication within a context); in this section we have pointed at a first semantic characterization of the other two meanings and in the coming next two sections we will explain them by making use of better-understood modal logic notions: the negation of global statements (proper classificatory counts-as) and the definition of a context (constitutive counts-as).

5 Modal Logic of Counts-as as Proper Contextual Classification

In the following section a modal logic is developed which implements the definition stated in Formula 10 above. By doing this we will capture the intuitions discussed in

Section 2 concerning the intuitive reading of counts-as statements in proper classificatory terms. At the same time we will maintain the possible worlds semantics of context exposed in Section 3 and developed in order to account for the purely classificatory view of counts-as.

5.1 Expansion of $\mathcal{L}_n$ and semantics

Language $\mathcal{L}_n$ is expanded as follows. The set of context indexes C is such that it always contains the special context index u denoting the universal (or global) context. We call this language $\mathcal{L}_n^u$.

Languages $\mathcal{L}_n^u$ are given a semantics via a special class of CXT frames, namely the class of CXT frames $\mathcal{F} = \langle W, \{W_i\}_{i \in C} \rangle$ such that $W \in \{W_i\}_{i \in C}$. That is, the frames in this class, which we call CXT$^\top$, always contain the global context among their contexts. The definition of the satisfaction relation for language $\mathcal{L}_n^u$ follows.

Definition 5. (Satisfaction based on CXT$^\top$ frames)
Let $\mathcal{M}$ be a model built on a CXT$^\top$ *frame.*

$$\mathcal{M}, w \models [u]\phi \text{ iff } \forall\, w' \in W : \mathcal{M}, w' \models \phi$$
$$\mathcal{M}, w \models [c]\phi \text{ iff } \forall\, w' \in W_c : \mathcal{M}, w' \models \phi$$

where u is the universal context index and c ranges on the context indexes in C. The obvious boolean clauses and the clauses for the dual modal operators are omitted.

The new clause states that the $[u]$ operator is interpreted on the universal 1-frame contained in each CXT$^\top$ frame. It is therefore nothing but a **S5** necessity operator.

5.2 Axiomatics

We call $\mathbf{Cxt^u}$ the logic characterizing the class of CXT$^\top$ frames. Logic $\mathbf{Cxt^u}$ results from the union $\mathbf{K45^{ij}_n} \cup \mathbf{S5_u} \cup \{\subseteq .ui)\}$, that is, from the union of $\mathbf{K45^{ij}_n}$ with the $\mathbf{S5_u}$ logic for the $[u]$ operator together with the interaction axiom $\subseteq .ui$ below. The axiomatics runs thus as follows:

$$
\begin{array}{rl}
(\mathtt{P}) & \text{all tautologies of propositional calculus} \\
(\mathtt{K}^i) & [i](\phi_1 \rightarrow \phi_2) \rightarrow ([i]\phi_1 \rightarrow [i]\phi_2) \\
(4^{ij}) & [i]\phi \rightarrow [j][i]\phi \\
(5^{ij}) & \neg[i]\phi \rightarrow [j]\neg[i]\phi \\
(\mathtt{T}^u) & [u]\phi \rightarrow \phi \\
(\subseteq .ui) & [u]\phi \rightarrow [i]\phi \\
(\mathtt{Dual}) & \langle i \rangle\, \phi \leftrightarrow \neg[i]\neg\phi
\end{array}
$$

$$
\begin{array}{rl}
(\mathtt{MP}) & \text{IF } \vdash \phi_1 \text{ AND } \vdash \phi_1 \rightarrow \phi_2 \text{ THEN } \vdash \phi_2 \\
(\mathtt{N}^i) & \text{IF } \vdash \phi \text{ THEN } \vdash [i]\phi
\end{array}
$$

where i, j denote elements of the set of indexes C and u denotes the universal context index in C. The interaction axiom $\subseteq .ui$ states something quite intuitive concerning the interaction of the $[u]$ operator with all other context operators: what holds in the global context, holds in every context. Soundness and completeness of this axiomatization w.r.t. $\text{CXT}^{\top}$ frames are proven in Section 9.

5.3 Proper classificatory counts-as formalized

Using a multi-modal logic $\mathbf{Cxt^u}$ on a language $\mathcal{L}_n^u$, the proper classificatory reading of counts-as statements can be formalized as follows.

Definition 6. (Proper classificatory counts-as: $\Rightarrow_c^{cl+}$)
"γ_1 counts as γ_2 in context c", with γ_1 and γ_2 objective formulae, is formalized in the logic $\mathbf{Cxt^u}$ on a multi-modal language $\mathcal{L}_n^u$ as:

$$\gamma_1 \Rightarrow_c^{cl+} \gamma_2 \ := \ [c](\gamma_1 \rightarrow \gamma_2) \wedge \neg[u](\gamma_1 \rightarrow \gamma_2)$$

Notice that this definition is nothing but the translation in the $\mathcal{L}_n^u$ language of Formula 10.

What properties of counts-as are lost interpreting it as proper contextual classification? And what properties are instead still valid? The following two propositions answer these questions.

Proposition 2. (Properties of $\Rightarrow_c^{cl+}$: invalidities)
The $\Rightarrow_c^{cl+}$ versions of reflexivity, strengthening of the antecedent, weakening of the consequent, transitivity and cautious monotonicity are not valid:

$$\gamma \Rightarrow_c^{cl+} \gamma \tag{13}$$
$$(\gamma_1 \Rightarrow_c^{cl+} \gamma_2) \rightarrow (\gamma_1 \wedge \gamma_3 \Rightarrow_c^{cl+} \gamma_2) \tag{14}$$
$$(\gamma_1 \Rightarrow_c^{cl+} \gamma_2) \rightarrow (\gamma_1 \Rightarrow_c^{cl+} \gamma_2 \vee \gamma_3) \tag{15}$$
$$((\gamma_1 \Rightarrow_c^{cl+} \gamma_2) \wedge (\gamma_2 \Rightarrow_c^{cl+} \gamma_3)) \rightarrow (\gamma_1 \Rightarrow_c^{cl+} \gamma_3) \tag{16}$$
$$((\gamma_1 \Rightarrow_c^{cl+} \gamma_2) \wedge (\gamma_1 \Rightarrow_c^{cl+} \gamma_3)) \rightarrow ((\gamma_1 \wedge \gamma_2) \Rightarrow_c^{cl+} \gamma_3) \tag{17}$$

We do not provide all the proofs, which can be obtained by constructing appropriate countermodels. We show a countermodel for Formula 16: $\forall w \in W, \mathcal{M}, w \models \gamma_1 \rightarrow \gamma_3$; $\forall w \in W_c$, $\mathcal{M}, w \models \gamma_1 \rightarrow \gamma_2$ and $\mathcal{M}, w \models \gamma_2 \rightarrow \gamma_3$; and $\exists w', w''$ s.t. $\mathcal{M}, w' \models \gamma_1 \wedge \neg\gamma_2 \wedge \gamma_3$ and $\mathcal{M}, w'' \models \neg\gamma_1 \wedge \gamma_2 \wedge \neg\gamma_3$.

It might be instructive to provide at this point also an intuitive example for the failure of transitivity. Before 9/11/2001, it was the case that many legal systems did not specify a legal notion of terrorism. In the context of the legal systems that did, the following were therefore proper contextual classifications since they were not holding in general: "the use or threat of action designed to influence the government and advance a political cause counts as terrorism" and "terrorism counts as a criminal activity". However, it could not be inferred from them that "the use or threat of action designed to influence the government and advance a political cause counts as a criminal activity" was a proper contextual classification, because what stated was anyway the case also in those legal systems disregarding a notion of terrorism. Intuitively, transitivity fails just because it is possible to constitute local middle terms, e.g., terrorism, for classifications which hold globally in the model.

Proposition 3. (Properties of $\Rightarrow^{cl+}_{c}$: validities)
In logic $\mathbf{Cxt^u}$ *the* $\Rightarrow^{cl+}_{c}$ *variants of Formulae 1-4 of Proposition 1 are valid:*

$$\gamma_2 \leftrightarrow \gamma_3 \;/\; (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \leftrightarrow (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3) \tag{18}$$
$$\gamma_1 \leftrightarrow \gamma_3 \;/\; (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \leftrightarrow (\gamma_3 \Rightarrow^{cl+}_{c} \gamma_2) \tag{19}$$
$$((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3)) \to (\gamma_1 \Rightarrow^{cl+}_{c} (\gamma_2 \wedge \gamma_3)) \tag{20}$$
$$((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_3 \Rightarrow^{cl+}_{c} \gamma_2)) \to ((\gamma_1 \vee \gamma_3) \Rightarrow^{cl+}_{c} \gamma_2) \tag{21}$$

Contextualized antisymmetry, i.e., Formula 7 of Proposition 1 holds in the following form:

$$(\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl+}_{c} \gamma_1) \to [c](\gamma_1 \leftrightarrow \gamma_2) \wedge \neg[u](\gamma_1 \leftrightarrow \gamma_2) \tag{22}$$

Cumulative transitivity (alias cut) is also valid:

$$((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge ((\gamma_1 \wedge \gamma_2) \Rightarrow^{cl+}_{c} \gamma_3)) \to (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3) \tag{23}$$

Conditional versions of antecedent strengthening, consequent weakening and transitivity are valid:

$$\neg[u](\gamma_1 \wedge \gamma_3 \to \gamma_2) \to ((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \to (\gamma_1 \wedge \gamma_3 \Rightarrow^{cl+}_{c} \gamma_2)) \tag{24}$$
$$\neg[u](\gamma_1 \to \gamma_2 \vee \gamma_3) \to ((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \to (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2 \vee \gamma_3)) \tag{25}$$
$$\neg[u](\gamma_1 \to \gamma_3) \to ((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl+}_{c} \gamma_3)) \to (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3) \tag{26}$$

We provide the deduction of Formula 24 as an example.

$$\begin{array}{rcl}
1. & (\mathtt{P}) & (\gamma_1 \to \gamma_2) \to (\gamma_1 \wedge \gamma_3 \to \gamma_2) \\
2. & (\mathtt{N}), (\mathtt{K}), (\mathtt{MP}), 1 & [c](\gamma_1 \to \gamma_2) \to [c](\gamma_1 \wedge \gamma_3 \to \gamma_2) \\
3. & (\mathtt{P}) & \neg[u](\gamma_1 \wedge \gamma_2 \to \gamma_3) \\
 & & \to (\neg[u](\gamma_1 \to \gamma_3) \to \neg[u](\gamma_1 \wedge \gamma_2 \to \gamma_3)) \\
4. & (\mathtt{P}), (\mathtt{MP}), (\text{Def. } 6), 2, 3 & \neg[u](\gamma_1 \wedge \gamma_3 \to \gamma_2) \\
 & & \to ((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \to (\gamma_1 \wedge \gamma_3 \Rightarrow^{cl+}_{c} \gamma_2))
\end{array}$$

Propositions 2 and 3, though very simple, are of key importance for putting our characterization of counts-as as proper contextual classification in perspective with other proposals. Such a comparison is elaborated in detail in [7].

Formulae 24-26 are also of interest since they show that some quite standard properties of contextual classifications are inherited by proper contextual classification in a conditionalized form, the condition being an assertion of invalidity ($\neg[u]$). Proper classificatory counts-as statements are still monotonic, provided that the strengthened version of the antecedent does not universally imply the consequent. Similarly they are still transitive, provided that the implication between γ_1 and γ_3 is not a validity of the model. It is worth emphasizing the importance of these results from the perspective of conceptual analysis and their clarifying power. An alleged intuitive example of transitivity for counts-as statements, in a proper classificatory sense, is such only if the appropriate condition is assumed to hold. Consider again the example about terrorism discussed above. The example could be in fact legitimately be read as an instance of transitivity once it is also accepted that "the use or threat of action designed to influence the government and advance a political cause counts as a criminal activity" is not something which is already globally the case. Similar considerations hold in particular for the conditionalized version of antecedent strengthening. This property will be further discussed in Section 7.1.

6 Modal Logic of Constitutive Counts-as

In this section a modal logic is developed which implements Definition 4. Again, the possible world semantics developed in order to account for the classificatory view of counts-as lies at the ground of the proposed framework.

6.1 Expanding $\mathcal{L}_n^u$

Language $\mathcal{L}_n^u$, which has been used in the previous section to deal with proper contextual classification, needs now further expansion to enable the necessary expressivity. The language is expanded along two lines.

First, the set of context indexes C contains now a set K of m atomic indexes c among which the universal context index u, and the set of the negations $-c$ of the atomic contexts, i.e., of the elements of K: $C = K \cup \{-c \mid c \in K\}$. The cardinality n of C is therefore equal to $2m$.

Second, the language needs also to contain a set $\mathbb{N}$ of nominals s disjoint from the set $\mathbb{P}$ of propositional atoms. Nominals are names for states in the model or, in other words, formulae that can be satisfied by only one state in the model. They can be freely combined with propositions to form well-formed formulae. The BNF is therefore extended as follows:

$$\phi ::= \top \mid p \mid s \mid \neg\phi \mid \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \phi_1 \rightarrow \phi_2 \mid [i]\phi \mid \langle i \rangle \phi.$$

Metavariables for nominals are written as $\nu_1, \nu_2, \ldots$. Modal languages containing nominals have been recently object of thorough study and are known as hybrid languages ([21]). The language obtained is called $\mathcal{L}_n^{u,-}$.

Nominals are needed in order to provide a sound and complete axiomatization of the logic based on the semantics presupposed by Definition 4. To be more precise, they are necessary in order to axiomatize the notion of complement of a context[3]. This will become evident by exposing the axiomatics (Section 6.3) and especially, from a technical point of view, in proving its completeness (Section 9).

[3] For this purpose nominals were first introduced by the so-called "Sofia school" of modal logic ([22,23]) in order to axiomatize the complement and the intersection of accessibility relations, especially in a dynamic logic setting. In fact, the axiomatics we present in Section 6.3 is strictly related with the systems studied in their works.

6.2 Semantics

Languages $\mathcal{L}_n^{u,-}$ are given a semantics via a special class of CXT frames, namely the class of CXT frames $\mathcal{F} = \langle W, \{W_i\}_{i \in C} \rangle$ such that there always exists a $W_u \in \{W_i\}_{i \in C}$ s.t. $W_u = W$; and such that for any atomic index $c \in K$ $W_u \backslash W_c \in \{W_i\}_{i \in C}$. That is, the frames in this class, which we call CXT$^{\top,\backslash}$, always contain the global context among their contexts and the complement of every atomic context.

The semantics for $\mathcal{L}_n^{u,-}$ is thus obtained interpreting the formulae on models built on CXT$^{\top,\backslash}$ frames. However, because of the introduction of nominals, the evaluation function $\mathcal{I}$ should be redefined as a function $\mathcal{I} : \mathbb{P} \cup \mathbb{N} \longrightarrow \mathcal{P}(W)$ satisfying the following constraints:

- For all nominals $s \in \mathbb{N}$, $\mathcal{I}(s)$ is a singleton set, that is, nominals always denote one and only one state in the model.
- For all states $w \in W$, there exists a nominal $s \in \mathbb{N}$ such that $\mathcal{I}(s) = w$, that is, each state has a name. In other words, the restriction of the interpretation function $\mathcal{I}$ on the set of nominals ($\mathbb{N} \rceil \mathcal{I}$) is a surjection on the set of all singletons of W.

The definition of the satisfaction relation for language $\mathcal{L}_n^{u,-}$ runs as follows.

Definition 7. (Satisfaction based on CXT$^{\top,\backslash}$ frames)
Let $\mathcal{M}$ be a model built on a CXT$^{\top,\backslash}$ *frame.*

$$\begin{aligned}
\mathcal{M}, w \models s \ &\textit{ iff } \ \mathcal{I}(s) = \{w\} \\
\mathcal{M}, w \models [u]\phi \ &\textit{ iff } \ \forall\, w' \in W_u : \mathcal{M}, w' \models \phi \\
\mathcal{M}, w \models [c]\phi \ &\textit{ iff } \ \forall\, w' \in W_c : \mathcal{M}, w' \models \phi \\
\mathcal{M}, w \models [-c]\phi \ &\textit{ iff } \ \forall\, w' \in W \backslash W_c : \mathcal{M}, w' \models \phi.
\end{aligned}$$

where u is the universal context index and c ranges on the context indexes in C, and s is a nominal. The obvious boolean clauses and the clauses for the dual modal operators are omitted.

The first clause states the satisfaction relation for nominals: a nominal s is true in a state w in model $\mathcal{M}$ iff the evaluation function associates w to s. Nominals are therefore objective formulae which are true in at most one world. The second clause, which was already introduced in Definition 5, states that the $[u]$ operator is interpreted on the universal frame contained in each CXT$^{\top,\backslash}$ frame. The third one is just the standard clause for contextual truth introduced in Definition 1. Finally, the last and new clause states that the $[-c]$ operators range over the complements of the sets W_c on which $[c]$ operators range instead.

Some observations are in order. First of all, let us comment upon the semantics of the $[-c]$-operators. In fact, the $[c]$ operator specifies a lower bound on what holds in context c ('something more may hold in c'), that is, a formula $[c]\phi$ means that ϕ *at least* holds in context c. The $[-c]$ operator, instead, specifies an upper bound on what holds in c ('nothing more holds in c'), and a $[-c]\neg\phi$ formula means therefore that ϕ *at most* holds in c, i.e., $\neg\phi$ *at least* holds in the complement of c. It becomes thus possible in CXT$^{\top,\backslash}$ frames to express context definitions by means of modal $\mathcal{L}_n^{u,-}$ formulae interpreted on $\textsc{Cxt}^{\top,\setminus}$ models. A set of objective formulae Γ defines context c in a $\textsc{Cxt}^{\top,\setminus}$ model $\mathcal{M}$ iff:

$$\mathcal{M} \models [c]\Gamma \wedge [-c]\neg\Gamma \tag{27}$$

where $\neg\Gamma$ has to be intended in the obvious sense of the disjunction of the negations of all formulae in Γ. Formula 27 is an object language modal translation of the property stated in Formula 11.

Proposition 4. *(Equivalence of Formulae 11 and 27)*
Let $\mathcal{M}$ be a $\textsc{Cxt}$ model and $\mathcal{M}'$ be a $\textsc{Cxt}^{\top,\setminus}$ such that: $\mathcal{M}'$ is based on a frame having the same domain of the frame on which $\mathcal{M}$ is based, and containing all its contexts; $\mathcal{M}'$ has the same evaluation function of $\mathcal{M}$. It is the case that, given a set of objective formulae Γ and a context $W_c \in \{W_i\}_{i \in C}$:

$$\mathcal{M}, w \models \Gamma \textit{ iff } w \in W_c \quad \textit{is equivalent to} \quad \mathcal{M}' \models [c]\Gamma \wedge [-c]\neg\Gamma.$$

Proof. The proof is based on the semantics provided in Definition 7. By construction of $\mathcal{M}'$, the clause "if $w \in W_c$ then $\mathcal{M}, w \models \Gamma$" is equivalent to "if $w \in W_c$ then $\mathcal{M}', w \models \Gamma$", and therefore equivalent to $\mathcal{M}' \models [c]\Gamma$. Analogously, the clause "if $w \notin W_c$ then $\mathcal{M}, w \not\models \Gamma$" is equivalent to "if $w \in W \backslash W_c$ then $\mathcal{M}', w \models \neg\Gamma$", and therefore equivalent to $\mathcal{M}' \models [-c]\neg\Gamma$.

It might be instructive to notice that in practice we are making use, in a different setting but with exactly analogous purposes, of a well-known technique developed in the modal logic of knowledge, i.e., the interpretation of modal operators on "inaccessible states" typical, for instance, of the "all that I know" epistemic logics ([24]). In our case, the set of inaccessible states is nothing but the complement of a context.

6.3 Axiomatics

To axiomatize the above semantics an extension of logic $\mathbf{K45^{ij}_n}$ is needed which can characterize nominals as names for modal states and, consequently, context complementation. The extension, which we call logic $\mathbf{Cxt^{u,-}}$, results from the union $\mathbf{K45^{ij}_n} \cup \mathbf{S5_u}$, that is, from the union of $\mathbf{K45^{ij}_n}$ with the $\mathbf{S5_u}$ logic for the $[u]$ operator together with a group of two axioms (`Least` and `Most`) and one rule (`Name`) which axiomatize nominals, and a group of two axioms (`Covering` and `Packing`) which axiomatize context complementation. The axiomatics runs as follows:

(P) all tautologies of propositional calculus
($\mathtt{K}^i$) $[i](\phi_1 \rightarrow \phi_2) \rightarrow ([i]\phi_1 \rightarrow [i]\phi_2)$
(4^{ij}) $[i]\phi \rightarrow [j][i]\phi$
(5^{ij}) $\neg[i]\phi \rightarrow [j]\neg[i]\phi$
($\mathtt{T}^u$) $[u]\phi \rightarrow \phi$
($\subseteq .ui$) $[u]\phi \rightarrow [i]\phi$
(Least) $\langle u \rangle\, \nu$
(Most) $\langle u \rangle\, (\nu \wedge \phi) \rightarrow [u](\nu \rightarrow \phi)$

$$
\begin{aligned}
(\texttt{Covering}) \quad & [c]\phi \wedge [-c]\phi \rightarrow [u]\phi \\
(\texttt{Packing}) \quad & \langle -c \rangle\, \nu \rightarrow \neg \langle c \rangle\, \nu \\
(\texttt{Dual}) \quad & \langle i \rangle\, \phi \leftrightarrow \neg[i]\neg\phi
\end{aligned}
$$

$$
\begin{aligned}
(\texttt{Name}) \quad & \textsc{If} \vdash \nu \rightarrow \theta \ \textsc{then} \vdash \theta \\
(\texttt{MP}) \quad & \textsc{If} \vdash \phi_1 \ \textsc{and} \vdash \phi_1 \rightarrow \phi_2 \ \textsc{then} \vdash \phi_2 \\
(\texttt{N}^i) \quad & \textsc{If} \vdash \phi \ \textsc{then} \vdash [i]\phi
\end{aligned}
$$

where i, j are metavariables for the elements of K, c denotes elements of the set of atomic context indexes C, u is the universal context index, ν ranges over nominals, and θ in rule `Name` denotes a formula in which the nominal denoted by ν does not occur. The proofs of soundness and completeness of the axiomatization w.r.t. $\textsc{Cxt}^{\top,\setminus}$ frames are provided in Section 9.

The new axioms and rules deserve some comments. Let us start with the axiomatization of nominals. Axiom `Least` states just that every nominal denotes *at least* one state. Vice versa, axiom `Most` states that nominals denote *at most* one state. Intuitively it says that, if there is a state named ν where ϕ holds, then ϕ holds if ν is the case. Finally, rule `Name`, which we borrowed from standard hybrid logic ([21]), states that all states are nominated. It does that by saying that if it is provable that a formula θ holds at an arbitrary state ν —the state is arbitrary since the rule requires ν not to occur in θ— then θ itself is provable since there is no world that falsifies it. From a technical point of view, as observed in [23], this rules states a sufficient condition for function $\mathbb{N}\rceil\mathcal{I}$) to be a surjection on the set of all singletons of W[4]. To sum up, axioms `Least` and `Most` with rule `Name` axiomatize the conditions holding on the interpretation function $\mathcal{I}$ as exposed in Section 6.2.

Let us now discuss the axioms that are more central to the modeling aim we are pursuing: axioms `Covering` and `Packing`. They characterize context complementation. Axiom `Covering` states that if some formula holds in both c and $-c$, than it holds globally. In other words, it states that the universal context is *covered* by the contexts denoted by c and, respectively, $-c$. Axiom `Packing` states instead that the contexts denoted by c and $-c$ are strongly disjoint, in the sense that they do not contain the same states or. They *pack* the universal context in two disjoint subcontexts. They are thus just modal formulations of the two properties characterizing the bipartition of a given set. Notice that nominals are necessary in the formulation of the `Packing` axiom. It is easy to see that, without the possibility of naming individual states, it would be impossible to axiomatize disjointness.

6.4 A remark: $\mathbf{Cxt}^{\mathbf{u},-}$ as hybrid logic

Before putting the formalism at work it might be instructive to make one last technical remark. In logic $\mathbf{Cxt}^{\mathbf{u},-}$ a family of $@_\nu$ operators is definable, by means of which it is possible to express that a formula ϕ holds in the state named by ν: $@_\nu\phi$. This operator

[4] Rule `Name` plays a central role in the completeness proof for $\textsc{Cxt}^{\top,\setminus}$ (see the proof of Lemma 9 in Section 9).

is known in hybrid logics ([21]) as the *satisfaction operator*. Its semantics is given in terms of the following satisfaction clause:

$$\mathcal{M}, w \models @_{\nu}\phi \text{ iff } \mathcal{M}, \mathcal{I}(\nu) \models \phi.$$

The property of “holding in a state” is thus a global property, that is, it is independent of the point of evaluation. The clause states more precisely that, whatever the state of evaluation is, it is the case that if ν holds then ϕ also holds. In fact, the satisfaction operator can be defined in any logic enabling nominals and a universal modality ([25], [26]) as follows:

$$@_{\nu}\phi := [u](\nu \rightarrow \phi) \qquad (28)$$

where $@_{\nu}$ is a nominal and ϕ a formula. Leaving technicalities aside, this means that logic $\mathbf{Cxt}^{\mathbf{u},-}$ has sufficient expressive means to represent statements of the type “in situation (or state) ν state-of-affairs ϕ holds”. This expressive capability of logic $\mathbf{Cxt}^{\mathbf{u},-}$ will turn out useful to represent intuitive reasoning patterns involving constitutive counts-as statements (see Proposition 6).

6.5 Constitutive Counts-as formalized

Using a multi-modal logic $\mathbf{Cxt}^{\mathbf{u},-}$ on a language $\mathcal{L}_n^{u,-}$, the constitutive reading of counts-as statements can now be formalized.

Definition 8. (Constitutive counts-as: $\Rightarrow^{co}_{c,\Gamma}$)
Given a set of formulae Γ such that $\gamma_1 \rightarrow \gamma_2 \in \Gamma$, the constitutive counts-as statement “γ_1 counts as γ_2 in the context c defined by Γ” is formalized in a multi-modal logic $\mathbf{Cxt}^{\mathbf{u},\backslash}$ on language $\mathcal{L}_n^{u,-}$ as follows:

$$\gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2 := [c]\Gamma \wedge [-c]\neg\Gamma \wedge \neg[u](\gamma_1 \rightarrow \gamma_2)$$

with γ_1 and γ_2 objective formulae.

The definition implements in modal logic the intuition summarized in Observation 2, and formalized in Definition 4: constitutive counts-as statements correspond to those non trivial classifications which are stated by the definition Γ of the context c. In fact the following can be proven.

Proposition 5. *(Equivalence of Definitions 8 and 4)*
Let $\mathcal{M}$ be a $\textsc{Cxt}^{\top,\backslash}$ frame and Γ a set of objective formulae. It is the case that: $\gamma_1 \rightarrow \gamma_2 \in \mathbb{CO}(\Gamma, W_c)$ iff $\gamma_1 \rightarrow \gamma_2 \in \{\gamma_1 \rightarrow \gamma_2 \in \Gamma \mid \mathcal{M} \models \gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2 \}$. To put it otherwise:

$$\mathbb{CO}(\Gamma, W_c) = \{\gamma_1 \rightarrow \gamma_2 \in \Gamma \mid \mathcal{M} \models \gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2\}$$

Proof. The proof follows from Proposition 4.

A detailed comment of Definition 8 is in order. The most important consequence of it is that it is possible to talk about constitutive counts-as only once a set Γ is given. As already stressed in Section 4.2, there is no formula that is constitutive in isolation. This logic of constitutive rules takes therefore the warning raised in [27] very seriously: “no logic of norms without attention to a system of which they form part" ([27], pag. 29). As a result, constitutive counts-as statements can also be viewed as forms of speech acts creating a context: given that $\gamma_1 \rightarrow \gamma_2$ is a formula of Γ, $\gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2$ could be read as "let it be that $\gamma_1 \rightarrow \gamma_2$ with all the statements of Γ and only of Γ or, using the terminology of [28], "fiat Γ and only Γ". On the other hand, a constitutive counts-as is false if either Γ does not define the context denoted by c, or if it expresses a classification which is valid in the model.

This is precisely the distinctive feature of constitutive counts-as with respect to its two classificatory relatives. While the classificatory versions of counts-as express what at least holds in a context (contextual classification) and, respectively, what at least hold in a context which is not globally true (proper contextual classification), the constitutive version expresses also what at most holds in a context, thereby making explicit what the context actually is in terms of a set of formulae of the language. We can have a constitutive counts-as statement only if it is known what the definition is of the context the statement refers to. In the classificatory versions of counts-as this knowledge is absent since it is only partially known what the context explicitly is. Classificatory and proper classificatory counts-as statements presuppose the existence of a context of which only some information is available. This issue is discussed in more detail in [7] where classificatory and proper classificatory counts-as statements are related with the notion of enthymeme, i.e., of argument with unstated premises.

From a technical point of view, this linguistic dependence amounts to the fact that expressions of the form $\gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2$ where $\gamma_1 \rightarrow \gamma_2 \not\in \Gamma$ are just undefined. Only the classifications that belong to Γ can be evaluated as constitutive counts-as. In other words $\Rightarrow^{co}_{c,\Gamma}$ conditionals are not "logical" in the sense of yielding a truth value for any pair of formulae (γ_1, γ_2). Because of this there is no logic, in a proper sense, of constitutive statements pertaining to one context description. Given a set of $\Rightarrow^{co}_{c,\Gamma}$ statements, nothing can be inferred about $\Rightarrow^{co}_{c,\Gamma}$ statements which are not already contained in the set Γ. It is therefore not possible to study $\Rightarrow^{co}_{c,\Gamma}$ conditionals from a structural perspective like it has been done for the other forms of counts-as in Propositions 1, 2 and 3.

How awkward this might sound it is perfectly aligned with the intuitions on the notion of constitution which backed Definition 8: constitutive counts-as are those classifications which are explicitly stated in the specification of the normative system. In a sense, constitutive statements are just given, and that is it. This does not mean, however, that constitutive statements cannot be used to perform reasoning. The following example depicts the most typical form of reasoning involving constitutive counts-as statements.

Proposition 6. *($\Rightarrow^{co}_{c,\Gamma}$ and $@_\nu$)*
The following formula is valid in $\mathrm{C}{\small\mathrm{XT}}^{\top,\backslash}$ *frames for any Γ containing $\gamma_1 \rightarrow \gamma_2$:*

$$\gamma_1 \Rightarrow^{co}_{c,\Gamma} \gamma_2 \rightarrow ((@_\nu \Gamma \wedge @_\nu \gamma_1) \rightarrow @_\nu \gamma_2) \tag{29}$$

Proof. Follows from Definition 4, Formula 28 and propositional logic.

This property shows how constitutive rules work in providing grounds for inferring the occurrence of new states-of-affairs: it is a rule of the normative system of Utrecht

University that if the promotor pronounces the PhD. student to be a doctor then this counts as the PhD. student to be a doctor ($\gamma_1 \Rightarrow^{co}_{c,\mathit{\Gamma}} \gamma_2$); the current situation ν falls under the rules of Utrecht University ($@_\nu \mathit{\Gamma}$) and in the current situation the promotor pronounces a PhD. student to be a doctor ($@_\nu \gamma_1$), hence in the current situation the PhD. student is a doctor ($@_\nu \gamma_2$). Formula 29 perfectly captures the logical pattern of "conventional generation" as it is described in [29]:

> "Act-token A of agent G conventionally generates act-token B [. . .] only if the performance of A [. . .], together with a rule R saying that A [. . .] counts as B, guarantees the performance of B" ([29], p. 25).

It is instructive to notice that, besides formula $\gamma_1 \Rightarrow^{co}_{c,\mathit{\Gamma}} \gamma_2$, what plays an essential role here is formula $@_\nu \mathit{\Gamma}$ (i.e., $[u](\nu \rightarrow \mathit{\Gamma})$), which states that situation ν is one of the situations in context c. Without the notion of context definition and the availability of nominals, this could not be expressed.

Complex reasoning patterns involving constitutive counts-as statements arise also in relation with the other two notions of counts-as. The following section investigates the logical relationships between the three different senses of counts-as.

7 Relating the many faces of counts-as

This section is devoted to pursuing the last goal mentioned in the quote from [8] mentioned in Section 1: "and then we may proceed to a quiet and systematic study of all concepts involved, which will exhibit their main properties and mutual relations."

The logical relations between $\Rightarrow^{co}_{c,\mathit{\Gamma}}$, $\Rightarrow^{cl+}_{c}$ and $\Rightarrow^{cl}_{c}$ can be studied in logic $\mathbf{Cxt}^{\mathbf{u},\backslash}$ which extends both $\mathbf{K45}^{\mathbf{ij}}_{\mathbf{n}}$, i.e., the logic in which $\Rightarrow^{cl}_{c}$ has been defined, and $\mathbf{Cxt}^{\mathbf{u}}$, i.e., the logic in which $\Rightarrow^{cl+}_{c}$ has instead been defined.

Proposition 7. ($\Rightarrow^{cl}_{c}$ vs $\Rightarrow^{cl+}_{c}$ vs $\Rightarrow^{co}_{c,\mathit{\Gamma}}$)
In logic $\mathbf{Cxt}^{\mathbf{u},\backslash}$ the following formulae are valid:

$$(\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \rightarrow (\gamma_1 \Rightarrow^{cl}_{c} \gamma_2) \tag{30}$$

$$(\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \rightarrow (\gamma_1 \wedge \gamma_3 \Rightarrow^{cl}_{c} \gamma_2) \tag{31}$$

$$((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl+}_{c} \gamma_3)) \rightarrow (\gamma_1 \Rightarrow^{cl}_{c} \gamma_3) \tag{32}$$

$$(\gamma_1 \Rightarrow^{co}_{c,\mathit{\Gamma}} \gamma_2) \rightarrow (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \tag{33}$$

provided that $\gamma_1 \rightarrow \gamma_2 \in \mathit{\Gamma}$.

Proofs are omitted and can be easily obtained by application of Definitions 2 and 6 and Proposition 1.

Let us have a look at the intuitive meaning of the formulae just proven. Formula 30 states something very simple: proper contextual classification implies contextual classification. This corresponds, in the model-theoretic notation used in Section 4, to the following inclusion relation: $\mathbb{CL}^{+}(W_c) \subseteq \mathbb{CL}(W_c)$.

Formulae 31 and 32 are particularly interesting. If we forget that the two operators $\Rightarrow^{cl+}_{c}$ and $\Rightarrow^{cl}_{c}$ denote two different notions and we read both expressions $\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2$

and $\gamma_1 \Rightarrow_c^{cl} \gamma_2$ just as "γ_1 counts as γ_2", these formulae would sound as statements of the property of antecedent strengthening and of the transitivity of "counts-as". However, our formal analysis based on the acknowledgment that counts-as hides different senses has shown that transitivity and antecedent strengthening hold for $\Rightarrow_c^{cl}$ but not for $\Rightarrow_c^{cl+}$. On the other hand, and this is what Proposition 7 shows, their logical interactions display patterns clearly reminiscent of those properties. In a sense, we showed that questions such as "is transitivity a meaningful property for a characterization of counts-as?" are flawed by the possibility of confusing under the label counts-as different notions which enjoy different logical behaviors. This is a concrete example of the methodological concerns raised in Section 4.3.

More specifically, Formula 31 expresses that given a counts-as statement interpreted as a proper classification, a contextual classification can be inferred having as antecedent a strengthened version of the antecedent of the first statement, and this although proper contextual classification does not enjoy antecedent strengthening. In other words, although $\Rightarrow_c^{cl+}$ does not enjoy antecedent strengthening, it is nonetheless grounds for performing monotonic reasoning via $\Rightarrow_c^{cl}$. Analogous considerations apply to Formula 32. Proper contextual classification does not enjoy transitivity but reasoning via transitivity remains valid shifting from $\Rightarrow_c^{cl+}$ to $\Rightarrow_c^{cl}$.

Finally, Formula 33 translates the following intuitive fact: the promulgation of a constitutive rule guarantees the possibility of applying specific classificatory rules. If it is a rule of Γ that self-propelled conveyances count as vehicles (constitutive sense) then self-propelled conveyances count as vehicles in the context c defined by Γ in a proper classificatory sense.

With respect to the relation between constitution and classification, another interesting consequence of Definition 6 is the following one.

Proposition 8. (Impossibility of $\Rightarrow_u^{cl+}$ and $\Rightarrow_{u,\Gamma}^{co}$)
Proper classificatory counts-as statements and constitutive counts-as statements are impossible with respect to the universal context u. In fact, the following formulae are valid:

$$(\gamma_1 \Rightarrow_u^{cl+} \gamma_2) \rightarrow \bot \tag{34}$$

$$(\gamma_1 \Rightarrow_{u,\Gamma}^{co} \gamma_2) \rightarrow \bot \tag{35}$$

provided that $\gamma_1 \rightarrow \gamma_2 \in \Gamma$.

The proof is easily obtained from Definition 6.

Intuitively, Formula 34 states that what holds in general can not be the product of constitution, it can not be a "new" classification. This is indeed a very intuitive property: the fact that apples are classified as fruit cannot be a proper classification because it is something that always holds. Formula 35 states something slightly different: if something holds globally then it can not be used to constitute a context. Universal truths hold in all contexts, and therefore, can not be specific of any context. To put it otherwise, the statement "apple count as fruits" can not be a constitutive rule. Notice that contextual classificatory statements are instead perfectly sound also with respect to the universal context. Formula $\gamma_1 \Rightarrow_u^{cl} \gamma_2$ is a satisfiable formula in logic $\mathbf{Cxt}^{\mathbf{u},\setminus}$.

Let us now take into consideration properties displaying more complex reasoning patterns.

Proposition 9. (From $\Rightarrow^{co}_{c,\Gamma}$ to $\Rightarrow^{cl}_{c}$ and $\Rightarrow^{cl+}_{c}$ via $\Rightarrow^{cl}_{u}$)
The following formulae are valid:

$$(\gamma_2 \Rightarrow^{co}_{c,\Gamma} \gamma_3) \rightarrow ((\gamma_1 \Rightarrow^{cl}_{u} \gamma_2) \rightarrow (\gamma_1 \Rightarrow^{cl}_{c} \gamma_3)) \tag{36}$$

$$(\gamma_2 \Rightarrow^{co}_{c,\Gamma} \gamma_3) \rightarrow (((\gamma_1 \Rightarrow^{cl}_{u} \gamma_2) \wedge \neg[u](\gamma_1 \rightarrow \gamma_3)) \rightarrow (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3)) \tag{37}$$

provided that $\gamma_1 \rightarrow \gamma_2 \in \Gamma$.

The proof is straightforward by application of Definitions 2 and 8, and Propositions 3 and 1. These properties represent typical forms of reasoning patterns involving constitutive rules.

Formula 36: if it is a rule of Γ that $\gamma_2 \rightarrow \gamma_3$ ("self-propelled conveyances count as vehicles") and it is always the case that $\gamma_1 \rightarrow \gamma_2$ ("cars count as self-propelled conveyances"), then $\gamma_1 \rightarrow \gamma_3$ ("cars count as vehicles") holds in the context c defined by normative system Γ. Formula 37: if it is a rule of Γ that $\gamma_2 \rightarrow \gamma_3$ ("conveyances transporting people or goods count as vehicles") and it is always the case that $\gamma_1 \rightarrow \gamma_2$ ("bikes count as conveyances transporting people or goods") but it is not always the case that $\gamma_1 \rightarrow \gamma_3$ ("bikes count as vehicles"), then $\gamma_1 \rightarrow \gamma_3$ ("bikes count as vehicles") holds as a constituted classification in the context c defined by normative system Γ. Notice that while "cars count as self-propelled conveyances" in Formula 36 is a classificatory counts-as, since it might still be the case that cars are globally classified as vehicles, "bikes count as vehicles" in Formula 37 is instead a proper classificatory counts-as since it is explicitly stated that such classification is not a validity. Formula 37 represents nothing but the form of the reasoning pattern that has been used as starting point of our analysis (Example 1).

The very remarkable aspect about these properties is that they neatly show how the three senses of counts-as all play a role in the kind of reasoning we perform with constitutive rules. In particular, they show that the constitutive sense, though enjoying extremely poor logical properties, grounds in fact all the rich reasoning patterns proper of classificatory reasoning.

7.1 The *transfer problem* in the light of $\Rightarrow^{cl}_{c}$, $\Rightarrow^{cl+}_{c}$ and $\Rightarrow^{co}_{c,\Gamma}$

The 'transfer problem' has been introduced in [17] as a landmark for testing the intuitive adequacy of formalizations of counts-as. It can be exemplified as follows: suppose that somebody brings it about —for instance by coercion— that a priest effectuates a marriage, does this count as the creation of a state of marriage? Does anything implying that a priest effectuates a marriage count as the creation of a state of marriage? In other words, is the possibility to create a marriage transferable to anybody who brings it about that the priest effectuates the ceremony? In our framework, these questions get a triple formulation, one for each of the different senses of counts-as.

The transfer problem and $\Rightarrow^{cl}_{c}$. In [17], the transfer problem has been used as grounds for the rejection of the property of antecedent strengthening for counts-as conditionals. It is beyond doubt that a characterization of counts-as which enjoys the strengthening of the antecedent also exhibits the transfer problem: if that property holds, then the fact that the performance of the ceremony counts as the creation of a state of marriage implies that also a coerced performance does. As already noticed in [6], contextual classification ($\Rightarrow^{cl}_{c}$), which enjoys the strengthening of the antecedent (Proposition 1), does exhibit the transfer problem: whatever situation in which a priest performs a marriage ceremony is classified as a situation in which a marriage state comes to be. And this is precisely what we intuitively expect given the notion of contextual classification as informally introduced in Section 2. In other words, contextual classification *should* exhibit the transfer problem or, to put it another way, it should display a *transfer property*: the bringing about of a state of marriage should be transferable to any state in which a priest performs the ceremony.

The transfer problem and $\Rightarrow^{cl+}_{c}$. It has been shown that the characterization of proper contextual classification ($\Rightarrow^{cl+}_{c}$) does not enjoy the strengthening of the antecedent (Proposition 2). Interestingly enough, it still exhibits the transfer problem, as shown in Proposition 3 where Formula 24 has been proven valid: $\neg[u](\gamma_1 \rightarrow \gamma_3) \rightarrow ((\gamma_1 \Rightarrow^{cl+}_{c} \gamma_2) \wedge (\gamma_2 \Rightarrow^{cl+}_{c} \gamma_3)) \rightarrow (\gamma_1 \Rightarrow^{cl+}_{c} \gamma_3)$.

Intuitively, this formula expresses what follows. If the fact that a priest effectuates a marriage (γ_1) under coercion of a third party (γ_3) is not globally classified as giving rise to a state of marriage (γ_2) —which is the case, given the intuitive reading of the scenario at issue— then it is safe to say that if the priest's performance of the marriage counts as (in a proper classificatory sense) a marriage, then a coerced performance of the marriage counts also as a marriage.

Notice that this is again something perfectly intuitive given the assumptions about proper contextual classification exposed in Section 2: if a context c makes a classification $\gamma_1 \rightarrow \gamma_2$ true, which does not hold in general, then also the strengthened version of it $\gamma_1 \wedge \gamma_3 \rightarrow \gamma_2$ is true in that context. Besides, if the strengthened version is also not true in general, it then follows that $\gamma_1 \wedge \gamma_3 \rightarrow \gamma_2$ is also a novel classification which is brought about by context c. Exhibiting the transfer problem is also for proper contextual classification not problematic.

From a technical point of view, Proposition 3 shows that a characterization of counts-as, which does not enjoy the strengthening of the antecedent, can still exhibit the transfer problem. This is equivalent to say that a notion of counts-as which genuinely rejects the transfer problem should not only reject antecedent strengthening, but some yet weaker property.

The transfer problem and $\Rightarrow^{co}_{c,\Gamma}$. The 'transfer property' does not hold, instead, for the constitutive reading of counts-as statements. In this view, counts-as statements represent the rules specifying a normative system. So, all that it is explicitly stated by the 'institution of marriage' is that if the priest performs the ceremony then the couple is married, while no rule belongs to that normative system which states that the action of a third party bringing it about that the priest performs the ceremony also counts as a marriage. Our formalization fully captures this feature. Let the 'marriage institution' c be sketched by the set of rules $\Gamma = \{p \rightarrow m\}$, i.e., by the rule "if the priest performs the ceremony, then the couple is married". Let then t represent the fact that a third party brings it about that p. For Definition 8 the counts-as $(t \wedge p) \Rightarrow^{co}_{c,\Gamma} m$ is just an undefined expression, because $((t \wedge p) \rightarrow m) \notin \Gamma$, that is, because the 'marriage institution' does not state such a classification. This seems to suggest that the transfer problem, rather than having to do with the structural properties of a logical connective, concerns instead whether a rule is part of the promulgations of a normative system or not, that is to say, whether a counts-as statement is a constitutive rule or not.

8 Conclusions

Moving from hints provided by the literature on legal and social theory concerning constitutive rules, the paper has analyzed counts-as statements as forms of contextual classifications. This analytical option, which we have studied from a formal semantics perspective, has delivered three semantically precise senses (Definitions 2, 6 and 8) in which counts-as statements can be interpreted, which we called *classificatory*, *proper classificatory* and *constitutive* readings. The three readings have then been formally analyzed making use of modal logic.

The classificatory reading resulted in a strong logic of counts-as conditionals enabling many properties which are typical of reasoning with concept subsumptions such as, in particular, reflexivity, strengthening of the antecedent and weakening of the consequent (Proposition 1). In fact, the logic obtained is nothing but a modal logic version of the contextual terminological logic we investigated in [9, 10].

The characterization of proper contextual classification resulted, instead, in a much weaker logic rejecting reflexivity, transitivity and antecedent strengthening (Proposition 2), but retaining cumulative transitivity (Proposition 3). Noticeably, this notion corresponds to the counts-as characterized in [17] once transitivity is substituted with cumulative transitivity. Finally, the notion of proper contextual classification has offered some new insights on the transfer problem (Section 7.1) showing that it cannot be genuinely avoided just by means of rejecting the strengthening of the antecedent in a conditional logic setting. This result motivated the investigation of a yet stronger form of counts-as which we developed in [30], and which stems nevertheless from the same analytical option backing the present work.

The formal analysis of constitutive counts-as (Definition 8) has neatly shown, with formal means, in what sense constitutive rules are never constitutive in isolation, but only as parts of systems of rules, and how constitutive rules work in providing grounds for attributing institutional properties to situations (Proposition 6). Constitutive counts-as has also been shown to imply the two classificatory readings (Proposition 7). Other logical interrelationships between the three notions of counts-as have also been studied (Propositions 8 and 9) showing also that the logical relations between them could actually be grounds for fallacies in the formal characterization of counts-as once the polysemy of the term "counts-as" is overlooked.

9 Appendix: Soundness and Completeness Results

The appendix provides soundness and completeness results for the logics introduced in the paper: $\mathbf{K45^{ij}_{n}}$, $\mathbf{Cxt^{u}}$ and $\mathbf{Cxt^{u,-}}$. Completeness will be proven via the canonical model technique.

9.1 Preliminaries

In all the logics considered the axiomatization of every modality $[i]$ contains all tautologies of propositional calculus, axiom K and is closed under rules MP and N. We will therefore make use of some general results of completeness theory for normal modal logics. We refer the reader to [21] for further details.

Recall first some facts about maximal consistent sets. Let Λ be a multi-modal normal logic. A maximal Λ-consistent set of formulae on a multi-modal language $\mathcal{L}_n$ is a set Φ s.t.: (a) $\bot$ is not derivable in Λ from Φ (i.e., Λ-consistency of Φ); (b) every set properly including Φ is Λ-inconsistent. Every maximal Λ-consistent set Φ is such that: $\Lambda \subseteq \Phi$; Φ is closed under rule MP; for all formulae ϕ either $\phi \in \Phi$ or $\neg\phi \in \Phi$; for all formulae $\phi, \psi : \phi \vee \psi \in \Phi$ iff $\phi \in \Phi$ or $\psi \in \Phi$.

We can now report the notion of canonical model for a logic Λ.

Definition 9. *(Canonical model for logic Λ)*
The canonical model $\mathcal{M}^{\Lambda}$ for a normal modal logic Λ in the multi-modal language $\mathcal{L}_n$ is the structure $\langle W^{\Lambda}, \{R_i^{\Lambda}\}_{1 \leq i \leq n}, \mathcal{I}^{\Lambda} \rangle$ where:

1. *The set W^{Λ} is the set of all maximal Λ-consistent sets.*
2. *The anonical relations $R_i^{\Lambda} \in \{R_i^{\Lambda}\}_{1 \leq i \leq n}$ are defined as follows: for all $w, w' \in W^{\Lambda}$, if for all formulae ϕ, $\phi \in w'$ implies $\langle i \rangle \phi \in w$, then $wR_i^{\Lambda}w'$.*
3. *The canonical interpretation $\mathcal{I}^{\Lambda}$ is defined by $\mathcal{I}^{\Lambda}(p) = \{w \in W^{\Lambda} \mid p \in w\}$.*

We briefly recall three key propositions of (modal) completeness theory. For the proofs we refer the reader to [21].

Lemma 1. *(Strong completeness = satisfiability of all consistent sets)*
A normal modal logic Λ is strongly complete w.r.t. a class of frames $\mathfrak{F}$ iff every Λ-consistent set of formulae is satisfiable on some $\mathcal{F} \in \mathfrak{F}$, i.e., it has a model $\mathcal{M}$ built on a frame $\mathcal{F}$ in class $\mathfrak{F}$.

Lemma 2. *(Existence Lemma)*
For any normal modal logic Λ and any state $w \in W^{\Lambda}$, it holds that: if $\langle i \rangle \phi \in w$ then there exists a state $w' \in W^{\Lambda}$ such that $wR_i^{\Lambda}w'$ and $\phi \in w'$.

Lemma 3. *(Truth Lemma)*
For any normal modal logic Λ and any formula ϕ, it holds that: $\mathcal{M}^{\Lambda}, w \models \phi$ iff $\phi \in w$.

Lemma 4. *(Canonical Model Theorem)*
Any normal modal logic Λ is strongly complete w.r.t. its canonical model $\mathcal{M}^{\Lambda}$.

We will also make use of the notion of point-generated subframe. Given a frame $\mathcal{F} = \langle W, \{R_i\}_{1 \leq i \leq n}\} \rangle$, a point-generated subframe $\mathcal{F}^w$ of a frame $\mathcal{F}$ is a structure $\langle W^w, \{R_i^w\}_{1 \leq i \leq n}\} \rangle$ such that: (a) W^w is the set of states $w' \in W$ such that there exists, for any R_i, a finite R_i-path from w to w'; (b) $R_i^w = R_i \cap (W^w \times W^w)$, i.e., each R_i^w is the restriction of R_i on W^w. The following result is of interest.

Lemma 5. *(Generated subframes preserve validity)*
Let $\mathfrak{F}$ be a class of frames and $g(\mathfrak{F})$ be the class of point-generated subframes of the frames in $\mathfrak{F}$. It holds that, for all formulae ϕ on language $\mathcal{L}_n$: $\mathfrak{F} \models \phi$ iff $g(\mathfrak{F}) \models \phi$.

Finally, we need a way to relate context frames (see Section 3.1), that is, structures of the type $\langle W, \{W_i\}_{i \in C}\rangle$ with relational structures of the type $\langle W, \{R_i\}_{i \in C}\rangle$. The bridge is offered by *locally universal* relations. A relation R_i on a set W is locally universal if:

- For all $R_i \in \{R_i\}_{i \in C}$ and $w \in W$, R_i is universal on $r_i(w)$;
- For all $w, w' \in W$, $r_i(w) = r_i(w')$, where r_i is a function associating to each state w the set of reachable states via relation R_i.

The following representation result holds for this family of relations.

Lemma 6. *(Representation of context frames)*
A relation R_i on W is locally universal iff there exists a set $W_i \subseteq W$ such that for all w, w', wR_iw' iff $w' \in W_i$.

Proof. The right to left direction is straightforward. From left to right: for every $w, w' \in W$ it holds, by the definition of function r that wR_iw' iff $w' \in r_i(w)$. Since R_i is locally universal, it holds that for every $w, w'' \in W$, $r_i(w) = r_i(w'')$. It is now enough to stipulate $W_i = r_i(w'')$ for any w'' to obtain the desired result: there exists a set $W_i \subseteq W$ such that for all w, w', wR_iw' iff $w' \in W_i$.

Leaving technicalities aside, the property of local universality forces relations in $\{R_i\}_{i \in C}$ to cluster the domain of the frame in sets of worlds (contexts), one for each accessibility relation, and then defines these accessibility relations in such a way that the sets of accessible worlds correspond, for each world in W, to the clusters.

9.2 Soundness and completeness of $\mathbf{K45^{ij}_n}$

The proof of soundness is routinary. It is well-known that inference rules MP and N preserve validity on any class of frames[5]. Providing the soundness of $\mathbf{K45^{ij}_n}$ w.r.t. CXT frames boils than down to checking the validity of axioms 4^{ij} and 5^{ij}.

Theorem 1. *(Soundness of $\mathbf{K45^{ij}_n}$ w.r.t.* CXT *frames)*
Logic $\mathbf{K45^{ij}_n}$ is sound w.r.t. the class of CXT *frames.*

Proof. The validity of 4^{ij} is proven showing that its contrapositive has no countermodel. Such countermodel $\mathcal{M}$ would contain a state w such that for a given formula ϕ, $\mathcal{M}, w \models \langle j \rangle \langle i \rangle \phi$ and $\mathcal{M}, w \models \neg \langle i \rangle \phi$. Hence, by the semantics, $\exists w' \in W_i$ s.t. $\mathcal{M}, w \models \phi$ and $\nexists w' \in W_i$ s.t. $\mathcal{M}, w \models \phi$, which is impossible. The validity of 5^{ij} is proven in the same way. Suppose there is a model $\mathcal{M}$ and a state w such that $\mathcal{M}, w \models \langle i \rangle \phi$ and $\mathcal{M}, w \models \neg [j] \langle i \rangle \phi$. Hence, by the semantics, $\exists w' \in W_i$ s.t. $\mathcal{M}, w \models \phi$ and $\nexists w' \in W_i$ s.t. $\mathcal{M}, w \models \phi$.

The proof of completeness is obtained in two steps.

1. First, via the canonical model, it is proven that logic $\mathbf{K45^{ij}_n}$ is complete with respect to the class of i-j transitive (if wR_iw' and $w'R_jw''$ then wR_jw''), and i-j euclidean (if wR_iw' and wR_jw'' then $w'R_jw''$) frames[6].

[5] See [21].

[6] In [31], frames with this property are called, respectively, hyper-transitive and hyper-euclidean.

2. Second, it is proven that if $\mathfrak{F}$ is the class of of i-j transitive and i-j euclidean frames, for every $\phi \in \mathcal{L}_n$: $\mathfrak{F} \models \phi$ iff $\textsc{Cxt} \models \phi$.

Theorem 2. *(Completeness of $\mathbf{K45^{ij}_n}$)*
Logic $\mathbf{K45^{ij}_n}$ is strongly complete w.r.t. the class of i-j transitive and i-j euclidean frames.

Proof. By Lemma 1, given a $\mathbf{K45^{ij}_n}$-consistent set Φ of formulae, it suffices to find a model state pair ($\mathcal{M}$, w) such that: (a) $\mathcal{M}, w \models \Phi$, and (b) the frame $\mathcal{F}$ on which $\mathcal{M}$ is based is i-j transitive and i-j euclidean. Let $\mathcal{M}^{\mathbf{K45^{ij}_n}} = \left\langle W^{\mathbf{K45^{ij}_n}}, \{R_i^{\mathbf{K45^{ij}_n}}\}_{i \in C}, \mathcal{I}^{\mathbf{K45^{ij}_n}} \right\rangle$ be the canonical model of logic $\mathbf{K45^{ij}_n}$, and let Φ^+ be any maximal consistent set in $W^{\mathbf{K45^{ij}_n}}$ extending Φ. By Lemma 3 it follows that $\mathcal{M}^{\mathbf{K45^{ij}_n}}, \Phi^+ \models \Phi$, which proves (a). It remains to be proven that $\left\langle W^{\mathbf{K45^{ij}_n}}, \{R_i^{\mathbf{K45^{ij}_n}}\}_{i \in C} \right\rangle$ enjoys i-j transitivity (b.1) and i-j euclidicity (b.2). To prove (b.1) consider three states $w, w', w'' \in W^{\mathbf{K45^{ij}_n}}$ such that $wR_j^{\mathbf{K45^{ij}_n}}w'$ and $w'R_i^{\mathbf{K45^{ij}_n}}w''$. Suppose then that $\phi \in w''$. As $w'R_i^{\mathbf{K45^{ij}_n}}w''$ and $wR_j^{\mathbf{K45^{ij}_n}}w'$, it follows that $\langle i \rangle \phi \in w'$ and then that $\langle j \rangle \langle i \rangle \phi \in w$. Now, w is a maximal consistent set of logic $\mathbf{K45^{ij}_n}$, it therefore contains formula $\langle j \rangle \langle i \rangle \phi \rightarrow \langle i \rangle \phi$ (i.e., the contrapositive of axiom 4^{ij}), hence $\langle i \rangle \phi \in w$ and thus $wR_i^{\mathbf{K45^{ij}_n}}w''$ which completes the proof of (b.1). Analogously, to prove (b.2) consider three states $w, w', w'' \in W^{\mathbf{K45^{ij}_n}}$ such that $wR_j^{\mathbf{K45^{ij}_n}}w'$ and $wR_i^{\mathbf{K45^{ij}_n}}w''$. Suppose then that $\phi \in w''$. It follows that $\langle i \rangle \phi \in w$ and since w is a maximal consistent set of logic $\mathbf{K45^{ij}_n}$, it therefore contains formula $\langle i \rangle \phi \rightarrow [j] \langle i \rangle \phi$ (i.e., axiom 5^{ij}) and hence $[j] \langle i \rangle \phi \in w$. From this and from $wR_i^{\mathbf{K45^{ij}_n}}w''$ it follows that $\langle i \rangle \phi \in w''$, that is to say, for any formula ϕ it is the case that: if $\phi \in w'$ then $\langle i \rangle \phi \in w''$. Now, by Definition 9, this implies that $w'R_i^{\mathbf{K45^{ij}_n}}w''$ which proves (b.2).

Lemma 7. *(Semantic equivalence for* $\textsc{Cxt}$ *frames)*
Consider the class $\mathfrak{F}$ of i-j transitive and i-j euclidean frames. For every $\phi \in \mathcal{L}_n$, $\mathfrak{F} \models \phi$ iff $\textsc{Cxt} \models \phi$. That is, $\textsc{Cxt}$ frames and $\mathfrak{F}$ frames define the same logic.

Proof. From right to left: for every ϕ, $\textsc{Cxt} \models \phi$ implies $\mathfrak{F} \models \phi$. The proof is obtained showing that if $\mathcal{F}$ is a $\textsc{Cxt}$ frame then it is i-j transitive and i-j euclidean. By Lemma 6, for all $w, w' \in W$, $w' \in W_i$ iff wR_iw'. To prove i-j transitivity, suppose that wR_iw' ($w' \in W_i$) and $w'R_jw''$ ($w'' \in W_j$). It follows therefore that wR_jw''. The proof of i-j euclidicity is perfectly analogous. Suppose that wR_iw' ($w' \in W_i$) and wR_jw'' ($w'' \in W_j$), hence $w'R_jw''$. From left to right: for every ϕ, $\mathfrak{F} \models \phi$ implies $\textsc{Cxt} \models \phi$. In this case, the proof is obtained by showing that every i-j transitive and i-j euclidean frame, which is also point-generated, is a context frame. By Lemma 5, it holds that for every ϕ, $\mathfrak{F} \models \phi$ iff $g(\mathfrak{F}) \models \phi$. Now, let $\mathcal{F}^w$ be any frame in $g(\mathfrak{F})$ generated by some state w. In order to prove the desired result, it suffices to show that every i-j transitive and i-j euclidean frame $\mathcal{F}^w$ generated by state w is a $\textsc{Cxt}$ frame. By Lemma 6, this is proven by showing that for every $R_i^w \in \{R_i^w\}_{i \in C}$, $w'R_i^ww''$ iff $w'' \in r_i^w(w)$. This amounts to prove that for every w', w'' if there exists an R_i-path from w to w' and from w to w'', then $w'R_iw''$ iff $w'' \in r_i(w)$. From left to right, if there exists an R_i-path from w to w' and $w'R_iw''$, then by transitivity (which is a special case of i-j transitivity) wR_iw'', that is, $w'' \in r_i(w)$. From right to left, if there exists an R_i-path from w to w' and $w'' \in r_i(w)$, then wR_iw'' and hence, by euclidicity, $w'R_iw''$.

Corollary 1. *(Completeness of* $\mathbf{K45^{ij}_n}$ *w.r.t.* $\textsc{Cxt}$ *frames)*
Logic $\mathbf{K45^{ij}_n}$ *is strongly complete w.r.t. the class of* $\textsc{Cxt}$ *frames.*

Proof. Follows directly from Theorem 2 and Lemma 7.

9.3 Soundness and completeness of $\mathbf{Cxt^u}$

On the grounds of the results of the previous section, the proof of soundness and completeness of $\mathbf{Cxt^u}$ w.r.t. $\textsc{Cxt}^{\top}$ can be easily obtained. Soundness boils down to prove that axioms T^u and $\subseteq .ui$ are valid in $\mathbf{Cxt^u}$ frames.

Theorem 3. *(Soundness of* $\mathbf{Cxt^u}$ *w.r.t.*$\textsc{Cxt}^{\top}$ *frames)*
Logic $\mathbf{Cxt^u}$ *is sound w.r.t. the class of* $\textsc{Cxt}^{\top}$ *frames.*

Proof. Trivial, given the interpretation of the $[u]$-operator as universal quantification on all the states in the domain W of the frame.

Let $\mathfrak{TE}^{\sim}$ be the class of frames satisfying the following properties: they are i-j transitive, i-j euclidean; they contain an equivalence relation R_u such that for all $i \in C$, $R_i \subseteq R_u$. Again, completeness w.r.t. the relevant class of frames is proven in two steps.

1. Logic $\mathbf{Cxt^u}$ is first proven to be complete w.r.t. the class of $\mathfrak{TE}^{\sim}$ frames.
2. It is then proven that for any formula ϕ on $\mathcal{L}_n$: $\mathfrak{TE}^{\sim} \models \phi$ iff $\textsc{Cxt}^{\top} \models \phi$.

Theorem 4. *(Completeness of* $\mathbf{Cxt^u}$*)*
Logic $\mathbf{Cxt^u}$ *is strongly complete w.r.t. the class* $\mathfrak{TE}^{\sim}$ *frames.*

Proof. By Lemma 1, given a $\mathbf{Cxt^u}$-consistent set Φ of formulae, it suffices to find a model state pair ($\mathcal{M}$, w) such that: (a) $\mathcal{M}, w \models \Phi$, and (b) the frame $\mathcal{F}$ on which $\mathcal{M}$ is based is i-j transitive and i-j euclidean and contains a universal relation. Claim (a) is proven by making use of Lemma 3. It remains to be proven that the frame $\langle W^{\mathbf{Cxt^u}}, \{R_i^{\mathbf{Cxt^u}}\}_{i \in C}\rangle$ of the canonical model enjoys i-j transitivity and i-j euclidicity (b.1) and that there exists a relation $R_u^{\mathbf{Cxt^u}} \in \{R_i^{\mathbf{Cxt^u}}\}_{i \in C}$ such that $R_u^{\mathbf{Cxt^u}}$ is an equivalence relation (b.2) and for every $i \in C$, $R_i \subseteq R_u$ (b.3). Claim (b.1) follows from Theorem 2 since $\mathbf{Cxt^u}$ extends $\mathbf{K45^{ij}_n}$. As to (b.2), it follows from (b.1) that each $R_i^{\mathbf{Cxt^u}}$ is transitive and euclidean and, therefore, so is $R_u^{\mathbf{Cxt^u}}$. The proof of the reflexivity of $R_i^{\mathbf{Cxt^u}}$ is then routinary. Finally, claim (b.3) needs to be proven. Consider two states $w, w' \in W^{\mathbf{Cxt^u}}$ such that $wR_i^{\mathbf{Cxt^u}}w'$. Suppose then that $\phi \in w'$. It follows that $\langle i \rangle \phi \in w$. Since w is a maximal $\mathbf{Cxt^u}$-consistent set, it contains formula $\langle i \rangle \phi \rightarrow \langle u \rangle \phi$ (i.e., the contrapositive of axiom $\subseteq .ui$) and therefore $\langle u \rangle \phi \in w$. Hence, by Definition 9, $wR_u^{\mathbf{Cxt^u}}w'$.

Lemma 8. *(Semantic equivalence for* $\textsc{Cxt}^{\top}$ *frames)*
For any formula ϕ *on* $\mathcal{L}_n$*:* $\mathfrak{TE}^{\sim} \models \phi$ *iff* $\textsc{Cxt}^{\top} \models \phi$*. That is,* $\textsc{Cxt}^{\top}$ *frames and* $\mathfrak{TE}^{\sim}$ *frames define the same logic.*

Proof. The proof is analogous to the proof of Lemma 7. The direction from right to left (for every ϕ, $\textsc{Cxt}^{\top} \models \phi$ implies $\mathfrak{TE}^{\sim} \models \phi$) is straightforwardly proven by observing that every $\textsc{Cxt}^{\top}$ frame represents a frame containing a universal relation R_u. In fact, a relation R_u is universal iff it holds that: for any $w, w' \in W$, wR_uw' iff $w' \in W$ (notice that this is a special case of Lemma 6). But every universal relation is an equivalence relation, which also includes all R_i's for any $i \in C$. That all $\textsc{Cxt}^{\top}$ frames are i-j transitive and i-j euclidean follows from Lemma 7. This completes the proof of the right-to-left direction. From left to right: for every ϕ, $\mathfrak{TE}^{\sim} \models \phi$ implies $\textsc{Cxt}^{\top} \models \phi$. Lemma 7 has proven that every i-j transitive and i-j euclidean frame generated by state w is a $\textsc{Cxt}$ frame. Consider now the relation R^w_u of the point-generated subframe $\mathcal{F}^w$ of a frame $\mathcal{F} \in \mathfrak{TE}^{\sim}$ containing an equivalence relation R_u such that for all $i \in C$, $R_i \subseteq R_u$. To obtain the desired result —via Lemma 5— it suffices to show that the relation R^w_u is universal on W^w, which is trivial.

Corollary 2. *(Completeness of $\mathbf{Cxt^u}$ w.r.t.* $\textsc{Cxt}^{\top}$ *frames)*
Logic $\mathbf{Cxt^u}$ is strongly complete w.r.t. the class of $\textsc{Cxt}^{\top}$ *frames.*

Proof. Follows directly from Theorem 4 and Lemma 8.

9.4 Soundness and completeness of $\mathbf{Cxt^{u,-}}$

The proof of soundness is routinary.

Theorem 5. *(Soundness of $\mathbf{Cxt^{u,-}}$ w.r.t.* $\textsc{Cxt}^{\top,\setminus}$ *frames)*
Logic $\mathbf{Cxt^u}$ is sound w.r.t. the class of $\textsc{Cxt}^{\top,\setminus}$ *frames.*

Proof. It is to show that axioms `Covering` and `Packing` are valid in $\textsc{Cxt}^{\top,\setminus}$ frames by just noticing that in $\textsc{Cxt}^{\top,\setminus}$ frames, for any atomic context index c, family $\{W_c, W_{-c}\}$ is a bipartition of the domain W: $W \subseteq W_c \cup W_{-c}$, i.e., family $\{W_c, W_{-c}\}$ is a covering of W; and $W_c \cap W_{-c} = \emptyset$, i.e., $\{W_c, W_{-c}\}$ is a packing of W.

Let $\mathfrak{TE}^{\sim,\setminus}$ be the class of frames satisfying the following properties: they are i-j transitive, i-j euclidean; they contain an equivalence relation R_u such that for all $i \in C$, $R_i \subseteq R_u$; the set of relations $\{R_i\}_{i \in C}$ is such that, for any atomic context index c and states $w, w' \in W$: wR_uw' implies wR_cw' or $wR_{-c}w'$; and wR_cw' implies not $wR_{-c}w'$. Again, completeness w.r.t. the $\textsc{Cxt}^{\top,\setminus}$ frames is proven in two steps.

1. Logic $\mathbf{Cxt^{u,-}}$ is first proven to be complete w.r.t. the class of $\mathfrak{TE}^{\sim,\setminus}$ frames.
2. It is then proven that for any formula ϕ on $\mathcal{L}_n$: $\mathfrak{TE}^{\sim,\setminus} \models \phi$ iff $\textsc{Cxt}^{\top,\setminus} \models \phi$.

For completeness we need to prove some facts about the canonical model of logic $\mathbf{Cxt^{u,-}}$. Before stating and proving the desired lemma consider first that, since logic $\mathbf{Cxt^{u,-}}$ extends logic $\mathbf{Cxt^u}$, we know by Theorem 4 that the canonical model of logic $\mathbf{Cxt^{u,-}}$ contains an equivalence relation $R_u^{\mathbf{Cxt^{u,-}}}$ such that for every $i \in C$, $R_i^{\mathbf{Cxt^{u,-}}} \subseteq R_u^{\mathbf{Cxt^{u,-}}}$. Recall also that every equivalence relation yields a partition on its domain. The cluster of the partition yielded by $R_u^{\mathbf{Cxt^{u,-}}}$ on $W^{\mathbf{Cxt^{u,-}}}$ containing state w is denoted by $r_u^{\mathbf{Cxt^{u,-}}}(s)$, that is, the set of states reachable by w via $R_u^{\mathbf{Cxt^{u,-}}}$.

Lemma 9. *(Properties of maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent sets)*
Let $\mathcal{M}^{\mathbf{Cxt}^{\mathbf{u},-}} = \left\langle W^{\mathbf{Cxt}^{\mathbf{u},-}}, \{R_i^{\mathbf{Cxt}^{\mathbf{u},-}}\}_{i \in C}, \mathcal{I}^{\mathbf{Cxt}^{\mathbf{u},-}} \right\rangle$ *be the canonical model of logic* $\mathbf{Cxt}^{\mathbf{u},-}$.

1. *All maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent sets in $W^{\mathbf{Cxt}^{\mathbf{u},-}}$ contain at least one nominal;*
2. *If a nominal is contained in a maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set $w \in W^{\mathbf{Cxt}^{\mathbf{u},-}}$ then it is not contained in any other maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set $w' \in W^{\mathbf{Cxt}^{\mathbf{u},-}}$ which is accessible from w via $R_u^{\mathbf{Cxt}^{\mathbf{u},-}}$. In other words, if two maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent sets contain the same nominal, and belong to the same cluster of the partition of $W^{\mathbf{Cxt}^{\mathbf{u},-}}$ yielded by $R_u^{\mathbf{Cxt}^{\mathbf{u},-}}$, then they are the same set.*
3. *Each nominal in $\mathbb{N}$ is contained in at least one maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set.*

Proof. Clause 1. Let Φ be a maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set of $\mathcal{L}_n^{u,-}$ formulae. To prove the first claim, suppose per absurdum that $\forall \nu \in \mathbb{N}$, $\neg\nu \in \Phi$. It follows that for every ν there exists a finite conjunction θ of formulae from Φ such that: $\vdash \nu \rightarrow \neg\theta$. Now, either ν occurs in θ and thus $\nu \in \Phi$, or ν does not occur in θ and therefore, by rule $\mathtt{Name}$, $\neg\theta \in \Phi$ which is impossible. Clause 2 is proven in two steps. (a) Given a nominal $\nu \in \Phi$, for any maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set Φ it is proven that for all ϕ: $\phi \in \Phi$ iff $[u](\nu \rightarrow \phi) \in \Phi$. (b) Given two maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent sets Φ and Φ', if $\nu \in \Phi, \Phi'$ and $\Phi R_u^{\mathbf{Cxt}^{\mathbf{u},-}} \Phi'$ then $\Phi = \Phi'$. Let us prove (a). From left to right. We assumed a nominal $\nu \in \Phi$, hence if $\phi \in \Phi$ then $\nu \wedge \phi \in \Phi$, being Φ a maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set. The set Φ also contains formula $\phi \rightarrow \langle u \rangle\, \phi$ (i.e., the contrapositive of axiom $\mathtt{T}^u$) and $\langle u \rangle\, (\nu \wedge \phi) \rightarrow [u](\nu \rightarrow \phi)$ (i.e., axiom $\mathtt{Most}$) from which it follows that $\langle u \rangle\, (\nu \wedge \phi) \in \Phi$ and hence that $[u](\nu \rightarrow \phi) \in \Phi$. From right to left: for any $\phi \in \Phi$, if $[u](\nu \rightarrow \phi) \in \Phi$ then by axiom $\mathtt{T}^u$ we obtain $\nu \rightarrow \phi \in \Phi$ and then by $\mathtt{MP}$ $\phi \in \Phi$. Let us prove (b) per absurdum. Suppose $\Phi \neq \Phi'$. Then there should exist a formula ϕ such that $\phi \in \Phi$ and $\phi \notin \Phi'$ and hence $\neg\phi \in \Phi'$. From (a) it follows that $[u](\nu \rightarrow \phi) \in \Phi$ and since $\Phi R_u^{\mathbf{Cxt}^{\mathbf{u},-}} \Phi'$ we obtain that $\nu \rightarrow \phi \in \Phi'$ and via $\mathtt{MP}$ $\phi \in \Phi'$, which is impossible. Clause 3 follows easily from Lemma 2 and the fact that every state $w \in W^{\mathbf{Cxt}^{\mathbf{u},-}}$ contains formula $\langle u \rangle\, \nu$ (axiom $\mathtt{Least}$).

The lemma concerns some key properties of the interpretation of nominals. Clause 1 guarantees that in the canonical model every maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent set contains a nominal, that is, that $\mathcal{I}^{\mathbf{Cxt}^{\mathbf{u},-}}$ is a surjection on the set of singletons of $W^{\mathbf{Cxt}^{\mathbf{u},-}}$. Clause 2 is particularly interesting. It states that the same nominal can in fact belong to different maximal $\mathbf{Cxt}^{\mathbf{u},-}$-consistent sets if these sets are not related via $R_u^{\mathbf{Cxt}^{\mathbf{u},-}}$. To put it otherwise, nominals behave as real names only if they refer to sets in a same cluster in the partition yielded by $R_u^{\mathbf{Cxt}^{\mathbf{u},-}}$. It follows that interpreting nominals on a generated subframe guarantees them to behave like names, and this is precisely enough for our purposes since generated subframes preserve validity (Lemma 5). Finally, Clause 3 states just that all nominals get a denotation.

Theorem 6. *(Completeness of $\mathbf{Cxt}^{\mathbf{u},-}$)*
Logic $\mathbf{Cxt}^{\mathbf{u},-}$ is strongly complete w.r.t. the class of $\mathfrak{TE}^{\sim,\setminus}$ frames, that is, frames satisfying the following clauses:

1. *They are i-j transitive, i-j euclidean.*
2. *They contain an equivalence relation R_u such that for all $i \in C$, $R_i \subseteq R_u$.*
3. *The set of relations $\{R_i\}_{i \in C}$ is such that, for any atomic context index c and states $w, w' \in W$: (3.a) wR_uw' implies wR_cw' or $wR_{-c}w'$; and (3.b) $wR_{-c}w'$ implies not wR_cw'.*

Proof. By Lemma 1, given a $\mathbf{Cxt^{u,-}}$-consistent set $\varPhi$ of formulae, it suffices to find a model state pair ($\mathcal{M}$, w) such that: (a) $\mathcal{M}, w \models \varPhi$, and (b) the frame $\mathcal{F}$ on which $\mathcal{M}$ is based satisfies clauses 1-3. Claim (a) is proven by making use of Lemma 3. It remains to be proven that the frame $\left\langle W^{\mathbf{Cxt^{u,-}}}, \{R_i^{\mathbf{Cxt^{u,-}}}\}_{i \in C} \right\rangle$ of the canonical model satisfies clauses 1-3. Clause 1 and Clause 2 are proven to be satisfied by Theorem 4 since $\mathbf{Cxt^{u,-}}$ extends $\mathbf{K45^{ij}_n}$ and $\mathbf{Cxt^u}$. Claims (3.a) and (3.b) of clause 3 remain to be proven. To prove claim (3.a) it has to be shown that: for any atomic context index c and states $w, w' \in W^{\mathbf{Cxt^{u,-}}}$, $wR_u^{\mathbf{Cxt^{u,-}}}w'$ implies $wR_c^{\mathbf{Cxt^{u,-}}}w'$ or $wR_{-c}^{\mathbf{Cxt^{u,-}}}w'$. Consider two states $w, w' \in W^{\mathbf{Cxt^{u,-}}}$ such that $wR_u^{\mathbf{Cxt^{u,-}}}w'$ and suppose that $\phi \in w'$. Since w is a maximal $\mathbf{Cxt^{u,-}}$-consistent set, it contains formula $\langle u \rangle \phi \rightarrow (\langle c \rangle \phi \vee \langle -c \rangle \phi)$ (i.e., the contrapositive of axiom `Covering`) and therefore $\langle c \rangle \phi \vee \langle -c \rangle \phi \in w$. For the properties of maximal consistent sets it follows that either $\langle c \rangle \phi \in w$ or $\langle -c \rangle \phi \in w$, and hence by Definition 9, either $wR_c^{\mathbf{Cxt^{u,-}}}w'$ or $wR_{-c}^{\mathbf{Cxt^{u,-}}}w'$, which proves (3.a). As to (3.b), it should be proven that for any atomic context index c and states $w, w' \in W^{\mathbf{Cxt^{u,-}}}$, $wR_{-c}^{\mathbf{Cxt^{u,-}}}w'$ implies not $wR_c^{\mathbf{Cxt^{u,-}}}w'$. Suppose that $wR_{-c}^{\mathbf{Cxt^{u,-}}}w'$. By Clause 1 in Lemma 9 we know that w' should contain at least one nominal. Since all nominals denote at least one state (Clause 3 in Lemma 9) we can pick a nominal ν and suppose it to be the nominal contained in w'. By Clause 2 of this theorem, from $wR_{-c}^{\mathbf{Cxt^{u,-}}}w'$ it follows that $wR_u^{\mathbf{Cxt^{u,-}}}w'$ and from this, by Clause 2 in Lemma 9, we know that there is no $w'' \in r_u^{\mathbf{Cxt^{u,-}}}(w)$ such that $\nu \in w''$. By Definition 9 it follows that $\langle -c \rangle \nu \in w$. Now, w is a maximal $\mathbf{Cxt^{u,-}}$-consistent set and it contains thus formula $\langle -c \rangle \nu \rightarrow \neg \langle c \rangle \nu$ (i.e., axiom `Packing`). It follows that $\neg \langle c \rangle \nu \in w$ and it is therefore not the case that $wR_c^{\mathbf{Cxt^{u,-}}}w'$, which proves claim (3.b).

Lemma 10. *(Semantic equivalence for* $\textsc{Cxt}^{\top,\setminus}$ *frames)*
For any formula ϕ on $\mathcal{L}_n$: $\mathfrak{TE}^{\sim,\setminus} \models \phi$ iff $\textsc{Cxt}^{\top,\setminus} \models \phi$. That is, $\textsc{Cxt}^{\top,\setminus}$ *frames and* $\mathfrak{TE}^{\sim,\setminus}$ *frames define the same logic.*

Proof. The proof is analogous to the proof of Lemmata 7 and 8. From right to left: for every ϕ, $\textsc{Cxt}^{\top,\setminus} \models \phi$ implies $\mathfrak{TE}^{\sim,\setminus} \models \phi$. The results follow by the application of Proposition 6. From $W = W_c \cup W_{-c}$ for any atomic context identifier c, it follows that for every $w, w' \in W$, wR_uw' implies wR_cw' or $wR_{-c}w'$. And from $W_c \cap W_{-c} = \emptyset$ for any atomic context identifier c, it follows that for every $w, w' \in W$, $wR_{-c}w'$ implies not wR_cw'. From left to right: for every ϕ, $\mathfrak{TE}^{\sim,\setminus} \models \phi$ implies $\textsc{Cxt}^{\top,\setminus} \models \phi$. It suffices to show that every point-generated subframe of any $\mathfrak{TE}^{\sim,\setminus}$ frame is a $\textsc{Cxt}^{\top,\setminus}$ frame. The desired result follows then from Lemma 5. Consider a frame $\mathcal{F}^w \in g(\mathfrak{TE}^{\sim,\setminus})$ generated by state w. We show that $\mathcal{F}^w$ is a $\textsc{Cxt}^{\top,\setminus}$ frame. Building on the proofs of Lemma 7 and on the fact that $\mathfrak{TE}^{\sim,\setminus}$ already contain a universal relation, it just needs to be shown that for any atomic index c: (a) $W^w \subseteq r_c(w) \cup r_{-c}(w)$ and (b) $r_c(w) \cap r_{-c}(w) \subseteq \emptyset$. Both claims are straightforwardly proven by observing that for any atomic context index c and states $w', w'' \in W^w$: $w' R^w_u w''$ (i.e., $w'' \in W^w$) implies $w' R^w_c w''$ (i.e., $w'' \in r_c(w)$) or $w' R^w_{-c} w''$ (i.e., $w'' \in r_{-c}(w)$); and $w' R^w_c w''$ (i.e., $w'' \in r_c(w)$) implies not $w' R^w_{-c} w''$ (i.e., $w'' \notin r_{-c}(w)$).

Corollary 3. *(Completeness of* $\mathbf{Cxt}^{\mathbf{u},-}$ *w.r.t.* $\mathrm{CXT}^{\top,\setminus}$ *frames)*
Logic $\mathbf{Cxt}^{\mathbf{u},-}$ *is strongly complete w.r.t. the class of* $\mathrm{CXT}^{\top,\setminus}$ *frames.*

Proof. Follows directly from Theorem 6 and Lemma 10.

Acknowledgments

We would like to thank the anonymous reviewers of DEON’06, Prof. Andrew Jones for the inspiring discussions that motivated the research presented in this work, and Prof. Marek Sergot for the useful comments during the edition of DEON’06.

References

1. Searle, J.: Speech Acts. An Essay in the Philosophy of Language. Cambridge University Press, Cambridge (1969)
2. Searle, J.: The Construction of Social Reality. Free Press (1995)
3. Bulygin, E.: On norms of competence. Law and Philosophy 11 (1992) 201–216
4. Peczenik, A.: On Law and Reason. Kluwer, Dordrecht (1989)
5. Jones, A.J.I., Sergot, M.: Deontic logic in the representation of law: towards a methodology. Artificial Intelligence and Law 1 (1992)
6. Grossi, D., Meyer, J.-J.Ch., Dignum, F.: Modal logic investigations in the semantics of counts-as. In: Proceedings of the Tenth International Conference on Artificial Intelligence and Law (ICAIL’05), ACM (2005) 1–9
7. Grossi, D., Meyer, J.-J.Ch., Dignum, F.: Classificatory aspects of counts-as: An analysis in modal logic. Journal of Logic and Computation **16** (2006) 613–643 Oxford University Press.
8. Tarski, A.: The semantic conception of truth and the foundations of semantics. Philosophy and Phenomenological Research **4** (1944) 341–376
9. Grossi, D., Dignum, F., Meyer, J.-J.Ch.: Contextual taxonomies. In Leite, J., Toroni, P., eds.: Post-proceedings of CLIMA V, 5th International Workshop on Computational Logic in Multi-Agent Systems. LNAI 3487, Springer-Verlag (2005) 33–51
10. Grossi, D., Dignum, F., Meyer, J.-J.Ch.: Contextual terminologies. In Toni, F., Torroni, P., eds.: Post-proceedings of CLIMA VI, 6th International Workshop on Computational Logic in Multi-Agent Systems. LNAI 3900, Springer-Verlag (2006) 284–302
11. Stalnaker, R.: On the representation of context. In: Journal of Logic, Language, and Information. Volume 7., Kluwer (1998) 3–19
12. Ghidini, C., Giunchiglia, F.: Local models semantics, or contextual reasoning = locality + compatibility. Artificial Intelligence **127** (2001) 221–259
13. Ricciardi, M.: Constitutive rules and institutions. Paper presented at the meeting of the Irish Philosophical Club, Ballymascanlon (1997)
14. Gabbay, D., Kurucz, A., Wolter, F., Zakharyaschev, M.: Many-dimensional modal logics. Theory and applications. Elsevier (2003)

15. Buvač, S.V., Mason, I.A.: Propositional logic of context. Proceedings AAAI'93 (1993) 412–419
16. Buvač, S., Buvač, S.V., Mason, I.A.: The semantics of propositional contexts. Proceedings of the 8th ISMIS. LNAI-869 (1994) 468–477
17. Jones, A.J.I., Sergot, M.: A formal characterization of institutionalised power. Journal of the IGPL **3** (1996) 427–443
18. Gelati, J., Rotolo, A., Sartor, G., Governatori, G.: Normative autonomy and normative co-ordination: Declarative power, representation, and mandate. Artif. Intell. Law **12** (2004) 53–81
19. Boella, G., van der Torre, L.: Constitutive norms in the design of normative multiagent systems. In Toni, F., Torroni, P., eds.: CLIMA VI. LNAI 3900, Springer-Verlag (2005) 303–319
20. Tarski, A.: The establishment of scientific semantics. In: Logic, Semantics, Metamathematics. Hackett, Indianapolis (1983) 401–408
21. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge University Press, Cambridge (2001)
22. Passy, S., Tinchev, T.: An essay in combinatorial dynamic logic. Information and Computation **93** (1991) 263–332
23. Passy, S., Tinchev, T.: Pdl with data constants. Information Processing Letters **20** (1985) 35–41 Elsevier Science Publishers.
24. Levesque, H.J.: All I Know: A study in autoepistemic logic. Artificial Intelligence (1990) 263–309
25. Goranko, V., Passy, S.: Using the universal modality: Gains and questions. Journal of Logic and Computation **2** (1992) 5–30 Oxford University Press.
26. Areces, C., Blackburn, P., Marx, M.: The computational complexity of hybrid temporal logics. Logic Journal of the IGPL **8** (2000) 653–679 Oxford University Press.
27. Makinson, D.: On a fundamental problem of deontic logic. In McNamara, P., Prakken, H., eds.: Norms, Logics and Information Systems. New Studies in Deontic Logic and Computer Science. Volume 49 of Frontiers in Artificial Intelligence and Applications. IOS Press, Amsterdam (1999) 29–53
28. Smith, B.: Fiat objects. Topoi **2** (2001) 131–148
29. Goldman, A.I.: A Theory of Human Action. Princeton University Press, Princeton (1976)
30. Grossi, D., Meyer, J.J.C., Dignum, F.: Counts-as: Classification or constitution? an answer using modal logic. In Goble, L., Meyer, J.-J.Ch., eds.: Proceedings of DEON 2006. LNAI 4048 (2006) 115–130
31. Nayak, P.: Representing multiple theories. In: Proceedings of the 12th National Conference on Artificial Intelligence, Volume 2. Seattle, WA, USA, AAAI Press (1994) 1154–1160

On the Logic of Normative Systems*

Thomas Ågotnes[1], Wiebe van der Hoek[2], Juan A. Rodríguez-Aguilar[3], Carles Sierra[3], Michael Wooldridge[2]

[1] Department of Computer Engineering, Bergen University College
Norway
tag@hib.no

[2] Department of Computer Science, University of Liverpool
UK
{wiebe,mjw}@csc.liv.ac.uk

[3] Artificial Intelligence Research Institute IIIA, Spanish Council for Scientific Research CSIC, Spain
{jar,carles}@iiia.csic.es

Abstract. We introduce *Normative Temporal Logic* (NTL), a logic for reasoning about normative systems. NTL is a generalisation of the well-known branching-time temporal logic CTL, in which the path quantifiers A ("on all paths...") and E ("on some path...") are replaced by the indexed deontic operators O_η and P_η, where for example $\mathsf{O}_\eta\varphi$ means "φ is obligatory in the context of normative system η". After defining the logic, we give a sound and complete axiomatisation, and discuss the logic's relationship to standard deontic logics. We present a symbolic representation language for models and normative systems, and identify four different model checking problems, corresponding to whether or not a model is represented symbolically or explicitly, and whether or not we are given an interpretation for the normative systems named in formulae to be checked. We show that the complexity of model checking varies from P-complete up to EXPTIME-hard for these variations.

1 Introduction

Normative systems, or social laws, have been widely promoted as an approach to coordinating multi-agent systems [Shoham and Tennenholtz, 1996]. Crudely, a normative system defines a set of constraints on the behaviour of agents, corresponding to obligations, which may or may not be observed by agents. A number of formalisms have been proposed for reasoning about normative behaviour in multi-agent systems, typically based on deontic logic [Meyer and Wieringa, 1993]. However the computational properties of such formalisms – in particular, their use in the practical design and synthesis of normative systems and the complexity of reasoning with them – has received little attention. In this paper, we rectify this omission. We present Normative Temporal Logic (NTL), a logic for reasoning about normative systems, which is closely related to the well-known

[*] The content of this paper has previously appeared in Proc. IJCAI 2007.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/921

and widely-used branching time logic CTL [Emerson, 1990]. In NTL, the universal and existential path quantifiers of CTL are replaced by indexed deontic operators O_η and P_η, where $\mathsf{O}_\eta\varphi$ means that "φ is obligatory in the context of the normative system η", and $\mathsf{P}_\eta\varphi$ means "φ is permissible in the context of the normative system η". Here, φ is a temporal logic expression over the usual CTL temporal operators $\bigcirc$, $\diamondsuit$, $\square$, and $\mathcal{U}$ (every temporal operator must be preceded by a deontic operator, cf. CTL syntax), and η denotes a normative system. In NTL, obligations and permissions are thus, first, *contextualised* to a normative system η and, second, have a *temporal* dimension. It has been argued that the latter can help avoid some of the paradoxes of classical deontic logic. NTL generalises CTL because by letting $\eta_\emptyset$ denote the empty normative system, the universal path quantifier A can be interpreted as $\mathsf{O}_{\eta_\emptyset}$; much of the technical machinery developed for reasoning with CTL can thus be adapted for NTL [Emerson, 1990; Clarke *et al.*, 2000]. NTL is in fact a descendent of the *Normative* ATL (NATL) logic introduced in [Wooldridge and van der Hoek, 2005]: however, NTL is *much* simpler (and we believe more intuitive) than NATL, and we are able to present many more technical results for the logic: we first give a sound and complete axiomatisation, and then discuss the logic's relationship to standard deontic logics. We introduce a symbolic representation language for normative systems, and investigate the complexity of model checking for NTL, showing that it varies from P-complete in the simplest case up to EXPTIME-hard in the worst. We present an example to illustrate the approach, and present some brief conclusions.

2 Normative Temporal Logic

Kripke Structures: Let $\Phi = \{p, q, \ldots\}$ be a finite set of atomic *propositional variables*. A *Kripke structure* (over Φ) is a quad $\mathcal{K} = \langle S, S^0, R, V\rangle$, where: S is a finite, non-empty set of *states*, with $S^0 \subseteq S$ ($S^0 \neq \emptyset$) being the *initial states*; $R \subseteq S \times S$ is a total binary relation on S, which we refer to as the *transition relation*[1]; and $V : S \to 2^\Phi$ labels each state with the set of propositional variables true in that state. A *path* over R is an infinite sequence of states $\pi = s_0, s_1, \ldots$ which must satisfy the property that $\forall u \in \mathbb{N}$: $(s_u, s_{u+1}) \in R$. If $u \in \mathbb{N}$, then we denote by $\pi[u]$ the component indexed by u in π (thus $\pi[0]$ denotes the first element, $\pi[1]$ the second, and so on). A path π such that $\pi[0] = s$ is an *s-path.*

Normative Systems: In this paper, a normative system is *a set of constraints on the behaviour of agents in a system.* More precisely, a normative system defines, for every possible system transition, whether or not that transition is considered to be legal or not. Different normative systems may differ on whether or not a transition is legal. Formally, a normative system η (w.r.t. a Kripke structure $\mathcal{K} = \langle S, S^0, R, V\rangle$) is simply a subset of R, such that $R \setminus \eta$ is a total relation. The requirement that $R \setminus \eta$ is total is a *reasonableness* constraint: it prevents normative systems which lead to states with no successor. Let $N(R) = \{\eta \mid (\eta \subseteq R) \;\&\; (R \setminus \eta \text{ is total})\}$ be the set of normative systems over R. The intended

[1] A relation $R \subseteq S \times S$ is total iff $\forall s\, \exists s' : (s, s') \in R$.

interpretation of a normative system η is that $(s, s') \in \eta$ means transition (s, s') is forbidden in the context of η; hence $R \setminus \eta$ denotes the *legal* transitions of η. Since it is assumed η is reasonable, we are guaranteed that a legal outward transition exists for every state. If π is a path over R and η is a normative system over R, then π is *η-conformant* if $\forall u \in \mathbb{N}$, $(\pi[u], \pi[u+1]) \notin \eta$. Let $\mathcal{C}_\eta(s)$ be the set of η-conformant s-paths (w.r.t. some R).

Since normative systems are just *sets* (of disallowed transitions), we can *compare* them, to determine, for example, whether one is *more liberal* (less restrictive) than another: if $\eta \subset \eta'$, then η places fewer constraints on a system than η', hence η is more liberal. Notice that, assuming an *explicit* representation of normative systems, (i.e., representing a normative system η directly as a subset of R), checking such properties can be done in polynomial time. We can also operate on them with the standard set theoretic operations of union, intersection, etc. Taking the union of two normative systems η_1 and η_2 may yield (depending on whether $R \setminus (\eta_1 \cup \eta_2)$ is total) 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). Care must be taken when operating on normative systems in this way to ensure the resulting system is reasonable.

Syntax of NTL: The language of NTL is a generalisation of CTL: the only issue that may cause confusion is that, within this language, we refer explicitly to normative systems, which are *semantic* objects. We will therefore assume a stock of syntactic elements Σ_η which will denote normative systems. To avoid a proliferation of notation, we will use the symbol η both as a syntactic element for normative systems in the language, and the same symbol to denote the corresponding semantic object. An *interpretation* for symbols Σ_η with respect to a transition relation R is a function $I : \Sigma_\eta \to N(R)$. When R is a transition relation of Kripke structure $\mathcal{K}$ we say that I is an interpretation over $\mathcal{K}$. We will assume that the symbol $\eta_\emptyset$ always denotes the *emptyset* normative system, i.e., the normative system which forbids *no* transitions. Note that this normative system will be reasonable for *any* Kripke structure. Thus, we require that for all I: $I(\eta_\emptyset) = \emptyset$. The syntax of NTL is defined by the following grammar:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi \vee \varphi \mid \mathsf{P}_\eta \bigcirc \varphi \mid \mathsf{P}_\eta(\varphi \,\mathcal{U}\, \varphi) \mid \mathsf{O}_\eta \bigcirc \varphi \mid \mathsf{O}_\eta(\varphi \,\mathcal{U}\, \varphi)$$

where $p \in \Phi$ and $\eta \in \Sigma_\eta$. Sometimes we call α occurring in an expression $\mathsf{O}_\eta \alpha$ or $\mathsf{P}_\eta \alpha$ a *temporal formula* (although such an α is not a well-formed formula).

Semantic Rules: The semantics of NTL are given with respect to the satisfaction relation "$\models$". $\mathcal{K}, s \models_I \varphi$ holds when $\mathcal{K}$ is a Kripke structure, s is a state in $\mathcal{K}$, I an interpretation over $\mathcal{K}$, and φ a formulae of the language, as follows:

$\mathcal{K}, s \models_I \top$;
$\mathcal{K}, s \models_I p$ iff $p \in V(s)$ (where $p \in \Phi$);
$\mathcal{K}, s \models_I \neg\varphi$ iff not $\mathcal{K}, s \models_I \varphi$;
$\mathcal{K}, s \models_I \varphi \vee \psi$ iff $\mathcal{K}, s \models_I \varphi$ or $\mathcal{K}, s \models_I \psi$;
$\mathcal{K}, s \models_I \mathsf{O}_\eta \bigcirc \varphi$ iff $\forall \pi \in \mathcal{C}_{I(\eta)}(s) : \mathcal{K}, \pi[1] \models_I \varphi$;

$\mathcal{K}, s \models_I \mathsf{P}_\eta \bigcirc \varphi$ iff $\exists \pi \in \mathcal{C}_{I(\eta)}(s) : \mathcal{K}, \pi[1] \models_I \varphi$;
$\mathcal{K}, s \models_I \mathsf{O}_\eta(\varphi \,\mathcal{U}\, \psi)$ iff $\forall \pi \in \mathcal{C}_{I(\eta)}(s), \exists u \in \mathbb{N}$, s.t. $\mathcal{K}, \pi[u] \models_I \psi$ and $\forall v, (0 \leq v < u) : \mathcal{K}, \pi[v] \models_I \varphi$
$\mathcal{K}, s \models_I \mathsf{P}_\eta(\varphi \,\mathcal{U}\, \psi)$ iff $\exists \pi \in \mathcal{C}_{I(\eta)}(s), \exists u \in \mathbb{N}$, s.t. $\mathcal{K}, \pi[u] \models_I \psi$ and $\forall v, (0 \leq v < u) : \mathcal{K}, \pi[v] \models_I \varphi$

The remaining classical logic connectives ("$\wedge$", "$\rightarrow$", "$\leftrightarrow$") are assumed to be defined as abbreviations in terms of $\neg, \vee$, in the conventional manner. We write $\mathcal{K} \models_I \varphi$ if $\mathcal{K}, s_0 \models_I \varphi$ for all $s_0 \in S^0$, $\mathcal{K} \models \varphi$ if $\mathcal{K} \models_I \varphi$ for all I, and $\models \varphi$ if $\mathcal{K} \models \varphi$ for all $\mathcal{K}$. The remaining CTL temporal operators are defined:

$$\mathsf{O}_\eta \Diamond \varphi \equiv \mathsf{O}_\eta(\top \,\mathcal{U}\, \varphi) \quad \mathsf{P}_\eta \Diamond \varphi \equiv \mathsf{P}_\eta(\top \,\mathcal{U}\, \varphi)$$
$$\mathsf{O}_\eta \Box \varphi \equiv \neg \mathsf{P}_\eta \Diamond \neg \varphi \quad \mathsf{P}_\eta \Box \varphi \equiv \neg \mathsf{O}_\eta \Diamond \neg \varphi$$

Recalling that $\eta_\emptyset$ denotes the empty normative system, we obtain the conventional path quantifiers of CTL as follows: $\mathsf{A}\alpha \equiv \mathsf{O}_{\eta_\emptyset}\alpha$, $\mathsf{E}\alpha \equiv \mathsf{P}_{\eta_\emptyset}\alpha$.

Properties and Axiomatisation: The following Proposition makes precise the expected property that *a less liberal system has more obligations (and less permissions) than a more liberal system.*

Proposition 1. *Let $\mathcal{K}$ be a Kripke structure, I be an interpretation over $\mathcal{K}$, and $\eta_1, \eta_2 \in \Sigma_\eta$: If $I(\eta_1) \subseteq I(\eta_2)$ then $\mathcal{K} \models_I \mathsf{O}_{\eta_1}\varphi \rightarrow \mathsf{O}_{\eta_2}\varphi$ and $\mathcal{K} \models_I \mathsf{P}_{\eta_2}\varphi \rightarrow \mathsf{P}_{\eta_1}\varphi$.*

We now present a sound and complete axiomatisation for NTL and some of its variants. First, let NTL$^-$ be NTL without the empty normative system $\eta_\emptyset$. Formally, NTL$^-$ is defined exactly as NTL, except for the requirement that Σ_η contains the $\eta_\emptyset$ symbol and the corresponding restriction on interpretations. An axiom system for NTL$^-$, denoted $\vdash^-$, is defined by axioms and rules (Ax1)–(R2) in Figure 1. NTL$^-$ can be seen as a *multi-dimensional* variant of CTL, where there are several indexed versions of each path quantifier.

Going on to NTL, we add axioms (Obl) and (Perm) (Figure 1); the corresponding inference system is denoted $\vdash$. We then, have the following chain of implications in NTL (the second element in the chain is a variant of the deontic axiom discussed below). 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:

$$\vdash (\mathsf{A}\varphi \rightarrow \mathsf{O}_\eta\varphi) \qquad \vdash (\mathsf{O}_\eta\varphi \rightarrow \mathsf{P}_\eta\varphi) \qquad \vdash (\mathsf{P}_\eta\varphi \rightarrow \mathsf{E}\varphi)$$

Finally, let NTL$^+$ be the extension of NTL obtained by extending the logical language with propositions on the form $\eta \equiv \eta'$ and $\eta \sqsubset \eta'$ ($\sqsubseteq$ can then be defined), interpreted in the obvious way (e.g., $\mathcal{K}, s \models_I \eta \sqsubset \eta'$ iff $I(\eta) \subset I(\eta')$). An axiom system for NTL$^+$, denoted $\vdash^+$, is obtained from $\vdash^-$ by adding the schemes (Obl+) and (Perm+) (Figure 1).

(Ax1) All validities of propositional logic

(Ax2) $\mathsf{P}_\eta \Diamond \varphi \leftrightarrow \mathsf{P}_\eta(\top \,\mathcal{U}\, \varphi)$

(Ax2b) $\mathsf{O}_\eta \Box \varphi \leftrightarrow \neg \mathsf{P}_\eta \Diamond \neg \varphi$

(Ax3) $\mathsf{O}_\eta \Diamond \varphi \leftrightarrow \mathsf{O}_\eta(\top \,\mathcal{U}\, \varphi)$

(Ax3b) $\mathsf{P}_\eta \Box \varphi \leftrightarrow \neg \mathsf{O}_\eta \Diamond \neg \varphi$

(Ax4) $\mathsf{P}_\eta \bigcirc (\varphi \vee \psi) \leftrightarrow (\mathsf{P}_\eta \bigcirc \varphi \vee \mathsf{P}_\eta \bigcirc \psi)$

(Ax5) $\mathsf{O}_\eta \bigcirc \varphi \leftrightarrow \neg \mathsf{P}_\eta \bigcirc \neg \varphi$

(Ax6) $\mathsf{P}_\eta(\varphi \,\mathcal{U}\, \psi) \leftrightarrow (\psi \vee (\varphi \wedge \mathsf{P}_\eta \bigcirc \mathsf{P}_\eta(\varphi \,\mathcal{U}\, \psi)))$

(Ax7) $\mathsf{O}_\eta(\varphi \,\mathcal{U}\, \psi) \leftrightarrow (\psi \vee (\varphi \wedge \mathsf{O}_\eta \bigcirc \mathsf{O}_\eta(\varphi \,\mathcal{U}\, \psi)))$

(Ax8) $\mathsf{P}_\eta \bigcirc \top \wedge \mathsf{O}_\eta \bigcirc \top$

(Ax9) $\mathsf{O}_\eta \Box(\varphi \rightarrow (\neg \psi \wedge \mathsf{P}_\eta \bigcirc \varphi)) \rightarrow (\varphi \rightarrow \neg \mathsf{O}_\eta(\gamma \,\mathcal{U}\, \psi))$

(Ax9b) $\mathsf{O}_\eta \Box(\varphi \rightarrow (\neg \psi \wedge \mathsf{P}_\eta \bigcirc \varphi)) \rightarrow (\varphi \rightarrow \neg \mathsf{O}_\eta \Diamond \psi)$

(Ax10) $\mathsf{O}_\eta \Box(\varphi \rightarrow (\neg \psi \wedge (\gamma \rightarrow \mathsf{O}_\eta \bigcirc \varphi))) \rightarrow (\varphi \rightarrow \neg \mathsf{P}_\eta(\gamma \,\mathcal{U}\, \psi))$

(Ax10b) $\mathsf{O}_\eta \Box(\varphi \rightarrow (\neg \psi \wedge \mathsf{O}_\eta \bigcirc \varphi)) \rightarrow (\varphi \rightarrow \neg \mathsf{P}_\eta \Diamond \psi)$

(Ax11) $\mathsf{O}_\eta \Box(\varphi \rightarrow \psi) \rightarrow (\mathsf{P}_\eta \bigcirc \varphi \rightarrow \mathsf{P}_\eta \bigcirc \psi)$

(R1) If $\vdash \varphi$ then $\vdash \mathsf{O}_\eta \Box \varphi$ (generalization)

(R2) If $\vdash \varphi$ and $\vdash \varphi \rightarrow \psi$ then $\vdash \psi$ (modus ponens)

(Obl) $\mathsf{O}_{\eta_\emptyset} \alpha \rightarrow \mathsf{O}_\eta \alpha$

(Perm) $\mathsf{P}_\eta \alpha \rightarrow \mathsf{P}_{\eta_\emptyset} \alpha$

(Obl+) $\eta \sqsubseteq \eta' \rightarrow (\mathsf{O}_\eta \alpha \rightarrow \mathsf{O}_{\eta'} \alpha)$

(Perm+) $\eta \sqsubseteq \eta' \rightarrow (\mathsf{P}_{\eta'} \alpha \rightarrow \mathsf{P}_\eta \alpha)$

Fig. 1. The three systems NTL$^-$ ((Ax1)–(R2), derived from an axiomatisation of CTL); NTL ((Ax1)–(R2),(Obl),(Perm)); NTL$^+$ ((Ax1)–(R2),(Obl+),(Perm+)). α stands for a temporal formula.

Theorem 1 (Soundness and Completeness). *The inference mechanism* $\vdash^-$ *is sound and complete with respect to validity of* NTL$^-$ *formulas, i.e., for every formula* φ *in the language of* NTL$^-$*, we have* $\models \varphi$ *iff* $\vdash^- \varphi$*. The same holds for* $\vdash$ *with respect to formulas from* NTL *and* $\vdash^+$ *with respect to* NTL$^+$*.*

Proof. All three cases are proven by adjusting the technique presented in [Emerson, 1990]. For the NTL$^-$ *case, the tableau-based construction of [Emerson, 1990] immediately carries through: we will encounter, for every generated state, successors of different dimensions. For the case of* NTL*, which includes the symbol* $\eta_\emptyset$*, we have to add clauses corresponding to (Obl) and (Perm) to the construction of the closure cl*(φ) *of a formula* φ*: if* $\mathsf{O}_{\eta_\emptyset}\alpha$ *(respectively,* $\mathsf{P}_\eta \alpha$*) is in cl*(φ) *then also* $\mathsf{O}_\eta \alpha$ *(respectively,* $\mathsf{P}_{\eta_\emptyset}\alpha$*) should be in cl*($\varphi$)*. In the case of* NTL$^+$*, we have to close off cl*(φ) *under the implications of axioms (Obl+) and (Perm+).*

Going beyond NTL$^+$, we can impose further structure on Σ_η and its interpretations. For example, we can add unions and intersections of normative systems by requiring Σ_η to include symbols $\eta \sqcup \eta'$, $\eta \sqcap \eta'$ whenever it includes η and η', and require interpretations to interpret $\sqcup$ as set union and $\sqcap$ as set intersection. As discussed above, we must then further restrict interpretations such that $R \setminus (I(\eta_1) \cup I(\eta_2))$ always is total. This would give us a kind of calculus of normative systems. Let $\mathcal{K}$ be a Kripke structure and I be an interpretation with the mentioned properties:

$$\begin{array}{ll} \mathcal{K} \models_I \mathsf{P}_{\eta \sqcup \eta'}\varphi \rightarrow \mathsf{P}_\eta\varphi & \mathcal{K} \models_I \mathsf{P}_\eta\varphi \rightarrow \mathsf{P}_{\eta \sqcap \eta'}\varphi \\ \mathcal{K} \models_I \mathsf{O}_\eta\varphi \rightarrow \mathsf{O}_{\eta \sqcup \eta'}\varphi & \mathcal{K} \models_I \mathsf{O}_{\eta \sqcap \eta'}\varphi \rightarrow \mathsf{O}_\eta\varphi \end{array}$$

(all of which follow from Proposition 1). Having such a calculus allows one to reason about the composition of normative systems.

Relationship with Deontic Logic: The two main differences between the language of NTL and the language of conventional deontic logic (henceforth "deontic logic") are, first, *contextual* deontic operators allowing a formula to refer to several different normative systems, and, second, *temporal* operators. *All* deontic expressions in NTL refer to time: $\mathsf{P}_\eta \bigcirc \varphi$ ("it is permissible in the context of η that φ is true at the next time point"); $\mathsf{O}_\eta \square \varphi$ ("it is obligatory in the context of η that φ always will be true"); etc. Deontic logic contains no notion of time. In order to compare our temporal deontic statements with those of deontic logic we must take the temporal dimension to be implicit in the latter. Two of the perhaps most natural ways of doing that is to take "obligatory" ($\mathsf{O}\varphi$) to mean "*always* obligatory" ($\mathsf{O}_\eta \square \varphi$), or "obligatory at the *next point in time*" ($\mathsf{O}_\eta \bigcirc \varphi$), respectively, and similarly for permission. In either case, all the principles of *Standard Deontic Logic* (SDL) hold also for NDL, viz., $\mathsf{O}(\varphi \rightarrow \psi) \rightarrow (\mathsf{O}\varphi \rightarrow \mathsf{O}\psi)$ (K); $\neg\mathsf{O}\bot$ (D); and from φ infer $\mathsf{O}\varphi$ (N). The two mentioned temporal interpretations of the (crucial) deontic axiom D are (both NTL validities):

$$\neg\mathsf{O}_\eta \square \bot \text{ and } \neg\mathsf{O}_\eta \bigcirc \bot$$

With these translations, all of the most commonly discussed so-called paradoxes of deontic logic also holds in NTL. However, it has been argued (cf., e.g., [Meyer and Wieringa, 1993]) that one of the causes behind some of the instances of the paradoxes (particularly those involving contrary-to-duty obligations) is that the language of conventional deontic logic is too weak, and that by incorporating temporal operators some instances of the paradoxes can be avoided.

3 Symbolic Representations

In practice, explicit state representations of Kripke structures are rarely if ever used when reasoning about systems, because of the *state explosion problem*: given a system with n Boolean variables, the system will typically have 2^n states. Instead, practical reasoning tools provide *succinct*, *symbolic* representation languages for defining Kripke structures. We present such a language for defining models, and also introduce an associated symbolic language for defining normative systems.

A Symbolic Language for Models: We present the SIMPLE REACTIVE MODULES LANGUAGE (SRML), a "stripped down" version of Alur and Henzinger's REACTIVE

MODULES LANGUAGE (RML) [Alur and Henzinger, 1999], which was introduced in [Hoek *et al.*, 2006]. SRML represents the core of RML, with some "syntactic sugar" removed to simplify the presentation and semantics. The basic idea is to present a Kripke structure $\mathcal{K}$ by means of a number of symbolically represented agents, where the choices available to every agent are defined by a number of rules, defining which actions are available to the agent in every state; a transition (s, s') in $\mathcal{K}$ corresponds to a *tuple of actions, one for each agent in the system.* Here is an example of an agent definition in SRML (agents are referred to as "modules" in (S)RML):

```
module $toggle$ controls $x$
  init
  $\ell_1 : \top \leadsto x' := \top$
  $\ell_2 : \top \leadsto x' := \bot$
  update
  $\ell_3 : x \leadsto x' := \bot$
  $\ell_4 : (\neg x) \leadsto x' := \top$
```

This module, named *toggle*, controls a single Boolean variable, x. The choices available to the agent are defined by the `init` and `update` rules[2]. The `init` rules define the choices available to the agent with respect to the initialisation of its variables, while the `update` rules define the agent's choices subsequently. In this example, there are two `init` rules and two `update` rules. The `init` rules define two choices for the initialisation of variable x: assign it the value $\top$ or the value $\bot$. Both of these rules can fire initially, as their conditions ($\top$) are always satisfied; in fact, only one of the available rules will ever *actually* fire, corresponding to the "choice made" by the agent on that decision round. The effect of firing a rule is to execute the assignment statements on the r.h.s. of the rule, which modify the agent's controlled variables. (The "prime" notation for variables, e.g., x', means "the value of x afterwards".) Rules are identified by *labels* (ℓ_i); these labels do not form part of the original RML language, and in fact play no part in the semantics of SRML – they are used to identify rules in normative systems, as we shall see below. We assume a distinguished label "▯" for rules, which is used to identify rules that should never be made illegal by any normative system. With respect to `update` rules, the first rule says that if x has the value $\top$, then the corresponding action is to assign it the value $\bot$, while the second rule says that if x has the value $\bot$, then it can subsequently be assigned the value $\top$. In sum, the module non-deterministically chooses a value for x initially, and then on subsequent rounds toggles this value. In this example, the `init` rules are non-deterministic, while the `update` rules are deterministic. An SRML *system*, ρ, is a set of such modules, where the controlled variables of modules are mutually disjoint.

The Kripke structure $\mathcal{K}_\rho = \langle S_\rho, S^0_\rho, R_\rho, V_\rho \rangle$ corresponding to SRML system ρ is given as follows: the state set S_ρ and valuation function V_ρ corresponds to states (valuations of variables) that could be reached by ρ, with initial states S^0_ρ being states that could be generated by `init` rules; the transition relation R_ρ

[2] To be more precise, the rules are *guarded commands*.

is defined by $(s, s') \in R_\rho$ iff there exists a tuple of `update` rules, one for each module in the system, such that each rule is enabled in s and s' is obtained from executing this collection of rules on s.

A Symbolic Language for Normative Systems: We now introduce the SRML *Norm Language* (SNL) for representing normative systems, which corresponds to the SRML language for models. The general form of an SNL normative system definition is:

$$\begin{array}{l} \texttt{normative-system}\ id \\ \quad \chi_1\ \texttt{disables}\ \ell_{1_1},\ldots,\ell_{1_k} \\ \qquad \cdots \\ \quad \chi_m\ \texttt{disables}\ \ell_{m_1},\ldots,\ell_{m_k} \end{array}$$

Here, $id \in \Sigma_\eta$ is the name of the normative system; these names will be used to refer to normative systems in formulae of NTL. The body of the normative system is defined by a set of *constraint rules.* A constraint rule

$$\chi\ \texttt{disables}\ \ell_1,\ldots,\ell_k$$

consists of a condition part χ, which is a propositional logic formula over the variables of the system, and a set of rule labels $\{\ell_1,\ldots,\ell_k\}$ (we require $\square \notin \{\ell_1,\ldots,\ell_k\}$). If χ_i is satisfied in a particular state, then *any* SRML *rule with a label that appears on the r.h.s. of the constraint rule will be illegal in that state, according to this normative system.* An SNL *interpretation* is then simply a set of SNL normative systems, each with a distinct name.

Given SNL normative systems η_1 and η_2, for some SRML system ρ, we say: η_1 is *at least as liberal* as η_2 in system ρ if for every state $s \in S_\rho$, every rule that is legal according to η_2 is legal according to η_1; and they are *equivalent* if for every state $s \in S_\rho$, the set of rules legal according to η_1 and η_2 are the same.

Theorem 2. *The problem of testing whether* SNL *normative system* η_1 *is at least as liberal as* SNL *normative system* η_2 *is* PSPACE-*complete, as is the problem of testing equivalence of such systems.*

Proof. We do the proof for checking equivalence; the liberality case is similar. For membership of PSPACE, consider the complement problem: guess a state s, check that $s \in S_\rho$, (reachability of states in RML is in PSPACE [Alur and Henzinger,]) and check that there is some rule legal in s according to η_2 is not legal in s according to η_1, or vice versa. Hence the complement problem is in NPSPACE, and so the problem is in PSPACE. For PSPACE-hardness, we reduce the problem of propositional invariant checking over (S)RML modules [Alur and Henzinger,]. Given an SRML system ρ and propositional formula φ, define normative systems η_1 and η_2 as follows (where ℓ does not occur in ρ):

$$\begin{array}{ll} \texttt{normative-system}\ \eta_1 & \texttt{normative-system}\ \eta_2 \\ \quad \neg\varphi\ \texttt{disables}\ \ell & \qquad \bot\ \texttt{disables}\ \ell \end{array}$$

According to η_2, ℓ is always enabled; thus η_1 will be equivalent to η_2 iff φ holds across all reachable states of the system.

4 Model Checking

Model checking is an important computational problem for any modal or temporal logic [Clarke *et al.*, 2000]. We consider two versions of the model checking problem for NTL, depending on whether the model is presented explicitly or symbolically. For each of these cases, there are two further possibilities, depending on whether we are given an interpretation I for normative systems named in formulae or not. If we are given an interpretation for the normative systems named in the formula, then NTL model checking essentially amounts to a conventional verification problem: showing that, under the given interpretation, the model and associated normative systems have certain properties. However, the *uninterpreted* model checking problem corresponds to the *synthesis* of normative systems: we ask whether *there exist* normative systems that would have the desired properties.

Explicit State Model Checking: The *interpreted explicit state model checking problem* for NTL is as follows.

> Given a Kripke structure $\mathcal{K} = \langle S, S^0, R, V \rangle$, interpretation $I : \Sigma_\eta \to N(R)$ and formula φ of NTL, is it the case that $\mathcal{K} \models_I \varphi$?

The CTL model checking problem is P-complete [Schnoebelen, 2003]. The standard dynamic programming algorithm for CTL model checking may be easily adapted for interpreted explicit state NTL model checking, and has the same worst case time complexity. More interesting is the case where we are *not* given an interpretation. The *uninterpreted explicit state model checking problem* for NTL is as follows.

> Given a Kripke structure $\mathcal{K} = \langle S, S^0, R, V \rangle$ and formula φ of NTL, does there exist an interpretation $I : \Sigma_\eta \to N(R)$ such that $\mathcal{K} \models_I \varphi$?

Theorem 3. *The uninterpreted explicit state model checking problem for* NTL *is* NP*-complete.*

Proof. For membership in NP, simply guess an interpretation I and verify that $\mathcal{K} \models_I \varphi$. Since interpretations are polynomial in the size of the Kripke structure and formula, guessing can be done in (nondeterministic) polynomial time, and checking is the interpreted explicit state model checking problem. Hence the problem is in NP. For NP-hardness, we reduce SAT. Given SAT instance φ over variables $x_1, \ldots, x_k$, for each variable x_i, create two variables $t(x_i)$ and $f(x_i)$, and define a Kripke structure with $3k + 1$ states, as illustrated in Figure 2; state s_0 is the initial state, and state s_{3k} is a final state. Let φ^* denote the NTL formula obtained from φ by systematically replacing every variable x_i with $(\mathsf{P}_\eta \Diamond t(x_i))$. Define the formula to be model checked as:

$$\varphi^* \wedge \bigwedge_{1 \leq i \leq k} (\mathsf{P}_\eta \Diamond (t(x_i) \vee f(x_i))) \wedge \\ \bigwedge_{1 \leq i \leq k} (\mathsf{P}_\eta \Diamond t(x_i) \to \neg \mathsf{P}_\eta \Diamond f(x_i))(\mathsf{P}_\eta \Diamond f(x_i) \to \neg \mathsf{P}_\eta \Diamond t(x_i))$$

This formula is satisfied in the structure by some interpretation iff φ is satisfiable.

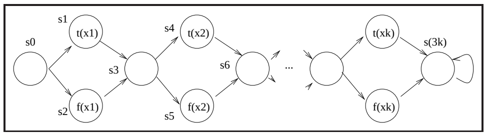


Fig. 2. Reduction for Theorem 3.

Symbolic Model Checking: As we noted above, explicit state model checking problems are perhaps of limited interest, since such representations are exponentially large in the number of propositional variables. Thus we now consider the SRML *model checking problem for* NTL. Again, we have two versions, depending on whether we are given an interpretation or not.

Theorem 4. *The interpreted* SRML *model checking problem for* NTL *is* PSPACE-*complete.*

Proof. PSPACE-hardness is by a reduction from the problem of propositional invariant verification for SRML [Alur and Henzinger,][3]. Given a propositional formula φ and an (S)RML system ρ, let $I = \{\eta_\emptyset\}$, and simply check whether $\mathcal{K}_\rho \models_I \mathsf{O}_{\eta_\emptyset} \square \varphi$. Membership of PSPACE is by adapting the CTL symbolic model checking algorithm of [Cheng, 1995].

Theorem 5. *The uninterpreted* SRML *model checking problem for* NTL *is* EXPTIME-*hard.*

Proof. By reduction from the problem of determining whether a given player has a winning strategy in the two-player game PEEK-G_4 [Stockmeyer and Chandra, 1979, p.158]. An instance of PEEK-G_4 is a quad $\langle X_1, X_2, X_3, \varphi \rangle$ where: X_1 and X_2 are disjoint, finite sets of Boolean variables – variables X_1 are under the control of agent 1, and X_2 are under the control of agent 2; $X_3 \subseteq (X_1 \cup X_2)$ are the variables true in the initial state of the game; and φ is a propositional formula over $X_1 \cup X_2$, representing the winning condition. The agents take try to make φ true, by taking it in turns to alter the value of at most one of their variables. The decision problem is to determine whether agent 2 has a winning strategy in a given game. The idea of the proof is to define an SRML system that such that the runs of the system correspond to plays of the given game instance, and then to define an NTL formula to be model checked, which names a normative system η, such that the transitions legal according to η correspond to a winning strategy for player 2. The construction of the SRML system follows that of the EXPTIME-completeness proof of ATL model checking in [Hoek *et al.*, 2006], with the difference that player 2's update rules are given labels (so that they may be disabled). The formula to model check then defines three properties: (i) if it is agent 2's turn, then according to η at most one of its possible moves is legal; (ii) all of agent 1's moves are legal according to η (i.e, agent 2 must win against

[3] In fact, the result of [Alur and Henzinger, 1999] is stated for RML, but the proof only makes use of features from SRML.

all of these); and (*iii*) the legal paths according to η must represent wins for agent 2.

5 Example: Traffic Norms

Consider a circular road, with two parallel lanes. Vehicles circulate on the two lanes clockwise. We consider two types of vehicles: cars, and ambulances. The road is discretised in a finite number of positions, each one represented as an instance of a proposition $at(lane\text{-}number, lane\text{-}position, vehicle\text{-}id)$. Thus $at(2, 5, car23)$ means agent $car23$ is on lane 2 at position 5 (lane 1 is the outer lane, lane 2 is the inner lane). We also refer to lane 1 as the left lane and to lane 2 as the right lane. At each time step, cars can either remain still or change their position by one unit, moving either straight or changing lane. Ambulances can remain still or change their position by one or two units, either straight or changing lanes at will. We are interested in normative systems that prevent crashes, and that permit ambulances take priority over private cars. So consider the following normative systems:

- η_1: Ambulances have priority over all other vehicles (i.e., cars stop to allow ambulances overtake them);
- η_2: Cars cannot use the rightmost (priority) lane;
- η_3: Vehicles have “right” priority (i.e., left lane has to give way to any car running in parallel on the right lane).

We modelled this scenario using an RML-based model checking system for ATL [Alur *et al.*, 2002]. Each vehicle is modelled as a module containing the rules that determine their physically legal movements, and global traffic control is modelled as a set of norms that constrain the application of certain rules. For example, here is the (somewhat simplified) definition of a car (we abuse notation to facilitate comprehension: for example addition and subtraction here are modulo-n operations, where n is the number of road positions, and the $at(\ldots)$ predicates are implemented as propositions):

```
module car-23 controls at(l,p,car-23)
init
  [] // initialise ...
update
 car-23-straight:
  at(l,p,car-23) &  not(at(l,p+1,car-1)) &
  ... & not(at(l,p+1,vehicle-n)) ->
   at(l,p+1,car-23)' := T, at(l,p,car-23)' := F;
 car-23-right:
  at(1,p,car-23) &  not(at(2,p+1,car-1)) &
  ... & not(at(2,p+1,vehicle-n)) ->
   at(2,p+1,car-23)' := T, at(1,p,car-23)' := F;
 car-23-left:
  at(2,p,car-23) & not(at(1,p+1,car-1)) &
  ... & not(at(1,p+1,vehicle-n)) ->
   at(1,p+1,car-23)' := T, at(2,p,car-23)' := F;
 car-23-still:
  T -> skip;
```

We can then define the norms described above using SNL; (again, we abuse notation somewhat in the interests of brevity; variables must be expanded out for each car and position, in the obvious way):

```
normative-system N1
 at(1,p,car-i) and at(1,p-1,amb-j) disables
  car-i-straight, car-i-left, car-i-right;

normative-system N2
 at(2,p,car-i) disables
  car-i-straight, car-i-still;
   at(1,p,car-i) disables car-i-right;

normative-system N3
    at(1,p,car-i) and at(2,p,car-j) disables
       car-i-straight, car-i-right;
```

Using a model checker, we can then evaluate properties of the system; e.g., if there is only one ambulance then we have $\mathsf{O}_{\eta_1 \cup \eta_2 \cup \eta_3} \Box \neg crash$.

6 Conclusions & Acknowledgments

Several issues present themselves for future work: tight bounds for complexity of uninterpreted symbolic model checking, the complexity of satisfiability, and a full implementation of a model checker encompassing the variations discussed in section 4 are the most obvious.

We gratefully acknowledge support from the Spanish research project Web(I)-2, TIC2003-08763-c02-00), the EU funded project OpenKnowledge, FP6-027253), the UK EPSRC project "Virtual Organisations for E-Science" and the Research Council of Norway (project 166525/V30).

References

Alur and Henzinger, . R. Alur and T. A. Henzinger. *Computer aided verification.* In press.

Alur and Henzinger, 1999. R. Alur and T. A. Henzinger. Reactive modules. *Form. Meth. Sys. Des.*, 15(11), 1999.

Alur *et al.*, 2002. R. Alur, T. A. Henzinger, and O. Kupferman. Alternating-time temporal logic. *JACM*, 49(5):672–713, 2002.

Cheng, 1995. A. Cheng. Complexity results for model checking. Tech. Rep. RS-95-18, Uni. Aarhus, 1995.

Clarke *et al.*, 2000. E. M. Clarke, O. Grumberg, and D. A. Peled. *Model Checking.* MIT Press 2000.

Emerson, 1990. E. A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Hand. of Theor. Comp. Sci. Vol. B*, pages 996–1072. Elsevier, 1990.

Hoek *et al.*, 2006. W. van der Hoek, A. Lomuscio, and M. Wooldridge. On the complexity of practical ATL model checking. In *Proc. AAMAS-2006*, 2006.

Meyer and Wieringa, 1993. J.-J. Ch. Meyer and R. J. Wieringa, eds. *Deontic Logic in Comp. Sci.*. Wiley, 1993.

Schnoebelen, 2003. P. Schnoebelen. The complexity of temporal logic model checking. In *Advances in Modal Logic Vol 4*, 2003.

Shoham and Tennenholtz, 1996. Y. Shoham and M. Tennenholtz. On social laws for artificial agent societies: Off-line design. In *Computational Theories of Interaction and Agency*. MIT Press, 1996.

Stockmeyer and Chandra, 1979. L. J. Stockmeyer and A. K. Chandra. Provably difficult combinatorial games. *SIAM Jnl of Comp.*, 8(2):151–174, 1979.

Wooldridge and van der Hoek, 2005. M. Wooldridge and W. van der Hoek. On obligations and normative ability. *Jnl Appl. Logic*, 4(3-4):396–420, 2005.

Prioritized Conditional Imperatives: Problems and a New Proposal[*]

Jörg Hansen

University of Leipzig, Institut für Philosophie
04107, Leipzig, Beethovenstraße 15, Germany
jhansen@uni-leipzig.de

Abstract. The sentences of deontic logic may be understood as describing what an agent ought to do when faced with a given set of norms. If these norms come into conflict, the best the agent can be expected to do is to follow a maximal subset of the norms. Intuitively, a priority ordering of the norms can be helpful in determining the relevant sets and resolve conflicts, but a formal resolution mechanism has been difficult to provide. In particular, reasoning about prioritized conditional imperatives is overshadowed by problems such as the 'order puzzle' that are not satisfactorily resolved by existing approaches. The paper provides a new proposal as to how these problems may be overcome.

Keywords: deontic logic, default logic, priorities, logic of imperatives

1 Drinking and Driving

Imagine you have been invited to a party. Before the event, you become the recipient of various imperative sentences:

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

Suppose that as a rule you do what your mother tells you – after all, she is the most important person in your life. Also, the last time you went to a party your best friend did the driving, so it really is your turn now. You can enjoy yourself without a drink, though it would be nice to have a drink with your acquaintance – your best friend would not mind if you had one drink, and your acquaintance does not care that you may be driving – but your mother would not approve of such a behavior. Making up your mind,

(4) You go to the party.

I think it is quite clear what you must do: obey your mother and your best friend, and hence do the driving and deny your acquaintance's request. However, it is not so clear what formal algorithm could explain this reasoning.

[*] I am grateful to Lou Goble, John F. Horty and Leon van der Torre for helpful comments and discussions in preparing this paper.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/914

An example of a similar form was first employed in epistemic logic,[1] and has been termed the 'order puzzle' (cf. Horty [22]). For the epistemic version, consider the following sentences:

(5) You remember from physics: if you are in a car, lightning won't strike you.
(6) The coroner tells you: he was struck by lightning.
(7) Your neighbor says: he must have been drinking and driving.

Suppose that driving includes being in a car, that you firmly believe in what you remember from physics, that you believe that information by medical officers is normally based on competent investigation, and that you usually don't question your neighbor's observations, but think that sometimes she is just speculating. It seems quite clear what happens: you keep believing what you remember from school, and don't doubt what the coroner told you, but question your neighbor's information, maybe answering: "This can't be true, as the authorities found he was struck by lightning, and you can't be struck by lightning in a car".

In both cases, the problem as to how the underlying reasoning can be formally reconstructed seems so far unsolved. Both involve a ranking, or priority ordering, of the sentences involved. Concentrating on the imperative side of things, in what follows, I will consider various proposals from the literature that have been put forward to explain the reasoning about such prioritized conditionals, discuss their strengths and weaknesses in relation to problems such as the one above, and finally propose a fresh solution that solves the problem.

2 Formal Preliminaries

To formally discuss problems such as the one presented above, I shall use a simple framework: let I be a set of objects, they are meant to be (conditional) imperatives. Two functions g and f associate with each imperative an antecedent and a consequent – these are sentences from the language of a basic logic that here will be the language $\mathscr{L}_{PL}$ of propositional logic.[2] $g(i)$ may be thought of as describing the 'grounds', or circumstances in which the consequent of i is to hold, and $f(i)$ as associating the sentence that describes what must be the case if the imperative i is satisfied, its 'deontic focus' or 'demand'.[3] In accordance with tradition (cf. Hofstadter and McKinsey [20]), I write $A \Rightarrow !B$ for an $i \in I$ with $g(i) = A$ and $f(i) = B$, and $!A$ means an unconditional imperative $\top \Rightarrow !A$. Note

[1] Cf. Rintanen [36] p. 234, who in turn credits Brewka with its invention.

[2] PL is based on a language $\mathscr{L}_{PL}$, defined from a set of proposition letters $Prop = \{p_1, p_2, ...\}$, Boolean connectives '$\neg$', '$\wedge$', '$\vee$', '$\rightarrow$', '$\leftrightarrow$' and brackets '(', ')' as usual. The truth of a $\mathscr{L}_{PL}$-sentence A is defined recursively using a valuation function $v : Prop \rightarrow \{1, 0\}$ (I write $v \models A$), starting with $v \models p$ iff $v(p) = 1$ and continuing as usual. If $A \in \mathscr{L}_{PL}$ is true for all valuations it is called a tautology. PL is the set of all tautologies, and this set is used to define provability, consistency and derivability (I write $\Gamma \vdash_{PL} A$) as usual. $\top$ is an arbitrary tautology, and $\bot$ is $\neg\top$.

[3] In analogy to Reiter's default logic one might add a third function e that describes exceptional circumstances in which the imperative is not to be applied. I will not address this additional complexity here.

that $A \Rightarrow !B$ is just the name for a conditional imperative that demands B to be made true in a situation where A is true – it is not an object that is assigned truth values. I write $m(i)$ for $\ulcorner g(i) \to f(i) \urcorner$ and call $m(i)$ the 'materialization' of i, as it represents the material implication that may be thought of as corresponding to the conditional imperative. For any $i \in I$ and $\Delta \subseteq I$, instead of $f(i)$, $g(i)$, $m(i)$, $f(\Delta)$, $g(\Delta)$ and $m(\Delta)$, I may use the superscripted i^f, i^g, i^m, Δ^f, Δ^g and Δ^m for better readability.

Let $\mathcal{I}$ be a tuple $\langle I, f, g \rangle$, let $W \subseteq \mathscr{L}_{PL}$ be a set of sentences, representing 'real world facts', and $\Delta \subseteq I$ be a subset of the imperatives: then we define

$$\mathit{Triggered}_{\mathcal{I}}(W, \Delta) =_{df} \{i \in \Delta \mid W \vdash_{PL} g(i)\}.$$

So an imperative $i \in \Delta$ is triggered if its antecedent is true given W. Tradition wants it that a conditional imperative can only be fulfilled or violated if its condition is the case.[4] So I define:

$$\mathit{Satisfied}_{\mathcal{I}}(W, \Delta) =_{df} \{i \in \Delta \mid W \vdash_{PL} i^g \wedge i^f\},$$
$$\mathit{Violated}_{\mathcal{I}}(W, \Delta) =_{df} \{i \in \Delta \mid W \vdash_{PL} i^g \wedge \neg i^f\},$$

An imperative in $\mathit{Satisfied}_{\mathcal{I}}(W, \Delta)$ [$\mathit{Violated}_{\mathcal{I}}(W, \Delta)$] is called satisfied [violated] given the facts W. It is of course possible that an imperative is neither satisfied nor violated given the facts W. If an imperative is triggered, but not violated, we call the imperative satisfiable:

$$\mathit{Satisfiable}_{\mathcal{I}}(W, \Delta) =_{df} \{i \in \mathit{Triggered}_{\mathcal{I}}(W, \Delta) \mid W \nvdash_{PL} \neg i^f\}.$$

Moreover, we define

$$\mathit{Obeyable}_{\mathcal{I}}(W, \Delta) =_{df} \{\Gamma \subseteq \Delta \mid \Gamma^m \cup W \nvdash_{PL} \bot\}.$$

So a subset Γ of Δ is obeyable given W iff it is not the case that for some $\{i_1, ..., i_n\} \subseteq \Gamma$ we have $W \vdash_{PL} (i_1^g \wedge \neg i_1^f) \vee ... \vee (i_n^g \wedge \neg i_n^f)$: otherwise we know that whatever we do, i.e. given any maxiconsistent subset V of $\mathscr{L}_{PL}$ that extends $W \subseteq V$, at least one imperative in Γ is violated.[5] We speak of a *conflict of imperatives* when the triggered imperatives cannot all be satisfied given the facts W, i.e. when $\mathit{Triggered}_{\mathcal{I}}(W, \Delta)^f \cup W \vdash_{PL} \bot$. More generally speaking I will also call imperatives conflicting if they are not obeyable in the given situation.

As prioritized conditional imperatives are our concern here, we let all imperatives in I be ordered by some priority relation $<\subseteq I \times I$. The relation $<$ is assumed to be a strict partial order on I, i.e. $<$ is irreflexive and transitive, and additionally we assume $<$ to be well-founded, i.e. infinite descending chains are excluded. For any $i_1, i_2 \in I$, $i_1 < i_2$ means that i_1 takes priority over i_2 (ranks higher than i_2, is more important than i_2, etc.). A tuple $\langle I, f, g \rangle$ will be called a *conditional imperative structure*, and $\langle I, f, g, < \rangle$ a *prioritized conditional imperative structure*. If all imperatives in I are unconditional, we may drop any reference to the relation g in the tuples and call these *basic imperative structures* and *prioritized imperative structures* respectively.

[4] Cf. Rescher [35], Sosa [40], van Fraassen [10]. Also cf. Greenspan [12]: "Oughts do not arise, it seems, until it is too late to keep their conditions from being fulfilled."

[5] Terms differ here, e.g. Downing [8] uses the term 'compliable' instead of 'obeyable'.

3 Deontic Concepts

Given a set of imperatives, one may truly or falsely state that their addressee must, or must not, perform some act or achieve some state of affairs according to what the addressee was ordered to do. For instance, in the 'drinking and driving' example from sec. 1 I think it is true that the agent ought to do the driving, as this is what the second-ranking imperative, uttered by the best friend, requires the agent to do, but that it would be false to say that the agent ought to drink and drive. Statements that something ought to be done or achieved are called 'normative' or 'deontic statements', and the ultimate goal is to find a logical semantics that models the situation and defines the deontic concepts in such a way that the formal results coincide with our natural inclinations in the matter.

3.1 Deontic operators for unconditional imperatives

For unconditional imperatives, such definitions are straightforward. Given a basic imperative structure $\mathcal{I} = \langle I, f \rangle$, a monadic deontic O-operator is defined by

$(td\text{-}m1)$ $\mathcal{I} \models OA$ if and only if (iff) $I^f \vdash_{PL} A$.

So obligation is defined in terms of what the satisfaction of all imperatives logically implies. With the usual truth definitions for Boolean operators, it can easily be seen that such a definition produces a normal modal operator, i.e. one that is defined by the following axiom schemes plus *modus ponens*:

(Ext) If $\vdash_{PL} A \leftrightarrow B$, then $OA \leftrightarrow OB$ is a theorem.
(M) $O(A \wedge B) \rightarrow (OA \wedge OB)$
(C) $(OA \wedge OB) \rightarrow O(A \wedge B)$
(N) $O\top$

Furthermore, $(td\text{-}m1)$ defines standard deontic logic *SDL*, which adds

(D) $OA \rightarrow \neg O \neg A$

iff the imperatives are assumed to be non-conflicting and so I^f is *PL*-consistent, i.e. $I^f \nvdash_{PL} \bot$. It is immediate that in the case of conflicts, $(td\text{-}m1)$ pronounces everything as obligatory, and in particular defines $O\bot$ true, thus making the impossible obligatory. If conflicts are not excluded, a solution is to only consider (maximal) subsets of the imperatives whose demands are consistent and define the O-operator with respect to these (I write $I \curlywedge \neg C$ for the set of all '$\neg C$-remainders', i.e. maximal subsets Γ of I such that $\Gamma^f \nvdash_{PL} \neg C$):

$(td\text{-}m2)$ $\mathcal{I} \models OA$ iff $\forall \Gamma \in I \curlywedge \bot : \Gamma^f \vdash_{PL} A$

Quite similarly, a dyadic deontic operator $O(A/C)$, meaning that A ought to be true given that C is true, can be defined with respect to the maximal subsets of imperatives that do not conflict in these circumstances:

$(td\text{-}d1)$ $\mathcal{I} \models O(A/C)$ iff $\forall \Gamma \in I \curlywedge \neg C : \Gamma^f \vdash_{PL} A$

So A is obligatory given that C is true if A is what the imperatives in any $\neg C$-remainder demand. In the case of conflicts, this definition produces a "disjunctive solution": e.g. if there are two imperatives $!A$ and $!B$ with $\vdash_{PL} C \rightarrow (A \rightarrow \neg B)$, then neither $O(A/C)$ nor $O(B/C)$ but $O(A \vee B/C)$ is true.[6]

[6] For alternative solutions to the problem of conflicts cf. Goble [11] and my [13], [14].

Often, we want to use the information that we have about the circumstances also for reasoning about the obligations in these circumstances. E.g. if the set of imperatives is $\{!(p_1 \vee p_2)\}$, ordering me to either send you a card or phone you, and I cannot send you a card, i.e. $\neg p_1$ is true, I should be able to conclude that I should phone you, and so $O(p_2/\neg p_1)$ should be true. Such 'circumstantial reasoning' is achieved by the following change to the truth definition:

(td-$d2$) $\quad \mathcal{I} \models O(A/C)$ iff $\forall \Gamma \in I \curlywedge \neg C : \Gamma^f \cup \{C\} \vdash_{PL} A$

With the usual truth conditions for Boolean operators, a semantics that employs (td-$d2$) has a sound and (weakly) complete axiom system PD that equals the system P of Kraus, Lehmann, Magidor [23], defined by these axiom schemes

(DExt) If $\vdash_{PL} A \leftrightarrow B$ then $O(A/C) \leftrightarrow O(B/C)$ is a theorem.
(DM) $O(A \wedge B/C) \rightarrow (O(A/C) \wedge O(B/C))$
(DC) $O(A/C) \wedge O(B/C) \rightarrow O(A \wedge B/C)$
(DN) $O(\top/C)$
(ExtC) If $\vdash_{PL} C \leftrightarrow D$ then $O(A/C) \leftrightarrow O(A/D)$ is a theorem.
(CCMon) $O(A \wedge D/C) \rightarrow O(A/C \wedge D)$
(CExt) If $\vdash_{PL} C \rightarrow (A \leftrightarrow B)$ then $O(A/C) \leftrightarrow O(B/C)$ is a theorem.
(Or) $O(A/C) \wedge O(A/D) \rightarrow O(A/C \vee D)$

with the additional (restricted) dyadic 'deontic' axiom scheme

(DD-R) If $\nvdash_{PL} \neg C$ then $\vdash_{PD} O(A/C) \rightarrow \neg O(\neg A/C)$

added (hence the name PD).[7]

3.2 Deontic operators for conditional imperatives

Unlike their unconditional counterparts, conditional imperatives have been found hard to reason about. G. H. von Wright [47] called conditional norms the "touchstone of normative logic", and van Fraassen [10] wrote with regard to logics for conditional imperatives: "There may be systematic relations governing this moral dynamics, but I can only profess ignorance of them."

Representing a conditional imperative as an unconditional imperative that demands a material conditional to be made true yields undesired results. Most notorious is the problem of contraposition: consider a set I with the only imperative $!(p_1 \rightarrow p_2)$, meaning e.g. 'if the police stops you, show your drivers licence'. (td-$d1$) makes true $O(p_2/p_1)$, but also $O(\neg p_1/\neg p_2)$, so if you can't present your drivers licence (you don't have one) you must see to it that the police does not stop you, which is hardly what the speaker meant you to do.One may think that such problems arise from the fact that antecedents of conditional imperatives often describe states of the affairs that the agent is not supposed to, and often cannot, control. But consider the set $\{!(p_1 \rightarrow p_2), !(\neg p_1 \rightarrow p_3)\}$, it yields $O(p_2/\neg p_3)$ with (td-$d1$). Here, p_2 is what the consequent of some imperative demands, so it supposedly describes something the agent can control. Now let

[7] For proofs, and an additional "credulous ought" that defines $O(A/C)$ true if the truth of A is required to satisfy all imperatives in *some* $\neg C$-remainder, cf. my [14].

the imperatives be interpreted as ordering me to wear my best suit if it does not rain, and a rain coat if it does: it is clear nonsense that I am obliged to wear a raincoat given that I can't wear my best suit (e.g. it is in the laundry). Such problems are the reason why we cautiously use special models for conditional imperatives (i.e. conditional imperative structures), and write $p_1 \Rightarrow !p_2$ instead of $!(p_1 \rightarrow p_2)$. But this only delegates the problem from the level of representation to that of semantics, where now new truth definitions must be found.

Let $\mathcal{I} = \langle I, f, g \rangle$ be a conditional imperative structure, and let us ignore for the moment the further complication of possible conflicts between imperatives. Then the following seems a natural way to define what ought to be the case in circumstances where C is assumed to be true:

$$(\textit{td-cd}1) \quad \mathcal{I} \models O(A/C) \ \text{ iff } \ [\textit{Triggered}_{\mathcal{I}}(\{C\}, I)]^f \vdash_{PL} A$$

So dyadic obligation is defined in terms what is necessary to satisfy all imperatives that are triggered in the assumed circumstances. E.g. if $I = \{p_1 \Rightarrow !p_2\}$, with its only imperative interpreted as "if you have a cold, stay in bed", then $O(p_2/p_1)$ truly states that I must stay in bed given that I have a cold.

Like in the unconditional case, it seems important to be able to use 'circumstantial reasoning', i.e. employ the information about the situation not only to determine if an imperative is triggered, but also for reasoning with its consequent. E.g. if the set of imperatives is $\{p_1 \Rightarrow !(p_2 \vee p_3)\}$, with its imperative interpreted as expressing "if you have a cold, either stay in bed or wear a scarf", one would like to obtain $O(p_3/p_1 \wedge \neg p_2)$, expressing that given that I have a cold and don't stay in bed, I must wear a scarf. So ($\textit{td-cd}1$) may be changed into

$$(\textit{td-cd}2) \quad \mathcal{I} \models O(A/C) \ \text{ iff } \ [\textit{Triggered}_{\mathcal{I}}(\{C\}, I)]^f \cup \{C\} \vdash_{PL} A.$$

Though the step from ($\textit{td-cd}1$) to ($\textit{td-cd}2$) seems quite reasonable, such definitions have also been criticized for defining the assumed circumstances as obligatory. E.g. if the set of imperatives is $\{p_1 \Rightarrow !p_2\}$, where the imperative is interpreted as expressing "if you hit someone, apologize to him", then (td-5) makes true $O(p_1 \wedge p_2/p_1)$, and hence also $O(p_1/p_1)$, so given that I hit someone, this is something I ought to do. The criticism looses much of its edge in the present setting, where one can point to the distinction between imperatives (there is no imperative that demands p_1) and ought sentences that describe what must be true when all triggered imperatives are satisfied in the supposed circumstances: then the truth of $O(p_1/p_1)$ seems no more paradoxical than the truth of $O\top$ that is accepted in most systems of deontic logic.

3.3 Further modifications

In Makinson & van der Torre's [25] more general theory of 'input/output logic', ($\textit{td-cd}1$) is termed 'simple-minded output', and ($\textit{td-cd}2$) is its 'throughput version'.[8]As the names suggests, the authors also discuss more refined operations, which again might be considered for reasoning about conditional imperatives. One modification addresses the possibility of 'reasoning by cases' that e.g. makes

[8] If I resembles the generating set G of input/output logic, then $O(A/C)$ means that A is an output given the input C (Makinson & van der Torre write $A \in \textit{out}(G, \{C\})$).

true $O(p_2 \vee p_4/p_1 \vee p_3)$ for a set of imperatives $I = \{p_1 \Rightarrow !p_2, p_3 \Rightarrow !p_4\}$. This may be achieved by the following definition, where $\mathscr{L}_{PL} \bot \neg C$ is the set of all maximal subsets of the language $\mathscr{L}_{PL}$ that are consistent with C:[9]

$(td\text{-}cd3)\quad \mathcal{I} \models O(A/C)$ iff $\forall V \in \mathscr{L}_{PL} \bot \neg C : [\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \vdash_{PL} A$

In the example, each set $V \subset \mathscr{L}_{PL}$ that is maximally consistent with $p_1 \vee p_3$ either contains p_1, then $p_1 \Rightarrow !p_2$ is triggered and so p_2 and also $p_2 \vee p_4$ is implied by $[\mathit{Triggered}_{\mathcal{I}}(V, I)]^f$, or it contains $\neg p_1$, but then it cannot also contain $\neg p_3$ and so must contain p_3, so $p_3 \Rightarrow !p_4$ is triggered and therefore p_4 and also $p_2 \vee p_4$ implied, so for all sets V, $p_2 \vee p_4$ is implied and so $O(p_2 \vee p_4/p_1 \vee p_3)$ made true.

In order to add 'circumstantial reasoning' to (td-$cd3$) – or, in Makinson & van der Torre's terms, for its 'throughput version' –, one might, in the vein of (td-$d2$) and (td-$cd2$), try this definition:

$(td\text{-}cd4^-)\ \mathcal{I} \models O(A/C)$ iff $\forall V \in \mathscr{L}_{PL} \bot \neg C : [\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \cup \{C\} \vdash_{PL} A$

But the definition seems too weak. Consider the set $\{p_1 \Rightarrow !(\neg p_2 \vee p_4), p_3 \Rightarrow !p_4\}$ and the situation $(p_1 \wedge p_2) \vee p_3$. We would expect a reasoning as follows: in this situation, either $p_1 \wedge p_2$ is true, so the first imperative is triggered but we cannot satisfy it by bringing about $\neg p_2$, and so must bring about p_4. Or p_3 is true, then the second imperative is triggered and we must again bring about p_4. So we must bring about p_4 in the given situation. But the definition fails to make true $O(p_4/(p_1 \wedge p_2) \vee p_3)$. Like Makinson and van der Torre [25], I therefore combine reasoning by cases with a stronger version of throughput:

$(td\text{-}cd4)\quad \mathcal{I} \models O(A/C)$ iff $\forall V \in \mathscr{L}_{PL} \bot \neg C : [\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \cup V \vdash_{PL} A$

As is easy to see, this resolves the difficulty: for $\{p_1 \Rightarrow !(\neg p_2 \vee p_4), p_3 \Rightarrow !p_4\}$, $O(p_4/(p_1 \wedge p_2) \vee p_3)$ is now true, as desired. However, this modification has a surprising consequence: it makes the reasoning about conditional imperatives collapse into reasoning about consequences of their materializations:

Observation 1 *By (td-cd4), $\mathcal{I} \models O(A/C)$ iff $m(I) \cup \{C\} \vdash_{PL} A$.*

Proof. For the right-to-left direction, for any imperative $i \in I$ and any set $V \in \mathscr{L}_{PL} \bot \neg C$, either V includes $g(i)$, so $i \in \mathit{Triggered}_{\mathcal{I}}(V, I)$ and therefore $[\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \vdash_{PL} g(i) \rightarrow f(i)$, or it does not include $g(i)$, but then it includes $\neg g(i)$ by maximality, hence $V \vdash_{PL} g(i) \rightarrow f(i)$. So $[\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \cup V \vdash_{PL} g(i) \rightarrow f(i)$. For the left-to-right direction, if $m(I) \cup \{C\} \nvdash_{PL} A$ then $m(I) \cup \{C\} \cup \{\neg A\}$ is consistent, so there is a $V \in \mathscr{L}_{PL} \bot \neg C$ such that $m(I) \cup \{C\} \cup \{\neg A\} \subseteq V$. It is immediate that for each $i \in \mathit{Triggered}_{\mathcal{I}}(V, I)$, $m(I) \cup V \vdash_{PL} f(i)$, so if $[\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \cup V \vdash_{PL} A$ then $m(I) \cup V \vdash_{PL} A$ and since $m(I) \subseteq V$ also $V \vdash_{PL} A$. Since V was consistent and included $\neg A$, it cannot also derive A, and so by contraposition $[\mathit{Triggered}_{\mathcal{I}}(V, I)]^f \cup V \nvdash_{PL} A$.

But such an equivalence makes all the problems discussed above for identifying conditional imperatives with unconditional imperatives that demand their mate-

[9] Makinson & van der Torre's [25] call the resulting operator 'basic output', of which a syntactical version was first presented by Świrydowicz [41] p. 32.

rializations reappear, in particular the problem of contraposition.[10] So it seems we must choose between 'reasoning by cases' and 'circumstantial reasoning'.[11]

Another modification that these authors consider is that of 'reusable output': when an imperative is triggered that demands A, and A is the trigger for some imperative $A \Rightarrow !B$, then we also ought to do B. Such a modification can easily be incorporated into a truth definition and its 'throughput' version:

$$(\textit{td-cd}5) \quad \mathcal{I} \models O(A/C) \;\text{ iff }\; [\textit{Triggered}^*_{\mathcal{I}}(\{C\}, I)]^f \vdash_{PL} A$$
$$(\textit{td-cd}6) \quad \mathcal{I} \models O(A/C) \;\text{ iff }\; [\textit{Triggered}^*_{\mathcal{I}}(\{C\}, I)]^f \cup \{C\} \vdash_{PL} A$$

where $\textit{Triggered}^*_{\mathcal{I}}(W, \Gamma)$ means the smallest subset of $\Gamma \subseteq I$ such that for all $i \in \Gamma$, if $[\textit{Triggered}^*_{\mathcal{I}}(W, \Gamma)]^f \cup W \vdash_{PL} g(i)$ then $i \in \textit{Triggered}^*_{\mathcal{I}}(W, \Gamma)$. Moreover, the two modifications of 'reasoning by cases' and 'reusable output' can be combined to produce the following definition and its 'throughput' variant:

$$(\textit{td-cd}7) \quad \mathcal{I} \models O(A/C) \;\text{ iff }\; \forall V \in \mathscr{L}_{PL} \bot \neg C : [\textit{Triggered}^*_{\mathcal{I}}(V, I)]^f \vdash_{PL} A$$
$$(\textit{td-cd}8) \quad \mathcal{I} \models O(A/C) \;\text{ iff }\; \forall V \in \mathscr{L}_{PL} \bot \neg C : [\textit{Triggered}^*_{\mathcal{I}}(V, I)]^f \cup V \vdash_{PL} A$$

The topic of 'reusable output' is discussed under the name of 'deontic detachment' in the literature on deontic logic, and there is no agreement whether such a procedure is admissible (Makinson [24] p. 43 argues in favor, whereas Sven Ove Hansson [17] p. 155 disagrees). E.g. let $I = \{!p_1, p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !\neg p_2\}$, and for its interpretation assume that it is imperative for the proper execution of your job that you develop novel methods, which make you eligible for a bonus, and that if you develop such novel methods you owe it to yourself to apply for the bonus, but that if you don't develop such methods you must not apply for the bonus. Truth definitions that accept 'deontic detachment' make true $O(p_2/\top)$, and so tell us that you ought to apply for the bonus, which seems weird since it may be that you never invent anything. However, proponents of deontic detachment may argue that in such a situation, $O(p_1 \wedge p_2/\top)$ should hold, i.e. you ought to invent new methods *and* apply for the bonus, and that the reluctance to also accept $O(p_2/\top)$ is – like the inference from "you ought to put on your parachute and jump" to "you ought to jump" – just a variant of Ross' Paradox that is usually considered harmless.

For $(\textit{td-cd}7)$ we once again obtain $O(p_2/\neg p_3)$ for $I = \{p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !p_3\}$: for any $V \in \mathscr{L}_{PL} \bot p_3$, $\neg p_3 \in V$, furthermore either $p_1 \in V$ and so $p_1 \Rightarrow !p_2 \in \textit{Triggered}^*_{\mathcal{I}}(V, I)$, or $\neg p_1 \in V$, then $\neg p_1 \Rightarrow !p_3 \in \textit{Triggered}^*_{\mathcal{I}}(V, I)$, and since $\{p_3\} \cup$

[10] $(\textit{td-cd}4^-)$ does not fare much better: though it does not include contraposition, it again makes $O(p_2/\neg p_3)$ true for $I = \{p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !p_3\}$, which is counterintuitive.

[11] Legal use of 'reasoning by cases', or *Wahlfeststellung*, is controversial. It means that if the defendant either committed crime α or crime β, the defendant would be convicted according to the milder law. A proponent would argue that since the defendant committed a crime (though it remains open which), justice demands that he should not go free, while the defense would argue that this violates the *in dubio pro reo* principle, since neither charge is sufficiently proved. After a *Reichsgericht* ruling in 1934 allowed *Wahlfeststellung* for cases in which the crimes in question were 'ethically and psychologically equivalent', the national-socialist lawmakers introduced a law prescribing its unrestricted application in 1935, considered ideological and lifted again by the Allied Control Council of Germany in 1946 (cf. [43]).

$\{\neg p_3\} \vdash_{PL} p_1$, again $p_1 \Rightarrow !p_2$ is in $\mathit{Triggered}^*_{\mathcal{I}}(V, I)$, hence $[\mathit{Triggered}^*_{\mathcal{I}}(V, I)]^f \vdash_{PL} p_2$ for all $V \in \mathscr{L}_{PL} \perp p_3$. But as we saw above, when interpreting the imperatives as 'if it rains, wear a raincoat' and 'if it does not rain, wear your best jacket', this result seems counterintuitive.[12] Note that (*td-cd*8) is again equivalent to $\mathcal{I} \models O(A/C)$ iff $m(I) \cup \{C\} \vdash_{PL} A$ and thus to (*td-cd*4) (cf. Makinson & van der Torre [25] observation 16; [26], p. 156):

Observation 2 *By* (*td-cd*8), $\mathcal{I} \models O(A/C)$ *iff* $m(I) \cup \{C\} \vdash_{PL} A$.

Proof. Similar to the proof of observation 1. For the left-to-right direction, use that for each $i \in \mathit{Triggered}^*_{\mathcal{I}}(V, I)$, $m(I) \cup V \vdash_{PL} f(i)$, which is immediate.

3.4 Operators for prioritized conditional imperatives

This paper focuses on prioritized conditional imperatives, and for these there is a further hurdle to finding the proper truth definitions for deontic concepts. Priorities are only required if the imperatives cannot all be obeyed – otherwise there is no reason not to obey all, and the priority ordering is not used. So the truth definitions must be able to deliver meaningful results for possibly conflicting imperatives. The intuitive idea is to use the information in the ordering to choose subsets of the set of imperatives under consideration that contain only the more important imperatives and leave out less important, conflicting ones, so that the resulting 'preferred subset' (or rather, subsets, since the choice may not always be determined by the ordering) only contains imperatives that do not conflict in the given situation. More generally, let $\mathcal{I}$ be a prioritized conditional imperative structure $\langle I, g, f, < \rangle$, and let Δ be a subset of I. Then $\mathscr{P}_{\mathcal{I}}(W, \Delta)$ contains just the subsets of Δ that are thus preferred given the world facts W. The above truth definitions can then be adapted such that they now describe something as obligatory iff it is so with respect to all the preferred subsets of the imperatives, i.e. they take on the following forms:

$$
\begin{array}{lr}
 & \mathcal{I} \models O(A/C) \;\text{ iff }\; \forall \Gamma \in \mathscr{P}_{\mathcal{I}}(\{C\}, I) : \\
(\mathit{td\text{-}pcd}1) & [\mathit{Triggered}_{\mathcal{I}}(\{C\}, \Gamma)]^f \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}2) & [\mathit{Triggered}_{\mathcal{I}}(\{C\}, \Gamma)]^f \cup \{C\} \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}3) & \forall V \in \mathscr{L}_{PL} \perp \neg C : [\mathit{Triggered}_{\mathcal{I}}(V, \Gamma)]^f \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}4) & \forall V \in \mathscr{L}_{PL} \perp \neg C : [\mathit{Triggered}_{\mathcal{I}}(V, \Gamma)]^f \cup V \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}5) & [\mathit{Triggered}^*_{\mathcal{I}}(\{C\}, \Gamma)]^f \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}6) & [\mathit{Triggered}^*_{\mathcal{I}}(\{C\}, \Gamma)]^f \cup \{C\} \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}7) & \forall V \in \mathscr{L}_{PL} \perp \neg C : [\mathit{Triggered}^*_{\mathcal{I}}(V, \Gamma)]^f \vdash_{PL} A\,, \\
(\mathit{td\text{-}pcd}8) & \forall V \in \mathscr{L}_{PL} \perp \neg C : [\mathit{Triggered}^*_{\mathcal{I}}(V, \Gamma)]^f \cup V \vdash_{PL} A\,.
\end{array}
$$

So e.g. (*td-pcd*1) defines A as obligatory if the truth of A is required to satisfy the triggered imperatives in any preferred subset. Of course, the crucial and as yet missing element is the decision procedure that determines the set $\mathscr{P}_{\mathcal{I}}(\{C\}, I)$ of preferred subsets. The next section discusses several proposals to define such subsets; a new proposal is presented in the section that follows it.

[12] With respect to their out_4-operation that corresponds to (*td-cd*7), Makinson & van der Torre [25] speak of a 'ghostly contraposition'.

4 Identifying the Preferred Subsets

4.1 Brewka's preferred subtheories

The idea that normative conflicts can be overcome by use of a priority ordering of the norms involved dates back at least to Ross [37] and is also most prominent in von Wright's work (cf. [45] p. 68, 80). However, it has turned out to be difficult to determine the exact mechanism by which such a resolution of conflicts can be achieved. This is true even when only unconditional imperatives are considered, and when special problems are left out of the picture, such that the ordering itself might be dependent on the facts (e.g. when the command of an officer in the field may override that of her superior due to unexpected circumstances), or be the subject of normative regulation (e.g. when we are commanded to obey the law of God more than the law of man). Discussing various proposals for resolution of conflicts between unconditional imperative, I have argued in [15] that an 'incremental' definition be used for determining the relevant subsets. Based on earlier methods by Rescher [34], such a definition was first introduced by Brewka [4] for reasoning with prioritized defaults. For any priority relation $<$, the idea is to consider all the 'full prioritizations' $\prec$ of $<$ (strict well orders that preserve $<$), and then work ones way from top to bottom by adding the $\prec$-next-higher imperative to the thus constructed 'preferred subtheory' if its demand is consistent with the given facts and the demands of the imperatives that were added before. For the present setting, the definition can be given as follows:

Definition 1 (Brewka's preferred subtheories).
Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, Δ be a subset of I, and $W \subseteq \mathscr{L}_{PL}$ be a set of PL-sentences. Then $\Gamma \in \mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, \Delta)$ iff (i) *$W \nvdash_{PL} \bot$, and* (ii) *Γ is obtained from a full prioritization $\prec$ by defining*

$$\Gamma_{[\prec\downarrow i]} = \begin{cases} \bigcup_{j \,\prec\, i} \Gamma_{[\prec\downarrow j]} \cup \{i\} & \textit{if } \; W \cup [\bigcup_{j \,\prec\, i} \Gamma_{[\prec\downarrow j]} \cup \{i\}]^f \nvdash_{PL} \bot, \textit{ and} \\ \bigcup_{j \,\prec\, i} \Gamma_{[\prec\downarrow j]} & \textit{otherwise,} \end{cases}$$

for any $i \in \Delta$, and letting $\Gamma = \bigcup_{i \in \Delta} \Gamma_{[\prec\downarrow i]}$.

Clause (i) ensures that for an inconsistent set of assumed 'facts', no set is preferred. Somewhat roundabout, owed to the possibility of infinite ascending subchains, clause (ii) then recursively defines a set $\Gamma \in \mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, \Delta)$ for each full prioritization $\prec$: take the $\prec$-first i (the exclusion of infinite descending subchains guarantees that it exists) and if $W \cup \{i^f\} \nvdash_{PL} \bot$ then let $\Gamma_{[\prec\downarrow i]} = \{i\}$; otherwise $\Gamma_{[\prec\downarrow i]}$ is left empty.[13] Similarly, any $\prec$-later i is tested for possible addition to the set $\bigcup_{j \,\prec\, i} \Gamma_{[\prec\downarrow j]}$ of elements that were added in the step for a $j \in \Delta$ that occurs $\prec$-prior to i. Γ is then the union of all these sets.

To see how this definition works, consider the set $I = \{!(p_1 \vee p_2), !\neg p_2, !\neg p_1\}$, with the ranking $!(p_1 \vee p_2) < !\neg p_1$ and $!\neg p_2 < !\neg p_1$. For an interpretation, let $!(p_1 \vee p_2))$ be your mother's request that you buy cucumbers or spinach for dinner, $!\neg p_1$ be your father's wish that no cucumbers are bought, and $!\neg p_2$ your

[13] As usual, the union of an empty set of sets is taken to be the empty set.

sister's desire that you don't buy any spinach. We have two full prioritizations $!(p_1 \vee p_2) < !\neg p_2 < !\neg p_1$ and $!\neg p_2 < !(p_1 \vee p_2) < !\neg p_1$ – let these be termed $\prec_1$ and $\prec_2$, respectively. The construction for $\prec_1$ adds the imperative $!(p_1 \vee p_2)$ in the first step and, since no conflict with the situation arises, $!\neg p_2$ in the second step. In the third and last step, nothing is added since $!\neg p_1$ conflicts with the demands of the already added imperatives. For $\prec_2$ the only difference is that the first two imperatives are added in inverse order. Thus $\mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, I) = \{\{!(p_1 \vee p_2), !\neg p_2\}\}$. Using (*td-pcd*2) we obtain $O(p_1 \wedge \neg p_2/\top)$, which means that you have to buy spinach and not cucumbers, thus fulfilling your parents' requests but not your sister's, which seems reasonable.

As I showed in [15], Brewka's method is extremely successful for dealing with unconditional imperatives. It is provably equivalent for such imperatives to methods proposed by Ryan [38] and Sakama & Inoue [39], and it avoids problems of other approaches by Alchourrón & Makinson [2], Prakken [31] and Prakken & Sartor [32]. Moreover, an equally intuitive maximization method proposed by Nebel [29], [30], that adds first a maximal number of the highest-ranking imperatives, then a maximal number of the second-ranking imperatives, etc., but for its construction requires the ordering to be based on a complete preorder, can be shown to be embedded in Brewka's approach for such orderings. So my aim will be to retain Brewka's method for the unconditional case. However, when it is applied without change to conditional imperatives, the algorithm may lead to incorrect results. E.g. consider a set I with two equally ranking imperatives $\{p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !\neg p_2\}$, meaning e.g. "if you go out, wear your boots" and "if you don't go out, don't wear your boots". Since the consequents contradict each other, an unmodified application of Brewka's method produces $\mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(\{p_1\}, I) = \{\{p_1 \Rightarrow !p_2\}, \{\neg p_1 \Rightarrow !\neg p_2\}\}$, which fails to make true $O(p_2/p_1)$ by any truth definition (*td-pcd*1-8): the right set contains no imperatives that are in any way triggered by p_1. So we cannot derive that you ought to wear your boots, given that you are going out. But intuitively there is no conflict, since the conflicting obligations arise in mutually exclusive circumstances only.

4.2 A naïve approach

A straightforward way to adopt Brewka's method to the case of conditional imperatives is to use not all imperatives for the construction, but only those that are triggered by the facts W, i.e. to use $\mathit{Triggered}_{\mathcal{I}}(W, \Delta)$ instead of Δ:

Definition 2 (The naïve approach).
Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, Δ be a subset of I, and $W \subseteq \mathscr{L}_{PL}$ be a set of PL-sentences. Then $\Gamma \in \mathscr{P}^{\mathrm{n}}_{\mathcal{I}}(W, \Delta)$ iff $\Gamma \in \mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, \mathit{Triggered}_{\mathcal{I}}(W, \Delta))$.

The change resolves our earlier problems with Brewka's method: consider again the set of imperatives $\{p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !\neg p_2\}$, where the imperatives were interpreted as ordering me to wear my boots when I go out, and not wear my boots when I don't. The new definition produces $\mathscr{P}^{\mathrm{n}}_{\mathcal{I}}(\{p_1\}, I) = \{\{p_1 \Rightarrow !p_2\}\}$, its only 'preferred' subset containing just the one imperative that is triggered given the facts $\{p_1\}$. By any truth definition (*td-pcd*1-8), $O(p_2/p_1)$ is now defined true, so given that you go out, you ought to wear your boots, which is as it should be.

The naïve approach is clearly a conservative extension of Brewka's original method to conditional imperatives: for sets Δ of unconditional imperatives, $\mathit{Triggered}_{\mathcal{I}}(\{\top\}, \Delta) = \Delta$. It is similar to Horty's proposal in [21] in that conflicts are only removed between imperatives that are triggered (though the exact mechanism differs from Horty's). When I nevertheless call it 'naïve', this is because there are conceivable counterexamples to this method. Consider the set of imperatives $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, ranked $!p_1 < \; p_1 \Rightarrow !p_2 < !\neg p_2$, and for an interpretation suppose that your job requires you to go outside p_1, that your mother, who is concerned for your health, told you to wear a scarf p_2 if you go outside, and that your friends don't want you to wear a scarf, whether you go outside or not. In the default situation $\top$ only the first imperative and the third are triggered, i.e. $\mathit{Triggered}_{\mathcal{I}}(\{\top\}, I) = \{!p_1, !\neg p_2\}$. Since their demands are consistent with each other, we obtain $\mathscr{P}^{\mathrm{n}}_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, !\neg p_2\}\}$, for which all truth definitions (*td-pcd*1-8) make $O(p_1 \wedge \neg p_2/\top)$ true. So you ought to go out and not wear a scarf, thus satisfying the first and the third imperative, but violating the second-ranking imperative. But arguably, if an imperative is to be violated, it should not be the second-ranking $p_1 \Rightarrow !p_2$, but the lowest ranking $!\neg p_2$ instead.

4.3 The stepwise approach

To avoid the difficulties of the 'naïve' approach, it seems we must not just take into account the imperatives that are triggered, but also those that become triggered when higher ranking imperatives are satisfied. To this effect, the following modification may seem reasonable:

Definition 3 (The stepwise approach).
Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, Δ be a subset of I, and $W \subseteq \mathscr{L}_{\mathrm{PL}}$ be a set of PL-sentences. Then $\Gamma \in \mathscr{P}^{\mathrm{s}}(W, \Delta)$ iff (i) *$W \nvdash_{PL} \bot$, and* (ii) *Γ is obtained from a full prioritization $\prec$ by defining*

$$\Gamma_{[\prec\downarrow i]} = \begin{cases} \bigcup_{j \prec i} \Gamma_{[\prec\downarrow j]} \cup \{i\} & \textit{if } \; i \in \mathit{Satisfiable}_{\mathcal{I}}(W \cup [\bigcup_{j \prec i} \Gamma_{[\prec\downarrow j]}]^f, \Delta), \textit{ and} \\ \bigcup_{j \prec i} \Gamma_{[\prec\downarrow j]} & \textit{otherwise,} \end{cases}$$

for any $i \in \Delta$, and letting $\Gamma = \bigcup_{i \in \Delta} \Gamma_{[\prec\downarrow i]}$.

So at each step one considers what happens if the imperatives that were included so far are satisfied, and adds the current imperative only if it is satisfiable given the truth of W and the satisfaction of these previous imperatives. Note that satisfiability of an imperative, like its satisfaction and violation, presupposes that the imperative is triggered. In contrast to the naïve approach, the new definition not only includes, at each step, those imperatives that are triggered and can be satisfied given the facts and the supposed satisfaction of the previously added imperatives: it also includes those that *become* triggered when a previously added imperative is satisfied.

The modification avoids the previous difficulty: consider again the set of imperatives $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, with the ranking $!p_1 < \ p_1 \Rightarrow !p_2 < !\neg p_2$. There is just one full prioritization, which for $W = \{\top\}$ yields in the first step the set $\{!p_1\}$, and in the second step $\{!p_1, p_1 \Rightarrow !p_2\}$, since $p_1 \Rightarrow !p_2$ is triggered when the previously added imperative $!p_1$ is assumed to be satisfied. In the third step, nothing is added: though the imperative $!\neg p_2$ is triggered, it cannot be satisfied together with the previously added imperatives. So we obtain $\mathscr{P}^{\mathrm{s}}_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, p_1 \Rightarrow !p_2\}\}$, and hence $O(p_1/\top)$, but not $O(p_1 \wedge \neg !p_2/\top)$, is defined true by all of (*td-pcd*1-8). Operators that accept 'deontic detachment' (as defined by *td-pcd*5-8) even make true $O(p_1 \wedge p_2/\top)$, and so in the given interpretation you must go out and wear a scarf, which now is as it should be.

However, a small change in the ordering shows that this definition does not suffice: let the imperatives now be ranked $p_1 \Rightarrow !p_2 < !p_1 < !\neg p_2$. (For the interpretation, assume that the conditional imperative to wear a scarf when leaving the house was uttered by some high-ranking authority, e.g. a doctor.) Then again $\mathscr{P}^{\mathrm{s}}_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, !\neg p_2\}\}$: in the first step, nothing is added since $p_1 \Rightarrow !p_2$ is neither triggered by the facts nor by the assumed satisfaction of previously added imperatives (there are none). In the next two steps, $!p_1$ and $!\neg p_2$ are added, as each is consistent with the facts and the satisfaction of the previously added imperatives. So again all of (*td-pcd*1-8) make true $O(p_1 \wedge \neg p_2/\top)$, i.e. you ought to go out and not wear a scarf, satisfying the second and third ranking imperatives at the expense of the highest ranking one. But surely, if one must violate an imperative, it should be one of the lower-ranking ones instead.

4.4 The reconsidering approach

The merits of the stepwise approach were that it did not only consider the imperatives that are triggered, but also those that *become* triggered when already added imperatives are satisfied. Such considerations applied to those imperatives that follow in the procedure. Yet the satisfaction of already added imperatives might also trigger higher-ranking imperatives, which by this method are not considered again. So it seems necessary, at each step, to reconsider also the higher-ranking imperatives. An algorithm that does that was first introduced for default theory by Marek & Truszczyński [28] p. 72, and later employed by Brewka in [5]; it can be reformulated for the present setting as follows:

Definition 4 (The reconsidering approach).
Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, Δ be a subset of I, and $W \subseteq \mathscr{L}_{\mathrm{PL}}$ be a set of PL-sentences. Then $\Gamma \in \mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(W, \Delta)$ iff (i) *$W \nvdash_{\mathrm{PL}} \bot$, and* (ii) *$\Gamma$ is obtained from a full prioritization $\prec$ by defining*

$$\Gamma_i \ = \ \bigcup_{j \prec i} \Gamma_\beta \ \cup \ min_{\prec}[Satisfiable_{\mathcal{I}}(W \cup [\bigcup_{j \prec i} \Gamma_j]^f, \Delta) \setminus \bigcup_{j \prec i} \Gamma_j]$$

for $i \in \Delta$, and letting $\Gamma = \bigcup_{i \in \Delta} \Gamma_i$.

The definition reconsiders at each step the whole ordering, and adds the $\prec$-first[14] imperative that has not been added previously and is satisfiable given both the

[14] For any ordering $<$ on some set Γ, $min_{<}\Gamma = \{i \in \Gamma \mid \forall i' \in \Gamma : \text{if } i' \neq i, \text{ then } i' \not< i\}$, and $max_{<}\Gamma = \{i \in \Gamma \mid \forall i' \in \Gamma : \text{if } i' \neq i, \text{ then } i \not< i'\}$, as usual.

circumstances C and the consequents of the previously added imperatives. Note that in 'Γ_i', i is used just as a index – it does not mean that i is considered for addition to the set at this step, and in fact it may be added at an earlier or later step (or not at all). To see how the definition works, consider again the example which the stepwise approach failed, i.e. the set of imperatives $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, with the ranking $p_1 \Rightarrow !p_2 < !p_1 < !\neg p_2$. We are interested in the preferred sets for the default circumstances $\top$, i.e. the sets in $\mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(\{\top\}, I)$. I is already fully prioritized, so there is just one such set. Applying the algorithm, we find the minimal (highest ranking) element in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\}, I)$ is $!p_1$, so this element is added in the first step. In the second step, we look for the minimal element in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\} \cup \{!p_1\}^f, I)$, other than the previously added $!p_1$. It is $p_1 \Rightarrow !p_2$, since the assumed satisfaction of all previously added imperatives triggers it, and its consequent can be true together with $\{\top\} \cup \{p_1\}$. So $p_1 \Rightarrow !p_2$ is added in this step. In the remaining third step, nothing is added: $!\neg p_2$ is not in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\} \cup \{!p_1, p_1 \Rightarrow !p_2\}^f, I)$, and all other elements in this set have been previously added. So $\mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, p_1 \Rightarrow !p_2\}\}$. Now all truth definitions (*td-pcd*1-8) make true $O(p_1/\top)$, but not $O(p_1 \wedge \neg !p_2/\top)$, and operators that accept 'deontic detachment' make true $O(p_1 \wedge p_2/\top)$. So, in the given interpretation, you must go out (as your job requires) and wear a scarf (as the doctor ordered you to do in case you go out), which is as it should be.

However, again problems remain. Reconsider the set $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, but let the ranking now be $p_1 \Rightarrow !p_2 < !\neg p_2 < !p_1$. Let $p_1 \Rightarrow !p_2$ stand for the doctor's order to wear a scarf when going outside, let $!\neg p_2$ stand for your friends' expectation that you don't wear a scarf, and let $!p_1$ represent your sister's wish that you leave the house. Construct the set in $\mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(\{\top\}, I)$ – since I remains fully prioritized, there is again just one such set. The minimal element in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\}, I)$ is $!\neg p_2$, and so is added in the first step. The minimal element in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\} \cup \{!\neg p_2\}^f, I)$, other than $!\neg p_2$, is $!p_1$ which therefore gets added in the second step. Nothing is added in the remaining step: $!\neg p_2$ and $!p_1$ have already been added, and $p_1 \Rightarrow !p_2$ is not in $\mathit{Satisfiable}_{\mathcal{I}}(\{\top\} \cup \{!\neg p_2, !p_1\}^f, I)$: though it is triggered by the assumed satisfaction of $!p_1$, its consequent is contradicted by the assumed satisfaction of $!\neg p_2$. So $\mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, !\neg p_2\}\}$. Hence all truth definitions (*td-pcd*1-8) again makes true $O(p_1 \wedge \neg p_2/\top)$, so you ought to go out without a scarf, again satisfying the second and third ranking imperatives at the expense of the first, which seems the wrong solution.

4.5 The fixpoint approach

To eliminate cases in which the 'reconsidering approach' still adds imperatives that can only be satisfied at the expense of violating a higher-ranking one, a 'fixpoint' approach may seem adequate. Such an approach was first proposed for default reasoning by Brewka & Eiter [6]. It tests each set that may be considered as preferred to see if it really includes all the elements that should be included: imperatives that are triggered given the facts and the assumed satisfaction of all imperatives in the set, and would be added by Brewka's [4] original method that adds the higher ranking imperatives first. The procedure translates as follows:

Definition 5 (The fixpoint approach).
Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, Δ be a subset of I, and $W \subseteq \mathscr{L}_{\mathrm{PL}}$ be a set of PL-sentences. Then

$$\Gamma \in \mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(W, \Delta) \quad \textit{iff} \quad \Gamma \in \mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, \textit{Triggered}_{\mathcal{I}}(W \cup \Gamma^f, \Delta)).$$

To see how this definition works, consider the above set of imperatives $I = \{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, with the ranking $p_1 \Rightarrow !p_2 < !\neg p_2 < !p_1$. It is immediate that the set $\{!p_1, !\neg p_2\}$ cannot be in $\mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{\top\}, I)$: if we assume all imperatives in this set to be satisfied, then all imperatives are triggered, i.e. $\textit{Triggered}_{\mathcal{I}}(\{\top\} \cup \{!p_1, !\neg p_2\}^f, I) = I$. By Brewka's original method, $\mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, I) = \{\{p_1 \Rightarrow !p_2, !p_1\}\}$: $<$ is already fully prioritized, and for this full prioritization the method adds $p_1 \Rightarrow !p_2$ in the first step, $!\neg p_2$ cannot be added in the second step since its consequent contradicts the consequent of the previously added $p_1 \Rightarrow !p_2$, and in the third step $!p_1$ is added. So since the considered set $\{!p_1, !\neg p_2\}$ is not in $\mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(W, I)$, it is not a 'fixpoint'. Rather, as may be checked, the only 'fixpoint' in $\mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{\top\}, I)$ is $\{p_1 \Rightarrow !p_2, !p_1\}$. For this set all truth definitions (*td-pcd*1-8) make true $O(p_1/\top)$, but no longer $O(p_1 \wedge \neg p_2/\top)$. Moreover, truth definitions like (*td-pcd*5-8) that allow 'deontic detachment' make true $O(p_1 \wedge p_2/\top)$. In the given interpretation this means that you must leave the house at your sisters request and wear a scarf, as the doctor ordered you to do in case you go out.

Though the construction now no longer makes true $O(p_1 \wedge \neg p_2/\top)$, its solution for the example, that determines the set $\{p_1 \Rightarrow !p_2, !p_1\}$ as the fixpoint of the set of imperatives $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$ with the ranking $p_1 \Rightarrow !p_2 < !\neg p_2 < !p_1$, seems questionable. Though this now includes the doctor's order, you now have no obligation anymore to satisfy the imperative that is second ranking, i.e. your friends' request that you don't wear a scarf; truth definitions (*td-pcd*4-8) even oblige you to violate it by wearing a scarf. Now consider the situation without the third ranking imperative $!p_1$: it can easily be verified that for a set $I = \{p_1 \Rightarrow !p_2, !\neg p_2\}$ the only fixpoint in $\mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{\top\}, I)$ is $\{!\neg p_2\}$. So for the reduced set, (*td-pcd*2) makes true $O(\neg p_2/\top)$, i.e. you ought to obey your friends' wish. That the satisfaction of this higher ranking imperative $!\neg p_2$ should no longer be obligatory when a lower ranking imperative $!p_1$ is added, seems hard to explain. If the ranking is taken seriously, I think one should still satisfy the higher ranking imperatives, regardless of what lower ranking imperatives are added.

However, there is another, perhaps even more severe problem with the fixpoint approach.[15] Consider a new set of imperatives $\{p_1 \Rightarrow !p_2, !(p_1 \wedge \neg p_2), !p_3\}$, with the ranking $p_1 \Rightarrow !p_2 < !(p_1 \wedge \neg p_2) < !p_3$. For an interpretation, let the first imperative be again the doctor's order to wear a scarf in case you go out, the second one be your friends' request to go out and not wear a scarf, and the third ranking imperative be the wish of your aunt that you write her a letter. Try to find a fixpoint for the default circumstances, i.e. some $\Gamma \in \mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{\top\}, I)$: either Γ contains the highest ranking imperative $p_1 \Rightarrow !p_2$ or it does not. If Γ contains it, then $p_1 \Rightarrow !p_2$ must be in $\textit{Triggered}_{\mathcal{I}}(\{\top\} \cup \Gamma^f, I)$. It can only be in there if also $!(p_1 \wedge \neg p_2)$ is in Γ, for otherwise $p_1 \Rightarrow !p_2$ could not be triggered. But no set that

[15] Both problems also arise for a new fixpoint approach by John F. Horty in [22].

is constructed by Brewka's method can include both of these imperatives, since their consequents contradict each other. So Γ does not contain $p_1 \Rightarrow !p_2$, contrary to our assumption. So assume Γ does not contain $p_1 \Rightarrow !p_2$. Whatever Γ is like, $\mathit{Triggered}_{\mathcal{I}}(\{\top\} \cup \Gamma^f, I)$ includes $!(p_1 \wedge \neg p_2)$. By Brewka's method, $!(p_1 \wedge \neg p_2)$ will only not be added to the set $\Gamma \in \mathscr{P}^{\mathrm{B}}_{\mathcal{I}}(\{\top\}, \mathit{Triggered}_{\mathcal{I}}(\{\top\} \cup \Gamma^f, I))$ if the consequents of previously added imperatives conflict with its consequent – but the only higher ranking imperative is $p_1 \Rightarrow !p_2$ and we already established that it is not in Γ. So $!(p_1 \wedge \neg p_2)$ is in Γ. But then $p_1 \Rightarrow !p_2$ is in $\mathit{Triggered}_{\mathcal{I}}(\{\top\} \cup \Gamma^f, I)$, and so is added to Γ in the first step of the construction, contrary to the assumption that it is not in Γ. So there is a *reductio ad absurdum* for both possible cases, hence there can be no $\Gamma \in \mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{\top\}, I)$, i.e. there is no fixpoint. So there is also no fixpoint that contains $!p_3$, and hence none of the truth definitions make $O(p_3/\top)$ true, and so you do not even have to write to your aunt. But even if the presence of both a higher ranking conditional imperative and a lower ranking, unconditional imperative to violate it poses a problem (why should it? after all, the lower ranking imperative is outranked), it is hard to see why the subject should be left off the hook for all other, completely unrelated obligations.[16]

4.6 Discussion

For a discussion of our results so far, let us return to the 'drinking and driving' example from the introduction. Let the three imperatives:

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

be represented by a prioritized conditional imperative structure $\mathscr{I} = \langle I, f, g, < \rangle$, where $I = \{(p_1 \Rightarrow !\neg p_2, p_3 \Rightarrow !p_2, p_3 \Rightarrow !p_1\}$ and $p_1 \Rightarrow !\neg p_2 < p_3 \Rightarrow !p_2 < p_3 \Rightarrow !p_1$. Let the set of facts be $\{p_3\}$, i.e. you go to the party. As we noted, Brewka's original method is not tailored to be directly employed to conditional imperatives. The next three approaches, the naïve, the stepwise and the reconsidering ones, produce $\mathscr{P}^{\mathrm{n}}_{\mathcal{I}}(\{p_3\}, I) = \mathscr{P}^{\mathrm{s}}_{\mathcal{I}}(\{p_3\}, I) = \mathscr{P}^{\mathrm{r}}_{\mathcal{I}}(\{p_3\}, I) = \{\{p_3 \Rightarrow !p_2, p_3 \Rightarrow !p_1\}\}$, which by all truth definitions (*td-pcd*1-8) makes true $O(p_1 \wedge p_2/p_3)$, so you ought to drink and drive. The fixpoint approach produces $\mathscr{P}^{\mathrm{f}}_{\mathcal{I}}(\{p_3\}, I) = \{\{p_1 \Rightarrow !\neg p_2, p_3 \Rightarrow !p_1\}\}$, so all truth definitions make true $O(p_1/p_3)$, which means you ought to drink. Truth definition with 'deontic detachment' like (*td-pcd*5-8) additionally make true $O(p_1 \wedge \neg p_2/p_3)$, so you ought to drink and not drive. But the natural reaction is to ignore the third ranking imperative and drive, as your best friend asked you to do. So it seems we have to look for a different solution.

[16] An independent approach by Makinson in [24], which, however, only considers non-prioritized conditionals, also fails in this case: for the default circumstances $\top$ it produces the set $\{!(p_1 \wedge \neg p_2), !p_3\}$. $p_1 \Rightarrow !p_2$ is not considered, since its only 'label' (roughly: a conjunction of the circumstances, the imperatives' antecedents that would trigger* it, and its consequent) is inconsistent (it is $\top \wedge (p_1 \wedge \neg p_2) \wedge p_2$). But it is requires explanation why the agent should not be allowed to obey $p_1 \Rightarrow !p_2$, rather than having to violate it by satisfying $!(p_1 \wedge \neg p_2)$.

Before we do that, I will, however, question again our intuition in this matter. John F. Horty [22] has recently used a structurally similar example to argue for just the opposite, that the solution by the fixpoint approach is correct, i.e. that (in our example) you should drink and not drive. His example is that of three commands, uttered by a colonel, a major and a captain to a soldier, Corporal O'Reilly. The Colonel, who does not like it too warm in the cabin, orders O'Reilly to open the window whenever the heat is turned on. The Major, who is a conservationist, wants O'Reilly to keep the window closed during the winter. And the Captain, who does not like it to be cold, orders O'Reilly to turn the heat on during the winter. The intended representation is again the prioritized conditional imperative structure employed above for the 'drinking and driving' example, where p_1 now means that the heat is turned on, p_2 means that the window is closed, and p_3 means that it is winter. As we have seen, the fixpoint approach yields the preferred subset $\{p_1 \Rightarrow !\neg p_2, p_3 \Rightarrow !p_1\}$, making true $O(p_1/p_3)$ with (*td-pcd*1-3), and $O(p_1 \wedge \neg p_2/p_3)$ with (*td-pcd*4-8), so O'Reilly must turn on the heat and then open the window, and thus violate the Major's order. Horty argues as follows in support of the choice of this set:

> "O'Reilly's job is to obey the orders he has been given exactly as they have been issued. If he fails to obey an order issued by an officer without an acceptable excuse, he will be court-martialed. And, let us suppose, there is only one acceptable excuse for failing to obey such an order: that obeying the order would, in the situation, involve disobeying an order issued by an officer of equal or higher rank. (...) So given the set of commands that O'Reilly has been issued, can he, in fact, avoid court-martial? Yes he can, by (...) obeying the orders issued by the Captain and the Colonel (...). O'Reilly fails to obey the Major's order, but he has an excuse: obeying the Major's order would involve disobeying an order issued by the Colonel."

Horty's principle seems quite acceptable: for each order issued to the agent, the agent may ask herself if obeying the order would involve disobeying an order of a higher ranking officer (then he is excused), and otherwise follow it. The result is a set of imperatives where each imperative is either obeyed, or disobeyed but the disobedience excused. When I nevertheless think the argument is not correct, it is because I think it confuses the *status quo* and the *status quo posterior*. Obeying the Major's order does not, in the initial situation, involve disobeying the Colonel's order. Only once O'Reilly follows the Captain's order and turns on the heat, it is true that he must obey the Colonel, open the window and thus violate the Major's order. But this does not mean that he should follow the Captain's order in the first place – as by doing so he brings about a situation in which he is forced, by a higher ranking order, to violate a command from another higher ranking officer. Quite to the contrary, I think that being forced to violate a higher ranking order when obeying a lower ranking one is a case where following the lower one 'involves' such a violation, and so the only order the agent is excused from obeying is the lowest ranking command.

Another notion seems of importance in such examples: that of coherence, or coherent interpretation, of the imperatives that are accepted by an agent. Suppose I am a trainee at a factory, and over my new workplace there is a large sign: “If the light flashes, press the red button. By order of the Director.” On the first day, the foreman tells me “Don’t you ever press the red button.” A bit later a colleague comes round and tells me “Let’s have some fun. Make the light flash. Just short-circuiting it does the job”. Obviously I have not been told not to make the light flash. By doing so, I will have to do what the sign tells me and press the red button, and thus violate the foreman’s explicit order on my first day. But I can reason as follows: ‘Surely, the foreman did not want to contradict the Director’s order. But it would amount to a contradiction if the light flashes and I do as he told me and not press the button, though the sign says otherwise. So what the foreman meant was probably this: don’t press the button if the light does *not* flash. So I can safely make the light flash as my colleague told me, and then press the button, thus making everybody happy.’ (The consequence of such reasoning would probably be that I lose my job, which might be what my colleague meant by ‘fun’.) Such coherent reinterpretation plays an important role in judicial reasoning. But our concern are sets of imperatives that may stem from various sources and contain explicit conflicts. It is the preference ordering that is supposed to take care of arising conflicts. And by closer examination of the situation, if the light flashes, the apparent conflict is resolved since the foreman’s order is overridden. Yet that does not mean that I have to accept an obligation to bring about such a situation. If some order is to get me to make the light flash, I think it would have to rank at least as high as the foreman’s command, e.g. if my colleague had uttered the imperative in some state of emergency.

Consider finally this variant: suppose that if I am attacked by a man, I must fight him (to defend my life, my family etc.). Furthermore, suppose I have pacifist ideals which include that I must not fight the man. Now you tell me to provoke him, which in the given situation means that he will attack me. Let self-defense rank higher than my ideals, which in turn rank higher than your request. Should I do as you request? By the reasoning advocated by Horty, there is nothing wrong with it: I satisfy your request, defend myself as I must, and though I violate my ideals, I can point out to myself that the requirement to fight back took priority. But I think if I really do follow your advice, I would feel bad. I think this would not just be some irrational regret for having to violate, as I must, my ideals, but true guilt for having been tempted into doing something I should not have done, namely provoking the man: it caused the situation that made me violate my ideals. So I think our intuitions in the ‘drinking and driving’ example and the other cases have been correct.

5 New Strategies and a New Proposal

In the face of the difficulties encountered so far, it seems necessary to address the issue of finding an appropriate mechanism for a resolution of conflicts between prioritized conditional imperatives in a more systematic manner.

5.1 Deontically Tailored Preferred Subsets

In the unconditional case, the reason to move from definition (td-$m1$) to (td-$m2$) was that when there are conflicts between imperatives, the former makes true the monadic deontic formula $O\bot$, i.e. the agent ought to do the logically impossible. This result was avoided by considering only maximal sets of imperatives with demands that are collectively consistent, i.e. sets that do not make $O\bot$ true. When faced with the question what dyadic deontic formula should not be true when conflicts are resolved for arbitrary situations C, the formula $O(\neg C/C)$ appears to be the dyadic equivalent: it would be weird if a mechanism for conflict resolution results in telling the agent to do something that contradicts the assumed facts.[17] So to define the set $\mathscr{P}_{\mathcal{I}}(\{C\}, I)$ for a truth definition (td-pcd1-8), we can modify Brewka's original method in such a way that it tests, at each step, for each of the constructed subsets, if the corresponding truth-definition (td-cd1-8) does not make $O(\neg C/C)$ true for this set.[18] Formally:

Definition 6 (Deontically Tailored Preferred Subsets). *Let $\mathcal{I} = \langle I, f, g, <\rangle$ be a prioritized conditional imperative structure, and $C \in \mathscr{L}_{PL}$ describe the given situation. Let (td-$pcd*$) be any of the truth definitions (td-pcd1-8). Then Γ is in the set $\mathscr{P}^{*}_{\mathcal{I}}(\{C\}, I)$ employed by this truth definition iff* (i) $\{C\} \nvdash_{PL} \bot$, *and* (ii) *Γ is obtained from a full prioritization $\prec$ by defining*

$$\Gamma_{[\prec\downarrow i]} = \begin{cases} \bigcup_{j\,\prec\, i} \Gamma_{[\prec\downarrow j]} \cup \{i\} & \text{if } \langle \bigcup_{j\,\prec\, i} \Gamma_{[\prec\downarrow j]} \cup \{i\}, f, g\rangle \nvDash O(\neg C/C) \text{ by } (td\text{-}cd*), \\ \bigcup_{j\,\prec\, i} \Gamma_{[\prec\downarrow j]} & \text{otherwise,} \end{cases}$$

for any $i \in I$, and letting $\Gamma = \bigcup_{i\in I} \Gamma_{[\prec\downarrow i]}$.

By this construction, each of the preferred subsets contains a maximal number of the imperatives such that they do not make true $O(\neg C/C)$ for the situation C and the truth definition that is employed, and so the resulting truth definition likewise avoids this truth. Such a construction of the preferred subsets might be considered 'tailored' to the truth definition in question, and any remaining deficiencies might be seen as stemming from the employed truth definition. But this being so, the method reveals a strong bias towards truth definitions that accept 'deontic detachment', and in particular truth definitions (td-pcd4-8):

Consider the set of imperatives $I = \{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$ with the ranking $!p_1 < p_1 \Rightarrow !p_2 < !\neg p_2$, that was used to refute the 'naïve approach'. As can be easily checked, $\mathscr{P}^{*}_{\mathcal{I}}(\{\top\}, I) = \{I\}$ for all truth definitions (td-pcd1-3). So by all these truth definitions, $O(p_1 \wedge \neg p_2/\top)$ is true. So they commit us to violating the second-ranking imperative, whereas intuitively, the third-ranking imperative should be violated instead. By contrast, all truth definitions (td-pcd5-8), that employ reusable output, and of course likewise (td-pcd4) that is equivalent to (td-pcd8), handle all given examples exactly as they should be. For the set $I =$

[17] For arguments why $O(\neg C/C)$ should be chosen, i.e. for their setting, the 'output' should be consistent with the 'input', rather than the formula $O(\bot/C)$ and thus consistency of output *simpliciter*, cf. Makinson & van der Torre [26] p. 158/159.

[18] The preferred subsets are thus a choice from the 'maxfamilies' defined in [26].

$\{!p_1, p_1{\Rightarrow}!p_2, !\neg p_2\}$ they produce for both, the ranking $!p_1 < \ p_1{\Rightarrow}!p_2 < !\neg p_2$ that was used to refute the 'naïve approach', and the ranking $p_1{\Rightarrow}!p_2 < !p_1 < !\neg p_2$ that was used to refute the stepwise approach, the set $\mathscr{P}^*_{\mathcal{I}}(\{\top\}, I) = \{\{!p_1, p_1{\Rightarrow}!p_2\}\}$, $* = 4, 5, 6, 7, 8$. Thus they all make true $O(p_1 \wedge p_2/\top)$, committing us to violate the lowest-ranking imperative only, as it should be for these examples. If the imperatives' ranking is instead $p_1{\Rightarrow}!p_2 < !\neg p_2 < !p_1$, that was used to refute both the 'reconsidering' and the 'fixpoint' approaches, then $\mathscr{P}^*_{\mathcal{I}}(\{\top\}, I)$ is $\{\{p_1{\Rightarrow}!p_2, !\neg p_2\}\}$, making $O(\neg p_2/\top)$ true by all these truth definitions, which thus commit us to satisfying the second ranking imperative, and not to violating it in favor of satisfying the third ranking imperative as these approaches did. Finally the set $I = \{p_1{\Rightarrow}!p_2, !(p_1 \wedge \neg p_2), !p_3\}$ with the ranking $p_1{\Rightarrow}!p_2 < !(p_1 \wedge \neg p_2) < !p_3$, that was also mishandled by the 'fixpoint approach', produces the set $\mathscr{P}^*_{\mathcal{I}}(\{\top\}, I) = \{\{p_1{\Rightarrow}!p_2, !p_3\}\}$. So it rejects the second ranking imperative, that commits to violating the higher ranking one, and keeps both others, as it should be. The 'drinking and driving' example is also handled correctly: the set $\{p_1{\Rightarrow}!\neg p_2, p_3{\Rightarrow}!p_2, p_3{\Rightarrow}!p_1\}$, with the ranking $p_1{\Rightarrow}!\neg p_2 < \ p_3{\Rightarrow}!p_2 < \ p_3{\Rightarrow}!p_1$ produces, for the situation p_3, the set $\mathscr{P}_{\mathcal{I}}(\{p_3\}, I) = \{\{p_1{\Rightarrow}!\neg p_2, p_3{\Rightarrow}!p_2\}\}$. So the third ranking imperative, that commits the agent to drinking and thus, by observation of the highest ranking imperative, prevents the agent from driving, is disregarded. Instead, the truth definitions make true $O(p_2/p_3)$, so the agent must do the driving if she goes to the party, as her best friend asked her to.

Is this the solution to our problems, then? Some uneasiness remains as to the quick way with which definitions (td-*pcd*1-3) were discharged as insufficient. Why should it not be possible to maintain, as these definitions do, that conditional imperatives only produce an obligation if they are factually triggered, while at the same time maintaining that the above examples should not be resolved the way they are by (td-*pcd*1-3)? The purpose of a truth definition for the deontic O-operator is to find a formal notion of 'ought' that reflects ordinary reasoning, and our intuitions on that matter may differ from our ideas as to what may constitute a good choice from a possibly conflicting set of prioritized conditional imperatives. I will now make a new proposal how to construct the 'preferable subsets', that keeps the positive results without committing us to prefer any of (td-*pcd*1-8) by virtue of their handling of prioritized imperatives alone.

5.2 Preferred Maximally Obeyable Subsets

What made Brewka's approach so successful is that it maximizes the number of higher ranking imperatives in the preferred subsets of a given set of unconditional imperatives: for each 'rank', a maximal number of imperatives are added that can be without making the set's demands inconsistent in the given situation. As was shown, Brewka's approach cannot be directly applied to conditional imperatives, since it makes no sense to test the demands of imperatives for inconsistencies if these imperatives may not be triggered in the same circumstances. Just considering triggered imperatives is also not enough, as was demonstrated for the 'naïve approach'. But if the maximization method is to include imperatives that are not (yet) triggered, then we must look for something else than inconsistency of demands to take the role of a threshold criterion for the maximization process.

To do so we should ask ourselves why, for the unconditional case, the aim was to find a maximal set of imperatives with demands that are collectively consistent with the situation. I think that by doing so we intend to give the agent directives that can be safely followed. While in the unconditional case this means that the agent can satisfy all the chosen imperatives, the situation is different for conditional imperatives: here an agent can also obey imperatives without necessarily satisfying them. If you tell me to visit you in case I go to Luxembourg next month, I can safely arrange to spend all of next month at home and still do nothing wrong. If we think not so much of imperatives, but of legal regulations, then I can obviously be a law-abiding citizen by simply failing to trigger any legal norm (even though this might imply living alone on an island): whether I do that or boldly trigger some of the regulations' antecedents and then satisfy those I have triggered seems not a question of logic, but of individual choice. So I think the threshold criterion to be used should be that of obeyability: we should maximize the set of imperatives the agent can thus obey, and only stop when the addition of an imperative means that, given the facts, it or some already added imperative, i.e. one that ranks higher or at least as high, can no longer be obeyed, and so will be violated.[19]

For a given set of conditional imperatives $\varDelta$ and a set of factual truths W, the subsets of imperatives that can be obeyed are described by $\mathit{Obeyable}_{\mathcal{I}}(W, \varDelta)$, i.e. they are those subsets $\varGamma \subseteq \varDelta$ such that $W \cup \varGamma^m \nvdash_{PL} \bot$. To maximize not by collective consistency of demands, but by collective obeyability, Brewka's original approach can therefore be changed as follows:

Definition 7 (Preferred Maximally Obeyable Subsets). *Let $\mathcal{I} = \langle I, f, g, < \rangle$ be a prioritized conditional imperative structure, $\varDelta$ be a subset of I, and $W \subseteq \mathscr{L}_{PL}$ be a set of PL-sentences. Then $\varGamma \in \mathscr{P}^{o}_{\mathcal{I}}(W, \varDelta)$ iff* (i) *$W \nvdash_{PL} \bot$, and* (ii) *$\varGamma$ is obtained from a full prioritization $\prec$ by defining*

$$\varGamma_{[\prec\downarrow i]} = \begin{cases} \bigcup_{j \prec i} \varGamma_{[\prec\downarrow j]} \cup \{i\} & \text{if } W \cup [\bigcup_{j \prec i} \varGamma_{[\prec\downarrow j]} \cup \{i\}]^m \nvdash_{PL} \bot, \text{ and} \\ \bigcup_{j \prec i} \varGamma_{[\prec\downarrow j]} & \text{otherwise,} \end{cases}$$

for any $i \in \varDelta$, and letting $\varGamma = \bigcup_{i \in \varDelta} \varGamma_{[\prec\downarrow i]}$.

The change from Brewka's original definition is only minute: we test not the demands of the imperatives for consistency, but their materializations. Note that this is a conservative extension of Brewka's method, since for any unconditional imperative i we have $\vdash_{PL} f(i) \leftrightarrow m(i)$. As can easily be seen, the new construction solves all of the previously considered difficulties, regardless of the chosen truth definition for the deontic O-operator:

- To refute a direct application of Brewka's original method, we used the set $I = \{p_1 \Rightarrow !p_2, \neg p_1 \Rightarrow !\neg p_2\}$ with no ranking imposed. $m(I)$ is consistent and so $\mathscr{P}^{o}_{\mathcal{I}}(\{p_1\}, I) = \{I\}$, making $O(p_2/p_1)$ true for all definitions (td-*pcd*1-8). So you ought to wear your boots in case you go out, as it should be.

[19] While S. O. Hansson, in [17] p. 200, also advocates a move from 'consistency' to 'obeyability', what is meant there is rather the step from (*td-m*2) to (*td-d*1).

- To refute the 'naïve approach', we used the set $I = \{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$ with the ranking $!p_1 < \ p_1 \Rightarrow !p_2 < !\neg p_2$. Since I is already fully prioritized, the construction produces just one maximally obeyable subset, which is $\{!p_1, p_1 \Rightarrow !p_2\}$, as its two imperatives get added in the first two steps, and nothing is added in the third since $m(I)$ is inconsistent. All of (td-*pcd*1-8) make true $O(p_1/\top)$, none makes true the non-intuitive formula $O(\neg p_2/\top)$, and the definitions (td-*pcd*5-8) that accept 'deontic detachment' make true $O(p_1 \wedge p_2/\top)$. So you must go out and wear a scarf, which is as it should be.
- To refute the stepwise approach we used $I = \{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$ with the ordering $p_1 \Rightarrow !p_2 < !p_1 < !\neg p_2$. Still $\mathscr{P}^{\text{o}}_{\mathcal{I}}(\top\}, I) = \{\{!p_1, p_1 \Rightarrow !p_2\}\}$, so the sentences made true by truth definitions (td-*pcd*1-8) likewise do not change, and in particular the non-intuitive formula $O(\neg p_2/\top)$ is still false, and definitions (td-*pcd*5-8) that accept 'deontic detachment' make true $O(p_1 \wedge p_2/\top)$, so again you must go out and wear a scarf, which is as it should be.
- To refute the reconsidering and the fixpoint approaches we used again the set $\{!p_1, p_1 \Rightarrow !p_2, !\neg p_2\}$, but the ranking was changed into $p_1 \Rightarrow !p_2 < !\neg p_2 < !p_1$. Now $\mathscr{P}^{\text{o}}_{\mathcal{I}}(\top\}, I) = \{\{p_1 \Rightarrow !p_2, !\neg p_2\}\}$. Truth definitions (td-*pcd*1-8) make true $O(\neg p_2/\top)$ but not $O(p_1/\top)$ so you must satisfy the second ranking imperative, but not the third ranking imperative, which is as it should be.
- Troublesome for the fixpoint approach was the set $\{p_1 \Rightarrow !p_2, !(p_1 \wedge \neg p_2), !p_3\}$, with the ranking $p_1 \Rightarrow !p_2 < !(p_1 \wedge \neg p_2) < !p_3$: no fixpoint could be made out in the set and so the approach produced no preferred subset, making everything obligatory. The preferred maximally obeyable subset is $\{p_1 \Rightarrow !p_2, !p_3\}$, eliminating the second ranking imperative that demands a violation of the first, and making $O(p_3/\top)$ true under all truth definitions (td-*pcd*1-8), which again is as it should be.
- Finally, consider the 'drinking and driving' example: the set of imperatives $\{p_1 \Rightarrow !\neg p_2, p_3 \Rightarrow !p_2, p_3 \Rightarrow !p_1\}$ produces, for the situation p_3, the set of preferred maximally obeyable subsets $\mathscr{P}^{\text{o}}_{\mathcal{I}}(\{p_3\}, I) = \{\{p_1 \Rightarrow !\neg p_2, p_3 \Rightarrow !p_2\}\}$, making true $O(p_2/p_3)$ for all truth definitions (td-*pcd*1-8), so given that I go to the party I must do the driving, which is as it should be.

As could be seen, all truth definitions now produce the 'right' results in the examples used. Moreover, since all truth definitions refer to the same preferred subsets $\mathscr{P}^{\text{o}}_{\mathcal{I}}(\{C\}, I)$, it is possible to index the O-operators according to the truth definition employed, and e.g. state truths like $O^1(A/C) \wedge O^5(B/C) \rightarrow O^7(A \wedge B/C)$, meaning that if, for any maximal set of imperatives that I can obey in the situation C, imperatives are triggered that demand A, and that if I satisfy all such triggered imperatives, I will have to do B, then obeying a maximal number of imperatives includes having to do $A \wedge B$. It may well be that natural language 'ought-statements' are ambiguous in the face of conditional demands, the discussion in sec. 3 suggested this. If maximal obeyability is accepted as the threshold criterion that limits what norms an agent can be expected to conform to in a given situation, then definition 7 leaves the philosophical logician with maximal freedom as to what deontic operator is chosen.

6 Further Research

6.1 Benchmark examples for non-prioritized imperatives

Inevitably there remains further work to do. First of all, it seems worthwhile to consider what happens if the imperatives are not prioritized, in the sense that either there is no ranking between them or that they all have the same priority. It is immediate that for such imperatives, the set of preferred subsets $\mathscr{P}^{\circ}_{\mathcal{I}}(W, \varDelta)$ for a prioritized conditional imperative structure $\mathcal{I} = \langle I, f, g, < \rangle$ and a subset of the imperatives $\varDelta$, equals $max_{\subseteq} Obeyable_{\mathcal{I}}(W, \varDelta)$, i.e. the preferred subsets are just all the maximally obeyable subsets of $\varDelta$, given the facts W. There exist a number of benchmark examples for non-prioritized conditional imperatives, given by Makinson in [24], and I list without proof the solutions we obtain for these examples for the O-operators defined here.

Source and name	**Imperatives**	**Non-truths**	**Truths**
von Wright [**?**] window closing	$r \Rightarrow !c$, $s \Rightarrow !\neg c$	$O(c \wedge \neg c/r \wedge s)$	$O(c \vee \neg c/r \wedge s)$
Chisholm [7] help and inform	$!h$, $h \Rightarrow !i$, $\neg h \Rightarrow !\neg i$	$O(h/\neg h)$, $O(i/\neg h)$	$O(h \wedge i/\top)$, $O(\neg i/\neg h)$
Forrester [9] gentle murderer	$!\neg k$, $k \Rightarrow !g$	$O(g/\top)$, $O(\neg k/k)$	$O(\neg k/\top)$, $O(g/k)$
Belzer [3] Reykjavik scenario	1. $!(\neg r \wedge \neg g)$, $r \Rightarrow !g$, $g \Rightarrow !r$ 2. $!\neg r$, $!\neg g$, $r \Rightarrow !g$, $g \Rightarrow !r$	$O(\neg g/r)$	$O(g/r)$
Prakken& Sergot [33] cigarettes from a killer	$!\neg k$, $!\neg c$, $k \Rightarrow !c$	$O(\neg k/k)$	$O(c/k)$ fails!
Prakken& Sergot [33] white fence and dog	$!\neg f$,$f \Rightarrow !(f \wedge w)$, $d \Rightarrow !(f \wedge w)$	$O(\neg f/f)$, $O(\neg f/f \wedge d)$	$O(f \wedge w/f)$, $O(f \wedge w/d \wedge f)$ $O(f \wedge w/d)$ fails!
van der Torre [42] apples and pears	1. $!(a \vee p)$, $!\neg a$ 2. $!(a \vee p)$ 3. $\neg p \Rightarrow !a$, $\neg a \Rightarrow !p$	$O(\neg a/a)$	$O(\neg a \wedge p/\top)$, $O(\neg a \wedge p/\neg a)$ $O(p/\neg a)^{\mathrm{I}}$ $O(p/\neg a)$
van der Torre [42] joining paths	$!a$, $!b$	$O(a \wedge b/\neg a \vee \neg b)$	$O(a \vee b/\neg a \vee \neg b)$
Makinson [24] Möbius strip	$a \Rightarrow !b$, $b \Rightarrow !c$, $c \Rightarrow !\neg a$	$O(\neg a/a)$	$O(c/a)$ fails!
Makinson [24] exclusive options	$c \Rightarrow !(a \wedge b)$, $\neg c \Rightarrow !(a \wedge \neg b)$		$O(a/\top)^{\mathrm{II}}$

[I] O-operators that accept 'circumstantial reasoning' only, i.e.(*td-pcd*2,4,6,8).

[II] O-operators that accept 'reasoning by cases' only, i.e. (*td-pcd*3,4,7,8).

So there are three benchmark examples for which our definitions fail:

In the first one, proposed by Prakken & Sergot [33] and termed 'cigarettes from a killer', the imperative $!\neg k$ is intended to mean that you should not kill a certain man, $!\neg c$ means that you should not offer this man a cigarette, and $k \Rightarrow !c$ means that if you kill the man, you should offer him a cigarette first. Prakken & Sergot argue that the solution should make true $O(c/k)$, as this applies the imperative that is more specific for the given circumstances, but none of the operators provides this result. A similar idea underlies the second example, also proposed by Prakken & Sergot [33] and termed above 'white fence and dog'. There is a general prohibition of fences $!\neg f$ except if there already is one – in that case it should be white, i.e. $f \Rightarrow !(f \wedge w)$ – or if the owner has a dog, in which case the owner should have a white fence, i.e. $d \Rightarrow !(f \wedge w)$. Again, Prakken & Sergot intend the more specific imperative to be applied in the situation where there is a dog, and so argue that $O(f \wedge w/d)$ should hold. It is true none of the operators defined above includes a 'specificity test', and I do not think that this is a defect. The legal principle *lex specialis derogat legi generali* is not universally applicable to all sets of norms, in particular if they may stem from various sources, and even in the realm of law it competes with other principles like *lex posterior*, *lex superiori*, or standard argument forms like teleological interpretation. But if in the given case the more specific imperative should take priority, we can use a priority ordering that includes $k \Rightarrow !c < !\neg c$ in case of the first example, and $d \Rightarrow !(f \wedge w) < !\neg f$ in the case of the second. Then all operators (*td*-*pcd*1-8) make true $O(c/k)$ and $O(f \wedge w/d)$, as intended.

The third example that the truth definitions fail is Makinson's [24] 'Möbius strip': here the set of imperatives is $\{a \Rightarrow !b, b \Rightarrow !c, c \Rightarrow !\neg a\}$. Makinson argues that intuitively, $O(b \wedge c/a)$ should hold. But as is immediate, any maximally obeyable set includes just two of the given imperatives, which does not suffice for the truth of $O(b \wedge c/a)$ for any of (*td*-*pcd*1-8). The argument why $O(b \wedge c/a)$ ought to be true seems to be that since the consequent of the third imperative $c \Rightarrow !\neg a$ is false in the supposed situation a, the agent cannot do anything about it even if its antecedent becomes true, and so this imperative should not be considered.[20] But is this argument sound? Even if the consequent is inevitably false, there will be a violation only if its antecedent is (made) true. Certainly, I do not think that the agent should, in such cases, be under an obligation to make the antecedent false – this would introduce a 'deontic contraposition' that, as we saw, is not generally desirable. But that does not mean that the agent should accept *an obligation* to make the antecedent true. Consider this example: a professor tells a student that next time he sees her, he must have some written paper to present. The student's mother, who is worried about his PhD not getting finished, wants him to see his professor. The fact is: he does not, and will not, have a written paper. Should he therefore have to go and see his professor? I think that it is entirely up to the agent which of the two imperatives he is going to obey, either attributing higher weight to the explicit order of his professor, or giving priority to alleviating his mother's worries. Similarly, in the case of the Möbius strip, it may be that the agent has reasons to think that she must rather disobey one of the first two imperatives than violate the third. Then the set $\{a \Rightarrow !b, b \Rightarrow !c\}$ is not an acceptable choice in the situation a, so $O(b \wedge c/a)$ should not be true, and so not providing this truth seems not a defect.

[20] Similarly, Greenspan [12] argues that "it seems that oughts are no longer in force when it is too late to see to it that their objects are fulfilled".

6.2 Theorems

Truth definitions (td-*pcd*1-8) define when a sentence of the form $O(A/C)$ is true or false with respect to a prioritized conditional structure $\mathcal{I}$ and a situation C. So I briefly consider what sentences are theorems, i. e. hold for all such structures, given the usual truth definitions for Boolean operators. It is immediate that for all truth definitions, (DExt), (DM), (DC), (DN) and (DD-R) are theorems. (DD-R) states that there cannot be both an obligation to bring about A and one to bring about $\neg A$ unless the situation C is logically impossible, so our truth definitions succeed in eliminating conflicts. All these theorems are 'monadic' in the sense that the situation C is kept fixed; in fact, they are the C-relative equivalents of standard deontic logic *SDL*. More interesting are theorems describing the relations between obligations in different circumstances. Obviously we have

(ExtC) If $\vdash_{PL} C \leftrightarrow D$ then $O(A/C) \leftrightarrow O(A/D)$ is a theorem

for all truth definitions, i.e. for equivalent situations C, the obligations do not change. As long as truth definitions are not sensitive to conflicts, e.g. for (*td-cd*1-8), we have 'strengthening of the antecedent', i.e. for these definitions

(SA) $O(A/C) \rightarrow O(A/C \wedge D)$

holds. When only maximally obeyable subsets are considered, i.e. for truth definitions (*td-pcd*1-8), both (SA) and the weaker 'rational monotonicity' theorem

(RM) $\neg O(\neg D/C) \wedge O(A/C) \rightarrow O(A/C \wedge D)$

are refuted e.g. by a set $I = \{!(p_1 \wedge p_2), !(p_1 \wedge \neg p_2), p_2 \Rightarrow \neg p_1\}$ of equally ranking imperatives: though $O(p_1/\top)$ is true and $O(\neg p_2/\top)$ false, $O(p_1/p_2)$ is false. However, for all definitions(*td-pcd*1-8), '(conjunctive) cautious monotonicity'

(CCMon) $O(A \wedge D/C) \rightarrow O(A/C \wedge D)$

holds, which states that if you should to two things and you do one of them, you still have the other one left.[21] Moreover, truth definitions (*td-pcd*2,4,6,8) validate the 'circumstantial extensionality' rule

(CExt) If $\vdash_{PL} C \rightarrow (A \leftrightarrow B)$ then $O(A/C) \leftrightarrow O(B/C)$ is a theorem

that corresponds to 'circumstantial reasoning'. All definitions that accept 'reasoning by cases', i.e. (*td-pcd*3,4,7,8), make

(Or) $O(A/C) \wedge O(A/D) \rightarrow O(A/C \vee D)$

a theorem. Note that (CExt) and (Or) derive

(Cond) $O(A/C \wedge D) \rightarrow O(D \rightarrow A/C)$,

which in turn derives (Or) in the presence of (DC), and that by adding (CExt) and (Or) we obtain again the system *PD* (cf. sec. 3). Finally, all definitions with 'deontic detachment', i.e. (*td-pcd*5,6,7,8), make

(Cut) $O(A/C \wedge D) \wedge O(D/C) \rightarrow O(A/C)$

a theorem. (Cut) is derivable given (Cond) (use Cond on the first conjunct $O(A/C \wedge D)$ to obtain $O(D \rightarrow A/C)$, agglomerate with $O(D/C)$, and from $O(D \wedge (D \rightarrow A)/C)$ derive $O(A/C)$), which syntactically mirrors the semantic

[21] This is B. Hansson's [16] theorem (19).

equivalence of definitions (*td-pcd*4) and (*td-pcd*8). Theoremhood of all of the above theorems for semantics that employ the respective truth definitions is easily proved and left to the reader (cf. my [14] and [15] as well as Makinson & van der Torre [25] for the general outline). Makinson & van der Torre's results also suggest that these theorems axiomatically define complete systems of deontic logic with respect to semantics that employ the respective truth definitions (*td-pcd*1-8), but this remains a conjecture that further study must corroborate.[22]

6.3 Questions of representation

One might wonder if it is always adequate to represent a natural language conditional imperative 'if ... then bring about that ___' by use of a set I containing an imperative i with a $g(i)$ that formalizes '...' and a $f(i)$ that formalizes '___'. This is because there is a second possibility: represent the natural language conditional imperative by an unconditional imperative $\ulcorner !(g(i) \rightarrow f(i))\urcorner$. We saw in sec. 3 that this is not generally adequate. But that does not mean that such a representation is not *sometimes* what is required. Consider the crucial imperatives in the previous examples: perhaps what your mother meant was simply 'don't drink and drive'; perhaps what the doctor meant was 'don't go out without a scarf'; perhaps the Colonel meant to tell O'Reilly not to do both, turn the heat on and keep the window closed; perhaps the sign wanted me to see to it that the light does not flash without the button being pressed, perhaps self-defense required me to see to it that I am not attacked without fighting back. These interpretations seem not wholly unreasonable, and if they are adequate, then the best representation would be by an imperative $\ulcorner !(g(i) \rightarrow f(i))\urcorner$ instead of $\ulcorner g(i) \Rightarrow ! f(i)\urcorner$. It is easy to see that with such a representation, all of the discussed methods would have resolved these examples.

What then are the conditions that make a representation by an unconditional imperative adequate? One test may be to ask: 'Would bringing about the absence of the antecedent condition count as satisfaction of the imperative?'. Would not drinking, not going out, not turning on the heat, making the light not flash, making the man not attack, count as properly reacting to the imperatives in question? It should be if what the imperatives demand is a material conditional, since then the conditional imperatives in question are equivalent to telling the agent 'either don't drink or don't drive, its your decision', 'either don't go out, or wear a scarf', 'either don't turn on the heat, or open the window', etc. Another test would be to examine if contraposition is acceptable. Can we say that your mother wanted you not to drink if you are going to drive, that the doctor wanted you to stay inside if you are not going to wear a scarf, that the Colonel wanted O'Reilly to turn off the heat if the window is closed, that the sign wants you to make the light not flash if the button is not pressed, that self-defense requires you to make the man not attack if you are not going to fight back? If the proper representation is by imperatives that demand a material conditional, then the answers should be affirmative. I do not think these are easy questions, however, and leave them to the reader to discuss and answer at his or her own discretion.

[22] For (*td-pcd*4,8), completeness of *PD* is immediate from the results in [14], [15].

6.4 The problem of permission

The definition of the deontic notion of permission in a context of conditional norms is troublesome.[23] For monadic deontic logic it is generally accepted to define (weak) permission through the absence of an obligation to the contrary, i.e. $PA =_{df} \neg O\neg A$. This has the additional effect of making $OA \vee P\neg A$ a tautology, and so there are not 'gaps' – any state of affairs is positively or negatively regulated. For dyadic operators, the analogue would be $P(A/C) =_{df} \neg O(\neg A/C)$. But this leads here to counterintuitive results: consider the set $I = \{p_1 \Rightarrow !p_2\}$, with the intended interpretation 'if you go out, wear your boots', and truth definitions (*td-pcd*1,2,3,5,6,7), i.e. those truth definitions that do not collapse into reasoning about the imperatives' materializations. For all these we have $\mathcal{I} \not\models O(p_1 \rightarrow p_2/\top)$, and so by the above definition we have $\mathcal{I} \models P(p_1 \wedge \neg p_2/\top)$. So you are permitted to go out without your boots. There are several proposals that overcome this difficulty. Von Wright, in his re-interpretation of deontic logic as rules for rational norm-giving from [46] onwards, has denied the interdefinability of obligation and permission altogether; his theory has the result that in the absence of explicit permissive norms we only have that OA implies PA, i.e. anything permitted is also obligatory. Quite similarly, Makinson & van der Torre [27] have proposed two definitions of conditional permission that, in the absence of explicit permissive norms, either make it coincide with obligation ('forward permission'), or come quite close to it, by demanding that by forbidding the behavior for the same condition, a conflict would be created for some situation ('backward permission'). All these approaches have the strange result that the less is obligatory, the less is allowed.[24] But surely one can, in some weak sense, say that given the presence of some (conditional) imperatives, an agent is still free to do A in a situation C, without saying that A is also obligatory in this situation. It is perhaps a better solution to define

$$\mathcal{I} \models P(A/C) \;\text{ iff }\; \exists \Gamma \in \mathscr{P}_{\mathcal{I}}(\{C\}, I) : \Gamma^m \cup \{C\} \nvdash_{PL} \neg A,$$

thus defining A as permissible in a situation C if there is a preferred maximally obeyable subset of the imperatives for which bringing about A does not cause a violation. For operators other than (*td-pcd*4,8), this definition is not 'gapless'. E.g. consider the set $I = \{!p_1, p_1 \Rightarrow !p_2\}$. For truth definitions that do not accept 'deontic detachment', i.e. (*td-pcd*1,2,3), we have neither $O(p_2/\top)$ nor $P(\neg p_2/\top)$: though we are not yet under an obligation to bring about p_2, we are also not permitted to bring about $\neg p_2$ and thus make satisfaction of the imperative impossible that ought to be triggered. Or consider conditional imperatives whose

[23] I do not consider here the problem of how permissive *norms*, or licenses, may be represented. For attempts to use a separate set of 'P-norms' alongside what is here the set of imperatives cf. Alchourrón & Bulygin [1], von Wright [46], Makinson [24] and Makinson & van der Torre [27].

[24] Consider the set $I = \{p_1 \Rightarrow !p_i \mid i > 1\}$, and for an interpretation suppose that I have no obligations in the rest of the world, but am a slave once I go to Australia. By 'forward' or 'backward' permission, $P(A/\neg p_1)$ is false for any A, i.e. I am not allowed to do anything if I do not go to Australia, and though $P(p_i/\top)$ holds for backward permission, it is only by virtue of p_i being obligatory down under.

consequent has become impossible to satisfy: for a set $I = \{p_1 \Rightarrow !p_2\}$ we do not have $O(\neg p_1/\neg p_2)$ for truth definitions other than (*td-pcd*4,8) since $p_1 \Rightarrow !p_2$ is not triggered in the situation $\neg p_2$, but it is also not permitted to trigger it, i.e. $P(p_1/\neg p_2)$ is not true. This deontic vagueness may indeed be adequate for such situations. Further study must determine if such a definition does not create counterintuitive results. But it is important to see that as far as reasoning about conditional norms is concerned, the old definitions of permission as the absence of prohibition, obligation as the absence of a permission to the contrary, and prohibition as the absence of permission, do no longer hold.

7 Conclusion

Reasoning about obligations when faced with different and possibly conditional imperatives is a part of everyday life. To avoid conflicts, these imperatives may be ordered by priority and then observed according to their respective ranks. The 'drinking and driving' case in the introduction presented an example of such natural reasoning. To provide a formal account is, however, additionally complicated by the fact that there are various and mutually exclusive intuitions about what belongs to the right definition of an 'obligation in the face of conditional imperatives', i.e. the definition of a deontic O-operator. Based on similar definitions of operators by Makinson & van der Torre [25], [26] for their 'input/output logic', but leaving the choice of the 'right' operator to the reader, I presented several proposals in sec. 3 for definitions of a dyadic O-operator, namely (*td-pcd*1-8). These were dependent on a choice of 'preferred subsets' among a given set of prioritized conditional imperatives. A particularly successful method to identify such subsets, but applying to unconditional imperatives only, was Brewka's [4] definition of 'preferred subtheories' within a theory. In sec. 4 I discussed various approaches that extend this method to conditional imperatives, but these failed to produce satisfactory results for a number of given examples. In sec. 5 I first examined an approach that 'tailors' the choice procedure to the truth definition for the deontic O-operator in question, where the only criterion is to avoid the truth of $O(\neg C/C)$ for possible circumstances C. Though this finally produced the intended results, it did so for truth definitions (*td-pcd*4-8) only, whereas counterexamples remained for any of the weaker truth definitions (*td-pcd*1-3). I then argued that the solution is to adapt Brewka's method in such a way that it constructs, instead of maximal subsets of imperatives that are collectively satisfiable by an agent, maximally *obeyable* subsets of the imperatives. I showed that this new proposal provides adequate solutions to all of the examples, and in particular the 'drinking and driving' example is resolved in a satisfactory fashion for all of the discussed deontic operators. In sec. 6 I demonstrated that the new proposal also includes satisfactory results for benchmark examples developed for non-prioritized conditional imperatives; I presented theorems of a deontic logic based on this proposal (though the question of their completeness had to be left open), and finally I showed that there are problems for the representation of conditional imperatives and difficulties for the definition of a deontic P-operator that further philosophical discussion and research must address.

References

1. Alchourrón, C. E. and Bulygin, E., "The Expressive Conception of Norms", in [19], 95–124.
2. Alchourrón, C. E. and Makinson, D., "Hierarchies of Regulations and Their Logic", in [19] 125–148.
3. Belzer, M., "Legal Reasoning in 3-D", *Proceedings of the First International Conference in Artificial Intelligence and Law*, Boston: ACM Press, 1987, 155–163.
4. Brewka, G., "Preferred Subtheories: An Extended Logical Framework for Default Reasoning", in: Sridharan, N. S. (ed.), *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence IJCAI-89, Detroit, Michigan, USA, August 20-25, 1989*, San Mateo, Calif.: Kaufmann, 1989, 1043–1048.
5. Brewka, G., "Reasoning about Priorities in Default Logic", in: Hayes-Roth, B. and Korf, R. E. (eds.), *Proceedings of the 12th National Conference on Artificial Intelligence, Seattle, WA, July 31st - August 4th, 1994*, vol. 2, Menlo Park: AAAI Press, 1994, 940–945.
6. Brewka, G. and Eiter, T., "Preferred Answer Sets for Extended Logic Programs", *Artificial Intelligence*, **109**, 1999, 297–356.
7. Chisholm, R. M., "Contrary-to-Duty Imperatives and Deontic Logic", *Analysis*, **24**, 1963, 33–36.
8. Downing, P., "Opposite Conditionals and Deontic Logic", *Mind*, **63**, 1959, 491–502.
9. Forrester, J. W., "Gentle Murder, or the Adverbial Samaritan", *Journal of Philosophy*, **81**, 1984, 193–197.
10. van Fraassen, B., "Values and the Heart's Command", *Journal of Philosophy*, **70**, 1973, 5–19.
11. Goble, L., "A Logic for Deontic Dilemmas", *Journal of Applied Logic*, **3**, 2005, 461–483.
12. Greenspan, P., "Conditional Oughts and Hypothetical Imperatives", *Journal of Philosophy*, **72**, 1975, 259–276.
13. Hansen, J., "Problems and Results for Logics about Imperatives", *Journal of Applied Logic*, **2**, 2004, 39–61.
14. Hansen, J., "Conflicting Imperatives and Dyadic Deontic Logic", *Journal of Applied Logic*, **3**, 2005, 484–511.
15. Hansen, J., "Deontic Logics for Prioritized Imperatives", *Artificial Intelligence and Law*, 2005, *forthcoming*.
16. Hansson, B., "An Analysis of Some Deontic Logics", *Nôus*, **3**, 1969, 373–398. Reprinted in [18], 121–147.
17. Hansson, S. O., *The Structure of Values and Norms*, Cambridge: Cambridge University Press, 2001.
18. Hilpinen, R. (ed.), *Deontic Logic: Introductory and Systematic Readings*, Dordrecht: Reidel, 1971.
19. Hilpinen, R. (ed.), *New Studies in Deontic Logic*,Dordrecht: Reidel, 1981.
20. Hofstadter, A. and McKinsey, J. C. C., "On the Logic of Imperatives", *Philosophy of Science*, **6**, 1938, 446–457.
21. Horty, J. F., "Reasoning with Moral Conflicts", *Noûs*, **37**, 2003, 557–605.
22. Horty, J. F., "Defaults with Priorities", 2006. Draft version of August 18, 2006, `http://www.umiacs.umd.edu/~horty/articles/2005-dp.pdf`.
23. Kraus, S., Lehmann, D. and Magidor, M., "Nonmonotonic Reasoning, Preferential Models and Cumulative Logics", *Artificial Intelligence*, **44**, 1990, 167–207.

24. Makinson, D., “On a Fundamental Problem of Deontic Logic”,in: McNamara, P. and Prakken, H. (eds.), *Norms, Logics and Information Systems*, Amsterdam: IOS, 1999, 29–53.
25. Makinson, D. and van der Torre, L., “Input/Output Logics”, *Journal of Philosophical Logic*, **29**, 2000, 383–408.
26. Makinson, D. and van der Torre, L., “Constraints for Input/Output Logics”, *Journal of Philosophical Logic*, **30**, 2001, 155–185.
27. Makinson, D. and van der Torre, L., “Permissions from an Input/Output Perspective”, *Journal of Philosophical Logic*, **32**, 2003, 391–416.
28. Marek, V. W. and Truszczyński, M., *Nonmonotonic Logic. Context-Dependent Reasoning*, Berlin: Springer, 1993.
29. Nebel, B., “Belief Revision and Default Reasoning: Syntax-Based Approaches”, in: Allen, J. A. and Fikes, R. and Sandewall, E. (eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference, KR ’91, Cambridge, MA, April 1991*, San Mateo: Morgan Kaufmann, 1991, 417–428.
30. Nebel, B., “Syntax-Based Approaches to Belief Revision”, in: Gärdenfors, P. (ed.), *Belief Revision*, Cambridge: Cambridge University Press, 1992, 52-88.
31. Prakken, H., *Logical Tools for Modelling Legal Argument*, Dordrecht: Kluwer, 1997.
32. Prakken, H. and Sartor, G., “Argument-based Logic Programming with Defeasible Priorities”, *Journal of Applied Non-classical Logics*, **7**, 1997, 25–75.
33. Prakken, H. and Sergot, M., “Contrary-to-duty obligations”, *Studia Logica*, **52**, 1996, 91–115.
34. Rescher, N., *Hypothetical Reasoning*, Amsterdam: North-Holland, 1964.
35. Rescher, N., *The Logic of Commands*, London: Routledge & Kegan Paul, 1966.
36. Rintanen, J., “Prioritized Autoepistemic Logic”, in: MacNish, C. and Pearce, D. and Pereira, L. M., *Logics in Artificial Intelligence, European Workshop, JELIA ’94, York, September 1994, Proceedings*, Berlin: Springer, 1994, 232–246.
37. Ross, W. D., *The Right and the Good*, Oxford: Clarendon Press, 1930.
38. Ryan, M., “Representing Defaults as Sentences with Reduced Priority”, in: Nebel, B. and Rich, C. and Swartout, W. (eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference, KR ’92, Cambridge, MA, October 1992*, San Mateo: Morgan Kaufmann, 1992, 649–660.
39. Sakama, C. and Inoue, K., “Representing Priorities in Logic Programs”, in: Maher, M. (ed.), *Joint International Conference and Syposium on Logic Programming JICSLP 1996, Bonn, September 1996*, Cambridge: MIT Press, 1996, 82–96.
40. Ș osa, E., “The Logic of Imperatives”, *Theoria*, **32**, 1966, 224–235.
41. Świrydowicz, K., “Normative Consequence Relation and Consequence Operations on the Language of Dyadic Deontic Logic”, *Theoria*, **60**, 1994, 27–47.
42. van der Torre, L., *Reasoning About Obligations*, Amsterdam: Thesis Publ., 1997.
43. Tröndle, H., “Die Wahlfeststellung”, in: *Strafgesetzbuch. Leipziger Kommentar*, vol. 1, 10th ed., Berlin: Walter de Gruyter, 1985, §1, margin nos. 59–63.
44. von Wright, G. H., “A New System of Deontic Logic”, *Danish Yearbook of Philosophy*, **1**, 1961, 173–182. Reprinted in [18], 105–115.
45. von Wright, G. H., *An Essay in Deontic Logic and the General Theory of Action*, Amsterdam: North Holland, 1968.
46. von Wright, G. H.: “Norms, Truth and Logic”, in: von Wright, G. H., *Practical Reason: Philosophical Papers vol. I*, Oxford: Blackwell, 1983, 130–209.
47. von Wright, G. H.: “Bedingungsnormen, ein Prüfstein für die Normenlogik”, in: Krawietz, W., Schelsky, H., Weinberger, O. and Winkler, G. (eds.), *Theorie der Normen*, Berlin: Duncker & Humblot, 1984, 447–456.

Spatially Distributed Normative Objects

Fabio Y. Okuyama[1], Rafael H. Bordini[2], and Antônio Carlos da Rocha Costa[3]

[1] Universidade Federal do Rio Grande do Sul, Brazil. okuyama@inf.ufrgs.br
[2] University of Durham, United Kingdom. R.Bordini@durham.ac.uk
[3] Universidade Católica de Pelotas, Brazil. rocha@atlas.ucpel.tche.br

Abstract. Organisational structures for multi-agent systems are usually defined independently of any spatial or temporal structure. Therefore, when the multi-agent system is situated in a spatial environment, there is usually a conceptual gap between the definition of the system's organisational structures and the definition of the environment. In this paper, we focus on a mechanism for the spatial distribution of an organization's normative information. Spatially distributing the normative information over the environment is a natural way to simplify the definition of organisational structures and the development of large-scale multi-agent systems. By distributing the normative information in different spatial locations, we allow agents to directly access the relevant information needed in each environmental context. We extend our previous work on a language for modelling multi-agent environments in order to allow for the definition of spatially distributed norms in the form of *normative objects*.

1 Introduction

The environment is an important part of a Multi-Agent System (MAS), specially for systems of situated agents. Situated multi-agent systems are usually designed as a set of agents, together with the environment where they operate, their social structures, and the possible interactions among these components. In previous works, we introduced a language that allows MAS designers to describe, at a high conceptual level, environments for situated multi-agent systems [11, 1]. The language is called ELMS, and was created to be part of a platform for the development of (social) simulations based on multi-agent systems.

In this paper, we present extensions to the ELMS language which allow the distribution of normative information over an environment, construing what we call *situated norms*. In particular, we introduce here the notion of spatially distributed *normative objects*, which facilitates the modelling of various real-world situations, particularly for simulation, but more generally the coordination of large-scale multi-agent systems too, through situated norms.

To understand the notions of normative object and situated norm, consider the posters one typically sees in public places (such as libraries or bars) saying "Please be quiet" or "No smoking in this area". Human societies often resort to this mechanism for decentralising the burden of regulating social behaviour; people then adopt such situated norms whenever they have visual access to such posters. This should be equally efficient for computational systems because it avoids the need for providing a complete,

In G. Boella et al. (edts), Proceedings of the International Workshop on Coordination, Organization, Institutions and Norms in Agent Systems (COIN), held with ECAI, Riva del Garda, Italy, 28 August 2006.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/903

exhaustive representation of all social norms in a single public structure, known to all agents, as it is usually the case in current approaches to agent organisations.

Another extension we have introduced to our environment description language is the notion of normative places, which are zones where the normative objects and situated norms are relevant. As an example, consider a research group where there are agents with the role *principal researcher* whose main objective is to supervise the research of agents playing the *research student* role; such research can be conducted both at the laboratory or at the library. The interactions at the laboratory are to be outlined in the spatial scene of the laboratory space. The information about how to behave in a library is defined in the library spatial scene, where all researchers will also assume the role of *library users*. Normative information relevant for each such site (and each place within each site) can be made accessible to the agents with the help of normative objects.

In summary, the extensions we introduce here support situated norms and leaves the necessary room for the inclusion of group structures that are spatially situated within a (simulated) physical environment. This is done using two means: first, *normative objects*, which are objects that can contain normative information; and second, a normative principle for *situated norms*, conceived as a special form of conditional rule, where an explicit condition on an agent's perception of a normative object appears: 'When playing the relevant role and being physically situated within the confines referred by a situated norm $\mathcal{N}$ expressed in a normative object previously perceived, the agent is required to reason about following norm $\mathcal{N}$; otherwise, it is excused from reason about it'. Also, normative objects may be directed towards a specific role in a given organisation. We can thus model things such as a sign saying that students are not allowed beyond the library desk (while members of staff are).

In the next section, we briefly present our platform and the various component languages we use to model multi-agent systems. In Section 3, we briefly review how an environment should be modelled using our approach. In Sections 4 and 5, we present and discuss the normative extensions that we introduce in this paper. We then illustrate our approach with an example in Section 6; the example is based on the scenario presented in [4]. We discuss related work in Section 7, then conclude the paper.

2 The MAS-SOC Platform

One of the main goals of the MAS-SOC simulation platform (**M**ulti-**A**gent **S**imulations for the **SOC**ial Sciences) is to provide a framework for the creation of agent-based simulations which do not require too much experience in programming from users, yet allowing the use of state-of-the-art agent technologies. In particular, it should allow for the design and implementation of simulations with *cognitive* agents.

In our approach, an agent's individual reasoning is specified in an extended version of AgentSpeak [13], as interpreted by ***Jason***, an *open source* agent platform[4] based on Java [2]. The extensions allow, among other things, the use of speech-act based agent communication, and there is ongoing work to allow the use of ontologies and of organisational structures as part of a ***Jason*** multi-agent system.

[4] Available at `http://jason.sf.net`.

The environments where agents are situated are specified in ELMS, a language we have designed for the description of multi-agent environments [11]. For more details on MAS-SOC, refer to [1]. We here concentrate on the ELMS extensions to describe basic organisational structures and social norms, and to relate an organisational structure and the relevant normative aspects to the spatial structures defined within the physical environment.

3 Modelling Physical Environments with ELMS

As presented in [11], we developed ELMS (**E**nvironment Description **L**anguage for **M**ulti-Agent **S**imulation) as a means to describe environments and to execute simulated environments. Agents in a multi-agent system interact with the environment where they are situated and interact with each other (possibly through the shared environment). Therefore, the environment has an important role in a multi-agent system, whether the environment is the Internet, the real world, or some simulated environment.

We understand as environment modelling, the modelling of external aspects that an agent needs as input to its reasoning and for deciding on its course of action. Further, it is necessary to model explicitly the physical actions and perceptions that the agents are capable of in a given environment. Below we briefly review how a physical environment is described using this language.

To define an environment using ELMS, the following classes of constructs can be used:

- **Agent Body:** the agent's characteristics that are perceptible to other agents. Agent "bodies" are defined by a set of properties that characterise it and are perceptible to other agents. Such properties are represented as *string*, *integer*, *float*, and *boolean* values. Each "body" is associated with a set of actions that the agent is allowed to perform and of environment properties that the agent can perceive.
- **Agent Sensorial Capabilities (Perception):** the environment properties that will be perceptible to each agent at a given time, and under given specific circumstances.
- **Agent Effective Capacities (Actions):** the actions that an agent is able to perform in order to change the current state of the environment. These actions are defined as assignments of values to the attributes of entities in the environment[5]. The production (i.e., instantiation) of previously defined resources (i.e., objects), and the consumption (i.e., deletion) of existing instances can also be part of an action description.
- **Physical Environment Objects (Resources):** the objects/resources that are present in the environment. Although objects and resources can have conceptual differences, they are represented by the same structure in ELMS. Agents interact with objects through their actions in the environment. Object structures are defined by a set of properties that are relevant to the modelling and may be perceived by an agent. In the same way as the properties of the "bodies" of the agents, the properties of objects are also represented by *string*, *integer*, *float*, and *boolean* values. Each object can also be associated with a set of reactions that may happen as consequence of an agent's actions.

[5] Note that agent bodies are also properties of the environment.

- **Object Reactions:** the objects can "react", under specific circumstances, in order to respond to actions performed by the agents in the environment. Such reactions are given as the assignment of values to properties, the creation of previously defined object instances, and the deletion of existing object instances.
- **Space Structure (Grid):** the space is (optionally) divided into cells forming a grid that represents the spatial structure of the environment. When a grid is used, it can be defined in 2 or 3 dimensions. As for resources, each cell can have reactions associated to it. Although the specified set of reactions apply to all of the cells, this does not mean that all cells will behave equally, since they may be in different contexts (i.e., each cell has independent attributes, thus having different contents and, clearly, different positions, which can all affect the particular reactions).

3.1 Notes on Environment Descriptions

- **Perceptions:** agents do not normally have complete access to the environment. Perception of the environment will not normally give complete and accurate information about the whole environment and the other agents in it. However, since such restriction is not imposed by the ELMS model itself, designers can choose to create fully accessible environments if this is appropriate for a particular application.
- **Actions:** actions defined here are assumed to be atomic, as the action chaining or planning is meant to be part of the "mind" of the agent
- **Reactions:** all object reactions triggered by some change in the environment are executed in a single simulation cycle. This is different from agent actions, as each agent can execute only one action per cycle.

Additionally to the constructs mentioned above, the following operational constructs are used in our approach to model the (simulated) physical environment.

- **Constructors:** Each agent and resource may need to be initialised at the moment of its instantiation. This is defined by a list of initial value assignment to its attributes.
- **Observables:** A list of environment properties whose values are to be displayed/logged; these are the specific properties of a simulation that the user wants to observe/analyse.

The simulation of the environment itself is done by a process that controls the access and changes made to the data structure that represents the environment; the process is called the *environment controller*. The data structure that represents the environment is generated by the ELMS interpreter from a specification in ELMS given as input. In each simulation cycle, the environment controller sends to all agents currently taking part in the simulation the percepts to which they have access (as specified in ELMS). Recall that ELMS environments are designed for cognitive agents, so perception is transmitted in messages as a list of ground logical facts. After sending perception, the process waits for the actions that the agents have chosen to perform in that simulation cycle and then execute the actions, changing the environment data structures accordingly.

4 Normative Objects and Situated Norms

Typically, environments will have some objects aimed at informing agents about norms, give some advice, or warn about potential dangers. For example, a poster fixed on a wall of a library asking for "silence" is an object of the environment, but also informs about a norm that should be respected within that space. Another example are traffic signs, which give advice about directions or regulate priorities in crossings. The existence of such signs, that we call *normative objects*, implies the existence of a regulating code in such context, that we call *situated norms*.

In the examples above, the norms are only meant to be followed within certain boundaries of space or time and lose their effect completely if those space and time restrictions are not met, which is the initial motivation for situated norms. Another important advantage of modelling some norms as situated norms is the fact that the spatial context where the norm is to be followed is immediately determined. Thus, the norm can be "pre-compiled" to its situated form, making it easier for the agents to operationalise the norm, and also facilitating the verification of norm compliance.

For example a norm that says "Be kind to the elderly", may be quite hard to operationalise and verify, in general. However, in a fixed spatial context such as a bus or train, with the norm contextualised as "Give up your seat for the elderly", or in a street crossing, with the norm contextualised as "Help elderly people to cross the street", the norm would be much more easily interpreted by the agents, and verified by any norm compliance checking mechanism.

It is important to remark that the norm-abiding behaviour is not related to the existence of a normative object. Beyond the existence of such object, it is necessary for the agent to perceive the normative object, and autonomous agents will also reason about whether to follow or not the norm stated by the normative object.

4.1 Normative Objects and Situated Norms in ELMS

In the extended version of ELMS, normative objects are "readable" by agents under specific individual conditions: an agent is able to read a specific rule if it has the specific ability to perceive the type of object in which the rule is written at its given location. In the most typical case, the condition is simply being physically close to the object.

Normative objects can be defined before the simulation starts, or can be created dynamically during the simulation. Each normative object can be placed in a normative place (see below), in the spatial grid of the environment. The conditions under which the normative objects can be perceived are defined by the simulation designer using the usual ELMS constructs for defining perception capabilities and their conditions.

The normative information in a normative object is "read" by an agent through its perception ability. Besides the norm itself, it may contain meta-information, e.g., which agent or institution created the norm. In ELMS, normative objects should have at least the following properties:

- **Type:** the type of the normative information contained in the object; it determines the level of importance (e.g., a warning, an obligation, a direction);

- **Issued by:** where the power underlying the norm comes from (e.g., an agent, a group, an institution).
- **Norm:** a string that represents the normative information; this should be in the format of AgentSpeak predicates in the case of MAS-SOC environments, or whatever format the targeted agents will be able to understand.
- **Placement:** the set of normative spaces where the normative information applies. If omitted, the object is assumed to be accessible from anywhere, but normally under conditions determined by the designer; see the next item.
- **Condition:** conditions under which the normative information can be perceived. The conditions can be associated with groups, roles, abilities, and current physical placement and orientation of agents and objects.
- **Id:** identification string for eventual deletion/edition of the normative object.

We now briefly describe how the agents will receive normative information from normative objects. Whenever the agent position is such that access to the normative object is accessible, and the **Condition** is satisfied, the agent will receive perception of the form:

rule([PLACE],[GROUP],[ISSUED BY],[NORM])
Ex: rule(home, family, parents, obligation(child,play(TOY),tidy(TOY)))

The example above can be read as: "This is rule in group *family*, issued by the *parents*, with application at the normative place *home* (see below), that says: if the action $play(TOY)$ is done by an agent of role $child$, then it is an obligation of that agent to do $tidy(TOY)$ as well".

A rule like that would not normally be posted on a sign in a family home, but it illustrates the more general idea of situated norms as norms that apply within given environmental locations.

4.2 Normative Places in ELMS

Considering the ideas discussed above about normative objects and situated norms, ELMS descriptions of an environment (based on the concepts of agent bodies, objects, and an optional grid) need to be extended with the notion of *normative places*, i.e., a set of cells where an organisational activity occurs under the conditions of a set of situated norms.

A *normative place* can be defined in ELMS simply by its name and the set of cells that are part of it. A normative place may have intersections with other normative places, or even be contained within another normative place. For example, a normative place "school" may have a large set of cells where some of those cells refer to a normative place "classroom" and others to its "library". A normative place allows for the definition of the spatial location where certain situated norms are valid and relevant, as it will be exemplified in the next section.

In order to facilitate the definition of repetitive normative place structures, classes of normative places can be first defined and then instantiated in specific positions of the grid. The place "home" in the previous section is an example of a normative place. Other examples of such definitions and instantiations are as follows:

```
<NORMATIVE-PLACE-TYPE NAME="library"/>

<NORMATIVE-PLACE-TYPE NAME="classroom"/>

<PLACE NAME="lib1" NORMATIVE-PLACE-TYPE="library">
     <CELL X="0" Y="0"/>
     <CELL X="0" Y="1"/>
</PLACE>

<PLACE NAME="cr1" NORMATIVE-PLACE-TYPE="classroom">
     <CELL X="2" Y="0">
</PLACE>
```

5 MAS-SOC Modelling of Organisations Governed by Situated Norms

As the MAS-SOC platform does not enforce a particular agent-oriented software engineering methodology, designers can use the one they prefer. It is possible to model a multi-agent system that will have an ELMS environment using any approach: starting from the system organisation (top-down), or starting from the agent interactions (bottom-up).

In both approaches, the modelling of the organisational structures and the agents' reasoning need fine tuning to achieve the desired results. To have a stable point on which to base the tuning-up of the agents' reasoning or the organisational model, we have suggest the use of an explicitly defined environment description written in the ELMS language and the concepts presented in the Section 3. The environment is an important part of an multi-agent system, and although it can be very dynamic, in regards to design it is usually the most "stable" part of the system.

Based on these observations, we suggest that the multi-agent system modelling starts with the environment definition, followed by the definition of the normative places. The environment modelling is achieved by:

1. Definition of which kinds of action each type of agent is able to perform in the environment. Actions typically produce effects over objects of the environment or other agents.
2. Based on the changes that the agents' effective capabilities are able to make in the environment and the objectives of the simulation, the size and granularity of the grid can be determined. For example, how many cells an agent can move within one action or simulation cycle, and in how many simulation cycles the agent would be able to traverse the simulated space.
3. Based on the granularity and size of the spatial environment, the sensorial capabilities of the agents can be modelled, defining for example in which range an agent can detect other agents or objects.
4. Based on an agent's sensorial capabilities and on its typical activities, it should be possible to define which attributes of that agent is important to declare as accessible to other agents. For example, if agents identify each other's role by the colour of their uniform, the "agent body" should have an attribute that represent the colour of the agent's uniform.

5. The types of objects or resources present in the environment should also be modelled based on which attributes will be perceptible by the agents and which actions can affect them.
6. Finally, instances of the agent and object classes should be placed in the environment, determining its initial state.

The definition of the environment should be followed by the definition of normative places and then by the definition of the spatially distributed normative objects, as follows:

1. Together with the object types placed in the environment, the types of normative places within the environment can also be defined.
2. By instantiating normative places into sets of cells, *normative places* are created.
3. Then, based the set of activities that can possibly be performed in each type of normative place, the norms that are relevant to that type of place can be defined.
4. Finally, the types of *normative objects* can be defined and instantiated in the normative places, defining the locations where situated norms can be perceived.

Using the environment as a basis, the agents' reasoning capabilities can then be defined so as to help agents achieve their objectives as well as the objective of the groups in which they participate. Also, the detailed definitions of possible organisational structures can be fine-tuned, in order to have the system achieving its overall objectives. In MAS-SOC, we use AgentSpeak to define the practical reasoning for each agent; in particular, we use the extended version of AgentSpeak as interpreted by ***Jason***; for details, see [3].

6 Example

Below we give an example showing how normative objects are defined using our approach. It is based on the scenario presented in [4], a scenario in which the agents are placed on an environment where they may eat the food they find, challenge other agents for their food, or move in search of food.

In this scenario, an agent owns any food item that is near to itself (at a distance of up to 2 cells). The agents can "see" food and other agents in a radius of 1 cell, but can sense food in a radius of 2 cells. The physical space is represented by a grid of 10×10 cells.

The norms used in that scenario essentially concern the respect for the ownership of a food item, which means they prescribe non-aggressive behaviour. In the original scenario, the norms were valid throughout the grid, but in this example norms are valid only within normative places, as indicated by normative objects.

A shortened version of the physical environment description in ELMS is given below.

```
<!DOCTYPE ENVIRONMENT SYSTEM "elms.dtd">
<ENVIRONMENT NAME = "NORMATIVE">
    <DEFGRID SIZEX="10" SIZEY = "10"/>

    <RESOURCE NAME="food">
```

```
       <STRING  ownner = "none">
    </RESOURCE>

    <AGENT_BODY NAME="agent">
        <INTEGER NAME = "id"> "SELF" </INTEGER>
        <PERCEPTIONS>
           <ITEM NAME = "vision"/>
           <ITEM NAME = "sense_food">
        </PERCEPTIONS>
        <ACTIONS>
           <ITEM NAME = "walk"/>
           <ITEM NAME = "attack"/>
           <ITEM NAME = "eat"/>
        </ACTIONS>
    </AGENT_BODY>

    <PERCEPTION NAME="vision">
        <CELL_ATT ATTRIBUTE="CONTENTS" ABSOLUTE="TRUE">
            <X> +0</X>
            <Y> +0</Y>
        </CELL_ATT>
        <CELL_ATT ATTRIBUTE="CONTENTS" ABSOLUTE="TRUE">
            <X> +1</X>
            <Y> +0</Y>
        </CELL_ATT>
        <!-- shortened-->
    </PERCEPTION>

    <PERCEPTION NAME="sense_food">
        <!-- shortened-->
    </PERCEPTION>

    <ACTION NAME="eat">
        <PARAMETER NAME = "FOOD_ID" TYPE="INTEGER" />
        <!-- shortened-->
    </ACTION>

    <ACTION NAME="walk">
         <!-- shortened-->
    </ACTION>

    <ACTION NAME="attack">
         <!-- shortened-->
    </ACTION>

   <INITIALIZATION>
         <!-- instantiation and placement of
              food and agents -->
   </INITIALIZATION>
</ENVIRONMENT>
```

In the code excerpt above, the grid size is defined, then `food` is defined as an environment resource, then a generic type of agent body is defined. The agent body is defined as being capable of two types of perception — vision and food sensing – and being able to perform three types of actions: walk, attack, and eat. The `vision` perception allows the agent to perceive the contents of the current cell and the 4 neighbouring cells, while `sense_food` allows it to perceive food within a 2-cell radius.

For this example, the grid is partitioned in four normative places of equal sizes, and the normative objects are defined and placed only in the upper-left and upper-right quadrants, as shown in the code excerpt below:

```
    <NORMATIVE-PLACE-TYPE NAME="food-protected"/>
```

```
<PLACE NAME="upper-left" NORMATIVE-PLACE-TYPE="food-protected">
   <CELL X="0" Y="0"/><CELL X="1" Y="0"/>
   <!-- shortened-->
   <CELL X="3" Y="4"/><CELL X="4" Y="4"/>
</PLACE>

<PLACE NAME="upper-right" NORMATIVE-PLACE-TYPE="food-protected">
   <CELL X="5" Y="0"/><CELL X="6" Y="0"/>
   <!-- shortened-->
   <CELL X="8" Y="4"/><CELL X="9" Y="4"/>
</PLACE>

<PLACE NAME="lower-left" NORMATIVE-PLACE-TYPE="null">
   <CELL X="0" Y="5"/><CELL X="1" Y="5"/>
   <!-- shortened-->
   <CELL X="3" Y="9"/><CELL X="4" Y="9"/>
</PLACE>

<PLACE NAME="lower-right" NORMATIVE-PLACE-TYPE="null">
   <CELL X="5" Y="4"/><CELL X="6" Y="4"/>
   <!-- shortened-->
   <CELL X="8" Y="9"/><CELL X="9" Y="9"/>
</PLACE>

<NORMATIVE_OBJECT ID="norm1" TYPE="prohibition" PLACE = "upper-left">
   <NORM>prohibited(true,attack(SELF,AGENT))</NORM>
</NORM_OBJ>

<NORMATIVE_OBJECT ID="norm2" TYPE="prohibition" PLACE = "upper-right">
   <NORM>prohibited(not(owned(FOOD,SELF)),eat(SELF,FOOD))</NORM>
</NORM_OBJ>
```

The norms in the above example are very simple, and are given simply to illustrate how they can be modelled in our approach. For instance, `norm1` says that an agent ought not to attack (steal food from) another agent, while `norm2` says that the agent ought not to eat a food item that is not owned by itself.

Clearly, the agents' behaviour will be different in the four quadrants of the environment:

- in the upper-left quadrant, an agent is barred from eating food that belongs to another agent (since the situated norm states that an agent is prohibited from stealing food);
- in the upper-right quadrant, agents are supposedly prohibited of doing that, but not effectively, since the situated norm only prohibits the eating of food that is not owned by the agent itself rather than the stealing of food, so an agent can eat food that previously belonged to another agent if it first manages to steal that food;
- the lower quadrants (both left and right) are lawless areas, where agents are completely free to attack each other and to eat anyone else's food.

Notice that `prohibited` is used as a conditional deontic operator, with two arguments: the first argument is a condition to be tested, the second argument is the action that is prohibited.

7 Related Work

The notion of artifacts [16] and coordination artifacts [12] resembles our notion of *normative objects*. As defined in [12], coordination artifacts are abstractions meant to improve the automation of coordination activities, being the building blocks to create effective shared collaborative working environments. They are defined as runtime abstractions that encapsulate and provide a coordination service to the agents. Artifacts [16] were presented as a generalisation of coordination artifacts. Artifacts are an abstraction to represent tools, services, objects and entities in a multi-agent environment.

As building blocks for environment modelling, artifacts encapsulate the features of the environment as services to be used by the agents. The main objective of a coordination artifacts is to be used as an abstraction of an environmental coordination service provided to the agents. However, coordination artifacts express normative rules only implicitly, through their practical effects on the actions of the agents, and so their normative impact does not require any normative reasoning from the part of the agents. In our work, rather than having a general notion of objects that by their (physical) properties facilitate coordination, *normative objects* are objects used specifically to store *symbolic* information that can be interpreted by agents, so that they can become aware of norms that should be followed within a well-defined location.

Our choice has the advantage of keeping open the possibility of agent autonomy, as suggested in [5]. Agents are, in principle, able to decide whether to follow the norms or not, when trying to be effective in the pursuit of their goals. This is something that is not possible if an agent's action can only happen if in accordance to norms enforced by coordination mechanisms.

Another important difference is that *normative objects* are spatially distributed over a physical environment, with a spatial scope where they apply, and closely tied to the part of the organisation that is physically located in that space. While the objective of the coordination artifacts is to remove the burden of coordination from the agents, our work tries to simplify the way designers can guide the behaviour of each individual agent as they move around an environment where organisations are spatially located; this allows agents to adapt the way they behave in different social contexts.

In [8], the authors present the AGRE model, an extension to the previous AGR model. These latest extensions allow the definition of structures that represents the physical space. The approach defines organisational structures (i.e., groups) and the physical structures (i.e., areas) as "specialisations" of a generic space. The social structures are not contextualised in the space as they are in our work, leaving the social and physical structures quite unrelated.

In ELMS, however, it is not possible to explicitly define social structures, even though it would be possible to implicitly define them through the norms. This is because the aim of ELMS is, as mentioned earlier, to allow for environmental infrastructures compatible with existing approaches to organisational modelling, not for the modelling of organisations as such; the combination of ELMS with existing approaches to modelling organisations is planned as future work.

Another important series of related work is that on Electronic Institutions [9]. The internal working of an electronic institutions is given (in a simplified view) as a state-machine where each state is called a "scene". Each scene specifies the set of roles that agents may perform in it, and a "conversation protocol" that the agents should follow when interacting in the scene. To traverse the series of scenes that constitute the operation of the electronic institution, agents must do a sequence of actions in each scene, and also to commit to certain actions in certain scenes, as the result of their having performed certain other actions in certain other scenes. Our notion of normative space was inspired by such notion of scene, through giving it a physical, spatial content.

Similar to the electronic institutions approach, there is work on *computational institutions* [14], which are defined as virtual organisations ruled by constitutive norms and regulative norms. In computational institutions, organisational modelling uses the abstraction of coordination artifacts as building blocks, in a way that is very similar to our use of normative objects in spatially distributed organizations, but still keeping implicit in coordination artifacts the normative content imposed on the agents.

8 Conclusions

In this paper we have extended the ELMS language for describing environments with the means to define normative structures that make part of an environment representation. There are currently many approaches to modelling and implementing multi-agent systems: some are top-down approaches with focus on the organisations, while bottom-up approaches focus on the agents. We believe that including environment modelling at the initial stages of both approaches would help the modelling and implementation of multi-agent systems. To help such modelling, we have proposed an approach with an explicit environment description which now also includes the notions of *situated norms*, *normative places*, and (spatially distributed) *normative objects*.

It is important to note that our work is not an approach for modelling the organisational dimension of a multi-agent system. With the definition of *normative places*, where group structures would be inserted, we intend to fill a conceptual gap between the usual ways in which organisations and physical environments are modelled. In future work, with the integration of current means for defining organisational structures with ELMS, and thus with the possibility of associating them to normative places, we hope to contribute to a more integrated approach to designing and implementing the various aspects of multi-agent systems: concentrating on one particular organisation section at a time, specially if it is an organisation section attached to a spatial location, makes it easier for designers to define the groups, roles and agent behaviour that should operate in that particular organisation section.

By distributing the normative information in the environment, it is possible to partition the environment in a functional way, thus helping the structured definition of large simulations, norms being associated only with the places where they are meant to be followed. It is also more efficient (by taking advantage of natural distribution) to have norms spread in an environment than having them in a repository made available for the whole society, as it is usually the case.

We believe that an explicit environment description is an important part of a multi-agent system because it is a stable point from where the agent reasoning and the organisational structures can be fine-tuned so as to facilitate the development of agents and organisations that can achieve their goals. The notion of *spatially distributed normative objects* that we have introduced here can be a good solution connecting definitions of organisations and definitions of environments. Additionally, distributing the organisational/normative information can facilitate the modelling of large organisations.

It is interesting to note that, being conditioned on the possibility of checking the existence of a normative object, the normative reasoning required from agents that deal with normative objects is necessarily of a non-monotonic nature, and the experience of programming such reasoning in AgentSpeak is something we plan to experiment with in the future. Also as future work, we intend to allow a normative place to be associated with group structures, creating a connection between the organisational structures and the physical environment. We plan to make possible such association for any existing approach to agent organisations, such as $\mathcal{M}$OISE$^+$ [10], OperA/OMNI [15], GAIA [17], or approaches based on electronic institutions [6, 7]. The recursive nature of normative places may not be compatible, however, with some of such approaches to organisation, where the (possibly implicit) system of normative rules has no provision for a recursive structure in its operation.

Acknowledgements

This work was partially supported by CNPq, CAPES, and FAPERGS. Rafael Bordini gratefully acknowledges the support of The Nuffield Foundation (grant number NAL/01065/G). The authors greatly benefited from the comments and suggestions that arose in the discussions during the presentation of the paper at the COIN@ECAI2006 workshop in Riva del Garda.

References

1. Rafael H. Bordini, Antônio Carlos da Rocha Costa, Jomi F. Hübner, Álvaro F. Moreira, Fabio Y. Okuyama, and Renata Vieira, 'MAS-SOC: a social simulation platform based on agent-oriented programming', *Journal of Artificial Societies and Social Simulation*, **8**(3), (2005).
2. Rafael H. Bordini, Jomi F. Hübner, et al., ***Jason**: A Java-based Interpreter for an Extended Version of AgentSpeak*, manual, release version 0.9 edn., July 2006. `http://jason.sourceforge.net/`.
3. Rafael H. Bordini, Jomi F. Hübner, and Renata Vieira, '***Jason*** and the Golden Fleece of agent-oriented programming', in *Multi-Agent Programming: Languages, Platforms and Applications*, eds., Rafael H. Bordini, Mehdi Dastani, Jürgen Dix, and Amal El Fallah Seghrouchni, chapter 1, Springer-Verlag, (2005).
4. C. Castelfranchi, R. Conte, and M. Paolucci, 'Normative reputation and the costs of compliance', *Journal of Artificial Societies and Social Simulation*, **1**(3), (1998). <http://www.soc.surrey.ac.uk/JASSS/1/3/3.html>.
5. Cristiano Castelfranchi, Frank Dignum, Catholijn M. Jonker, and Jan Treur, 'Deliberative normative agents: Principles and architecture', in *6th International Workshop on Intelligent Agents VI, Agent Theories, Architectures, and Languages (ATAL)*, Lecture Notes In Computer Science, Vol. 1757, pp. 364–378, Londo, (1999). Springer-Verlag.
6. Marc Esteva, David de la Cruz, and Carles Sierra, 'Islander: an electronic institutions editor.', in *AAMAS*, pp. 1045–1052. ACM, (2002).

7. Marc Esteva, Bruno Rosell, Juan A. Rodríguez-Aguilar, and Josep Lluís Arcos, ‘Ameli: An agent-based middleware for electronic institutions.’, in *AAMAS*, pp. 236–243. IEEE Computer Society, (2004).
8. Jacques Ferber, Fabien Michel, and José-Antonio Báez-Barranco, ‘Agre: Integrating environments with organizations.’, in *E4MAS*, pp. 48–56, (2004).
9. Andrés Garcia-Camino, Pablo Noriega, and Juan A. Rodríguez-Aguilar, ‘Implementing norms in electronic institutions.’, in *AAMAS*, eds., Frank Dignum, Virginia Dignum, Sven Koenig, Sarit Kraus, Munindar P. Singh, and Michael Wooldridge, pp. 667–673. ACM, (2005).
10. Jomi Fred Hübner, Jaime Simão Sichman, and Olivier Boissier, ‘$\mathcal{M}$OISE$^+$: Towards a structural, functional, and deontic model for MAS organization’, in *Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'2002), Bologna, Italy*, (2002).
11. Fabio Y. Okuyama, Rafael H. Bordini, and Antônio Carlos da Rocha Costa, ‘ELMS: An environment description language for multi-agent simulations’, in *Proceedings of the First International Workshop on Environments for Multiagent Systems (E4MAS), held with AAMAS-04, 19th of July*, eds., Danny Weyns, H. van Dyke Parunak, and Fabien Michel, number 3374 in Lecture Notes In Artificial Intelligence, pp. 91–108, Berlin, (2005). Springer-Verlag.
12. A. Omicini, A. Ricci, M. Viroli, C. Castelfranchi, and L. Tummolini, ‘Coordination artifacts: Environment-based coordination for intelligent agents’, in *AAMAS'04*, (2004).
13. Anand S. Rao, ‘AgentSpeak(L): BDI agents speak out in a logical computable language’, in *Proceedings of the Seventh Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'96), 22–25 January, Eindhoven, The Netherlands*, eds., Walter Van de Velde and John Perram, number 1038 in Lecture Notes in Artificial Intelligence, pp. 42–55, London, (1996). Springer-Verlag.
14. Rossella Rubino, Andrea Omicini, and Enrico Denti, ‘Computational institutions for modelling norm-regulated MAS: An approach based on coordination artifacts’, in *1st International Workshop “Agents, Norms and Institutions for Regulated Multi-Agent Systems” (ANI@REM 2005)*, eds., Gabriela Lindemann, Sascha Ossowski, Julian Padget, and Javier Vazquez-Salceda, AAMAS 2005, Utrecht, The Netherlands, (25 July 2005).
15. Javier Vázquez-Salceda, Virginia Dignum, and Frank Dignum, ‘Organizing multiagent systems.’, *Autonomous Agents and Multi-Agent Systems*, **11**(3), 307–360, (2005).
16. Mirko Viroli, Andrea Omicini, and Alessandro Ricci, ‘Engineering MAS environment with artifacts’, in *2nd International Workshop “Environments for Multi-Agent Systems” (E4MAS 2005)*, eds., Danny Weyns, H. Van Dyke Parunak, and Fabien Michel, pp. 62–77, AAMAS 2005, Utrecht, The Netherlands, (26 July 2005).
17. Michael Wooldridge, Nicholas R. Jennings, and David Kinny, ‘The GAIA methodology for agent-oriented analysis and design’, *Autonomous Agents and Multi-Agent Systems*, **3**(3), 285–312, (2000).

Specifying and Enforcing Norms in Artificial Institutions

Nicoletta Fornara[1], Marco Colombetti[1,2]

[1] Università della Svizzera italiana, via G. Buffi 13, 6900 Lugano, Switzerland
nicoletta.fornara@lu.unisi.ch
[2] Politecnico di Milano, piazza Leonardo Da Vinci 32, Milano, Italy
marco.colombetti@lu.unisi.ch

Abstract. In this paper we investigate two important and related aspects of the formalization of open interaction systems: how to specify norms, and how to enforce them by means of sanctions. The problem of specifying the sanctions associated with the violation of norms is crucial in an open system because, given that the compliance of autonomous agents to obligations and prohibitions cannot be taken for granted, norm enforcement is necessary to constrain the possible evolutions of the system, thus obtaining a degree of predictability that makes it rational for agents to interact with the system. In our model, norms are specified declaratively. When certain events take place, norms become active and generate pending commitments for the agents playing certain roles. Norms also specify the sanctions associated with their violation. In the paper, we analyze the concept of sanction in detail and propose a mechanism through which sanctions can be applied.

Keywords. Norms, Sanctions, Commitments, Artificial Institutions, Open Interaction Systems.

1 Introduction

In our previous works [1,2,3] we have presented a metamodel of artificial institutions called *OCeAN* (Ontology, CommitmEnts, Authorizations, Norms), which can be used to specify at a high level and in an unambiguous way *open interaction systems* where heterogeneous and autonomous agents may interact.

In our view open interaction systems and artificial institutions used to model them are a technological extension of human reality, that is, they are an instrument by which human beings can enrich the type and the frequency of their interactions and overcome geographical distance. Potential users of this kind of systems are artificial agents, that can be more or less autonomous in making decisions on behalf of their owners, and human beings using an appropriate interface. For example, it is possible to devise an electronic auction where the artificial agents are autonomous in deciding the amount of their bids, or an interaction system for the organization of conferences in which human beings (like the organizers, or the Program Committee members) act by means of artificial

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/909

agents that have a very limited level of autonomy. In any case it is important to remark that in every type of system there is always a stage when the software agents have to interface with their human owners to perform certain actions in the real world. For these reasons artificial institutions have to reflect, with the necessary simplifications, crucial aspects of their human counterparts. Therefore in devising our model we draw inspiration from an analysis of social reality [4] and from human legal theory [5].

In this paper we concentrate mainly on the operational specification of the normative component of artificial institutions. We will extend our *OCeAN* metamodel by dealing with the problems of giving a declarative specification of norms for open systems and of devising efficient and complete computational mechanisms for managing norms. In particular we aim at automating the detection of, and reaction to, the violations of norms. An important feature of our framework, with respect to other proposals [6,7,8,9,10] is that it gives a uniform solution to two crucial problems: the specification of norms and the definition of the semantics of an Agent Communication Language. Indeed, our model of norms relies on the notion of commitment [11], that has been previously introduced to express the meaning of a library of communicative acts [12]. We analyze in detail the problem of defining a mechanism for enforcing obligations and prohibitions by means of sanctions, that is, a treatment of the actions to be performed when a violation occurs, in order to deter agents from misbehaving and to secure and recover the system from an undesirable state. We speak of "obligation and prohibition enforcement" instead of "norm enforcement", as done in other approaches, because our proposal can be used to enforce obligations and prohibitions that derive either from predefined norms or from the autonomous performance of communicative acts. The problem of managing sanctions has been tackled in a few other works: for example, López y López et al. [9] propose to enforce norms using the "enforcement norms" that oblige agents entitled to do so to punish misbehaving agents; Vázquez-Salceda et al. [10] present, in the OMNI framework, a method to enforce norms described at a different level of abstraction; and Grossi et al. in [13] develop a high-level analysis of the problem of enforcing norms. Other interesting proposals introduce norms to regulate the interaction in open systems but, even when the problem of enforcement is considered to be crucial, do not investigate with sufficient depth why an agent ought to comply with norms and what would happen if compliance does not occur. For instance, Esteva et al. [7,8] propose ISLANDER, where a normative language with sanctions is defined but not discussed in detail, Boella et al. [14] model violations but do not analyze sanctions, and Artikis et al. [6] propose a model where the problem of norm enforcement using sanctions is mentioned but not fully investigated.

The paper is organized as follows: in Section 2 we briefly describe our metamodel for artificial institutions. In Section 3 the reasons why in open interaction frameworks it makes sense to allow for the violation of obligations and prohibitions are discussed, and then in Section 4 a proposal on how to enforce obligations and prohibitions by means of sanctions is presented. In Section 5 our model of norms is described and our previous construct of commitment is extended by adding the treatment of sanctions. In Section 6 we exemplify our proposal and finally in Section 7 we present the main conclusions that have been obtained.

2 The OCeAN metamodel

Our metamodel of artificial institutions as described in details in [1] consists mainly of the following components:

- The constructs necessary to define the *core ontology* of an institution, including: the notion of an *entity*, used to define the concepts introduced by the institution (e.g., the notion of a run of an auction with its attributes introduced by the institution of auctions); the notion of an *institutional action*, described by means of their preconditions and postconditions (e.g., the action of opening an auction, or declaring the current ask-price of an auction). The core ontology also defines the syntax of a list of *base-level actions*, like for instance the action of exchanging a message, whose function is to concretely execute institutional actions.
- Two fundamental concepts that are common to all artificial institutions and that are used in the definition of other constructs: the notions of a *role* and of an *event*. In particular roles are used in the specification of authorizations and norms, while the happening of events is used to bring about the activation of a norm or to specify the initial or final instance of a time interval.
- A *counts-as relation* that is necessary for the concrete performance of institutional actions. In particular, such relation relies on a set of *conventions* that bind the exchange of a certain message, under a set of contextual conditions, to the execution of an institutional action. Contextual conditions include *authorizations* that specify what agents are authorized to perform institutional actions. Authorizations are represented with the following notation: $Auth(role, iaction(parameters), conditions)$.
- The construct of *norm*, used to impose obligations and prohibitions to perform certain actions on agents interacting with the system. In our model, as will be described in Section 5, we have *declarative* norms that, when their activating event happens, are transformed into their operational counterpart, that is, a *commitment*.

3 Regimentation vs. Enforcement

In our model, as it will be discussed in more detail in Section 5, an active obligation is expressed by means of *commitments* to perform an action of a given type within a specified interval of time; similarly, an active prohibition is expressed by a commitment not to perform an action of a given type; moreover, every action is permitted unless it is explicitly forbidden. Note that a commitment can be created not only by the activation of a norm, but also by the performance of a communicative act [1], for instance by a promise.

In this section we briefly discuss the reasons why in open interaction systems it makes sense, and sometimes it is also inevitable, to allow for commitment violations, that happen when a prohibited action is performed or when an obligatory action is not performed within a predefined interval of time. The question is, Why should we give an agent the possibility to violate commitments? Why not adopt what in the literature is called "regimentation" [5], as proposed in [13], by introducing a control mechanism that does not allow agents to violate commitments?

To answer this question, it is useful to distinguish between *natural* (or physical) actions (like opening a door or physically delivering a product), whose effects take place thanks to nonconventional physical laws, and *institutional* actions (like opening an auction or transferring the property of a product), whose effects take place thanks to the common agreement of the interacting agents (more precisely, of their designers).

Regarding physical actions, it is important to remark that they cannot be regimented since, after they have been performed, they cannot be considered "void", that is, their effects cannot be annulled. Therefore it is impossible to use regimentation to prevent the violation of a prohibition to perform a given physical action.

Concerning institutional actions, the choice to allow for commitment violations or to impose regimentation is different in the case of obligations or prohibitions:

- Prohibitions can be expressed using two different mechanisms: (i) through the absence of authorization: in fact, when an agent performs a base-level action bound by a convention to an institutional action a_i, but the agent is not authorized to perform a_i, neither the "counts-as" relation nor the effects of a_i take place; (ii) through a commitment not to perform such an action: in this case, if the action is authorized, its effects take place but the corresponding commitment is violated. The solution to block the effects of certain actions by changing their authorizations during the life of the system is adopted for instance in AMELI (an infrastructure that mediates agent interactions by enforcing institutional rules) by means of *governors* [15], which filter the agents' actions letting only the allowed actions to be performed. However, this solution is not feasible when more than one institution contributes to the definition of an interaction system, as happens for example when the Dutch Auction and the Auction-House institutions contribute to the specification of an interaction system as presented in [2] and briefly recalled in Section 6. In such cases, an action authorized by an institution cannot be annulled by another institution, which at most can prohibit it.
- With respect to obligations, there is only one way to "regiment" the performance of an obliged action, that is, by making the system performing the obliged action instead of a misbehaving agent. But this solution is not always viable, especially when the agent has to set the values of some parameters of the action. For instance, the auctioneer of a Dutch Auction is repeatedly obliged to declare an ask price lower than the one previously declared, but

can autonomously decide the value of the decrement; therefore it would be difficult for the system to perform the action on behalf of the auctioneer. In any case it has to be taken into account that, even if the regimentation of obligations violates the autonomy of self-interested interacting agents, sometimes it can be adopted to recover the system from an undesirable state.

Finally it is important to remark that in an open system, where heterogeneous agents interact exhibiting self-interested behavior based on a hidden utility function, it is impossible to predict at design phase all the interesting and fruitful behaviors that may emerge. To reach an optimal solution for all participants [16] it may be profitable to allow agents to violate their obligations and prohibitions.

We therefore conclude that regimenting an artificial system so that violations of commitments are completely avoided is often impossible and sometimes even detrimental, since it may preclude interesting evolutions of the system towards results that are impossible to foresee at design time. It is also true, however, that in order to make the evolution of the system at least partially predictable, misbehavior must be reduced to a minimum. But then, how is it possible to deter agents from violating commitments? An operational proposal to tackle this problem, based on the notion of sanction, is described in the following sections.

4 Sanctions

In this section we briefly discuss the crucial role played by *sanctions* in the specification of an open interaction system. In the Merriam-Webster On Line Dictionary [1] a sanction is defined as "the detriment, loss of reward, or coercive intervention annexed to a violation of a law as a means of enforcing the law". In an artificial system, even if the utility function of the misbehaving agent is not known, sanctions can be devised:

- to deter agents from misbehaving bringing about a loss for them in case of violation, under the assumption that the interacting heterogeneous agents are human beings or artificial agents able to reason on sanctions;
- to compensate the institution or other damaged agents for their loss due to the misbehavior of the agents;
- to contribute to the security of the system, for example by prohibiting misbehaving agents to interact any longer with the system;
- to specify the acts that have to be performed to recover the system from an undesirable state [17].

When thinking about sanctions from an operational point of view, and in particular to the set of actions that have to be performed when a violation occurs, it is important to distinguish between two types of actions that differ mainly as far as their actors are concerned:

[1] <http://www.m-w.com>

- One crucial type of action that deserves to be analyzed in detail, and that is not taken into account in other proposals [9,10,8], consists of the actions that the misbehaving agent itself has to perform against a violation, and that are devised as a deterrent and/or a compensation for the violation. For instance, an unruly agent may have to pay a fine or compensate another agent for the damage. When trying to model this type of action it is important to take into account that it is also necessary to check that the compensating actions are performed and, if not, to sanction again the agent or, in some situations, to give it a new possibility to remedy the situation.
- Another type is characterized by the actions that certain agents are *authorized* to perform only against violations. In other existing proposals, for instance [9,10], which do not highlight the notion of authorization (or power [18]), those actions are simply the actions that certain agents are obliged to perform against violations. From our point of view, instead, the obligation to sanction a violation should be distinguished from the authorization to do so. The reason why authorizations are crucial is obvious: sanctions can only be issued by agents playing certain specific roles in an institution. But an authorization does not always carry an obligation with it.

In some situations, and in particular when the sanction is crucial for the continuation of the interaction, one may want to express the obligation for authorized agents to react to violations by defining an appropriate new norm. For instance, in the organization of a conference if a referee does not meet the deadline for submitting a review, the organizers are not only authorized, but also obliged to reassign the paper to another referee. The norm that may be introduced to oblige the agents entitled to do so to manage the violation is similar to the "enforcement norm" proposed in [9]: it has to be activated by a violation and its content has to coincide with the sanctions of the violated obligation or prohibition. This norm may in turn be violated, and it is up to the designer of the system to decide when to stop the potentially infinite chain of violations and sanctions, leaving some violation unpunished.

Regarding this aspect, to make it reasonable for certain agents (or for their owner) to interact with an open system, it has to be possible to specify that certain violations will definitely be punished (assuming that there are not software failures). One approach is to specify that the actor of the actions performed as sanctions for those violations is the *interaction-system* itself, that therefore needs to be represented in our model as a "special agent". By "special" we mean that such an agent will not be able to take autonomous decisions, and will only be able to follow the system specifications that are stated before the interaction starts. We call this type of agents *heteronomous* (as opposite to autonomous). Note that the given that the *interaction-system* can become, in an actual implementation, the actor of numerous actions performed as sanctions it would be better to implement it in a distributed manner in order to avoid that it becomes a possible bottleneck.

Examples of reasonable sanctions that can be inflicted by means of norms in an open artificial system are the decrement of the trust or reputation level of the agent (similar to the reduction of the driving licence points that is nowadays applied in some countries), the revocation of the authorization to perform certain actions or a change of role (similar to confiscation of the driving licence) or, as a final action, the expulsion of the agent from the system. Another type of sanction typical of certain contracts (i.e., sets of correlated commitments created by performing certain communicative acts) is the authorization for an agent to break its part of the contract, without incurring a violation, if the counterpart has violated its own commitments.

5 Norms

In an open system, norms are necessary to impose obligations and prohibitions to the interacting agents, in order to make the systems evolution at least partially predictable [19,20]. In particular, norms can be used to express interaction protocols as exemplified in [1,2], where the English Auction and the Dutch Auction are specified by indicating what agents can do, cannot do, and have to do at each state of the interaction. In this section we propose a development of the model of norms that we have presented in our previous works [1,2,3], which clearly separates the *declarative* form of norms from their *operational* counterpart, that is, *commitment*, and from the procedure to transform the former into the second.

Norms are taken as a specification of how a system ought to evolve. At design time, the main point is to guarantee that the system has certain crucial properties. This result can be achieved by formalizing obligations and prohibitions by means of logic and applying model checking techniques as studied in [21,22]. At run time, and from the point of view of the interacting agents, norms can be used to reason about the relative utility of future actions [23]. Still at run time, but from the point of view of the open interaction system, norms can be used to check whether the agents behavior is compliant with the specifications and able to suitably react to violations. Our model of norms is mainly suited for the last task.

Coherent with other approaches [7,6,8,9,10], in our view norms have to specify who is affected by them, who is the creditor, what are the actions that should or should not be performed, and what are the consequences of violating them. For instance, a norm of a university may state that a professor has to be ready to give exams any day from the middle to the end of February, otherwise the dean is authorized to lower the professors public reputation level.

From the point of view of the specification of a system, and in particular of its set of norms, it is crucial to abstract away from the actual set of agents that are interacting with the system at a given time, a result that can be achieved by using the notion of role in the definition of norms. Moreover, the time instant at which a norm becomes active is typically not known at design time, being related to the occurrence of certain events; for example, the agent playing the role of the auctioneer in an English auction is obliged to declare the current ask-price after receiving each bid by a participant. Whereas at during the system run time, norms must produce an unambiguous representation of the obligations and prohibitions that every agent has at every state of the interaction. For these reasons we propose a declarative description of norms expressed in terms of roles and times of events, which at run time can generate commitments relative to specific agents and time intervals. The main advantage of using commitments to express active obligations and permissions is that the same construct used to represent the activation of declarative norms is also used in our model of institutions to express the semantics of numerous communicative acts [1]. Interacting agents may therefore be designed to reason on just one construct to make them able to reason on all their obligations and prohibitions, derived both from norms and from the performance of communicative acts.

5.1 Declarative norms

First of all a norm is used to impose a certain behavior on certain agents in the system. Therefore a norm is applied to a set of agents, identified by means of the *debtors* attribute, on the basis of the roles they play in the system.

Another fundamental component of a norm is its *content*, which describes the actions that the debtors have to perform (if the norm expresses an obligation) or not to perform (if the norm expresses a prohibition) within a specified interval of time. In our model *temporal propositions*, which are defined by the Basic Institution (for a detailed treatment see [11]), are used to represent the content of commitments and, due to the strict connection between commitments and norms, are also used to represent the content of norms. A temporal proposition binds a *statement* about a state of affairs or about the performance of an action to a specific *interval of time* with a certain *mode* (that can be ∀ or ∃). Temporal propositions are represented with the following notation:

$$TP(statement, [t_{start}, t_{end}], mode, truth\text{-}value),$$

where the *truth-value* could be undefined (⊥), true or false. In particular when the *statement* represents the performance of an action and the *mode* is ∃, the norm is an obligation and the debtors of the norms have to perform the action within the interval of time. When the *statement* represents the non-performance of an action and the *mode* is ∀ the norm is a prohibition and the debtors of the norms should not perform the action within the interval of time. The time interval of the content is strictly connected to norms activation and deactivation events, that are described later on. In particular t_{start} is always equal to the time of occurrence of the event that activates the norm, and t_{end} is equal to the time of occurrence of the event that deactivates the norm. Regarding the verification of prohibitions, in order to be able to check that an action has not been performed during an interval of time it is necessary to rely on the closure assumption that if an action is not recorded as happened in the system, then it has not happened.

A norm becomes active when the *activation event* e_{start} happens and becomes inactive when the *deactivation event* e_{end} takes place. Activation can also depend on some Boolean *conditions*, that have to be true in order that the norm can become active; for instance an auctioneer may be obliged to open a run of an auction at time t_{start} if at least two participants are present.

An agent can reason whether fulfil or not to fulfil a norm on the basis of the sanctions (as discussed later) and of who is the *creditor* of the norm, as proposed also in [24,9]. For example, an agent with the role of auctioneer may decide to violate a norm imposed by the auction house if it is in conflict with another norm that regulates trade transactions in a certain country. The creditor of a declarative norm, given that it becomes the creditor of the commitments generated by the norm (as described in next section), is the only agent authorized to cancel such commitment [1]. In particular the operation of cancelling the commitment generated by the activation of a norm coincides with the operation of *exempting* an agent from obeying the norm in certain circumstances. Like for the *debtors* attribute, it is useful to express the creditor of declarative norms by means of their role. For instance, a norm may state that an employee is obliged to report to his director on the last day of each month; this norm will become active on the last day of each month and will be represented by means of a set of commitments, each having an actual employee as the debtor, and the employees director as the creditor.

Sometimes it may be useful to take the creditor of norms to be an *institutionalized agent*, that typically represents a human organization, like a university, a hospital, or a company, which can be regarded as the creditors of their bylaws. In the human world, an institutionalized agent is an abstract entity that can perform actions only through a human being, who is its legal representative and has the right *mandate* [25]. On the contrary, in an artificial system it is always possible to create an agent that represents an organization but can directly execute actions. Therefore we prefer to view an institutionalized agent as a special role that can be assigned to one and only one agent having the appropriate authorizations, obligations, and prohibitions.

In order to enforce norms it is necessary to specify sanctions. More precisely, as discussed in the previous section, it is necessary to specify what actions have to be performed, when a violation occurs, by the debtors of a norm and by the agent(s) in charge of norm enforcement. These two types of actions, that we respectively call *d-sanctions* (debtors sanctions) and *e-sanctions* (enforcers sanctions) are sharply dissimilar, and thus require a different treatment. More specifically, to specify a *d-sanction* means to describe an action that the violator should perform in order to extinguish its violation; therefore, a *d-sanction* can be specified through a temporal proposition representing an action. On the contrary, to specify an *e-sanction* means to describe what actions the norm enforcer is authorized to perform in the face of a violation; therefore, an *e-sanction* can be specified by representing a suitable set of authorizations.

Regarding *d-sanctions*, it is necessary to consider that a violating agent may have more than one possibility to extinguish its violation. For example, an agent may have to pay a fine of x euro within one month, and failing to do so may have to pay a fine of $2 * x$ euro within two months. In principle we may regard the second sanction as a compensation for not paying the first fine in due time, but this approach would require an unnecessarily complex procedure of violation detection. Given that any Boolean combination of temporal propositions is still a temporal proposition, and that the truth-value of the resulting temporal proposition can be obtained from the truth-values of its components using an extended truth table to manage the indefinite truth-value [26], a more viable solution consists in specifying every possible action with a different temporal proposition, and combining them using the OR operator.

In summary, in our model declarative norms are characterized by the following attributes having the specified domains:

debtors:	*role*;
creditor:	*role*;
content:	*temporal proposition*;
e_{start}:	*event-template*;
e_{end}:	*event-template*;
conditions:	*Boolean expression*;
d-sanctions:	*temporal proposition*;
e-sanctions:	*authorization*;

5.2 Commitments with Sanctions

In order to give an intuitive operational semantics to the declarative representation of norms introduced so far, we now describe an operational mechanism to transform them, at run time, into their operational counterpart, that is, into *commitments* relative to specific agent and time interval. The transformation of declarative norms in commitments is crucial in the actual evolution of the system because they are the mechanisms used to detect and react to violations. Moreover given that the activation event of norms may happen more than once in the life of the system, it is possible to distinguish between different activations and, in case, violations of the same norm. Given that our previous treatment of *commitment* [11,1] does not cover sanctions, in this section we extend it to cover this aspect.

In our model a special institution, the Basic Institution, defines the construct of commitment, which is represented with the following notation:

$$Comm(state, debtor, creditor, content).$$

The content of commitments is expressed using *temporal propositions* (briefly recalled in Section 5.1). The *state* of a commitment can change as an effect of the execution of institutional actions or of environmental events. Relevant events for the life cycle of commitments are due to the change of the truth-value of the commitments content: if the content becomes true the commitment becomes fulfilled, otherwise it becomes violated as described in Figure 1.

In our view an operational model of sanctions has to specify how to detect that a commitment has been violated, that the debtor of the violated commitment performs the compensating actions and that the agents entitled to enforce the norms have managed the violation by performing certain actions.

In our model, when the content of a commitment becomes false an event-driven routine (that as discussed in [2] can be implemented applying the *observer pattern* [27]) automatically changes the commitments state to violated. Regarding the necessity to check that the debtor performs the compensating actions, one solution may be to create a new commitment to perform those actions. A simpler and more elegant solution consists in adding two new attributes, *d-sanctions* and *e-sanctions*, to commitments, and two new states, *extinguished* and *irrecoverable*, to their life-cycle. The value of the *d-sanctions* attribute is a temporal proposition describing the actions that the debtor of the commitment has to perform, within a given interval of time, to remedy the violation. If the actions indicated in the *d-sanctions* attribute are performed, the truth-value of the related temporal proposition becomes true and an event driven routine automatically changes the state of the violated commitment to *extinguished*, as reported in Figure 1. Analogously, if the debtor does not perform those actions, at the end of the specified time interval the truth-value of the temporal proposition becomes false and the state of the commitment becomes *irrecoverable*. The actions that the agents entitled to do so are authorized to perform against the violation of the commitment are represented in the *e-sanctions* attribute. Note that whether such actions are or are not performed does not affect the life cycle of the commitment; this depends on the fact that the agent that violated a commitment cannot be held responsible for a possible failure of other agents to actually carry out the actions they are authorized to perform.

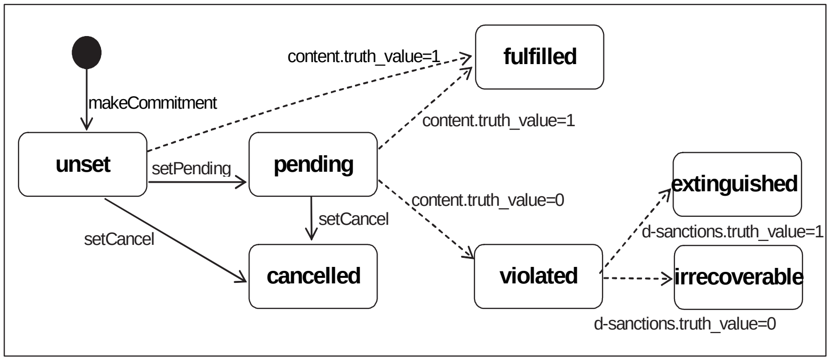


Fig. 1. The life-cycle of commitments.

Finally, for proper management of violation it may be necessary to trace the source of a commitment, either deriving it from the activation of a norm or from the performance of a communicative act. In order to represent this aspect we add to commitments an optional attribute called *source*. Our enriched notion of commitment is therefore represented with the following notation:

$Comm(state, debtor, creditor, content, d\text{-}sanctions, e\text{-}sanctions, source)$.

In our model we use *ECA-rules* (Event-Condition-Action rules) to specify that certain *actions* are executed when an event identified by an *event-templates* happens, provided that certain Boolean *conditions* are true; the *interaction-system* agent (see Section 4) is the actor of the actions performed by means of ECA-rules, and has to have the necessary authorization in order to perform them.

The following ECA-rule transforms at run time declarative norms into commitments: when the activation event (e_{start}) of the norm happens, the *makePendingComm* institutional action is performed and creates a pending commitment for each agent playing one of the roles specified in the *debtors* attribute of the norm:

```
on $e_{start}$
if $norm.conditions$ then
   do foreach $agent \mid agent.role$ in $norm.debtors$
         do $makePendingComm(agent, norm.creditor, norm.content,$
               $norm.d\text{-}sanctions, norm.e\text{-}sanctions, norm\text{-}ref)$
```

When a commitment is violated, another ECA-rule gives the authorizations expressed in the *e-sanctions* attributes to the relevant agents:

```
on $e$: $AttributeChange(comm.state, violated)$
if $true$ then
  do foreach $auth$ in $comm.e\text{-}sanctions$
     do $createAuth(auth.role, auth.iaction)$
```

The *createAuth(role,iaction)* institutional action creates the authorization for the agents playing a certain role to perform a certain institutional action. We assume that the *interaction-system* (the actor of ECA-rules) is always authorized to create new authorizations.

To guarantee that the *interaction-system* actually performs the actions specified in the *e-sanctions* attribute, it is possible to create an ECA-rule that reacts to commitments violation performing those actions:

```
on $e$: $AttributeChange(commitment.state, violated)$
if $true$ then
   do foreach $auth$ in $commitment.e\text{-}sanctions$
         if $auth.role = interaction\text{-}system$
         do $auth.iaction(parameters)$
```

6 Example

An interesting example that highlights the importance of a clear distinction between permission and authorization, which becomes relevant when more than one institution is used to specify the interaction system, is the specification of the Dutch Auction as discussed in [2].

One of the norms of the Dutch Auction obliges the auctioneer to declare a new ask-price (within λ seconds) lowering the previous one by a certain amount κ, on condition that δ seconds have elapsed from the last declaration of the ask-price without any acceptance act from the participants. If the auctioneer violates this norm the interaction-system is authorized to declare the ask-price and to lower the auctioneer's public reputation level (obviously there is no need of an authorization to change a private reputation level), while the auctioneer has to pay a fine (within h seconds) to extinguish its violation. Such a norm can be expressed in the following way:

$$\begin{array}{ll}
\textit{debtors}= & \textit{auctioneer};\\
\textit{creditor}= & \textit{auction-house};\\
\textit{content}= & TP(setAskPrice(DutchAuction.LastPrice\text{-}\kappa),\\
 & \quad [\textit{time-of}(e_{start}), \textit{time-of}(e_{end})], \exists, \bot);\\
e_{start}= & TimeEvent(DutchAuction.timeLastPrice + \delta);\\
e_{end}= & TimeEvent(\textit{time-of}(e_{start}) + \lambda);\\
\textit{conditions}= & DutchAuction.offer.value = null;\\
\textit{d-sanctions}= & TP(pay(\textit{ask-price}, \textit{interaction-system});\\
 & \quad [\textit{time-of}(e), \textit{time-of}(e) + h], \exists, \bot);\\
\textit{e-sanctions}= & Auth(\textit{interaction-system}, setAskPrice(value)),\\
\multicolumn{2}{l}{\quad Auth(\textit{interaction-system}, ChangeRep(auctioneer, value));}
\end{array}$$

where variable e refers to the event that happens if the commitment generated at run-time by this norm is violated.

At the same time, the seller of a product can fix the minimum price ($minPrice$) at which the product can be sold, for example by means of an act of proposal [26]. The auction house, by means of its auctioneer, sells the product in a run of the Dutch Auction where the auctioneer is authorized to lower the price to a predetermined *reservation price*. The reservation price fixed by the auction house can be lower than $minPrice$, for example because in previous runs of the auction the product remained unsold. If the auctioneer actually sells the product at a price ($winnerPrice$) lower that $minPrice$, the sale is valid but the auction house violates its commitment with the seller of the product and will incur the corresponding sanctions; for example, it may have to refund the seller, while the seller is authorized to lower the reputation of the auction house. This situation can be modelled by the following commitment between the seller and the auction house:

$$\begin{array}{ll}
\textit{state}= & \textit{pending};\\
\textit{debtor}= & \textit{auction-house};\\
\textit{creditor}= & \textit{seller};
\end{array}$$

content=	$TP(not\ setCurPrice(p) \mid p < minPrice,$ $[now, +\infty)], \exists, \bot)$
d-sanctions=	$TP(pay(seller, minPrice\text{-}winnerPrice),$ $[time\text{-}of(e), time\text{-}of(e){+}15days], \exists, \bot)$
e-sanctions=	$Auth(seller, ChangeReputation(auction\text{-}house, value))$

where variable e refers to the event that happens if the commitment is violated.

7 Conclusions

In this paper we have discussed the importance of formalizing and enforcing obligations and prohibitions in the specification of open interaction frameworks. We have proposed a normative component characterized by declarative norms, expressed in terms of roles and event times. The operational semantics of the declarative norms is defined by the commitments they generate through ECA-rules.

The innovative aspects of our proposal are the definition of different types of sanctions and of the operational mechanisms for monitoring the behavior of the agents and reacting to commitment violations. In particular, an interesting feature of our proposal is that the construct of commitment is uniformly used to model the semantics of communicative acts and of norms; thus artificial agents able to reason on commitments can deal with both ACL semantics and the normative component of the interaction system.

Differently from [9] our model of norms specifies the interval of time within which norms are active. Thanks to their transformation into commitments, it is possible to apply certain norms (whose activation event may happen many times) more than once in the life of the system. Another crucial aspect of our norms is that, differently from [9], they are activated by the occurrence of events and not simply if a certain state holds. Regarding the treatment of sanctions our model is more in-depth with respect to other proposals [9,10,13] because we distinguish the actions of the debtors from the actions of the other agents that are entitled to react to violations. In particular, regarding the actions of the debtors, we propose an effective solution for managing multiple sanctions, that is, multiple possibilities to compensate the violation (for example, paying an increasing amount of money), without entering in an infinite loop of checking violations and applying punishments. Regarding the sanctions applied by other agents, we discussed the reasons why a norm expresses what actions are authorized against violations and the reasons why some norms may be enforced by the interaction-system itself, which is treated as a special heteronomous agent.

Acknowledgements

We would like to thank Bertil Cottier professor of Law at Università della Svizzera italiana for helping us in improving our knowledge on legal aspects of sanctions.

References

1. Fornara, N., Viganò, F., Colombetti, M.: Agent communication and artificial institutions. Autonomous Agents and Multi-Agent Systems **14** (2007) 121–142
2. Viganò, F., Fornara, N., Colombetti, M.: An Event Driven Approach to Norms in Artificial Institutions. In Boissier, O., Padget, J., Dignum, V., Lindemann, G., Matson, E., Ossowski, S., Simao Sichman, J., Vázquez-Salceda, J., eds.: Coordination, Organization, Institutions and Norms in Multi-Agent Systems I, Proc. ANIREM'05/OOOP'05, Utrecht, The Netherlands, July 2005. Volume LNAI 3913., Springer Berlin (2006) 142–154
3. Fornara, N., Viganò, F., Verdicchio, M., Colombetti, M.: Artificial institutions: A model of institutional reality for open multiagent systems. Technical Report 4, Institute for Communication Technologies, Università della Svizzera Italiana (2006)
4. Searle, J.R.: The construction of social reality. Free Press, New York (1995)
5. Hart, H.L.A.: The Concept of Law. Clarendon Press, Oxford (1961)
6. Artikis, A., Sergot, M., Pitt, J.: Animated Specifications of Computational Societies. In C. Castelfranchi and W. L. Johnson, ed.: Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2002), ACM Press (2002) 535–542
7. Esteva, M., Padget, J., Sierra, C.: Formalizing a language for institutions and norms. In Meyer, J.J., Tambe, M., eds.: Intelligent Agents VIII : 8th International Workshop, ATAL 2001 Seattle, WA, USA, August 1-3, 2001 Revised Papers. Volume 2333 of LNCS., Springer (2002) 348–366
8. Garcia-Camino, A., Noriega, P., Rodriguez-Aguilar, J.A.: Implementing norms in electronic institutions. In: Proceedings of the 4th International Joint Conference on Autonomous agents and Multi-Agent Systems (AAMAS 2005), New York, NY, USA, ACM Press (2005) 667–673
9. López y López, F., Luck, M., d'Inverno, M.: A Normative Framework for Agent-Based Systems. In: Proceedings of the First International Symposium on Normative Multi-Agent Systems, Hatfield. (2005)
10. Vázquez-Salceda, J., Dignum, V., Dignum, F.: Organizing multiagent systems. Autonomous Agents and Multi-Agent Systems **11** (Nov 2005) 307–360
11. Fornara, N., Colombetti, M.: A commitment-based approach to agent communication. Applied Artificial Intelligence an International Journal **18** (2004) 853–866
12. Fornara, N., Viganò, F., Colombetti, M.: Agent communication and institutional reality. In van Eijk, R., Huget, M., Dignum, F., eds.: Agent Communication: International Workshop on Agent Communication, AC 2004, New York, NY, USA, July 19, 2004, Revised Selected and Invited Papers. Volume LNCS 3396., Springer (2005) 1–17
13. Grossi, D., Aldewereld, H., Dignum, F.: Ubi lex, ibi poena: Designing norm enforcement in e-institutions. In Dignum, V., Fornara, N., Noriega, P., eds.: Proceedings of the AAMAS06 Workshop on Coordination, Organization, Institutions and Norms in Agent Systems (COIN). (2006) 107–120
14. Boella, G., van der Torre, L.: Contracts as legal institutions in organizations of autonomous agents. In Dignum, V., Corkill, D., Jonker, C., Dignum, F., eds.: Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004), Los Alamitos, CA, USA, IEEE Computer Society (2004) 948–955
15. Esteva, M., Rodríguez-Aguilar, J.A., Rosell, B., Arcos, J.L.: AMELI: An Agent-based Middleware for Electronic Institutions. In Jennings, N.R., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings of the 3rd International Joint Conference on

Autonomous Agents and Multi-Agent Systems (AAMAS 2004), ACM Press (2004) 236–243

16. Zambonelli, F., Jennings, N.R., Wooldridge, M.: Developing multiagent systems: The Gaia methodology. ACM Transactions on Software Engineering and Methodology (TOSEM) **12(3)** (2003) 317–370
17. Vázquez-Salceda, J., Aldewereld, H., Dignum, F.: Implementing Norms in Multiagent Systems. In Lindemann, I.G., Denzinger, J., Timm, I.J., Unland, R., eds.: Multiagent System Technologies: Second German Conference (MATES 2004). Volume 3187 of LNAI., Berlin, Germany, Springer Verlag (2004) 313–327
18. Jones, A., Sergot, M.J.: A formal characterisation of institutionalised power. Journal of the IGPL **4** (1996) 429–445
19. Barbuceanu, M., Gray, T., Mankovski, S.: Coordinating with obligations. In Sycara, K.P., Wooldridge, M., eds.: Proceedings of the 2nd International Conference on Autonomous Agents (Agents'98), New York, ACM Press (1998) 62–69
20. Moses, Y., Tennenholtz, M.: Artificial social systems. Computers and AI **14** (1995) 533–562
21. Lomuscio, A., Sergot, M.: A formulation of violation, error recovery, and enforcement in the bit transmission problem. Journal of Applied Logic (Selected articles from DEON02 - London) **1** (2002) 93–116
22. Viganò, F.: A Framework for Model Checking Institutions. In: Proceedings of the ECAI Workshop on Model checking and Artificial Intelligence (MOCHART IV). (2006) 31–46 To appear. Available from: www.istituti.usilu.net/viganof.
23. López y López, F., Luck, M., d'Inverno, M.: Normative Agent Reasoning in Dynamic Societies. In Jennings, N.R., Sierra, C., Sonenberg, L., Tambe, M., eds.: Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004), ACM Press (2004) 535–542
24. Kagal, L., Finin, T.: Modeling Conversation Policies using Permissions and Obligations. In van Eijk, R., Huget, M., Dignum, F., eds.: Developments in Agent Communication. Volume 3396 of LNCS., Springer (2005) 123–133
25. Pacheco, O., Carmo, J.: A Role Based Model for the Normative Specification of Organized Collective Agency and Agents Interaction. Autonomous Agents and Multi-Agent Systems **6** (2003) 145–184
26. Fornara, N.: Interaction and Communication among Autonomous Agents in Multiagent Systems. PhD thesis, Faculty of Communication Sciences, University of Lugano, Switzerland (2003) http://doc.rero.ch.
27. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns. Addison Wesley (1995)

Ten Philosophical Problems in Deontic Logic

Jörg Hansen[1], Gabriella Pigozzi[2] and Leendert van der Torre[2]

[1] University of Leipzig, Institut für Philosophie
04107, Leipzig, Beethovenstraße 15, Germany
jhansen@uni-leipzig.de
[2] University of Luxembourg, Computer Science and Communications (CSC)
1359, Luxembourg, 6 rue Richard Coudenhove Kalergi, Luxembourg
{gabriella.pigozzi,leon.vandertorre}@uni.lu

Abstract. The paper discusses ten philosophical problems in deontic logic: how to formally represent norms, when a set of norms may be termed 'coherent', how to deal with normative conflicts, how contrary-to-duty obligations can be appropriately modeled, how dyadic deontic operators may be redefined to relate to sets of norms instead of preference relations between possible worlds, how various concepts of permission can be accommodated, how meaning postulates and counts-as conditionals can be taken into account, and how sets of norms may be revised and merged. The problems are discussed from the viewpoint of input/output logic as developed by van der Torre & Makinson. We argue that norms, not ideality, should take the central position in deontic semantics, and that a semantics that represents norms, as input/output logic does, provides helpful tools for analyzing, clarifying and solving the problems of deontic logic.

Keywords. Deontic logic, normative systems, input/output logic

Introduction

Deontic logic is the field of logic that is concerned with normative concepts such as obligation, permission, and prohibition. Alternatively, a deontic logic is a formal system that attempts to capture the essential logical features of these concepts. Typically, a deontic logic uses Ox to mean that it is obligatory that x, (or it ought to be the case that x), and Px to mean that it is permitted, or permissible, that x. The term 'deontic' is derived from the ancient Greek *déon*, meaning that which is binding or proper.

So-called Standard Deontic Logic (SDL) is a normal propositional modal logic of type KD, which means that it extends the propositional tautologies with the axioms $K : O(x \to y) \to (Ox \to Oy)$ and $D : \neg(Ox \wedge O\neg x)$, and it is closed under the inference rules *modus ponens* $x, x \to y/y$ and Necessitation x/Ox. Prohibition and permission are defined by $Fx = O\neg x$ and $Px = \neg O \neg x$. SDL is an unusually simple and elegant theory. An advantage of its modal-logical setting is that it can easily be extended with other modalities like epistemic or temporal operators and modal accounts of actions.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/941

Not surprisingly for such a highly simplified theory, there are many features of actual normative reasoning that SDL does not capture. Notorious are the so-called 'paradoxes of deontic logic', which are usually dismissed as consequences of the simplifications of SDL. E.g. Ross's paradox [48], the counterintuitive derivation of "you ought to mail or burn the letter" from "you ought to mail the letter", is typically viewed as a side effect of the interpretation of 'or' in natural language. Many researchers seem to believe that the subject of deontic logic may be more or less finished, and we can focus on the use of deontic logic in computer science and agent theory, since there is nothing important left to add to it. In our view, this is far from the truth. On the contrary, there is a large number of important open problems in this field of research.

In this paper we discuss ten philosophical problems in deontic logic. All of these problems have been discussed in previous literature, and solutions have been offered, but we believe that all of them should be considered open and thus meriting further research. These problems are how deontic logic relates or applies to given sets of norms (imperatives, rules, aims) (sec. 1), what it means that a set of norms should be coherent (sec. 2), how conflicts of norms can be taken into account (sec. 3), how deontic logic should react to contrary-to-duty situations in which some norms are invariably violated (sec. 4), how to interpret dyadic deontic operators that formalize 'it ought to be that x on conditions α' as $O(x/\alpha)$ (sec. 5), how explicit permissions relate to, and change, an agent's obligations (sec. 6), how meaning postulates – norms that define legal terms – and constitutive norms, that create normative states of affairs, can be modeled (sec. 7 and 8), and how normative systems may be revised (sec. 9) and merged (sec. 10). Our choice is motivated by our aim at providing ourselves with models of normative reasoning of actual agents which may be human beings or computers, but the list of open problems is by no means final. Other problems may be considered equally important, such as how a hierarchy of norms (or of the norm-giving authorities) is to be respected, or how general norms relate to individual obligations, but we hope that our discussion provides the tools, and encourages the reader, to take a fresh look at these other problems, too.

To illustrate the problems, we use Makinson & van der Torre's input/output logic as developed in [42], [43], [44], and we therefore assume familiarity with this approach (cf. [45] for a good introduction). Input/output logic takes a very general view at the process used to obtain conclusions (more generally: outputs) from given sets of premises (more generally: inputs). While the transformation may work in the usual way, as an 'inference motor' to provide logical conclusions from a given set of premises, it might also be put to other, perhaps non-logical uses. Logic then acts as a kind of secretarial assistant, helping to prepare the inputs before they go into the machine, unpacking outputs as they emerge, and, less obviously, coordinating the two. The process as a whole is one of logically assisted transformation, and is an inference only when the central transformation is so. This is the general perspective underlying input/output logic. It is one of logic at work rather than logic in isolation; not some kind of non-classical logic, but a way of using the classical one.

1 Jørgensen's dilemma

While normative concepts are the subject of deontic logic, it is quite difficult how there can be a logic of such concepts at all. Norms like individual imperatives, promises, legal statutes, moral standards etc. are usually not viewed as being true or false. E.g. consider imperative or permissive expressions such as "John, leave the room!" and "Mary, you may enter now": they do not describe, but demand or allow a behavior on the part of John and Mary. Being non-descriptive, they cannot meaningfully be termed true or false. Lacking truth values, these expressions cannot – in the usual sense – be premise or conclusion in an inference, be termed consistent or contradictory, or be compounded by truth-functional operators. Hence, though there certainly exists a logical study of normative expressions and concepts, it seems there cannot be a logic of norms: this is Jørgensen's dilemma ([30], cf. [41]).

Though norms are neither true nor false, one may state that *according to the norms*, something ought to be (be done) or is permitted: the statements "John ought to leave the room", "Mary is permitted to enter", are then true or false descriptions of the normative situation. Such statements are sometimes called normative statements, as distinguished from norms. To express principles such as the principle of conjunction: $O(x \wedge y) \leftrightarrow (Ox \wedge Oy)$, with Boolean operators having truth-functional meaning at all places, deontic logic has resorted to interpreting its formulas Ox, Fx, Px not as representing norms, but as representing such normative statements. A possible logic of normative statements may then reflect logical properties of underlying norms – thus logic may have a "wider reach than truth", as von Wright [54] famously stated.

Since the truth of normative statements depends on a normative situation, like the truth of the statement "John ought to leave the room" depends on whether some authority ordered John to leave the room or not, it seems that norms must be represented in a logical semantics that models such truth or falsity. But semantics used to model the truth or falsity of normative statements mostly fail to include norms. Standard deontic semantics evaluates deontic formulas with respect to sets of worlds, in which some are ideal or better than others – Ox is then defined true if x is true in all ideal or the best reachable worlds. In our view, norms, not ideality, should take the central position from which normative statements are evaluated. Then the following question arises, pointedly asked by D. Makinson in [41]:

Problem 1. How can deontic logic be reconstructed in accord with the philosophical position that norms are neither true nor false?

In the older literature on deontic logic there has been a veritable 'imperativist tradition' of authors that have, deviating from the standard approach, in one way or other, tried to give truth definitions for deontic operators with respect to given sets of norms.[3] The reconstruction of deontic logic as logic about imperatives

[3] Cf. among others S. Kanger [32], E. Stenius [53], T. J. Smiley [51], Z. Ziemba [62], B. van Fraassen [15], Alchourrón & Bulygin [1] and I. Niiniluoto [47].

has been the project of one of the authors beginning with [19]. Makinson & van der Torre's input/output logic [42] is another reconstruction of a logic of norms in accord with the philosophical position that norms direct rather than describe, and are neither true nor false. Suppose that we have a set G (meant to be a set of conditional norms), and a set A of formulas (meant to be a set of given facts). The problem is then: how may we reasonably define the set of propositions x making up the output of G given A, which we write $\mathit{out}(G, A)$? In particular, if we view the output as descriptions of states of affairs that ought to obtain given the norms G and the facts A, what is a reasonable output operation that enables us to define a deontic O-operator that describes the normative statements that are true given the norms and the facts, we say: the normative consequences given the situation? One such definition is the following:

$$G, A \models Ox \ \text{ iff } \ x \in \mathit{out}(G, A)$$

So Ox is true iff the output of G under A includes x. Note that this is rather a description of how we think such an output should or might be interpreted, whereas 'pure' input/output logic does not discuss such definitions. For a simple case, let G include a conditional norm that states that if a is the case, x should obtain (we write $(a, x) \in G$).[4] If a can be inferred from A, i.e. if $a \in Cn(A)$, and z is logically implied by x, then z should be among the normative consequences of G given A. An operation that does this is simple-minded output out_1:

$$\mathit{out}_1(G, A) \ = \ Cn(G(Cn(A)))$$

where $G(B) = \{y \mid (b, y) \in G \text{ and } b \in B\}$. So in the given example, Oz is true given $(a, x) \in G$, $a \in Cn(A)$ and $z \in Cn(x)$.

Simple-minded output may, however, not be strong enough. Sometimes, legal argumentation supports reasoning by cases: if there is a conditional norm (a, x) that states that an agent must bring about x if a is the case, and a norm (b, x) that states that the same agent must also bring about x if b is the case, and $a \vee b$ is implied by the facts, then we should be able to conclude that the agent must bring about x. An operation that supports such reasoning is basic output out_2:

$$\mathit{out}_2(G, A) \ = \ \cap\{Cn(G(V)) \mid v(A) = 1\}$$

where v ranges over Boolean valuations plus the function that puts $v(b) = 1$ for all formulae b, and $V = \{b \mid v(b) = 1\}$. It can easily be seen that now Ox is true given $\{(a, x), (b, x)\} \subseteq G$ and $a \vee b \in Cn(A)$.

It is quite controversial whether reasoning with conditional norms should support 'normative' or 'deontic detachment', i.e. whether it should be accepted that if one norm (a, x) commands an agent to make x true in conditions a, and another norm (x, y) directs the agent to make y true given x is true, then the agent has an obligation to make y true if a is factually true. Some would argue that as long as the agent has not in fact realized x, the norm to bring about y is not 'triggered'; others would maintain that obviously the agent has an obligation to make $x \wedge y$ true given that a is true. If such detachment is viewed

[4] As has become usual, an unconditional norm that commits the agent to realizing x is represented by a conditional norm $(\top, x)$, where $\top$ means an arbitrary tautology.

as permissible for normative reasoning, then one might use reusable output out_3 that supports such reasoning:

$$out_3(G, A) \;=\; \cap\{Cn(G(B)) \mid A \subseteq B = Cn(B) \supseteq G(B)\}$$

An operation that combines reasoning by cases with deontic detachment is then reusable basic output out_4:

$$out_4(G, A) \;=\; \cap\{Cn(G(V)) : v(A) = 1 \text{ and } G(V) \subseteq V\}$$

Finally, it is often required to reconsider the facts when drawing conclusions about what an agent must do: suppose there is an unconditional norm $(\top, x \vee y)$ to bring about $x \vee y$, but that the agent cannot realize x as the facts include $\neg x$. We would like to say that then the agent must bring about y, as this is the only possible way left to satisfy the norm. To do this, one may use the throughput versions out_n^+ of any of the output operations $out_1, out_2, out_3, out_4$,

$$out_n^+(G, A) \;=\; out_n(G^+, A),$$

where $G^+ = G \cup I$ and I is the set of all pairs (a, a) for formulae a. The choice of the throughput versions might appear questionable, since each makes Ox true in case $x \in Cn(A)$, i.e. it makes the unalterable facts obligatory.

It may turn out that further modifications of the output operation are required in order to produce reasonable results for normative reasoning. Also, the proposal to employ input/output logic to reconstruct deontic logic may lead to competing solutions, depending on what philosophical views as to what transformations should be acceptable one subscribes to. All this is what input/output logic is about. However, it should be noted that input/output logic succeeds in representing norms as entities that are neither true nor false, while still permitting normative reasoning about such entities.

2 Coherence

Consider norms which on one hand require you to leave the room, while on the other requiring you not to leave the room at the same time. In such cases, we are inclined to say that there is something wrong with the normative system. This intuition is captured by the SDL axiom $D : \neg(Ox \wedge O\neg x)$ that states that there cannot be co-existing obligations to bring about x and to bring about $\neg x$, or, using the standard cross-definitions of the deontic modalities: x cannot be both, obligatory and forbidden, or: if x is obligatory then it is also permitted. But what does this tell us about the normative system?

Since norms do not bear truth values, we cannot, in any usual sense, say that such a set of norms is inconsistent. All we can consider is the consistency of the output of a set of norms. We like to use the term *coherence* with respect to a set of norms with consistent output, and define:

(1) A set of norms G is coherent iff $\perp \notin out(G, A)$.

However, this definition seems not quite sufficient: one might argue that one should be able to determine whether a set of norms G is coherent or not regardless of what arbitrary facts A might be assumed. A better definition would be $(1a)$:

($1a$) A set of norms G is coherent iff there exists a set of formulas A such that $\bot \notin out(G, A)$.

For ($1a$) it suffices that there exists a situation in which the norms can be, or could have been, fulfilled. However, consider the set of norms $G = \{(a, x), (a, \neg x)\}$ that requires both x to be realized and $\neg x$ to be realized in conditions a: it is immediate that e.g. for all output operations $out_n^{(+)}$, we have $\bot \notin out_n^{(+)}(G, \neg a)$: no conflicting demands arise when $\neg a$ is factually assumed. Yet something seems wrong with a normative system that explicitly considers a fact a only to tie to it conflicting normative consequences. The dual of ($1a$) would be

($1b$) A set of norms G is coherent iff for all sets of formulas A, $\bot \notin out(G, A)$.

Now a set G with $G = \{(a, x), (a, \neg x)\}$ would no longer be termed coherent. ($1b$) makes the claim that for no situation A, two norms $(a, x), (b, y)$ would ever come into conflict, which might seem too strong. We may wish to restrict A to sets of facts that are consistent, or that are not in violation of the norms. The question is, basically, how to distinguish situations that the norm-givers should have taken care of, from those that describe misfortune of otherwise unhappy circumstances. A weaker claim than ($1b$) would be ($1c$):

($1c$) A set of norms G is coherent iff for all a with $(a, x) \in G$, $\bot \notin out(G, a)$.

By this change, consistency of output is required just for those factual situations that the norm-givers have foreseen, in the sense that they have explicitly tied normative consequences to such facts. Still, ($1c$) might require further modification, since if a is a foreseen situation, and so is b, then also $a \vee b$ or $a \wedge b$ might be counted as foreseen situations for which the norms should be coherent.

However, there is a further difficulty: let G contain a norm $(a, \neg a)$ that, for conditions in which a is unalterably true, demands that $\neg a$ be realized. We then have $\neg a \in out_n(G, a)$ for the principal output operations out_n, but not $\bot \in out_n(G, a)$. Certainly the term 'incoherent' should apply to a normative system that requires the agent to accomplish what is – given the facts in which the duty arises – impossible. But since not every output operation supports 'throughput', i.e. the input is not necessarily included in the output, neither (1) nor its variants implies that the agent can actually realize all propositions in the output, though they might be logically consistent. We might therefore demand that the output is not consistent *simpliciter*, but consistent with the input:

(2) A set of norms G is coherent iff $\bot \notin out(G, A) \cup A$.

But with definition (2) we obtain the questionable result that for any case of norm-violation, i.e. for any case in which $(a, x) \in G$ and $(a \wedge \neg x) \in Cn(A)$, G must be termed incoherent – Adam's fall would only indicate that there was something wrong with God's commands. One remedy would be to leave aside all those norms that are invariably violated, i.e. instead of $out(G, A)$ consider $out(\{(a, x) \in G \mid (a \wedge \neg x) \notin Cn(A)\}, A)$ – but then a set G such that $(a, \neg a) \in G$ would not be incoherent. It seems it is time to formally state our problem:

Problem 2. When is a set of norms to be termed 'coherent'?

As can be seen from the discussion above, input/output logic provides the tools to formally discuss this question, by rephrasing the question of coherence of the norms as one of consistency of output, and of output with input. Both notions have been explored in the input/output framework as 'output under constraints':

Definition (Output under constraints) *Let G be a set of conditional norms and A and C two sets of propositional formulas. Then G is coherent in A under constraints C when $out(G, A) \cup C$ is consistent.*

Future study must define an output operation, determine the relevant states A, and find the constraints C, such that any set of norms G would be appropriately termed coherent or incoherent by this definition.

3 Normative conflicts and dilemmas

There are essentially two views on the question of normative conflicts: in the one view, they do not exist. In the other view, conflicts and dilemmas are ubiquitous.

According to the view that normative conflicts are ubiquitous, it is obvious that we may become the addressees of conflicting normative demands at any time. My mother may want me to stay inside while my brother wants me to go outside with him and play games. I may have promised to finish a paper until the end of a certain day, while for the same day I have promised a friend to come to dinner – now it is late afternoon and I realize I will not be able to finish the paper if I visit my friend. Social convention may require me to offer you a cigarette when I am lighting one for myself, while concerns for your health should make me not offer you one. Legal obligations might collide - think of the recent case where the SWIFT international money transfer program was required by US anti-terror laws to disclose certain information about its customers, while under European law that also applied to that company, it was required not to disclose this information. Formally, let there be two conditional norms (a, x) and (b, y): unless we have that either $(x \rightarrow y) \in Cn(a \wedge b)$ or $(y \rightarrow x) \in Cn(a \wedge b)$ there is a possible situation $a \wedge b \wedge \neg(x \wedge y)$ in which the agent can still satisfy each norm individually, but not both norms collectively. But to assume the former for any two norms (a, x) and (b, y) is clearly absurd.[5] So any logic about norms must take into account possible conflicts. But standard deontic logic SDL includes D: $\neg(Ox \wedge O\neg x)$ as one of its axioms, and it is not quite immediate how deontic reasoning could accommodate conflicting norms. The problem is thus:

Problem 3a. How can deontic logic accommodate possible conflicts of norms?

In an input/output setting one could say that there exists a conflict whenever $\bot \in Cn(out(G, A) \cup A)$, i.e. whenever the output is inconsistent with the input: then the norms cannot all be satisfied in the given situation. There appear to be two ways to proceed when such inconsistencies cannot be ruled out.[6]. For both, it is necessary to recur to the the notion of a $maxfamily(G, A, A)$, i.e. the

[5] Nevertheless, Lewis' [36], [37] and Hansson's [24] deontic semantics imply that there exists a 'system of spheres', in our setting: a sequence of boxed contrary-to-duty norms $(\top, x_1), (\neg x_1, x_2), (\neg x_1 \wedge \neg x_2, x_3), \ldots$ that satisfies this condition.

[6] For the concepts underlying the 'some-things-considered' and 'all-things-considered' O-operators defined below cf. Horty [28] and Hansen [20], [21]

family of all maximal $H \subseteq G$ such that $out(H, A) \cup A$ is consistent. On this basis, input/output logic defines the following two output operations $out^{\cup}$ and $out^{\cap}$:

$$\begin{aligned} out^{\cup}(G, A) &= \bigcup\{out(H, A) \mid H \in maxfamily(G, A, A)\} \\ out^{\cap}(G, A) &= \bigcap\{out(H, A) \mid H \in maxfamily(G, A, A)\} \end{aligned}$$

Note that $out^{\cup}$ is a non-standard output operation that is not closed under consequences, i.e. we do not generally have $Cn(out^{\cup}(G, A)) = out^{\cup}(G, A)$. Finally we may use the intended definition of an O-operator

$$G, A \models Ox \text{ iff } x \in out(G, A)$$

to refer to the operations $out^{\cup}$ and $out^{\cap}$, rather than the underlying operation $out(G, A)$ itself, and write $O^{\cup}x$ and $O^{\cap}x$ to mean that $x \in out^{\cup}(G, A)$ and $x \in out^{\cap}(G, A)$, respectively. Then the 'some-things-considered', or 'bold' O-operator $O^{\cup}$ describes x as obligatory given the set of norms G and the facts A if x is in the output of some $H \in maxfamily(G, A, A)$, i.e. if some subset of non-conflicting norms, or: some coherent normative standard embedded in the norms, requires x to be true. It is immediate that neither the SDL axiom $D : \neg(Ox \wedge O\neg x)$ nor the agglomeration principle $C : Ox \wedge Oy \rightarrow O(x \wedge y)$ holds for $O^{\cup}$, as there may be two competing standards demanding x and $\neg x$ to be realized, while there may be none that demands the impossible $x \wedge \neg x$. On the other hand, the 'all-things-considered', or 'sceptic', O-operator $O^{\cap}$ describes x as obligatory given the norms G and the facts A if x is in the outputs of all $H \in maxfamily(G, A, A)$, i.e. it requires that x must be realized according to all coherent normative standards. Note that by this definition, both SDL theorems D and C are validated.

The opposite view, that normative conflicts do not exist, appeals to the very notion of obligation: it is essential for the function of norms to direct human behavior that the subject of the norms is capable of following them. To state a norm that cannot be fulfilled is a meaningless use of language. To state two norms which cannot both be fulfilled is confusing the subject, not giving him or her directions. To say that a subject has two conflicting obligations is therefore a misuse of the term 'obligation'. So there cannot be conflicting obligations, and if things appear differently, a careful inspection of the normative situation is required that resolves the dilemma in favor of the one or other of what only appeared both to be obligations. In particular, this inspection may reveal a priority ordering of the apparent obligations that helps resolve the conflict (this summarizes viewpoints prominent e.g. in Ross [49], von Wright [59], [60], and Hare [25]). The problem that arises for such a view is then how to determine the 'actual obligations' in face of apparent conflicts, or, put differently, in the face of conflicting 'prima facie' obligations.

Problem 3b. How can the resolution of apparent conflicts be semantically modeled?

Again, both the $O^{\cup}$ and the $O^{\cap}$-operator may help to formulate and solve the problem: $O^{\cup}$ names the conflicting *prima facie* obligations that arise from a set of norms G in a given situation A, whereas $O^{\cap}$ resolves the conflict by telling the agent to do only what is required by all maximal coherent subsets of the norms: so there might be conflicting 'prima facie' $O^{\cup}$-obligations, but no conflicting 'all things considered' $O^{\cap}$-obligations. The view that a priority ordering helps to resolve conflicts seems more difficult to model. A good approach appears to be to let the priorities help us to select a set $\mathscr{P}(G, A, A)$ of preferred maximal subsets $H \in \mathit{maxfamily}(G, A, A)$. We may then define the $O^{\cap}$-operator not with respect to the whole of $\mathit{maxfamily}(G, A, A)$, but only with respect to its selected preferred subsets $\mathscr{P}(G, A, A)$. Ideally, in order to resolve all conflicts, the priority ordering should narrow down the selected sets to $\mathit{card}(\mathscr{P}(G, A, A)) = 1$, but this generally requires a strict ordering of the norms in G. The demand that all norms can be strictly ordered is itself subject of philosophical dispute: some moral requirements may be incomparable (this is Sartre's paradox, where the requirement that Sartre's student stays with his ailing mother conflicts with the requirement that the student joins the resistance against the German occupation), while others may be of equal weight (e.g. two simultaneously obtained obligations towards identical twins, of which only one can be fulfilled). The difficult part is then to define a mechanism that determines the preferred maximal subsets by use of the given priorities between the norms. There have been several proposals to this effect, not all of them successful, and the reader is referred to the discussions in Boella & van der Torre [8] and Hansen [22], [23].

4 Contrary-to-duty reasoning

Suppose we are given a code G of conditional norms, that we are presented with a condition (input) that is unalterably true, and asked what obligations (output) it gives rise to. It may happen that the condition is something that should not have been true in the first place. But that is now water under the bridge: we have to "make the best out of the sad circumstances" as B. Hansson [24] put it. We therefore abstract from the deontic status of the condition, and focus on the obligations that are consistent with its presence. How to determine this in general terms, and if possible in formal ones, is the well-known problem of contrary-to-duty conditions as exemplified by the notorious contrary-to-duty paradoxes. Chisholm's paradox [13] consists of the following four sentences:

(1) It ought to be that a certain man go to the assistance of his neighbors.
(2) It ought to be that if he does go, he tell them he is coming.
(3) If he does not go then he ought not to tell them he is coming.
(4) He does not go.

Furthermore, intuitively, the sentences derive (5):

(5) He ought not to tell them he is coming.

Chisholm's paradox is a contrary-to-duty paradox, since it contains both a primary obligation to go, and a secondary obligation not to call if the agent does not go. Traditionally, the paradox was approached by trying to formalize each of the sentences in an appropriate language of deontic logic, and then consider the sets $\{Ox, O(x \rightarrow z), O(\neg x \rightarrow \neg z), \neg x\}$, or $\{Ox, x \rightarrow Oz, \neg x \rightarrow O\neg z, \neg x\}$, or $\{Ox, O(x \rightarrow z), \neg x \rightarrow O\neg z, \neg x\}$ or $\{Ox, x \rightarrow Oz, O(\neg x \rightarrow \neg z), \neg x\}$. But whatever approach is taken, it turned out that either the set of formulas is traditionally inconsistent or inconsistent in SDL, or one formula is a logical consequence – by traditional logic or in SDL – of another formula. Yet intuitively the natural-language expressions that make up the paradox are consistent and independent from each other: this is why it is called a paradox. Though the development of dyadic deontic operators as well as the introduction of temporally relative deontic logic operators can be seen as a direct result of Chisholm's paradox, the paradox seems so far unsolved. The problem is thus:

Problem 4. How do we reason with contrary-to-duty obligations which are in force only in case of norm violations?

In the input/output logic framework, the strategy for eliminating excess output is to cut back the set of generators to just below the threshold of yielding excess. To do that, input/output logic looks at the maximal non-excessive subsets, as described by the following definition:

Definition (Maxfamilies) *Let G be a set of conditional norms and A and C two sets of propositional formulas. Then maxfamily(G, A, C) is the set of maximal subsets $H \subseteq G$ such that $out(H, A) \cup C$ is consistent.*

For a possible solution to Chisholm's paradox, consider the following output operation $out^{\cap}$:

$$out^{\cap}(G, A) \ = \ \bigcap\{out(H, A) \mid H \in maxfamily(G, A, A)\}$$

So an output x is in $out^{\cap}(G, A)$ if it is in output $out(H, A)$ of all maximal norm subsets $H \subseteq G$ such that $out(H, A)$ is consistent with the input A. Let a deontic O-operator be defined in the usual way with regard to this output:

$$G, A \models O^{\cap}x \ \text{ iff } \ x \in out^{\cap}(G, A)$$

Furthermore, tentatively, and only for the task of shedding light on Chisholm's paradox, let us define an entailment relation between norms as follows:

Definition (Entailment relation) *Let G be a set of conditional norms, and (a, x) be a norm whose addition to G is under consideration. Then (a, x) is entailed by G iff for all sets of propositions A, $out^{\cap}(G \cup \{(a, x)\}, A) = out^{\cap}(G, A)$.*

So a (considered) norm is entailed by a (given) set of norms if its addition to this set would not make a difference for any set of facts A. Finally, let us use the following cautious definition of 'coherence from the start' (also called 'minimal coherence' or 'coherence per se'):

A set of norms G is 'coherent from the start' iff $\bot \notin out(G, \top)$.

Now consider a 'Chisholm norm set' $G = \{(\top, x), (x, z), (\neg x, \neg z), \}$, where $(\top, x)$ means the norm that the man must go to the assistance of his neighbors, (x, z) means the norm that it ought to be that if he goes he ought to tell them he is coming, and $(\neg x, \neg z)$ means the norm that if he does not go he ought not to tell them he is coming. It can be easily verified that the norm set G is 'coherent from the start' for all standard output operations $out_n^{(+)}$, since for these either $out(G, \top) = Cn(\{x\})$ or $out(G, \top) = Cn(\{x, z\})$, and both sets $\{x\}$ and $\{x, z\}$ are consistent. Furthermore, it should be noted that all norms in the norm set G

are independent from each other, in the sense that no norm $(a, x) \in G$ is entailed by $G \setminus \{(a, x)\}$ for any standard output operation $out_n^{(+)}$: for $(\top, x)$ we have $x \in out^{\cap}(G, \top)$ but $x \notin out^{\cap}(G \setminus \{(\top, x)\}, \top)$, for (x, z) we have $z \in out^{\cap}(G, x)$ but $z \notin out^{\cap}(G \setminus \{(x, z)\}, x)$, and for $(\neg x, \neg z)$ we have $\neg z \in out^{\cap}(G, \neg x)$ but $\neg z \notin out^{\cap}(G \setminus \{(\neg x, \neg z)\}, \top)$. Finally consider the 'Chisholm fact set' $A = \{\neg x\}$, that includes as an assumed unalterable fact the proposition $\neg x$, that the man will not go to the assistance of his neighbors: we have $maxfamily(G, A, A) = \{G \setminus \{(\top, x)\}\} = \{\{(x, z), (\neg x, \neg z), \}\}$ and either $out(G \setminus \{(\top, x)\}, A) = Cn(\{\neg z\})$ or $out(G \setminus \{(\top, x)\}, A) = Cn(\{\neg x, \neg z\})$ for all standard output operations $out_n^{(+)}$, and so $O^{\cap}\neg z$ is true given the norm and fact sets G and A, i.e. the man must not tell his neighbors he is coming.

5 Descriptive dyadic obligations

Dyadic deontic operators, that formalize e.g. 'x ought to be true under conditions a' as $O(x/a)$, were introduced over 50 years ago by G. H. von Wright [56]. Their introduction was due to Prior's paradox of derived obligation: often a primary obligation Ox is accompanied by a secondary, 'contrary-to-duty' obligation that pronounces y (a sanction, a remedy) as obligatory if the primary obligation is violated. At the time, the usual formalization of the secondary obligation would have been $O(\neg x \rightarrow y)$, but given Ox and the axioms of standard deontic logic SDL, $O(\neg x \rightarrow y)$ is derivable for any y. A bit later, Chisholm's paradox showed that formalizing the secondary obligation as $\neg x \rightarrow Oy$ produces similarly counterintuitive results. So to deal with such contrary-to-duty conditions, the dyadic deontic operator $O(x/a)$ was invented.

The perhaps best-known semantic characterization of dyadic deontic logic is Bengt Hansson's [24] system *DSLD*3, axiomatized by Spohn [52]. Hansson's idea was that the circumstances (the conditions a) are something which has actually happened (or will unalterably happen) and which cannot be changed afterwards. Ideal worlds in which $\neg a$ is true are therefore excluded. But some worlds may still be better than others, and there should then be an obligation to make "'the best out of the sad circumstances". Consequently, Hansson presents a possible worlds semantics in which all worlds are ordered by a preference (betterness) relation. $O(x/a)$ is then defined true if x is true in the best a-worlds. Here, we intend to employ semantics that do not make use of any prohairetic betterness relation, but that models deontic operators with regard to given sets of norms and facts, and the question is then

Problem 5. How to define dyadic deontic operators with regard to given sets of norms and facts?

Input/output logic assumes a set of (conditional) norms G, and a set of invariable facts A. The facts A may describe a situation that is inconsistent with the output $out(G, A)$: suppose there is a primary norm $(\top, a) \in G$ and a secondary norm $(\neg a, x) \in G$, i.e. $G = \{(\top, a), (\neg a, x)\}$, and $A = \{\neg a\}$. Though

$a \in out(G, A)$, it makes no sense to describe a as obligatory since a cannot be realized any more in the given situation – no crying over spilt milk. Rather, the output should include only the consequent of the secondary obligation x – it is the best we can make out of these circumstances. To do so, we return to the definitions of $maxfamily(G, A, A)$ as the set of all maximal subsets $H \subseteq G$ such that $out(H, A) \cup A$ is consistent, and the set $out^{\cap}(G, A)$ as the intersection of all outputs from $H \in maxfamily(G, A, A)$, i.e. $out^{\cap}(G, A) = \bigcap\{out(H, A) \mid H \in maxfamily(G, A, A)\}$. We may then define:

$$G \models O(x/a) \;\text{ iff }\; x \in out^{\cap}(G, \{a\})$$

Thus, relative to the set of norms G, $O(x/a)$ is defined true if x is in the output under a of all maximal sets H of norms such that their output under $\{a\}$ is consistent with a. In the example where $G = \{(\top, a), (\neg a, x)\}$ we therefore obtain $O(x/\neg a)$ but not $O(a/\neg a)$ as being true, i.e. only the consequent of the secondary obligation is described as obligatory in conditions $\neg a$.

In the above definition, the antecedent a of the dyadic formula $O(x/a)$ makes the inputs explicit: the truth definition does not make use of any facts other than a. This may be unwanted; one might consider an input set A of *given* facts, and employ the antecedent a only to denote an additional, *assumed* fact. Still, the output should contradict neither the given nor the assumed facts, and the output should include also the normative consequences x of a norm (a, x) given the assumed fact a. This may be realized by the following definition:

$$G, A \models O(x/a) \;\text{ iff }\; x \in out^{\cap}(G, A \cup \{a\})$$

So, relative to a set of norms G and a set of facts A, $O(x, a)$ is defined true if x is in the output under $A \cup \{a\}$ of all maximal sets H of norms such that their output under $A \cup \{a\}$ is consistent with $A \cup \{a\}$.

Hansson's description of dyadic deontic operators as describing defeasible obligations that are subject to change when more specific, namely contrary-to-duty situations emerge, may be the most prominent view, but it is by no means the only one. Earlier authors like von Wright [57] [58] and Anderson [4] have proposed more normal conditionals, which in particular support 'strengthening of the antecedent' SA $O(x/a) \rightarrow O(x/a \wedge b)$. From an input/output perspective, such operators can be accommodated by defining

$$G, A \models O(x/a) \;\text{ iff }\; x \in out(G, A \cup \{a\})$$

It is immediate that for all standard output operations $out_n^{(+)}$ this definition validates SA. The properties of dyadic deontic operators that are, like the above, semantically defined within the framework of input/output logic, have not been studied so far. The theorems they validate will inevitably depend on what output operation is chosen (cf. [23] for some related conjectures).

6 Permissive norms

In formal deontic logic, permission is studied less frequently than obligation. For a long time, it was naively assumed that it can simply be taken as a dual of obligation, just as possibility is the dual of necessity in modal logic. Permission is then defined as the absence of an obligation to the contrary, and the modal operator P defined by $Px =_{def} \neg O \neg x$. Today's focus on obligations is not only in stark contrast how deontic logic began, for when von Wright [55] started modern deontic logic in 1951, it was the P-operator that he took as primitive, and defined obligation as an absence of a permission to the contrary. Rather, more and more authors have come to realize how subtle and multi-faceted the concept of permission is. Much energy was devoted to solving the problem of 'free choice permission', where one may derive from the statement that one is permitted to have a cup of tea or a cup of coffee that it is permitted to have a cup of tea, and it is permitted to have a cup of coffee, or for short, that $P(x \vee y)$ implies Px and Py (cf. [31]. Von Wright, in his late work starting with [61], dropped the concept of inter-definability of obligations and permissions altogether by introducing P-norms and O-norms, where one may call something permitted only if it derives from the collective contents of some O-norms and at most one P-norm. This concept of 'strong permission' introduced deontic 'gaps': whereas in standard deontic logic SDL, $O \neg x \vee Px$ is a tautology, meaning that any state of affairs is either forbidden or permitted, von Wright's new theory means that in the absence of explicit P-norms only what is obligatory is permitted, and that nothing is permitted if also O-norms are missing. Perhaps most importantly, Bulygin [12] observed that an authoritative kind of permission must be used in the context of multiple authorities and updating normative systems: if a higher authority permits you to do something, a lower authority can no longer prohibit it. Summing up, the understanding of permission is still in a less satisfactory state than the understanding of obligation and prohibition. The problem can be phrased thus:

Problem 6. How to distinguish various kinds of permissions and relate them to obligations?

¿From the viewpoint of input/output logic, one may first try to define a concept

of negative permission in the line of the classic approach. Such a definition is the following:

$$G, A \models P^{neg}x \text{ iff } \neg x \notin out(G, A)$$

So something is permitted by a code iff its negation is not obligatory according to the code and in the given situation. As innocuous and standard as such a definition seems, questions arise as to what output operation out may be used. Simple-minded output out_1 and basic output out_2 produce counterintuitive results: consider a set of norms G of which one norm $(work, tax)$ demands that if I am employed then I have to pay tax. For the default situation $A = \{\top\}$ then $P^{neg}(a \wedge \neg x)$ is true, i.e. it is by default permitted that I am employed and do not pay tax. Stronger output operations out_3 and out_4 that warrant reusable output exclude this result, but their use in deontic reasoning is questionable for other reasons.

In contrast to a concept of negative permission, one may also define a concept of 'strong' or 'positive permission'. This requires a set P of explicit permissive norms, just as G is a set of explicit obligations. As a first approximation, one may say that something is positively permitted by a code iff the code explicitly presents it as such. But this leaves a central logical question unanswered as to how explicitly given permissive and obligating norms may generate permissions that – in some sense – follow from the explicitly given norms. In the line of von Wrights later approach, we may define:

$$G, P \models P^{stat}(x/a) \ \text{ iff } \ x \in out(G \cup \{(b, y)\}, a) \text{ for some } (b, y) \in P \cup \{(\top, \top)\}$$

So there is a permission to realize x in conditions a if x is generated under these conditions either by the norms in G alone, or the norms in G together with some explicit permission (b, y) in P. We call this a 'static' version of strong permission. For example, consider a set G consisting of the norm $(work, tax)$, and a set P consisting of the sole license $(18y, vote)$ that permits all adults to take part in political elections. Then all of the following are true: $P^{stat}(tax/work)$, $P^{stat}(vote/18y)$, $P^{stat}(tax/work \wedge male)$ and also $P^{stat}(vote/\neg work \wedge 18y)$ (so even unemployed adults are permitted to vote).

Where negative permission is liberal, in the sense that anything is permitted that does not conflict with ones obligations, the concept of static permission is quite strict, as nothing is permitted that does not explicitly occur in the norms. In between, one may define a concept of 'dynamic permission' that defines something as permitted in some situation a if forbidding it for these conditions would prevent an agent from making use of some explicit (static) permission. The formal definition reads:

$$G, P \models P^{dyn}(x/a) \ \text{ iff } \ \neg y \in out(G \cup \{(a, \neg x)\}, b) \text{ for some } y \text{ and conditions } b \text{ such that } G, P \models P^{stat}(y/b)$$

Consider the above static permission $P^{stat}(vote/\neg work \wedge 18y)$ that even the unemployed adult populations is permitted to vote, generated by the sets $P = \{(18y, vote)\}$ and $G = \{(work, tax)\}$. We might also like to say, without reference to age, that the unemployed are protected from being forbidden to vote, and in this sense are permitted to vote, but $P^{stat}(vote/\neg work)$ is not true. And we might like to say that adults are protected from being forbidden to vote unless they are employed, and in this sense are permitted to be both unemployed and take part in elections, but also $P^{stat}(\neg work \wedge vote/18y)$ is not true. Dynamic permissions allow us to express such protections, and make both $P^{dyn}(vote/\neg work)$ and $P^{dyn}(\neg work \wedge vote/18y)$ true: if either $(\neg work, \neg vote)$ or $(18y, (\neg work \rightarrow \neg vote))$ were added to G we would obtain $\neg vote$ as output in conditions $\neg work \wedge 18y)$ in spite of the fact that, as we have seen, $G, P \models P^{stat}(vote/\neg work \wedge 18y)$.

There are, ultimately, a number of questions for all these concepts of permissions that have been further explored in [44]. Other kinds of permissions have been discussed from an input/output perspective in the literature, too, for example permissions as exceptions of obligations [8]. But it seems input/output logic is able to help clarify the underlying concepts of permission better than traditional deontic semantics.

7 Meaning postulates and intermediate concepts

To define a deontic operator of individual obligation seems straightforward if the norm in question is an individual command or act of promising. For example, if you are the addressee α of the following imperative sentence

(1) You, hand me that screwdriver, please.

and you consider the command valid, then what you ought to do is to hand the screwdriver in question to the person β uttering the request. In terms of input/output logic, let x be the proposition that α hands the screwdriver to β: with the set of norms $G = \{(\top, x)\}$, the set of facts $A = \{\top\}$, and the truth definition Ox iff $x \in out(A, G)$: then we obtain that Ox is true, i.e. it is true that it ought to be that α hands the screwdriver to β.

Norms that belong to a legal system are more complex, and thus more difficult to reason about. Consider, for example

(2) An act of theft is punished by a prison sentence not exceeding 5 years or a fine.

Things are again easy if you are a judge and you know that the accused in front of you has committed an act of theft – then you ought to hand out a verdict that commits the accused to pay a fine or to serve a prison sentence not exceeding 5 years. But how does the judge arrive at the conclusion that an act of theft has been committed? ‘Theft’ is a legal term that is usually accompanied by a legal definition such as the following one:

(3) Someone commits an act of theft if that person has taken a movable object from the possession of another person into his own possession with the intention to own it, and if the act occurred without the consent of the other person or some other legal authorization.

It is noteworthy that (3) is not a norm in the strict sense – it does not prescribe or allow a behavior – but rather a stipulative definition, or, in more general terms, a *meaning postulate* that constitutes the legal meaning of theft. Such sentences are often part of the legal code. They share with norms the property of being neither true nor false. The significance of (3) is that it decomposes the complex legal term ‘theft’ into more basic legal concepts. These concepts are again the subject of further meaning postulates, among which may be the following:

(4) A person in the sense of the law is a human being that has been born.
(5) A movable object is any physical object that is not a person or a piece of land.
(6) A movable object is in the possession of a person if that person is able to control the uses and the location of the object.
(7) The owner of an object is – within the limits of the law – entitled to do with it whatever he wants, namely keep it, use it, transfer possession or ownership of the object to another person, and destroy or abandon it.

Not all of definitions (4)-(7) may be found in the legal statutes, though they may be viewed as belonging to the normative system by virtue of having been accepted in legal theory and judicial reasoning. They constitute ‘intermediate concepts': they link legal terms (person, movable object, possession etc.) to words describing natural facts (human being, born, piece of land, keep an object etc.).

Any proper representation of legal norms must include means of representing meaning postulates that define legal terms, decompose legal terms into more basic legal terms, or serve as intermediate concepts that link legal terms to terms that describe natural facts. But for deontic logic, with its standard possible worlds semantics, a comprehensive solution to the problem of representing meaning postulates is so far lacking (cf. Lindahl [39]). The problem is thus:

Problem 7. How can meaning postulates and intermediate terms be modeled in semantics for deontic logic reasoning?

The representation of intermediate concepts is of particular interest, since such concepts arguably reduce the number of implications required for the transition from natural facts to legal consequences and thus serve an economy of expression (cf. Lindahl & Odelstad [40]). Lindahl & Odelstad use the term 'ownership' as an example to argue as follows: let $F_1, ..., F_p$ be descriptions of some situations in which a person α acquires ownership of an object γ, e.g. by acquiring it from some other person β, finding it, building it from owned materials, etc., and let $C_1, ..., C_n$ be among the legal consequences of α's ownership of γ, e.g. freedom to use the object, rights to compensation when the object is damaged, obligations to maintain the object or pay taxes for it etc. To express that each fact F_i has the consequence C_j, $p \times n$ implications are required. The introduction of the term $\mathit{Ownership}(x, y)$ reduces the number of required implications to $p + n$: there are p implications that link the facts $F_1, ..., F_p$ to the legal term $\mathit{Ownership}(x, y)$, and n implications that link the legal term $\mathit{Ownership}(x, y)$ to each of the legal consequences $C_1, ..., C_n$. The argument obviously does not apply to all cases: one implication $(F_1 \vee ... \vee F_p) \rightarrow (C_1 \wedge ... \wedge C_n)$ may often be sufficient to represent the case that a variety of facts $F_1, ..., F_p$ has the same multitude of legal consequences $C_1, ..., C_n$. However, things may be different when norms that link a number of factual descriptions to the same legal consequences stem from different normative sources, may come into conflict with other norms, can be overridden by norms of higher priority, or be subject of individual exemption by norms that grant freedoms or licenses: in these cases, the norms must be represented individually. So it seems worthwhile to consider ways to incorporate intermediate concepts into a formal semantics for deontic logic.

In an input/output framework, a first step could be to employ a separate set T of theoretical terms, namely meaning postulates, alongside the set G of norms. Let T consists of intermediates of the form (a, x), where a is a factual sentence (e.g. that β is in possession of γ, and that α and β agreed that α should have γ, and that β hands γ to α), and x states that some legal term obtains (e.g. that α is now owner of γ). To derive outputs from the set of norms G, one may then use $A \cup \mathit{out}(T, A)$ as input, i.e. the factual descriptions together with the legal statements that obtain given the intermediates T and the facts A.

It may be of particular interest to see that such a set of intermediates may help resolve possible conflicts in the law. Let $(\top, \neg \mathit{dog})$ be a statute that forbids dogs on the premises, but let there also be a higher order principle that no blind person may be required to give up his or her guide dog. Of course the conflict may be solved by modifying the statute (e.g. add a condition that the dog in question is not a guide dog), but then modifying a statute is usually not something a judge, faced with such a norm, is allowed to do: the judge's duty is solely to consider the statute, interpret it according to the known or supposed will of the norm-giver, and apply it to the given facts. The judge may then come to the conclusion that a fair and considerate norm-giver would not have meant the statute to apply to guide dogs, i.e. the term "dog" in the statute is a theoretical term whose extension is smaller than the natural term. So the statute must be re-interpreted as reading $(\top, \neg tdog)$ with the additional intermediate $(dog \wedge \neg guidedog, tdog) \in T$, and thus no conflict arises for the case of blind persons that want to keep their guide dog. While this seems to be a rather natural view of how judicial conflict resolution works (the example is taken from an actual court case), the exact process of creating and modifying theoretical terms in order to resolve conflicts must be left to further study.

8 Constitutive norms

Constitutive norms like counts-as conditionals are rules that create the possibility of or define an activity. For example, according to Searle [50], the activity of playing chess is constituted by action in accordance with these rules. Chess has no existence apart from these rules. The institutions of marriage, money, and promising are like the institutions of baseball and chess in that they are systems of such constitutive rules or conventions. They have been identified as the key mechanism to normative reasoning in dynamic and uncertain environments, for example to realize agent communication, electronic contracting, dynamics of organizations, see, e.g., [9].

Problem 8. How to define counts-as conditionals and relate them to obligations and permissions?

For Jones and Sergot [29], the counts-as relation expresses the fact that a state of affairs or an action of an agent "is a sufficient condition to guarantee that the institution creates some (usually normative) state of affairs". They formalize this introducing a conditional connective $\Rightarrow_s$ to express the "counts-as" connection holding in the context of an institution s. They characterize the logic of $\Rightarrow_s$ as a conditional logic, with axioms for agglomeration $((x \Rightarrow_s y) \,\&\, (x \Rightarrow_s z)) \supset (x \Rightarrow_s (y \wedge z))$, left disjunction $((x \Rightarrow_s z) \,\&\, (y \Rightarrow_s z)) \supset ((x \vee y) \Rightarrow_s z)$ and transitivity $((x \Rightarrow_s y) \,\&\, (y \Rightarrow_s z)) \supset (x \Rightarrow_s z)$. The flat fragment can be phrased as an input/output logic as follows [7].

Definition 1. *Let L be a propositional action logic with $\vdash$ the related notion of derivability and Cn the related consequence operation $Cn(x) = \{y \mid x \vdash y\}$. Let CA be a set of pairs of L, $\{(x_1, y_1), \ldots, (x_n, y_n)\}$, read as '$x_1$ counts as y_1', etc.*

Moreover, consider the following proof rules conjunction for the output (AND), disjunction of the input (OR), and transitivity (T) defined as follows:

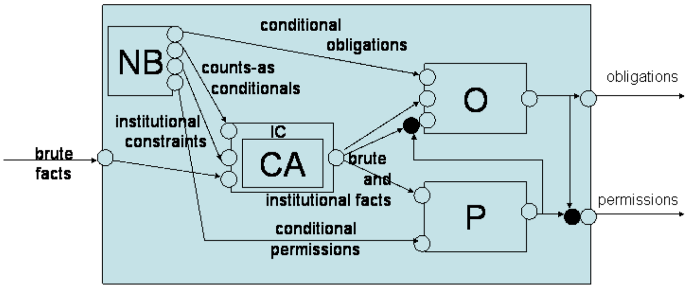
$$\frac{(x, y_1), (x, y_2)}{(x, y_1 \wedge y_2)} AND \qquad \frac{(x_1, y), (x_2, y)}{(x_1 \vee x_2, y)} OR \qquad \frac{(x, y_1), (y_1, y_2)}{(x, y_2)} T$$

For an institution s, the counts-as output operator out_{CA} is defined as closure operator on the set CA using the rules above, together with a silent rule that allows replacement of logical equivalents in input and output. We write $(x, y) \in out_{\mathrm{CA}}(CA, s)$. Moreover, for $X \subseteq L$, we write $y \in out_{\mathrm{CA}}(CA, s, X)$ if there is a finite $X' \subseteq X$ such that $(\wedge X', y) \in out_{\mathrm{CA}}(CA, s)$, indicating that the output y is derived by the output operator for the input X, given the counts-as conditionals CA of institution s. We also write $out_{\mathrm{CA}}(CA, s, x)$ for $out_{\mathrm{CA}}(CA, s, \{x\})$.

Example 1. If for some institution s we have $CA = \{(a, x), (x, y)\}$, then we have $out_{CA}(CA, s, a) = \{x, y\}$.

There is presently no consensus on the logic of counts-as conditionals, probably due to the fact that the concept is not studied in depth yet. For example, the adoption of the transitivity rule T for their logic is criticized by Artosi *et al.* [5]. Jones and Sergot say that “we have been unable to produce any counter-instances [of transitivity], and we are inclined to accept it”.[7]

The main issue in defining constitutive norms like counts-as conditionals is defining their relation with regulative norms like obligations and permissions. Boella and van der Torre [7] use the notion of a logical architecture combining several logics into a more complex logical system, also called logical input/output nets (or lions).



The notion of logical architecture naturally extends the input/output logic framework, since each input/output logic can be seen as the description of a ‘black box’. In the above figure there are boxes for counts-as conditionals (CA), institutional constraints (IC), obligating norms (O) and explicit permissions (P). The norm base (NB) component contains sets of norms or rules, which are used in the other components to generate the component’s output from its input. The

[7] Neither of these authors considers replacing transitivity by cumulative transitivity (CT): $((x \Rightarrow_s y) \,\&\, (x \wedge y \Rightarrow_s z)) \supset (x \Rightarrow_s z)$, that characterizes operations out_3, out_4 of input/output logic.

figure shows that the counts-as conditionals are combined with the obligations and permissions using iteration, that is, the counts-as conditionals produce institutional facts, which are input for the norms. Roughly, if we write $out(CA, G, A)$ for the output of counts-as conditionals together with obligations, $out(G, A)$ for obligations as before, then $out(CA, G, A) = out(G, out_{CA}(CA, A))$.

There are many open issues concerning constitutive norms, since their logical analysis has not attracted much attention yet. How to distinguish among various kinds of constitutive norms? How are constitutive norms (x counts as y) distinguished from classifications (x is a y)? What is the relation with intermediate concepts?

9 Revision of a set of norms

In general, a code G of regulations is not static, but changes over time. For example, a legislative body may want to introduce new norms or to eliminate some existing ones. A different (but related) type of change is the one induced by the fusion of two (or more) codes as it is addressed in the next section.

Little work exists on the logic of the revision of a set of norms. To the best of our knowledge, Alchourrón and Makinson were the first to study the changes of a legal code [2,3]. The addition of a new norm n causes an enlargement of the code, consisting of the new norm plus all the regulations that can be derived from n. Alchourrón and Makinson distinguish two other types of change. When the new norm is incoherent with the existing ones, we have an *amendment* of the code: in order to coherently add the new regulation, we need to reject those norms that conflict with n. Finally, *derogation* is the elimination of a norm n together with whatever part of G implies n.

In [2] a "hierarchy of regulations" is assumed. Few years earlier, Alchourrón and Bulygin [1] already considered the *Normenordnung* and the consequences of gaps in this ordering. For example, in jurisprudence the existence of precedents is an established method to determine the ordering among norms.

However, although Alchourrón and Makinson aim at defining change operators for a set of norms of some legal system, the only condition they impose on G is that it is a non-empty and finite set of propositions. In other words, a norm x is taken to be simply a formula in propositional logic. Thus, they suggest that "the same concepts and techniques may be taken up in other areas, wherever problems akin to inconsistency and derogation arise" ([2], p. 147).

This explains how their work (together with Gärdenfors' analysis of counterfactuals) could ground that research area that is now known as *belief revision*. Belief revision is the formal studies of how a set of propositions changes in view of a new information that may cause an inconsistency with the existing beliefs. Expansion, revision and contraction are the three belief change operations that Alchourrón, Gärdenfors and Makinson identified in their approach (called AGM) and that have a clear correspondence with the changes on a system of norms we mentioned above. Hence, the following question needs to be addressed:

Problem 9. How to revise a set of regulations or obligations? Does belief revision offer a satisfactory framework for norms revision?

Some of the AGM axioms seem to be rational requirements in a legal context, whereas they have been criticized when imposed on belief change operators. An example is the *success* postulate, requiring that a new input must always be accepted in the belief set. It is reasonable to impose such a requirement when we wish to enforce a new norm or obligation. However, it gives rise to irrational behaviors when imposed to a belief set, as observed for instance in [16].

On the other hand, when we turn to a proper representation of norms, like in the input/output logic framework, the AGM principles prove to be too general to deal with the revision of a normative system. For example, one difference between revising a set of propositions and revising a set of regulations is the following: when a new norm is added, coherence may be restored modifying some of the existing norms, not necessarily retracting some of them. The following example will clarify this point:

Example. If we have $\{(\top, a), (a, b)\}$ and we have that c is an exception to the obligation to do b, then we need to retract (c, b). Two possible solutions are $\{(\neg c, a), (a, b)\}$ or $\{(\top, a), (a \wedge \neg c, b)\}$.

Future research must investigate whether general patterns in the revision of norms exist and how to formalize them.

10 Merging sets of norms

In the previous section we have seen that the change over time of a system of norms raises questions that cannot be properly answered within the belief revision framework. We now want to turn to another type of change, that is the aggregation of regulations. This problem has been only recently addressed in the literature and therefore the findings are still very partial.

The first noticeable thing is the lack of general agreement about where the norms that are to be aggregated come from:

1. some works focus on the merging of conflicting norms that belong to the same normative system [14];
2. other works assume that the regulations to be fused belong to different systems [11]; and finally
3. some authors provide patterns of possible rules to be combined, and consider both cases (1) and (2) above [18].

The first situation seems to be more a matter of coherence of the whole system rather than a genuine problem of fusion of norms. However, such approaches have the merit to reveal the tight connections between fusion of norms, non-monotonic logics and defeasible deontic reasoning. The initial motivation for the study of belief revision was the ambition to model the revision of a set of regulations. On the contrary, the generalization of belief revision to *belief merging* is exclusively dictated by the goal to tackle the problem — arising in computer science — of combining information from different sources. The pieces of information are represented in a formal language and the aim is to merge them in an (ideally) unique knowledge base.[8]

Problem 10. Can the belief merging framework deal with the problem of merging sets of norms?

If (following Alchourrón and Makinson) we assume that norms are unconditional, then we could expect to use standard merging operators to fuse sets of norms. Yet, not only once we consider conditional norms, as in the input/output logic framework, problems arise again. But also, most of the fusion procedures proposed in the literature seem to be inadequate for the scope.

To see why this is the case, we need to explain the merging approach in few words. Let us assume that we have a finite number of belief bases $K_1, K_2, \dots, K_n$ to merge. IC is the belief base whose elements are the integrity constraints (i.e., any condition that we want the final outcome to satisfy). Given a multi-set $E = \{K_1, K_2, \dots, K_n\}$ and IC, a merging operator $\mathcal{F}$ is a function that assigns a belief base to E and IC. Let $\mathcal{F}_{IC}(E)$ be the resulting collective base from the IC fusion on E.

Fusion operators come in two types: model-based and syntax-based. The idea of a model-based fusion operator is that models of $\mathcal{F}_{IC}(E)$ are models of IC, which are preferred according to some criterion depending on E. Usually the preference information takes the form of a total pre-order on the interpretations induced by a notion of distance $d(w, E)$ between an interpretation w and E.

Syntax-based merging operators are usually based on the selection of some consistent subsets of E [6,34]. The bases K_i in E can be inconsistent and the result does not depend on the distribution of the wffs over the members of the group.[9]

Finally, the model-based aggregation operators for bases of equally reliable sources can be of two sorts. On the one hand, there are majoritarian operators that are based on a principle of distance-minimization [38]. On the other hand, there are egalitarian operators, which look at the distribution of the distances in E [33]. These two types of merging try to capture two intuitions that often guide the aggregation of individual preferences into a social one. One option is to let the majority decide the collective outcome, and the other possibility is to equally distribute the individual dissatisfaction.

Obviously, these intuitions may well serve in the aggregation of individual knowledge bases or individual preferences, but have nothing to say when we try to model the fusion of sets of norms. Hence, for this purpose, syntactic merging operators may be more appealing. Nevertheless, the selection of a coherent subset

[8] See [35] for a survey on logic-based approaches to information fusion.

[9] [34] refers the term 'combination' to the syntax-based fusion operators to distinguish them from the model-based approaches.

depends on additional information like an order of priority over the norms to be merged, or some other meta-principles.

As the application of belief merging to the aggregation of sets of norms turned out to be unfeasible, an alternative approach is to generalize existing belief change operators to merging rules. This is the approach followed in [11], where merging operators defined using a consolidation operation and possibilistic logic are applied to the aggregation of conditional norms in an input/output logic framework. However, at this preliminary stage, it is not clear whether such methodology is more fruitful for testing the flexibility of existing operators to tackle other problems than the ones they were created for, or if this approach can really shed some light to the new riddle at hand.

A different perspective is taken in [18]. Here, real examples from the Belgian-French bilateral agreement preventing double taxation are considered. These are fitted into a taxonomy of the most common legal rules with exceptions, and the combination of each pair of norms is analyzed. Moreover, both the situations in which the regulations come from the same system and those in which they come from different ones are contemplated, and some general principles are derived. Finally, a merging operator for rules with abnormality propositions is proposed. A limit of Grégoire's proposal is that only the aggregation of rules with the same consequence is taken into account and, in our opinion, this neglects other sorts of conflicts that may arise, as we see now.

The call for non-monotonic reasoning in the treatment of contradictions is also in Cholvy and Cuppens' [14]. A logic to reason when several contradictory norms are merged is presented. The proposal assumes an order of priority among the norms to be merged and this order is also the way to solve the incoherence. Even though this is quite a strong assumption, Cholvy and Cuppens' work take into consideration a broader type of incoherence than in [18]. In their example, an organization that works with secret documents has two rules. R_1 is "It is obligatory that any document containing some secret information is kept in a safe, when nobody is using this document". R_2 is "If nobody has used a given document for five years, then it is obligatory to destroy this document by burning it". As they observe, in order to deduce that the two rules are conflicting, we need to introduce the constraint that keeping a document and destroying it are contradictory actions. That is, the notion of coherence between norms can involve information that are not norms.

11 Conclusion: Deontic logic in context

In this paper we discussed problems of deontic logic that should be considered open and how input/output logic may be useful for analyzing these problems and finding fresh solutions. Jørgensen's dilemma might be overcome by distinguishing operations with norms, like the output $out(G, A)$ of a set of norms G under conditions A, from truth definitions that define what ought to obtain or be done given these norms and conditions. Coherence of a set of norms might be defined with respect to output under constraints, meaning that the set of norms should not generate output for certain conditions that is inconsistent with these constraints. Normative conflicts may be overcome by considering coherent subsets of norms and their output, or such subsets that are preferred given a priority ordering of the norms. Likewise, contrary-to-duty obligations, that obtain in conditions that represent violations, may be modeled by considering only output that is consistent with the input, i.e. the given conditions. Input/output logic provides two possible definitions of dyadic deontic operators, which reconstruct past discussions on whether such operators should be defeasible (in particular in contrary-to-duty conditions), or support strengthening of the antecedent that derives $O(x/a \wedge b)$ from $O(x/a)$. Input/output logic may take into account not just sets of obligating norms, but also explicit permissions, and thus helps shed light on the distinction between weak (negative) permission, where something is permitted if it does not conflict with the norms, and strong (positive) permission which requires an explicit license by the norm-givers. Meaning postulates and intermediate terms, common in legal reasoning but largely ignored by traditional deontic literature, can be taken into account by considering generators T that link natural facts to theoretical terms occurring in the norms, and for counts-as conditionals we may use a separate set of generators (normative institutions) that models how norms are created given an input of natural facts. Finally the questions of how to revise and merge given sets of norms may be approached by preparing the generators (norms) with the aid of standard revision and merging operators.

Lately, normative systems and deontic logic have received widespread attention in multiagent systems and artificial intelligence. A normative multiagent system is "a multiagent system together with normative systems in which agents can decide whether to follow the explicitly represented norms, and the normative systems specify how and in which extent the agents can modify the norms" [10]. Deontic logic, that attempts to formalize the normative consequences given a set of norms and a given situation, can be a helpful tool for devising such systems. In such a general setting, a setting of 'deontic logic in context', many new problems arise: how do deontic truths feature in agent planning and decision making? how do they interact with agent desires, goals, preferences and intentions? how do they feature in communication? how do we model the change of obligations over time, when agents violate or discharge their obligations, when the underlying norms are modified or retracted or when new norms come into existence? The clarification and solution of the problems outlined above, and others, may serve as a first step to make deontic logic fit to become a working component in such a larger setting.

Acknowledgments

The authors wish to thank David Makinson for helpful discussions in the preparation of this paper.

References

1. Alchourrón, C. E. and Bulygin, E., "The Expressive Conception of Norms", in [27] 95–124.
2. Alchourrón, C. E. and Makinson, D., "Hierarchies of Regulations and Their Logic", in [27] 125–148.
3. Alchourrón, C. E. and Makinson, D., "On the Logic of Theory Change: Contraction Functions and Their Associated Revision Functions", *Theoria*, **48**, 1982, 14–37.
4. Anderson, A. R., "On the Logic of Commitment", *Philosophical Studies*, **19**, 1959, 23–27.
5. Artosi, A., Rotolo, A. and Vida, S., "On the logical nature of count-as conditionals", in: *Procs. of LEA 2004 Workshop*, 2004.
6. Baral, C., Kraus, S., Minker, J. and Subrahmanian, V. S., "Combining knowledge bases consisting of first-order theories", *Computational Intelligence*, **8**, 1992, 45–71.
7. Boella, G. and van der Torre, L., "A Logical Architecture of a Normative System", in [17] 24–35.
8. Boella, G. and van der Torre, L., "Permissions and Obligations in Hierarchical Normative Systems", in: *Proceedings of the 9th International Conference on Artificial Intelligence and Law, ICAIL 2003, June 24-28, Edinburgh, Scotland, UK*, ACM, 2003, revised version to appear in *Artificial Intelligence and Law*.
9. Boella, G. and van der Torre, L., "Constitutive Norms in the Design of Normative Multiagent Systems", in: *Computational Logic in Multi-Agent Systems, 6th International Workshop, CLIMA VI*, LNCS 3900, Springer, 2006, 303–319.
10. Boella, G., van der Torre, L. and Verhagen, H., "Introduction to normative multiagent systems", *Computational and Mathematical Organization Theory*, **12(2-3)**, 2006, 71–79.
11. Booth, R., Kaci, S. and van der Torre, L., "Merging Rules: Preliminary Version", in: *Proceedings of the NMR'06*, 2006.
12. Bulygin, E., "Permissive Norms and Normative Concepts", in: Martino, A. A. and Socci Natali, F. (eds.), *Automated Analysis of Legal Texts*, Amsterdam: North Holland, 1986, 211–218.
13. Chisholm, R., "Contrary-to-duty imperatives and deontic logic", *Analysis*, **24**, 1963, 3336.
14. Cholvy, L. and Cuppens, F., "Reasoning about Norms Provided by Conflicting Regulations", in [46] 247–264.
15. van Fraassen, B., "Values and the Heart's Command", *Journal of Philosophy*, **70**, 1973, 5–19.
16. Gabbay, D., Pigozzi, G. and Woods, J., "Controlled Revision — An algorithmic approach for belief revision", *Journal of Logic and Computation*, **13**, 2003, 3–22.
17. Goble, L. and Meyer, J.-J. C. (eds.), *Deontic Logic and Artificial Normative Systems. 8th International Workshop on Deontic Logic in Computer Scicence, DEON 2006, Utrecht, July 2006, Proceedings*, Berlin: Springer, 2006.
18. Grégoire, E., "Fusing legal knowledge", in: *Proceedings of the 2004 IEEE INt. Conf. on Information Reuse and Integration (IEEE-IRI'2004)*, 2004, 522–529.
19. Hansen, J., "Sets, Sentences, and Some Logics about Imperatives", *Fundamenta Informaticae*, **48**, 2001, 205–226.
20. Hansen, J., "Problems and Results for Logics about Imperatives", *Journal of Applied Logic*, **2**, 2004, 39–61.
21. Hansen, J., "Conflicting Imperatives and Dyadic Deontic Logic", *Journal of Applied Logic*, **3**, 2005, 484–511.

22. Hansen, J., “Deontic Logics for Prioritized Imperatives”, *Artificial Intelligence and Law*, *forthcoming*, 2005.
23. Hansen, J., “Prioritized Conditional Imperatives: Problems and a New Proposal”, *Autonomous Agents and Multi-Agent Systems*, 2007, submitted.
24. Hansson, B., “An Analysis of Some Deontic Logics”, *Nôus*, **3**, 1969, 373–398, reprinted in [26] 121–147.
25. Hare, R. M., *Moral Thinking*, Oxford: Clarendon Press, 1981.
26. Hilpinen, R. (ed.), *Deontic Logic: Introductory and Systematic Readings*, Dordrecht: Reidel, 1971.
27. Hilpinen, R. (ed.), *New Studies in Deontic Logic*, Dordrecht: Reidel, 1981.
28. Horty, J. F., “Nonmonotonic Foundations for Deontic Logic”, in: Nute, D. (ed.), *Defeasible Deontic Logic*, Dordrecht: Kluwer, 1997, 17–44.
29. Jones, A. and Sergot, M., “A formal characterisation of institutionalised power”, *Journal of IGPL*, **3**, 1996, 427443.
30. Jørgensen, J., “Imperatives and Logic”, *Erkenntnis*, **7**, 1938, 288–296.
31. Kamp, H., “Free Choice Permission”, *Proceedings of the Aristotelian Society*, **74**, 1973/74, 57–74.
32. Kanger, S., “New Foundations for Ethical Theory: Part 1”, duplic., 42 p., 1957, reprinted in [26] 36–58.
33. Konieczny, S., *Sur la Logique du Changement: Révision et Fusion de Bases de Connaissance*, Ph.D. thesis, University of Lille, France, 1999.
34. Konieczny, S., “On the difference between merging knowledge bases and combinig them”, in: *Proceedings of KR’00*, vol. 8, Morgan Kaufmann, 2000, 135–144.
35. Konieczny, S. and Grégoire, E., “Logic-based approaches to information fusion”, *Information Fusion*, **7**, 2006, 4–18.
36. Lewis, D., *Counterfactuals*, Oxford: Basil Blackwell, 1973.
37. Lewis, D., “Semantic Analyses for Dyadic Deontic Logic”, in: Stenlund, S. (ed.), *Logical Theory and Semantic Analysis*, Dordrecht: Reidel, 1974, 1 – 14.
38. Lin, J. and Mendelzon, A., “Merging databases under constraints”, *International Journal of Cooperative Information Systems*, **7**, 1996, 55–76.
39. Lindahl, L., “Norms, Meaning Postulates, and Legal Predicates”, in: Garzón Valdés, E. (ed.), *Normative Systems in Legal and Moral Theory. Festschrift for Carlos E. Alchourrón and Eugenio Bulygin*, Berlin: Duncker & Humblot, 1997, 293–307.
40. Lindahl, L. and Odelstad, J., “Intermediate Concepts in Normative Systems”, in [17] 187–200.
41. Makinson, D., “On a Fundamental Problem of Deontic Logic”, in [46], 29–53.
42. Makinson, D. and van der Torre, L., “Input/Output Logics”, *Journal of Philosophical Logic*, **29**, 2000, 383–408.
43. Makinson, D. and van der Torre, L., “Constraints for Input/Output Logics”, *Journal of Philosophical Logic*, **30**, 2001, 155–185.
44. Makinson, D. and van der Torre, L., “Permissions from an Input/Output Perspective”, *Journal of Philosophical Logic*, **32**, 2003, 391–416.
45. Makinson, D. and van der Torre, L., “What is Input/Output Logic”, in: Löwe, B., Malzkom, W. and Räsch, T. (eds.), *Foundations of the Formal Sciences II : Applications of Mathematical Logic in Philosophy and Linguistics (Papers of a conference held in Bonn, November 10-13, 2000)*, Trends in Logic, vol. 17, Dordrecht: Kluwer, 2003, 163–174, reprinted in this volume.
46. McNamara, P. and Prakken, H. (eds.), *Norms, Logics and Information Systems*, Amsterdam: IOS, 1999.

47. Niiniluoto, I., “Hypothetical Imperatives and Conditional Obligation”, *Synthese*, **66**, 1986, 111–133.
48. Ross, A., “Imperatives and Logic”, *Theoria*, **7**, 1941, 53–71, Reprinted in *Philosophy of Science* **11**:30–46, 1944.
49. Ross, W. D., *The Right and the Good*, Oxford: Clarendon Press, 1930.
50. Searle, J., *Speech Acts: an Essay in the Philosophy of Language*, Cambridge (UK): Cambridge University Press, 1969.
51. Smiley, T. J., “The Logical Basis of Ethics”, *Acta Philosophica Fennica*, **16**, 1963, 237–246.
52. Spohn, W., “An Analysis of Hansson’s Dyadic Deontic Logic”, *Journal of Philosophical Logic*, **4**, 1975, 237–252.
53. Stenius, E., “The Principles of a Logic of Normative Systems”, *Acta Philosophica Fennica*, **16**, 1963, 247–260.
54. von Wright, G., *Logical Studies*, London: Routledge and Kegan, 1957.
55. von Wright, G. H., “Deontic Logic”, *Mind*, **60**, 1951, 1–15.
56. von Wright, G. H., “A Note on Deontic Logic and Derived Obligation”, *Mind*, **65**, 1956, 507–509.
57. von Wright, G. H., “A New System of Deontic Logic”, *Danish Yearbook of Philosophy*, **1**, 1961, 173–182, reprinted in [26] 105–115.
58. von Wright, G. H., “A Correction to a New System of Deontic Logic”, *Danish Yearbook of Philosophy*, **2**, 1962, 103–107, reprinted in [26] 115–119.
59. von Wright, G. H., *Norm and Action*, London: Routledge & Kegan Paul, 1963.
60. von Wright, G. H., *An Essay in Deontic Logic and the General Theory of Action*, Amsterdam: North Holland, 1968.
61. von Wright, G. H., “Norms, Truth and Logic”, in: von Wright, G. H. (ed.), *Practical Reason: Philosophical Papers vol. I*, Oxford: Blackwell, 1983, 130–209.
62. Ziemba, Z., “Deontic Syllogistics”, *Studia Logica*, **28**, 1971, 139–159.

Towards a Logic of Graded Normativity and Norm Adherence

(*Draft version, 03/11/2007*)

Matthias Nickles

AI/Cognition Group, Computer Science Department
Technical University of Munich
Boltzmannstr. 3, D-85748 Garching b. Muenchen, Germany
nickles@cs.tum.edu

Abstract. A key focus of contemporary agent-oriented research and engineering is on *open multiagent systems* composed of truly autonomous, interacting agents. This poses new challenges, as entities in open systems are usually more or less mentally opaque (e.g., possibly insincere), and can enter and leave the system at will. Thus interactions among such black- or gray-box entities usually imply more or less severe contingencies in behavior: Among other issues, in principle, the adherence of agents to norms cannot be guaranteed in such systems. As a response to this issue, this paper proposes a logic-based approach based on the notion of (possibly probabilistic) behavioral expectations, which are stylized either as adaptive (i.e., predictive) or normative (i.e., prescriptive). Some features of this approach are the enabling of "soft norms" which are automatically weakened to some degree if contradicted at runtime, and the possibility to quantify norm adherence using the measurement of norm deviance.

Keywords. Norms, Modal Logic, Computational Expectations, Social AI, Belief Revision

1 Introduction

A key focus of contemporary agent-oriented research and engineering is on *open multiagent systems* composed of truly autonomous, interacting agents. This poses new challenges, as agents in open systems are usually more or less mentally opaque (e.g., possibly insincere), and can enter and leave the system at will. Among other issues, in principle, the adherence of agents to norms cannot be guaranteed in such systems. As a response, this paper proposes a logic-based approach to the modeling of open multiagent systems in form of probabilistic *behavioral expectations*, which are stylized either as adaptive (i.e., predictive) or normative (i.e., prescriptive), and which can be adapted dynamically at runtime. More concretely, we propose first a probabilistic logic for the representation of agent actions and other events, event sequences, and beliefs and intentions. Then, we define (event-related) normative and adaptive expectations on top of this logic. Doing so, our proposal is a more or less direct logic-based variant of our approach proposed in (Brauer, Nickles, Rovatsos, Weiß & Lorentzen 2002, Lorentzen

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/926

& Nickles 2002, Nickles, Rovatsos & Weiß 2005), adopting the sociological viewpoint regarding expectations and norms which was introduced in (Luhmann 1995*a*). With the approach presented in (Castelfranchi & Lorini 2003) it has in common that expectations are based on intentions and beliefs regarding future events, but otherwise the two approaches are unrelated.

Being a normative expectation is not sufficient to constitute a full-fledged social *norm* (Boella, van der Torre & Verhagen 2007) by itself (mainly because expectations do not by themselves lead to their announcement and enforcement), but all social norms are necessarily grounded in normative expectations (Luhmann 1995*b*, Lorentzen & Nickles 2002): Because only the behavior of an autonomous agent within some shared environment is visible for an observer, while his mental state remains obscure, *beliefs* and *demands* directed to the respective other agent can basically be stylized only as mutable behavioral expectations which are *fulfilled* or *disappointed* in future events (Luhmann 1995*b*). In the case of disappointment, an expectation can either be revised in order to consider the new perception accurately (so-called *adaptive* expectations), or the expecter decides to keep this expectation even contra-factually (so-called *normative* expectations), or to revise (resp. maintain) it only to a certain degree (*adaptive-normative* expectations). In the two latter cases, the expectation holder likely also decides to take action in order to make further disappointments of this expectation less probable (by, e.g., sanctioning unexpected - so-called *deviant* - behavior). And in any case, the expectation can be strengthened/weakened if an expected repeatable event turns out to be useful/useless afterwards.

Thus, we define normative expectations (and thus norms) via the degree of *resistance* to environmental (e.g., social) dynamics *in the course of time*, wrt. how somebody else should behave from the viewpoint of the expectation-holder. In addition, expectations can address the behavior of the expecter *himself* also, which can be useful for the expecter in order to model his self-commitments (intentions regarding own actions), and to communicate them to other agents in form of uttered expectations.

In order to make expectations expected (and thus socially relevant, e.g., as a norm), any kind of expectation needs of course to be *communicated* to others and to be armed with sanctions, if necessary, but concrete ways to do so are outside the scope of this paper.

As for the modeling of norms, we are mainly interested in representing behavioral norms as mental attitudes of some norm giver, such as the designer of the multiagent system (MAS). By representing even the designer of an agent-based application as an agent conceptually, we suggest that the designer of open MAS should not and can not be granted the omniscient, almighty position, as it is the case in by far most current frameworks (e.g., (Ndumu, Collins, Owusu, Sullivan & Lee 1999, Bellifemine, Poggi & Rimassa 2000, Bauer & Müller 2003)). Rather, we see her in the role of a *primus inter pares* among other agents, who, although equipped with more power than “real” agents, should aim for her goals socially (= communicatively) in interaction with the other agents as far as possible. In addition, the openness of open MAS suggests that the development of such systems can only be done in an evolutionary manner, with the need to monitor the system and to improve its model even after deployment during runtime. A way to put the conceptualization of system designers as agents into practice in

a semi-automatic manner is to assign the designer an intelligent, agent-like case tool, as we have proposed in (Brauer et al. 2002, Nickles, Rovatsos & Weiß 2005).
The rest of this paper is organized as follows: The next section introduces a formal language for the representation of mental attitudes of expectation-holders. Section 3 then defines expectations using the means of this language, and Section 4 outlines how expectations can be computed and adapted during runtime. Section 5 concludes.

2 A logic with modalities for intentions and uncertain beliefs

We use the languages proposed in (Cohen & Levesque 1990), Bacchus' logic of uncertain beliefs (Bacchus 1990) and own works (Nickles, Fischer & Weiß 2005, Fischer & Nickles 2005) as a basis. Most aspects of these formalisms have the advantage that they are well established and researched in Distributed AI in the context of modeling agent beliefs and intentions.

The deliberatively rich language[1] $\mathcal{L}^{probDL}$ we propose allows for the representation of

- **event sequences and test expressions.**
- **uncertain beliefs** denoting the an agent believes something with a certain degree. This requires us to use a different semantics compared to the standard belief-intention logics (i.e., for non-gradual beliefs).
- **agent intentions.** This is done in the same way as described above for standard Kripke-type belief-intention logics. Note that intentions might encode demands the agent is self-committed to and which are directed to other agents
(e.g., using $Int(me, otheragent.\ done(someaction))$).
- **normative and non-normative expectations, and norm deviance.** These major enhancements are spelled out in the next section.

2.1 Syntax

Definition 1. *The syntax of well-formed* $\mathcal{L}^{probDL}$ *formulas* $F, F_1, F_2, \ldots$ *and processes* $\alpha, \beta, \ldots$ *is given by*

$$\begin{aligned}F, F_1, F_2, \ldots ::= p \quad | \quad \top \quad | \quad \bot \quad | \quad \neg F_1 \quad | \quad F_1 \wedge F_2 \quad | \quad F_1 \vee F_2 \quad | \quad F_1 \rightarrow F_2 \\ | \quad F_1 \leftrightarrow F_2 \quad | \quad \exists x F_1 \quad | \quad \forall x F_1 \quad | \quad < Pred > (x_1, ..., x_n) \\ | \quad done(\alpha) \quad | \quad happens(\alpha) \quad | \quad now(< Time >) \\ | \quad Bel(a, F) \quad | \quad Bel(a, F, d) \quad | \quad Bel(a, F|c, d) \quad | \quad Int(a, F) \\ Expect(agent, normativity, event|context, strength) \quad | \quad \Delta(event|context, deviance)\end{aligned}$$

$$\alpha, \beta, \ldots ::= action \quad | \quad a_i.action \quad | \quad any \quad | \quad \alpha; \beta \quad | \quad \alpha \cup \beta \quad | \quad \alpha * \quad | \quad F?$$

Here,

[1] which is expected to be easily tailorable to concrete application needs in order to reduce complexity.

- $a, a_i \in$ *Agents (cf. below), being agents;*
- $x, y, z, i, j, k, x_i, y_i, z_i$ *being variables;*
- $< Pred >$ *being a predicate symbol;*
- $action, a_i.action \in A$*, with* A *denoting the set of elementary actions; every elementary action can be indexed with an agent (e.g.,* $agent_3.assert$ *represents the communication act "assert" uttered by* $agent_3$*);*
- *any is an arbitrary action;*
- $< Time >$ *is a time point, denoted as a natural number;*
- $\alpha; \beta$ *denotes sequential process combination (i.e., (sub-)process* α *is followed by* β*);*
- $\alpha \cup \beta$ *denotes non-deterministic choice between* α *and* β*;*
- $\alpha*$ *denotes zero or more iterations of* α*;*
- $F?$ *is a test operation (i.e., the process proceeds if* F *holds true);*
- $Bel(a, F, d)$ *denotes that agent* a *(sincerely) believes with degree* d *that* F*, with* d *being a real valued probability measure and* $0 \leq d \leq 1$*;*
- $Bel(a, F)$ *is a shortcut for* $Bel(a, F, 1)$*;*
- $Int(a, F)$ *denotes that agent* a *(sincerely) intends that* F*;*
- $Bel(a, F|c, d)$ *and* $Bel(a, F|c) := Bel(a, F|c, 1)$ *are shortcuts denoting* contextualized *beliefs (cf. below);*
- $Expect(agent, normativity, event|context, strength)$ *(see next section), and*
- $\Delta(event|context, deviance)$ *(see next section).*

See below for the meaning of $done$, $happens$, and now. Further, let $Agents = \{a_i\}$ be the set of all agents (those currently in the MAS under observation, as well as all other possible agents), $\Theta = \{\alpha, \beta, \gamma...\}$ the set of all syntactically valid processes, and $\Phi = \{F, F_1, F_2, ...\}$ the set of all $\mathcal{L}^{probDL}$ formulas.

Meta-$\mathcal{L}^{probDL}$ statements are given in natural-like language (e.g., "*If* $\models F$ *then* $\models Bel(a, F, 1)$"). In addition, we will sometimes write calculations and other functional expressions directly within $\mathcal{L}^{probDL}$ formulas for simplicity, like in $Bel(a, F, f(1)/2)$.

2.2 Model-based semantics

We propose the usual Kripke-style semantics of belief and intentions, with the "possible worlds" (future as well as past) consisting of finite sequences of events here. Some of these worlds are worlds the agents considers being true (i.e., consistent with the agents belief), others are those worlds the agents wants to become true, thus consistent with the agents intentions [2]. Event sequences (or *courses*) are simply timely ordered, atomic events such as agent actions, with finite length, denoted as $event_1; ...; event_n$. If we additionally assume some meaningful correlation or even causation among events, we speak about (event) *processes*.

A $\mathcal{L}^{probDL}$ model is a structure $(\Theta, Events, Agents, Trajectories, B, I, \Phi)$, where Θ is the universe of discourse, $Agents$ is a set of agents as specified above, $Events$ is a

[2] For simplicity, we do not explicitly introduce goals and desires. Instead, we assume that for the purpose of this work that long-term and, in case of failure, possibly re-established intentions could act as such.

set of events, $Trajectories \subseteq \{(n,e) : n \in \mathbb{N}, e \in Events\}$ is a linearly ordered, denumerable set of worlds in form of event sequences, $B : Agents \times Situations \rightarrow [0;1]$ is a personalized and discrete probability measure over the worlds at all particular time points (so-called *situations*), with $Situations = \{(w,i) : i \in \mathbb{N}, w \in Trajectories\}$ and $\sum_{s \in Situations} B(agent, s) = 1$ for some particular agent $agent$, $I \subseteq Trajectories \times Agents \times \mathbb{N} \times Trajectories$ is the accessibility relation for the agent's intentions (extended with time points, cf. below), and Φ is a predicate interpreting function. B and I are serial, and B is in addition euclidian and transitive.
This structure defines an agent-specific probability distribution over $2^{Situations}$
by $B(agent, S \subseteq Situations) = \sum_{s \in S} B(agent, s)$.

Let M be a $\mathcal{L}^{probDL}$ model, σ a sequence of events (possible world), $n \in \mathbb{N}$ (a time point), ν be a set of variable bindings, relating elements of Θ, $Events$ and σ to variables. The satisfaction of some $\mathcal{L}^{probDL}$ formula F by M, σ, ν is written as $M, \sigma, \nu, n \models F$. To express that an event occurs between two time points n and m, we write $M, \sigma, \nu, n \rhd \alpha \lhd m$ (the exact meaning of this is given below).

With this, the model theoretic semantics of $\mathcal{L}^{probDL}$ is then given by the following rules:

1. $M, \sigma, \nu, n \models now(< Time >)$ iff $\nu(< Time >) = n$.
2. $M, \sigma, \nu, n \models happens(\alpha)$ iff $\exists m, m \geq n : M, \sigma, \nu, n \rhd \alpha \lhd m$ (i.e., α is here an event or event sequence which happens after time n.
3. $M, \sigma, \nu, n \models done(\alpha)$ iff $\exists m, m \leq n : M, \sigma, \nu, m \rhd \alpha \lhd n$ (i.e., α is here an event or event sequence which happened just before time n.
4. $M, \sigma, \nu, n \models Bel(a, F, d)$ iff $B(a, \{s : M, \sigma, \nu, n \models F\}) = d$. This expresses that agent a believes F with strength d if and only if the personalized probability measure B equals d for all situations where F holds.
   Since the relation B in $\mathcal{L}^{probDL}$ models is a probability measure, $d = 1 - d'$ if $Bel(a, F, d) \wedge Bel(a, \neg F, d')$.
5. $M, \sigma, \nu, n \models Int(a, F)$ iff for all $\sigma^*, (\sigma, n, \nu(a), \sigma^*) \in I$. This rule states that F follows from the agents intentions iff F is true in all possible worlds (event sequences) accessible via I, at time n. Observe that it is not required that F is brought about by a. The intention to perform or let someone else perform some action can trivially be expressed with $Int(agent_1, done(agent_2.action))$.

The following defines the occurrence conditions for single and compound events:

1. $M, \sigma, \nu, n \rhd \alpha \lhd n + i$ iff $\nu(\alpha) = \alpha_1; ...; \alpha_i$ and $\sigma_{n+j} = \alpha_j, 1 \leq j \leq i$. This means that the event sequence α happens next to time point n in world σ.
2. $M, \sigma, \nu, n \rhd \alpha \cup \beta \lhd m$ iff $M, \sigma, \nu, n \rhd \alpha \lhd m$ or $M, \sigma, \nu, n \rhd \beta \lhd m$ (i.e., either α or β occurs in the time interval $n...m$).
3. $M, \sigma, \nu, n \rhd \alpha; \beta \lhd m$ iff $\exists k, n \leq k \leq m : (M, \sigma, \nu, n \rhd \alpha \lhd k) \wedge (M, \sigma, \nu, n \rhd \beta \lhd m)$ (i.e., β follows α).
4. $M, \sigma, \nu, n \rhd F? \lhd m$ iff $M, \sigma, \nu, n \models F$. I.e., the test expression $F?$ occurs iff F is true.

5. $M, \sigma, \nu, n \rhd \alpha * \lhd m$ iff $\exists n_1, ..., n_k, n_1 = n, n_k = m \forall i, 1 \leq i \leq m : M, \sigma, \nu, n_i \rhd \alpha \lhd n_{i+1}$. This states that $\alpha*$ occurs iff a sequence of αs occurs.

Like in temporal logic, we can express that F will eventually be true using $\exists \alpha \in A : \mathit{happens}(\alpha; F?)$. $\neg \exists \alpha \in A : \mathit{happens}(\alpha; \neg F?)$ says that F holds always.

We define logically contextualized beliefs by $Bel(a, F|c, d), c \in \varPhi \equiv Bel(a, c \rightarrow \exists \alpha : \mathit{happens}(\alpha; F?), d)$, and beliefs contextualized with event sequences α with $Bel(a, F|\alpha, d), \alpha \in \{\alpha_1; ...; \alpha_n : \alpha_i \in A\} \equiv Bel(a, \mathit{happens}(\alpha; F?), d)$.
Sometimes, we abbreviate $Bel(a, \mathit{done}(\alpha), ...)$ with $Bel(a, \alpha, ...)$.

The semantics of $< \mathit{Pred} > (...), \neg, \wedge, =, \vee, \top, \bot, \exists, \forall, \rightarrow$ and $\leftrightarrow$ is as usual in FOL with equality.

We provide a partial axiomatization, focusing on belief and intention modalities. The $\mathcal{L}^{probDL}$ belief axioms schema includes the well-known K45 (aka weak S5 plus consistency) modal logic axioms schema, adapted for personalized, probabilistic beliefs:

Axioms 1.1:

***K* (closure under consequence))** $(Bel(a, F) \wedge Bel(a, F \rightarrow F')) \rightarrow Bel(a, F')$
***D* (consistency)** $\neg Bel(a, \bot)$
***4* (closure under positive introspection)** $Bel(a, F) \rightarrow Bel(a, Bel(a, F))$
***5* (closure under negative introspection)** $\neg Bel(a, F) \rightarrow Bel(a, \neg Bel(a, F))$

Sometimes, a belief logic also includes the *necessity rule*: "If $\models F$ then $\models Bel(a, F, 1)$", which we do not adopt. I.e., our agents need not to be aware of valid formulas.

Contrary to the famous approach (Cohen & Levesque 1990), which focuses mainly on the interaction of intentions and goals, but in accordance with (Herzig & Longin 2002), we think that the relationship of intentions to the agent's belief is most important. It is governed by the following Bel-Int bridge axioms:

Axioms 1.2:

BelInt1 $Int(a, F) \rightarrow \neg Bel(a, F)$. Agent a intends F to become true only if she does not already believe that F is true already.
BelInt2 $Int(a, Bel(a, \mathit{event}|\mathit{context}))$
$\wedge \neg Bel(a, \mathit{event}|\mathit{context}) \Rightarrow Int(a, \mathit{event})$.
***BelInt2'* (alternatively)** $Int(a, Bel(a, \mathit{event}|\mathit{context}, e))$
$\wedge \neg Bel(a, \mathit{event}|\mathit{context}, e) \Rightarrow Int(a, \mathit{occurs}(\mathit{event}|\mathit{context}, e))$.
This probabilistic version of $Rel_{IntBel2}$ in (Herzig & Longin 2002) would expresses that disbelief in the occurrence of an event with probability e while intending to belief the event occurs with this probability forces the agent to intend the event to occur with probability e
(denoted as $Int(\mathit{agent}, \mathit{occurs}(\mathit{event}|\mathit{context}, e))$). This also expresses that in

case the agent has no particular belief regarding the occurrence of this event, she can bring about her introspective intention to belief in the event even without intending the event itself (e.g., by exploring new perceptions, or by improving her reasoning process). This axiom becomes very important later in the context of normative and adaptive-normative expectations. Unfortunately, the modality $Int(...occurs)$ is not part of $\mathcal{L}^{probDL}$, and maybe shouldn't be, since it is not clear what "shall occur with a certain probability" means exactly. Since we feel that adding such a modality could be problematic, we provided a simpler, harmless variant in form of BelInt2.

BelInt3 $Int(a, F) \rightarrow Int(a, Bel(a, F))$. Note that the opposite direction should not hold: There are other means than intentions to change one's belief, e.g., exploration.

BelInt4 $Bel(a, Int(a, F)) \rightarrow Int(a, F)$. This allows for introspection regarding an agent's intentions.

The following schematic axioms deal with uncertain belief:

Axioms 1.3:

$$Bel(a, F, d) \rightarrow d \geq 0$$
$$Bel(a, F, d) \wedge Bel(a, \neg F, d') \rightarrow d' = 1 - d$$
$$Bel(a, F \wedge F', d) \wedge Bel(a, F \wedge \neg F', d') \wedge Bel(a, F, d'') \rightarrow d'' = d + d'$$
$$\forall x : Bel(a, F, 1) \rightarrow \bigwedge Bel(a, \forall x : F, 1)$$

For δ and $Expect$, cf. the next section.

Being interested in open systems with truly autonomous agents only, we deliberatively do not propose any axioms which would enforce sincerity, collaboration or other properties of benevolence.

3 Expectations as combined mental attitudes with temporal dynamics

Expectations can be weighted in two ways, namely, w.r.t. their *strength* and w.r.t. their *normativity* (or inversely, their *adaptability*). The strength of an expectation indicates its “degree of expectedness” (also called *expectability*): the weaker (stronger) the expectation is, the less likely is or should be its expected fulfilment (violation). Against that, the normativity of an expectation (*both* weak *and* strong expectations) indicates its deliberate “degree of changeability”: the more normative (adaptive) an expectation is, the smaller (greater) is the change in its strength when being contradicted by unpredicted actual actions. With that, the strength of a lowly normative expectation tends to change faster, whereas the strength of a highly normative expectation is maintained in the longer term even if it is obviously inconsistent with reality (e.g., some other agents' activities). Fully normative expectations ($normativity = 1$) ignore the actual occurrences of their modeled events completely, as long as they are not adapted “manually”, whereas fully adaptive expectations ($normativity = 0$) follow the resp. beliefs of the expecting agents, given these beliefs follow themselves any incoming new information regarding the expected events. Thus it is assumed that there is a continuous transition

from weak to strong strength and from low to high normativity. The difference between the probability and the expectability (normativity-biased probability) of a certain event is called *deviance*. So, we can model both gradual and, to some degree, auto-adaptive normative expectations - in contrast to, e.g., binary-style modalities like obligation and permission as in deontic logic.
Some examples (adopted from (Brauer et al. 2002)) of combinations of expectation strength, normativity and deviance:

rules that govern criminal law (strong/non-adaptable/rather low deviance in western countries: even hundreds of actual murders will not alter the respective laws, and most people think of murder as a rather exceptional event);

habits (strong/adaptable,low deviance: before the times of fast food, people took full service in restaurants for granted, but as fast food became popular, they were willing to abandon this expectation);

adherence to public parking regulations (strong/hardly adaptable/high deviance: almost everyone violates them even if they are, in principle, rigid);

and *shop clerk friendliness* (weak/adaptable/indefinite deviance: most people expect bad service but are willing to change their view once encountering friendly staff).

Thus, the term "expectation" is inherently ambiguous, as it deliberatively combines subjective, demanding expectations (reflecting the goals and intentions of the expecting agent) and the empirical likeliness of events (desired or not). In this regard it is worth to state that even the strengths of fully-adaptive expectations are not necessarily probabilities (from a frequentist point of view), because expectations are maintained ("expected") as a part of the belief a subjective observer has, and do not necessarily take into account enough "real world" facts to determine expectation strengths objectively when he sets up his expectations. So, not only (adaptive-)normative, but also fully-adaptive expectations could theoretically be used to represent individual, contra-factual preferences ("desired probabilities", so to say) instead of likelihoods. But such contra-factual yet non-normative expectations converge immediately to probabilities, since they are "willing to learn", so to say.

Starting from these observations, we define the semantics of a so-called *normative* ($normativitiy = 1$) or *adaptive-normative* ($0 < normativity < 1$) expectation held by some agent as his intention to make (or keep) the strength of his belief regarding the (re-)occurrence of the expected event identical with the strength of this expectation. This can be weaker than to intend a certain probability of the event, but as we will see later, in the most common case we actually get by with defining (adaptive-)normative expectations as the intention to make the environment conforming to the expected state to some degree. In contrast expectations without any normativity, simply corresponding to uncertain beliefs, are called *adaptive expectations*.
At this, "intending a probability" can be understood as either aiming at bringing about a certain frequency of a repeatable event, or as the will to provide occurrence conditions for the event that make it probable to a certain degree.
Formally, an agent's expectation (denoted as $Expect$) is a mental attitude, represented as a logic modality, and defined as follows:

Definition 2.

$$Expect(agent, \psi, event|context, e) :\Leftrightarrow \begin{cases} Bel(agent, event|context, e) \\ \ \vee Int(agent, Bel(agent, event|context, e))) \\ \textit{if } \psi > 0 \\ Bel(agent, event|context, e) \textit{ otherwise} \end{cases}$$

Hereby, e is the expectability, and $\psi \in [0; 1]$ is the normativity of the expectation. $\psi = 0$ leads to the special case of an adaptive expectation.
$event$ can theoretically be any proposition, but focusing on actions, if we use $event$, it should in fact be $done(event)$ (the $done$ operator omitted for simplicity).
For convenience, we set
$Expect_t(agent, \psi, event|context, e) \equiv Expect(agent, \psi, event|context, e) \wedge now(< t >)$ to denote expectations held at a certain time.
$\psi = 0$ leads to the special case of an adaptive expectation.

$Bel(agent, event|context, b)$ denotes that $agent$ believes that $event$ occurs with probability b in $context$ [3], and $Int(agent, p)$ denotes that agent intends p to become true (if $agent$ is not capable to bring about the desired fact or action directly by herself, this shall include the intention to make other agents bring about p etc., i.e., to use them like a tool)
We write $Expect(agent, event|context, e)$ as an abbreviation of
$Expect(agent, 0, event|context, e)$, and $Expect_t$ for $Expect$, when the time point t at which the expectation is held matters and can not be derived from the context (for ψ, Int and Bel analogously). Note that t is not the time point at which the event (should) occur(-s). If we would like to express that some event will or should happen at a certain time, we would have to encode this time within $context$.

The exact normativity (except from distinguishing if it is above zero or not) is not used in Definition 2, because the normativity prescribes how an expectability auto-evolves *in the course of time* with new information, if the expectability it is not set “manually”. If the normativity is zero, the expectation is set equal to the belief of the expecter immediately. Otherwise, the expectability adopts gradually to the belief when both differ, with a “learning rate” of the expectation inverse to the normativity.

Our definition of expectation is build straightforwardly upon probabilistic versions of the KD45 and belief-intention axioms usually used for multi-modal logics of mental attitudes (e.g. (Herzig & Longin 2002)), and is related to Sadek’s *want* attitude (Sadek 1992).

Given the agent’s belief (e.g., obtainable from an expectation via the so-called *deviance*, cf. below), the following proposition obviously holds, given
$Expect(agent, \psi, event|context, e)$:

Observation 1:

[3] We can also use this syntax to denote *expected expectations*: $Expect(agent_1, ..., Expect(agent_2, ...)...)$.

$Int(agent, Bel(agent, event|context, e))$
 if $(\psi > 0 \wedge Bel(agent, event|context, ne), ne \neq e)$
$Bel(agent, event|context, e)$ otherwise

If we would either drop the usual $Bel(p) \rightarrow \neg Int(p)$ axiom in belief-intention logics, or introduce alternatively *maintenance intentions* (Bratman 1987) (denoted as Int^M), Definition 2 would change to

Definition 3. *(alternatively to Definition 2)*

$$Expect^{alt}(agent, \psi, event|context, e) :\Leftrightarrow \begin{cases} Int^M(agent, Bel(agent, event|context, e))) \\ \textit{if } \psi > 0 \\ Bel(agent, event|context, e) \textit{ otherwise} \end{cases}$$

The agent can achieve the intention to revise his belief in several ways, possibly even concurrently.

i. Change the world This is considered to be the usual way to enforce adaptive-normative and normative expectations, either by execution of the expected events by the expecting agent herself, or by bringing about the intended events indirectly (e.g., by asking other agents to do so).

ii. Explore The agent can try to obtain new perceptions in order to change his belief by exploration. Here, the (adaptive-)normative expectation serves as a kind of hypotheses, and the agents strives after new evidence in order to support or refute it.

iii. Wait This is actually not covered by the original intention at time t, but is a way to automatically decrease the "strength" of the intention (i.e., the degree and duration of the self-commitment) in consecutive time steps instead: If the normativity is below 1, in the longer term the expectation *learns* (i.e., adapts to the current probability), provided the probabilities of a certain event remain stable enough to be learnable (cf. 4). Practically, this happens if the expectation holder failed to decrease the deviance actively (due to insufficient social power, for example). The adaptation of the expectability to the probability in this case can nevertheless be desired, and it can even be a prerequisite for the enforcement of less flexible and thus likely more important expectations.

iv. Ignore the deviance Here, the agents simply believes that the expected event will occur, possibly ignoring reality thereby:
$Bel(agent, event|context, e) \wedge Expect(agent, \psi, event|context, e)$ holds *in any case* then.
Such deliberative ignorance appears to be irrational for intelligent agents, but is a common attitude of human agents and obviously somewhat functional for them. In any case, the identification of certain expectations with beliefs regardless of deviance might be reasonable for artificial agents in case the event belief is obtained from an unreliable source.
A less debatable use for such deliberative ignorance is to set the normativity greater zero in order to filter out ("flatten") temporal and insignificant fluctuations of probabilities.

In all cases except from iv., we assume that the expectability is equal to the probability (in case the normativity is zero).

Note that even for the cases i.-iii. so far no assumptions have been made on how e has been obtained - an agent is basically free to hold any expectabilities she likes / is interested in from her subjective and possibly irrational viewpoint.

Definition 4. *The* deviance Δ *of an event regarding a certain expectation (or vice versa of an expectation regarding an event) is defined with*

$\Delta(event|context) = e - p$,
given that $Bel(agent, event|context, p)$ *and* $Expect(agent, \psi, event|context, e)$ *holds.*

We integrate deviance measures into $\mathcal{L}^{probDL}$ using
$M, \sigma, \nu, n \models \Delta(event|context, e{-}p)$ iff $\exists e, p, \psi : M, \sigma, \nu, n \models Bel(a, F|context, p) \wedge$ $M, \sigma, \nu, n \models Expect(agent, event|context, \psi, e)$.
Sometimes we use $\Delta(event|context) = e - p$ as a syntactic variant.

A deviance can intuitively be seen as an indicator of the effort that would be required to make a normatively expected event happen, and as a measure for the compliance of the event-generating agent with the expectation, whereas the normativity is intuitively a kind of "stamina" of the intention (the strength of a self-commitment. Please remember in this regard, that we allow intentions also to be denoted as desired behavior of other agents).

Trivially, the deviance can be used to retrieve a probability p from an expectability.

There is also a conjunction with the *utilities* of events: If the normativity is larger zero, the utility for the agent to reach the specified probability is certainly larger zero also. The expectability *might* correspond to the utility of the event in this case (but this is to state a heuristic only, suggesting further research).

Observation 2:

Except from the case iv. above (belief despite ignorance of event occurrences)

$$Int(agent, \forall t_i, t \leq t_i \leq t + h : \Delta_{t+i}(event|context) = 0)$$

holds at time step t. At this, h is a possibly infinite intention horizon which determines how long the expectation is maintained, and Δ_{t+i} is defined analogously to $Expect_t$.

Finally, we want to further simply the semantics in case the probability of an intended event is irrelevant:

Observation 3:

$(Expect(agent, \psi, event|context, e) \wedge Bel(agent, event|context, en), en)$,
$\rightarrow Int(agent, event)$, if $en < e$

4 Computational adaptation of expectations at runtime

The expectability of an event is a function of event probability and normativity, whereby the normativity can be interpreted as the "stubbornness" of the expectation, or, inversely, its flexibility. After the expectabilities and normativities of adaptive-normative expectations have been obtained from goals and intentions, they are exposed to reality, so to say. One driving force for the run time adaptation of such expectations is the active influencing of the domain of the expected events in order to enforce normative and adaptive-normative expectations, another is to let such expectations adopt to empirical expectations passively. The following shows how this can be done in dependance from the normativity. As important special cases, the following definition covers expectations with normativity zero and one also.

To this end, it is assumed that for an event $event|context$ corresponding to a certain EN node an initial expectation strength $\theta(event, context) = P_0(event|context)$ exists. We define thereby for convenience $Bel_t(a, F|context, d) \equiv Bel(a, F|context, d) \wedge now(< t >)$ and $P_t(a|context) = d \Leftrightarrow Bel_t(a, F|context, d)$, denoting a probability stated at time t (not the probability of an event happening at time t) (cf. Definition 2 for $Expect_t$). Given a normativity ψ_t and a probability $P_t(event|context)$ (e.g., in form of a belief) obtained empirically at time step t, the expectation strength at this time step can be calculated recursively as follows. This way to calculate $Expect_t$ is not obligatory, other ways to calculate adaptive-normative expectations could be reasonable too, depending from the concrete application.

Definition 5. *With* $E_t(agent, \psi_t, event|context) = e \leftrightarrow Expect_t(agent, \psi_t, event|context, e)$, $E_t(agent, event|context, \psi_t) =$

$$\begin{cases} \theta(event, context) \text{ \textit{if} } t < 1 \\ E'_{t+1}(agent, \psi_t, event|context) \text{ \textit{otherwise}} \end{cases}$$

with $E'_t(agent, \psi_t, event|context) =$

$$\begin{cases} E'_{t-1}(agent, \psi_t, event|context) \\ \qquad -\Delta'_{t-1}(event|context)(1 - \psi_t) \\ \quad \text{\textit{if} } t > 0 \\ \theta(event, context) \text{ \textit{otherwise}} \end{cases}$$

$\Delta'_t(event|context)$ is calculated as
$E'_t(agent, \psi_t, event|context) - P_t(event|context)$[4].

This (non-mandatory) way to calculate $Expect_t$ reminds of the econometrics technique of *Exponential Smoothing* used for the smoothing and extrapolation of non-linear

[4] Calculating $Expect_t(...)$ using $Expect'_{t+1}(...)$ is done just in order to get rid of the delay of one time step in the adaptation of $Expect_t(...)$ to $P_t(...)$ that would exist otherwise.

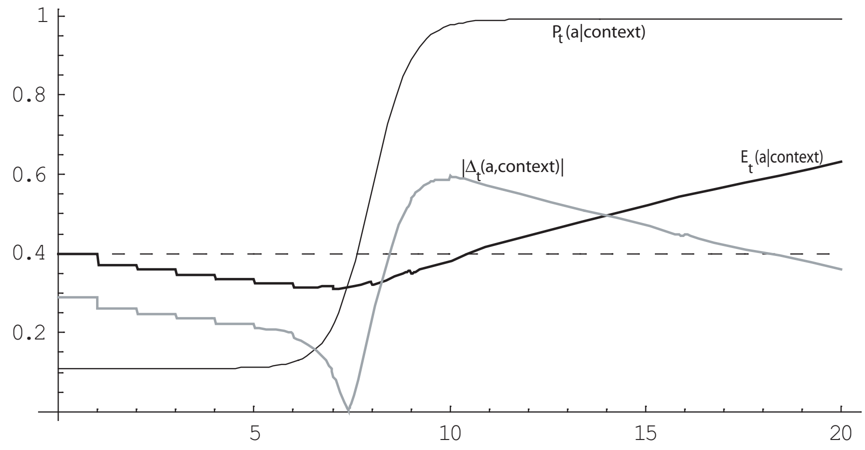


Fig. 1. Unattended adaptation of an expectability ($t \to$)

time series. It calculates a flattened version (with a flattening degree depending on the normativity) of the graph of $P_t(event|context)$, and lets $Expect_t(agent, \psi_t, event|context)$ converge to $P_t(event|context)$ at least if $P_t(event|context)$ remains constant with increasing t, and ψ_t remains constant also. The normativity (i.e., the expectation adaptation rate) itself does not change.

If, e.g., $\psi_t = 1$, the expectation strength e in $Expect_t(agent, \psi_t, event|context), e = \theta(contex, event)$ will remain constant, whatever the empirical evidence is. In contrast, if $\psi_t(agent, contex, event) = 0$, $Expect_t(agent, \psi_t, event|context, e) withe = P_t(event|context)$ applies at all time steps.

Example:

Figure 1 shows the time and normativity dependent expectabilities (abbreviated with E_t) of an event a, with $\psi_{0..20} = 0.95$ and $\theta(a, context) = 0.4$. Being a fictive event, the potential effect the announcement of these values to the event generator (a communication partner of the agent, for example) would have, is not considered. The *agent* parameter has been omitted.

5 Conclusion

Most traditional formalisms for normative systems clearly restrict the agents' autonomy without any concept of flexibility. In demarcation from such approaches, we aim at avoiding this by accepting autonomy even from social norms as a necessary characteristic of agency that must not be ruled out, and sometimes even can not be ruled out at all, as it is typical for truly open multiagent systems. Based on Luhmann's theory of social systems and our previous works (Brauer et al. 2002), this is in the line of Castelfranchi's view: A socially oriented perspective of engineering order in agent systems is needed and most effective (Castelfranchi 2000). In addition to that, this sociological grounding also makes our approach different from approaches that apply sociological concepts and terminology in a comparatively superficial and more or less ad-hoc manner. Thus we hope that the introduction of adaptive-normative expectations opens a new perspective of multiagency and normative systems.

References

Bacchus, F. (1990), *Representing and Reasoning with Probabilistic Knowledge*, MIT Press, Cambridge, Massachusetts.

Bauer, B. & Müller, J. (2003), Using UML in the context of agent-oriented software engineering, *in* P. Giorgini, J. Müller & J. Odell, eds, 'Agent-oriented software engineering. Proceedings of the Fourth International Workshop (AOSE-2003)', Lecture Notes in Artificial Intelligence, Vol. 2935, Springer-Verlag, pp. 1–24.

Bellifemine, F., Poggi, A. & Rimassa, G. (2000), Developing multi-agent systems with JADE, *in* C. Castelfranchi & Y. Lespérance, eds, 'Intelligent Agents VII, Proceedings of the Seventh International Workshop on Agent Theories, Architectures, and Languages (ATAL-2000)', Springer-Verlag.

Boella, G., van der Torre, L. & Verhagen, H. (2007), Introduction to normative multiagent systems, *in* G. Boella, L. van der Torre & H. Verhagen, eds, 'Proceedings of the Dagstuhl Seminar on Normative Multiagent Systems 2007'.

Bratman, M. (1987), *Intentions, Plans and Practical Reasoning*, Harvard University Press, Cambridge, MA.

Brauer, W., Nickles, M., Rovatsos, M., Weiß, G. & Lorentzen, K. (2002), Expectation-oriented analysis and design, *in* M. Wooldridge, G. Weiß & P. Ciancarini, eds, 'Agent-oriented software engineering. Proceedings of the Second International Workshop (AOSE-2001)', Lecture Notes in Artificial Intelligence, Vol. 2222, Springer-Verlag, pp. 226–244.

Castelfranchi, C. (2000), Engineering social order, *in* 'Working Notes of the First International Workshop on Engineering Societies in the Agents' World (ESAW-00)'.

Castelfranchi, C. & Lorini, E. (2003), Cognitive anatomy and functions of expectations, *in* 'Proceedings of IJCAIŠ03 Workshop on Cognitive Modeling of Agents and Multi-Agent Interactions'.

Cohen, P. & Levesque, H. (1990), 'Intention is choice with commitment', *Artificial Intelligence* **42**, 213–261.

Fischer, F. & Nickles, M. (2005), Towards a notion of generalised statement-level trust, Technical Report FKI-249-05, Institut für Informatik, Technical University of Munich.

Herzig, A. & Longin, D. (2002), A logic of intention with cooperation principles and with assertive speech acts as communication primitives, *in* 'Proceedings of the first international joint conference on Autonomous agents and multiagent systems (AAMAS 2002)'.

Lorentzen, K. & Nickles, M. (2002), Ordnung aus Chaos - Prolegomena zu einer Luhmann'schen Modellierung deentropisierender Strukturbildung in Multiagentensystemen, *in* T. Kron, ed., 'Luhmann modelliert. Ans"atze zur Simulation von Kommunikationssystemen', Leske & Budrich.

Luhmann, N. (1995*a*), *Social systems*, Stanford University Press, Palo Alto, CA. Originally published in 1984.

Luhmann, N. (1995*b*), *Social Systems*, Stanford University Press, Palo Alto, CA. (originally published in 1984).

Ndumu, D., Collins, J., Owusu, G., Sullivan, M. & Lee, L. (1999), 'ZEUS: A toolkit for building distributed multi-agent systems', *Agentlink News* **2**, 6–9.

Nickles, M., Fischer, F. & Weiß, G. (2005), Communication attitudes: A formal approach to ostensible intentions, and individual and group opinions, *in* W. van der Hoek & M. Wooldridge, eds, 'Proceedings of the Third International Workshop on Logic and Communication in Multiagent Systems (LCMAS 2005)'.

Nickles, M., Rovatsos, M. & Weiß, G. (2005), 'Expectation-oriented modeling', *Engineering Applications of Artificial Intelligence (EAAI)* **18**.

Sadek, M. (1992), A study in the logic of intention, *in* 'Proceedings of the third international conference on principles of knowledge representation and reasoning (KR'92)'.

Towards a General Framework for Modelling Roles

Valerio Genovese

Università di Torino

Abstract. Role is a widespread concept, it is used in many areas like MAS, Programming Languages, Organizations, Security and OO modelling. Unfortunately, it seems that the literature is not actually able to give a uniform definition of roles, there exist several approaches that model roles in many different (or even opposite) ways. In this draft we start to define a meta-model for roles. Our aim is to build a formal framework through which we can describe different roles appeared in the literature or implemented in up and running computer systems. In particular we give a new definition of role's foundation introducing *sessions*, which are a formal instrument to talk about role's states and we show how sessions may be useful to model many different role's accounts.

keywords: Roles, Organizations, Object Oriented Modelling, Multi-Agent Systems, Security.

1 Introduction

The notion of role is a modelling concept strictly linked with interaction between entities. In natural language, we notice that terms like "student", "employee" or "president" are linked with a person who plays them and a *context* in which the player interact, the term "student" refers to a person that is a student in a specific university [1]. In a certain way, we can view roles as a way to model an interaction, but problems arise because it is not completely clear how many different types of interactions exist and is possible to represent in the OO paradigm.

There are many definitions of roles, each one with a plausible approach based on intuition, practical needs and, sometimes, on a formal account. In security, roles are seen as a way to distribute permissions [2], in organizational models roles gives *powers* to their players in order to access an institution, in MAS roles could be seen as descriptions of the behaviour which is expected by agents who play them [3], in ontology research roles are an anti-rigid notion founded on a player and a context [4], and many more. Even in the same field of research, there exist in the literature completely different notions of role which are in contrast with each other. Roles are not so easy to grasp, it seems that each different approach underlines a particular part of a common phenomenon not definable in a unique way.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/925

The main goal of this work is to provide a flexible formal model for roles, which is able to grasp the basic primitives behind the different role's accounts in the literature, rather than a definition. If it is possible to define such a model,then we can study the key properties of roles in different practical implementations.

The paper is organized as follows. In section 2 we introduce the model. In section 3 we describe different roles approaches with the new formalism to show its generality, in particular we analyze social roles [4], roles as non instantiable types [5].

2 A Logical Model for Roles

Examining the literature, from an ontological point of view, we find that roles come in an universal and individual flavour, so also in our model we decide to stick to this approach in order to be able to cover a wider range of roles' definitions. To describe the model we use an OO vocabulary, because is would be easier to eventually extend our role's definition into an implementative solution. We prefer to define the formalism as much general as possible, this gives us an unconstrainted model where special constraints are added later in order to describe different approaches.

2.1 Universal Level

Definition 1 An *universal model* is a tuple

$$< \mathsf{D}, \mathsf{Contexts}, \mathsf{Players}, \mathsf{Roles}, \mathsf{Attr}, \mathsf{Op}, \mathsf{Constraints} >$$

where:

- D is a domain of classes
- $\mathsf{Contexts} \subseteq$ D is a set of institutions
- $\mathsf{Player} \subseteq$ D is a set of potential players or *actors*
- $\mathsf{Roles} \subset$ D is a finite set of *roles* $\{R_1, ..., R_n\}$
- Attr is a set of *attributes*
- Op is a set of *operations*
- $\mathsf{Constraints}$ is a set of *Constraints*

The static model has also a few functions and relations:

- PL $\subseteq$ $\mathsf{Players}$ x Roles: this relation states, at the universal level, which are the players that can play a certain role.
- RO $\subseteq$ Roles x $\mathsf{Contexts}$: each role is linked with one or more contexts.
- AS $\subseteq$ D x Attr: is an attribute assignment relationship, through which we can assign to each class its attributes.
- OS $\subseteq$ D x Op: is an operation assignment relationship, through which we can assign to each class its operations.
- RH $\subseteq$ Roles x Roles is a partial order relationship called role hierarchy, also written as $\geq_{RH}$. If $r <_{RH} r^{'}$, we say that r inherits all Attr and Op which belong to $r^{'}$.

- PH $\subseteq$ $\mathsf{Players}$ x $\mathsf{Players}$ is a partial order relationship called player hierarchy, also written as $\geq_{PH}$. If $p <_{PH} p^{'}$, we say that p inherits all Attr and Op which belong to $p^{'}$.
- CH $\subseteq$ $\mathsf{Contexts}$ x $\mathsf{Contexts}$ is a partial order relationship called context hierarchy, also written as $\geq_{CH}$. Is $c <_{CH} c^{'}$, we say that c inherits from $c^{'}$.

At this point we can add into $\mathsf{Constraints}$ some logical rules in order to model different role notion. For example in powerJava each role is linked with one and only one context [6], so we can express this through the following constraint:

$$\forall x, y, z(x \in \mathsf{Roles}\ y, z \in \mathsf{Contexts}\ x\mathsf{RO}y \wedge x\mathsf{RO}z \rightarrow y = z) \qquad (1)$$

2.2 Individual level

Definition 1 A *snapshot model* is a tuple

$$< \mathsf{O}, \mathsf{Institutions}, \mathsf{Actors}, \mathsf{R\_Instances}, \mathsf{Sessions}, \mathsf{Attr}, \mathsf{Op}, \mathsf{Val} >$$

where:

- O is a *domain* of objects which instantiate classes in D.
- $\mathsf{Institutions} \subseteq \mathsf{O}$ is a set of institution which instantiate classes in $\mathsf{Contexts}$.
- $\mathsf{Actors} \subseteq \mathsf{O}$ is a set of (potential) actors, which instantiate universals in $\mathsf{Players}$.
- $\mathsf{R\_Instances} \subset \mathsf{O}$ is a set of *roles instances*.
- $\mathsf{Sessions}$ is a set of *sessions*, which keep the state of an interaction between actors and $\mathsf{institutions}$.
- Attr is a set of *attributes*
- Op is a set of *operations*
- Val is a set of *values*

The snapshot model has also a few functions:

- I_{Roles} is a *role assignment* that assigns to each role R a relation on $\mathsf{Institutions}$ x Actors x $\mathsf{Sessions}$ x $\mathsf{R\_Instances}$.
- π_{Attr} is a *projection function* that assigns to each role R a subset from Attr, which are the attributes of role R assigned by the relationship AS at the universal level.
- π_{Op} is a *projection function* that assigns to each role R a subset from Op, which are the operations of role R assigned by the relationship OS at the universal level.
- I_{Attr} is an *assignment function* which it takes as arguments an object P $\in$ O, and an attribute p $\in \pi_{Attr}(R)$, if p has a value v $\in$ Val it returns it, $\emptyset$ otherwise.

When an object x is the individual of the universal y, we say that x instantiates y and, in order to express this in a formal way, we write $x :: y$. Intuitively, a snapshot represents the state of a system at a certain particular point in time.

The role assignment function I_{Roles} gives us the *role instances*: if $i{::}x$ is an institution, $a{::}y$ an actor, s a session, and o$::R$ a role, such that $(R, a) \in \mathsf{RO}$ and $(y, R) \in \mathsf{PL}$, the fact $(i,a,s,o) \in I_{Roles}(R)$ is to be read as: "the object o represents agent a playing the role R in institution i in session s". We will often write R(i,a,s,o) for this statement, and we call o the role instance. When it is not interesting to talk about sessions we can write R(i,a,o).

Suppose we have a role employee, and that the value of the attribute salary is 1000 € usually, instead of writing $I_{Attr}(\mathsf{employee}, \mathsf{salary}) = \mathsf{1000}$, we write

$$\mathsf{salary}(\mathsf{employee}) = \mathsf{1000}$$

We have used terms like institutions and actors from the literature on roles in organizations, but these terms must be taken rather broadly.Institutions suggests organizations like governments and banks, and we indeed have such applications in mind, with actors being people holding certain positions within such institutions. But the model is intended to capture a much wider range of phenomenons: institution may be folders in a file system or any object structured in roles , and actors its users, operations or attributes their permissions, and roles a way of organizing these permissions. Or even further away from the metaphor, an institution may be a relation (such as 'love') in an ER model, with roles of *lover* and *lovee* filled by actors.

We have tried to formulate the present definition in a way that is a compromise between simplicity and generality, which allows us to focus on facets of the model that are specific of roles without being hindered too much by formal details. The way we defined a snapshot leaves a lot of room for formulating further constraints that may or may not be reasonable to assume, depending on the particular role's definition we have in mind. Here are a number:

1. *Identity of role instances.* Should a role played by an actor be seen as an object per se, i.e. as a "qua-individual", or the fact that an actor plays a role simply extend or change the properties of the actor itself? The choice translates in a constraint on the roles in Roles. If we see qua-individuals as objects per se, this corresponds to the constraint that:

$$R(i, a, s, x) \rightarrow x \neq a$$

   which is valid for powerJava [6], but also for social roles [4]. The opposite of this constraint is that roles simply change the objects themselves - qua individuals as such do not exist:

$$R(i, a, s, x) \rightarrow x = a$$

   which is the natural option in an RBAC model, for example, in such a case we can write simply R(i,a,s) because,as already said, I_{Roles} maps R to Institutions x Actors x Sessions.

2. *Combinations of Roles.* In powerJava, each actor can play a role at most one time within a single institution, i.e.

$$R(i, a, x) \wedge R(i, a, y) \rightarrow x = y$$

In this assumption that allows for the use of 'role casting' in powerJava to refer to role instances: an expression of the form *(i.R)a* can be used to denote the unique object x such that *R(i,a,x)*.
Variants on these constraint can be formulated as well.
If an actor can play at most one role within an institution translates to the fact that for each R $\neq R^{'}$:

$$R(i,a,x) \rightarrow \neg R^{'}(i,a,x)$$

3. *Inheritance of attributes*. In the model, both roles and objects have properties. A natural constraint is that role-instances all the properties that are defined for that role:

   $$R(i,a,x) \rightarrow (\mathsf{attr} \in \pi_{Attr}(R) \rightarrow \exists v : \mathsf{attr}(\mathsf{x}) = v)$$

   With respect to the question if the role-player should 'inherit' all the properties of the original player object there are different possible answers.
   For example, in powerJava, no such inheritance is assumed at all - the properties of the role instance are precisely those of the role, and we have that:

   $$R(i,a,x) \rightarrow (\mathsf{attr} \in \pi_{Attr}(R) \leftrightarrow \exists v : \mathsf{attr}(\mathsf{x}) = v)$$

   But other options are possible as well. For example, one alternative view is that roles can be best seen as 'views' on a certain object, providing only a *subset* of the properties of the original object. A constraint which reflects that view has it that the role-player only has the properties that are defined for the original object as well as for the role:

   $$R(i,a,x) \rightarrow (\mathsf{attr}(x) = v \leftrightarrow (\mathsf{attr}(R) \wedge \mathsf{attr}(a) = v))$$

   The opposite view is that roles *add* properties to the players. For example, in the Zope security model (like also in RBAC) we have the following:

   $$R(i,a,x) \rightarrow (\mathsf{attr}(x) = v \leftrightarrow (\mathsf{attr}(R) \vee \mathsf{attr}(a) = v))$$

4. *Dependence of roles on institutions*. In our model it is presupposed that the identity of a role instance depends not only on the role and the actor involved, but on an 'institution' as well. This is often, but not always, appropriate. We can mimic the case were the introduction on institutions is unnecessary with the introduction of a 'trivial' institution, and let Institutions contains only this trivial institution, as we do in section 3 when we model RBAC [2].
5. *Context coherence*. From an organizational point of view, there cannot be a student role's player without a teacher one, also it wouldn't be sensible to talk about the context family without someone who plays the role of husband and another one being the wife. To express this constraint we can state, for example, the following integrity rule:

   $$\exists y :: Family \leftrightarrow husband(y,x,o) \wedge wife(y,z,p)$$

   Which means that in the snapshot exists $y \in Institutions$ if and only if there exist two role instances o_1 and o_2 which represent respectively an *husband* and a *wife* played by actors x and z in y.

2.3 The dynamic model

There are two kinds of ways in which our model can change. In the next section, we turn to the question of changing the properties of objects in the model. But first we look at the ways that roles can be added and deleted. For simplicity, in this section we always write the role instance as *R(i,a,o)*, without directly talk about sessions.

Definition 1 A *dynamic model* is a tuple

$$< \mathsf{S}, \mathsf{Actions}, \mathsf{Requirements}, \mathsf{I}_{\mathsf{Actions}}, \mathsf{I}_{\mathsf{Roles_t}}, \pi_{\mathsf{Req}}(\mathsf{t}), \mathsf{I}_{\mathsf{Requirements_t}}, \mathsf{I}_{\mathsf{RClosure}} >$$

where:

- S is a set of *snapshots*.
- $\mathsf{Actions}$ is a set of actions.
- $\mathsf{Requirements}$ is a set of requirements.
- $\mathsf{I}_{\mathsf{Actions}}$ maps each action from $\mathsf{Actions}$ to a function on S. $I_{Actions}(a)$ tells us how a snapshot changes as a result of executing action a. If $\mathsf{Sessions} \neq \emptyset$ the change takes place if and only if all role instances in the resulting snapshot are coherent with their corresponding sessions (for a complete discussion about sessions see section 2.4).
- About I_{Roles_t} we say that $R_t(i, a, o)$ is true if there exists, at a time t, the *role instance* $R(i, a, o)$.
- $\pi_{Req}(t, R)$ returns a subset of $\mathsf{Requirements}$ present at a given time t for the role R, which are the requirements that must be fulfilled in order to create the R's role instance.
- $I_{Requirements_t}$ is a function that, given (i,a,R,t) returns True if the actor a fills the requirement in $\pi_{Req}(t, R)$ to play the role R in the institution i, False otherwise. We often write $Req_t(i, a, R)$.
- $I_{RClosure}(a, t)$ given an actor a it returns all its roles played by a at time t.
- $I_{RPlayers}(R, t)$ given a role R it returns all its players at time t.

Intuitively, the snapshots in S represent the state of a system at a certain time. We suppose that, for every time t, given an object p we can always say if it exist or not via the $?_t$ operator, so that $?_t(p)$ is true, iff p exists at time t, false otherwise.

A particular case of a dynamic model is something that we can call somewhat unelegantly a *role addition-deletion model.* It has actions corresponding to role assignment for each R, i and a, which are supposed to capture the effect of adding the role R within institution i to actor a, and actions that represent the taking away from a the role R in institution i.

Of course, these actions will not be arbitrary. We first identify a number of minimal properties that the action of role assignment need to satisfy.

Definition 1 ***(role assignment)*** *let M be a snapshot.*

$$< \mathsf{O}, \mathsf{Institutions}, \mathsf{Actors}, \mathsf{Roles}, \mathsf{Attr}, \mathsf{Op}, \mathsf{Val} >$$

Let $i \in \mathsf{Institutions}$, $a \in \mathsf{Actors}$, and $R \in \mathsf{Roles}$. There are two possibilities, if we want to assign role R to actor a: either if fails, or it succeeds. In the latter case, the resulting snapshot:

$$\mathsf{M}^{'} =< \mathsf{O}^{'}, \mathsf{Institutions}^{'}, \mathsf{Actors}^{'}, \mathsf{Roles}^{'}, \mathsf{Attr}^{'}, \mathsf{Op}^{'}, \mathsf{Val}^{'} >$$

should satisfy the following properties:

- A role assignment may add at most one new object to the domain (namely the newly introduced qua-individual). $O^{'} = O \cup \{o\}$, where o may or may not already be in O.
- $\mathsf{Institutions}^{'} = \mathsf{Institution}$ or $\mathsf{Institutions}^{'} = \mathsf{Institution} \cup \{\mathsf{o}\}$.
- $\mathsf{Actors}^{'} = \mathsf{Actors}$ or $\mathsf{Actors}^{'} = \mathsf{Actors} \cup \{\mathsf{o}\}$.
- $\mathsf{Roles}^{'} = \mathsf{Roles}$, $\mathsf{Attr}^{'} = \mathsf{Attr}$, $\mathsf{Val}^{'} = \mathsf{Val}$, $Op^{'} = Op$. The sets of roles, attributes, operations and their possible values do not change.
- $I^{'}_{Roles}$ is just like I_{Roles}, except that $I^{'}_{Roles}(R) = I_{Roles} \cup \{(i, a, o)\}$
- $I^{'}_{Attr}$ is just like $I^{'}_{Attr}$ with respect to the properties of objects different from i, a, and o.

For role addition and deletion we use, respectively $Req_t(i, a, R), R, i \hookrightarrow a$, and $Req_t(i, a, R), R, i \hookleftarrow a$. Then using the notation of dynamic logic we write:

$$[Req_t(i, a, R)?; R, i \hookrightarrow a]\phi$$

to express that, if actor a fills the requirements at time t ($Req_t(i, a, R)$ is True), after assigning role R within institution i, ϕ is true. If there are no particular Requirements (i.e. $\pi_{Req}(t, R) \in \emptyset$) we can omit Req_t. The above definition gives us the possibility to model that a role assignment introduces a role instance:

$$[R, i \hookrightarrow a]\exists x R(i, a, x)$$

or the fact that if a does not play the role R within institution i, then the role assignment introduces exactly one role instance:

$$(\neg\exists x R(i, a, x)) \rightarrow [R, i \hookrightarrow a]\exists! x R(i, a, x)$$

And many more.

Definition 1 ***(object deletion)*** An object does not exist after deleting it*:*

$$[\mathsf{delete(o)}]\neg\mathsf{exists(o)}$$

If we delete a role-instance, then we also delete the role from the actor, and similarly for institutions and actors:

$$[\mathsf{delete(i)}]\neg\mathsf{R(i, a, x)}$$

$$[\mathsf{delete(a)}]\neg\mathsf{R(i, a, x)}$$

$$[\mathsf{delete(x)}]\neg\mathsf{R(i, a, x)}$$

For example, Depke *et al.* [7], “A role (instance) will be deleted when the agent is destroyed, i.e., its lifetime is dependent on that of the base agent.”:

$$R(i,a,x) \rightarrow [delete(a)]\neg \mathsf{exists}(\mathsf{x})$$

We might also want to say something about: if an agent has certain properties *because he plays this role,* then upon object deletion, also all properties associated with that role must be removed from the actor.

Definition 1 ***(role deletion)***

$$< \mathsf{O}, \mathsf{Institutions}, \mathsf{Actors}, \mathsf{Roles}, \mathsf{Attr}, \mathsf{Op}, \mathsf{Val} >$$

Let $i \in \mathsf{Institutions}$, $a \in \mathsf{Actors}$, and $R \in \mathsf{Roles}$. Role deletion has different consequences depending on if the role instances have their own identity or not. In the latter case role deletion could be defined in the following way:

$$[R, i \hookleftarrow a] \equiv [delete(x)]$$

where x is the unique role instance linked with the institution i and played by a.

The second, and more subtle case needs to be taken into account when:

$$R(i,a,x) \rightarrow a = x$$

In such a case, we cannot simply remove the role instance x because this would mean to delete the actor once he stops playing role R. We know that when an object plays a role that has no identity it directly acquires new properties, properties that in our model are expressed through attr and Op. The constraint that represent such type of inheritance is, (as already mentioned in section 2.2):

$$R(i,a,x) \rightarrow (\mathsf{attr}(x) = v \leftrightarrow (\mathsf{attr}(R) \vee \mathsf{attr}(a) = v))$$

the same holds for Op. A way to formalize the fact that an actor relinquishes a role without an identity is:

$$[R, i \hookleftarrow a](\pi_{Attr}(a) \cap \pi_{Attr}(R) = \emptyset \wedge \pi_{Op}(a) \cap \pi_{Op}(R) = \emptyset$$

The above formula expresses that an actor who stops playing a role loses all the Attr and Op acquired by the role R.

Methods There are other ways to change the model as well - objects may assign new values to their attributes. Again, the effects of such changes may depend on choices made earlier (e.g. in the case of delegation, changing the attribute value of an object may change the value of that attribute also in some roles he plays)

Here, we will focus on the case in which the attribute-values can be changed by the *objects themselves*. What we will do is to define *methods* of objects with which they can change attributes of their own or those of others. Actually, to simplify the model, we define one single primitive: $\mathsf{set}(\mathsf{o}_1, \mathsf{o}_2, \mathsf{attr}, \mathsf{v})$, which means that o_1 sets the value of attr on o_2 to v.

Now, we will of course have that:

$$[\mathsf{set}(\mathsf{o}_1, \mathsf{o}_2, \mathsf{attr}, \mathsf{v})]\mathsf{attr}(\mathsf{o}_2) = \mathsf{v}$$

which means that if the action of setting this attribute succeeds, then the relevant object will indeed have this value for that attribute.

Now, one interesting question is how to define constraints on attributes access.

Power One observation of the work of Boella and Torre [8] is that certain aspects of the notion of *power* can be captured by how features of one agent can be changed be the actions of another, this approach promote what in software engineering is called modularity. In the present terminology, an object has *power* over another object if that object can change the values of attributes of other object. Or, formally, o_1 has power over o_2 if and only if:

$$\langle \mathsf{set}(\mathsf{o}_1, \mathsf{o}_2, \mathsf{attr}, \mathsf{v}) \rangle \top$$

It is important to underline that o_1 can have power over o_2 in three situations:

$$R(o_1, x, o_2) \vee (R(i, x, o_2) \wedge R(i, x, o_1)) \vee R(o_2, x, o_1) \vee o_1 = o_2$$

so o_1 and o_2 can be role instances or institutions.

In the work of Boella at al. [8], roles are seen as a way of organizing and assigning such powers. This idea is in particular realized in powerJava, in which the powers of players and role-instances are formally restricted by both the Java compiler as well as by the way that roles are represented in powerJava. Clearly objects can call the public methods of other objects, and thereby, possibly, change some of the attributes of an object. Roles add one extra dimension to that: linking a role to a player within an institution may give to the role instance access to methods that can change features of the institution over and above those that we given by the original model. In other words, role instances have powers over the institution within which the role is played.

2.4 Sessions

We explicitly introduce the concept of session because we argue that is strictly linked with the role's notion. As already said, we talk about sessions when is possible to keep the state of an interaction between entities. Sessions in our model are a set of objects Depending on what we want to model, we can look at sessions from at least three different points of view:

1. A session can collapse into one role instance.
2. A session can collapse into the actor (as we will see in Section 3.3).
3. A session can have its own identity and can link different role instances.

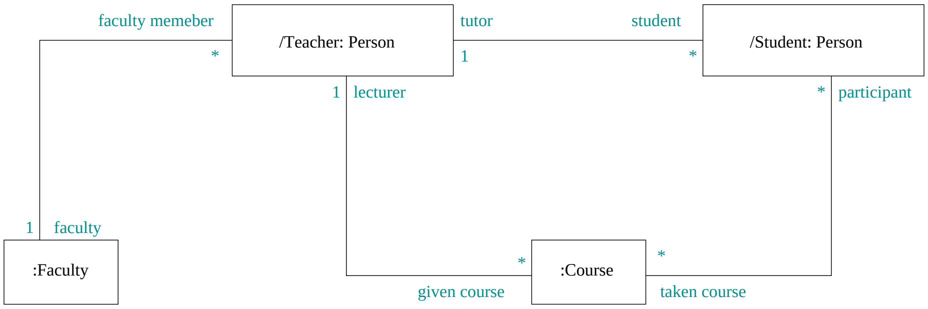
In powerJava the state of the interaction between a player and an institution is kept by the role instance:

$$R(i, a, s, x) \to s = x$$

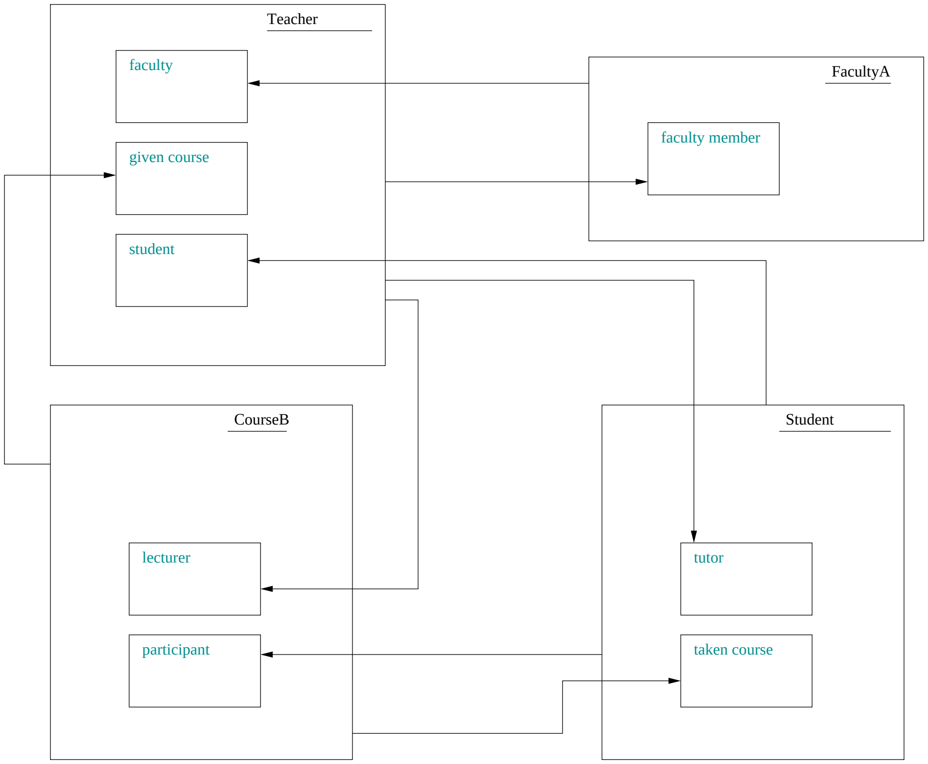
In such a case we represent the role instance as R(i,a,x) because the session collapses into the role instance. The second approach is taken into account in Section 3.3 where we try to model RBAC.

The third definition gives the possibility to unify the state of the interaction between different roles instances which participate in the same relationship or which are part of the same organizational model.

We start with an example to show how roles can be linked in a relationship:



In UML, roles serve two purposes: they label association ends, and they act as type specifiers in the scope of a collaborations (so-called classifier roles) [5]. We can translate this UML diagram into our model modelling the labels of association ends as roles and the classifier role as the object which plays the role instances to engage the relationship :



Where, for example, Teacher, which instantiate the class /Teacher:Person, is an Institution because it offers roles, but is also an actor because it plays roles which belong to other external institution.

Quoting Steimann [5] we can state that, for example:"...the role of a teacher unifies the roles of a faculty member, a lecturer and a tutor...". The problem here is that the three role are scattered between different objects (because they label associations ends) and this make difficult to link one role instance with another played by the same actor. For instance, suppose that *faculty member* and *tutor* have an attribute num_courses which counts the number of courses held by the *Teacher*, if *Teacher* stops playing *lecturer* in CourseB, num_courses in both *faculty member* and *tutor* should be decreased by one. There could also be a case where an action carried out as *tutor* can modify a *lecturer's* attribute.

In general, when the state of a role instances x does not depend only on the player and the institution, but also on other roles y and z, we say that x, y and z share the same session s, in other words:

$$R(i, a, s, x) \wedge R(i, a, s', y) \wedge R(i, a, s'', z) \rightarrow s = s' = s''$$

Giving a session s is possible to define a set of integrity rules that each role instance, participating in the session, have to respect. Referring to the above example, we can state that all role instances linked with Teacher have to share the same session:

$$\forall x, y, z\ \exists! s : faculty\_member(FacultyA, Teacher, s, x) \wedge$$

$$tutor(Student, Teacher, s, y) \wedge$$

$$lecturer(CourseB, Teacher, s, z)$$

Then we can define the following integrity rule associate with s:

$$\forall z, p, q\ \ :$$

$$p :: Faculty\ \wedge\ q :: /Student : Person\ \wedge z :: /Teacher : Person \wedge$$

$$faculty\_member(p, z, s, x) \wedge$$

$$tutor(q, z, s, y)$$

$$\rightarrow$$

$$\mathsf{num\_courses(x)} = \mathsf{num\_courses(y)} = \alpha$$

Where α is the number of $\mathsf{Teacher}$ instances played by z.

3 Different role's accounts

3.1 Social roles

This model is able to describe portion of the Social role's properties identified by Masolo *et al.* [4]

The key features of social roles

1. **Roles are 'properties'**: Quoting the referred article: "… *different* entities can *play* the *same* role". In order to link this sentence with our model we need to specify at which level we are reasoning, the sentence should be interpreted as [1]: "Different player *Universal* can play the same role *Universal*". In our model this is represented as:

$$\{Human, Frog\} \subseteq \mathsf{Players}$$

$$Fantasy\_Village \in \mathsf{Contexts}$$

$$Prince \in \mathsf{Roles}$$

$$\mathsf{RO} = (< Prince, Fantasy\_Village >)$$

$$\mathsf{PL} = (< Human, prince >, < Frog, Prince >)$$

2. **Roles are anti-rigid and they have 'dynamic' properties**:

[1] for an analysis at the Individual level see *"A role can be played by different entities, simultaneously"*

- *An entity can change role*: At individual level an actor can delete the role instance R(i,a,o), and play another role in place of it.
- *An entity can play the same role several times, simultaneously*: In our model an actor a who plays a role r in an institution i can have assigned only one *role instance*. However it is not clear what does it means to play the same role several times simultaneously, Masolo *et al.* conjecture that an actor can play simultaneously two different specific roles which are all specializations of a more general one. This point can be modelled in our formalism using role hierarchies, with sessions is also possible have a player which plays two role instances of the same role class R simultaneously.
- *A role can be played by different entities, simultaneously*: This sentence can be considered from two different angles: "Two player individual (Mario,Tom) can play the same role (employee) in an institution (bank)" in such a case we have two role instances $employee(bank, Mario, x)$ and $employee(bank, Tom, y)$ where $x \neq y$.
- *A role can be played by different entities, at different times*: The same role instance cannot be played by different entities, but we can have two different times $t^{'} \leq t$ in which:

$$(R_t(i, a, o) \wedge \neg R_t(i, b, c)) = true$$

$$(\neg R_{t'}(i, a, o) \wedge R_{t'}(i, b, c)) = true$$

3. **Roles have a relational nature**: "In other words we define the term role as a *founded* concept. In general, we say that α is founded on a property β if, necessarily, any *definition* of α ineliminably involves β, which is external to α". In our model, the role class R is definitionally dependent on another entity C if RO relation has a couple $< R, C >$ where C is a context. If we want to represent that *all* roles are founded on a context:

$$R \in \mathsf{Roles} \leftrightarrow \exists! C \in \mathsf{Contexts} : < R, C > \in \mathsf{RO}$$

4. **Roles are linked to contexts**: As already said above, the same happens in our model.

The key-properties for an entity to be a *role* are *anti-rigidity* and *foundation*. Foundation, as already mentioned, is an intrinsic property of our role model (think about *role instance*), the same holds for anti-rigidity, hence an object a playing a role R maintains its identity even after the role instance $R(i, a, o)$ ceases to exists. In other words we can represent the following *integrity rule*:

$$AR(R) = \forall a, o, t(R_t(i, a, o) \rightarrow \exists t^{'}(?_{t'}(a) \wedge \neg R_{t'}(i, a, o)))$$

Where AR stands for anti-rigidity.

3.2 Steimann's approach

In *object-oriented* and *conceptual modelling*, the representation of roles needs to take into account various modelling issues: multiple and dynamic classification, multiple inheritance, objects changing their attributes and behaviours, etc. Steimann introduces roles as 'first-class citizens' underlining the weaknesses which arise from others modelling approaches. To formalise his approach he defines a role-oriented modelling language called LODWICK [5].

In LODWICK roles are a kind of relationship's placeholder and playing a role for an actor means to *fill* that role in a relationship (i.e., to join the relationship taking the place held by the role filled). We already showed in Section 2.2.1 how we can simulate the idea of roles as placeholders in relationships, thanks to the fact that a role is strictly linked with a context and a player.

Here we would like to analyse how Steimann evaluates the adequacy of Lodwick's role concept, and then show how his approach could be modelled in our logical role's account. To do this several role's features are taken from different works in literature by Steimann and then discussed from the LODWICK point of view, it is interesting to notice that our model is able to describe all of them, even when they are in contradiction. We list all the features and to quote the replies that Steimann gives comparing LODWICK with them.

1. *A role comes with its own properties and behaviour*: "Yes. Roles are types, only that they cannot be instantiated. However, since the absolute properties of a role are inherited to the types filling them, they influence the properties of the instances playing them."
   This sentence can be translated in the following way:

   $$R(i, a, x) \rightarrow a = x$$

   $$R(i, a, x) \rightarrow (\forall \mathsf{attr} \in \pi_{Attr}(R) \rightarrow \exists v : \mathsf{attr}(\mathsf{a}) = v)$$

   Where the first predicate state that roles have no identity, and the second one express the fact that the properties of R influence a. In our formalism is also possible to model the case where roles are types but they can be instantiated:

   $$R(i, a, o) \rightarrow a \neq o$$

   in that case a interacts through o with i, and the property of the role instance are [2]:

   $$R(i, a, x) \rightarrow (\mathsf{attr}(x) = v \leftrightarrow (\mathsf{attr}(R) \vee \mathsf{attr}(a) = v))$$

2. *Roles depend on relationships:* "Yes. Roles occupy the places of relationships, and the relative part of a role's intension captures which relationships an object must participate in to be considered playing the role."
   Also in our model roles can be strictly linked with relationships, the fact that playing a certain role causes the player to be engaged in a relationship is implicit in our account, because the role is a link between two entities which

[2] The same holds for Op.

let the actor interact with the institution. Informally, we can say that the role instance implicitly defines a one way association ($\mathsf{actor} \rightarrow \mathsf{institution}$). It is also possible to model a situation where playing a role means to engage in a two way relationship, for example in the following situation:

$$Man, Woman \subseteq \mathsf{Players}$$

$$Man, Woman \subseteq \mathsf{Contexts}$$

$$husband, wife \in \mathsf{Roles}$$

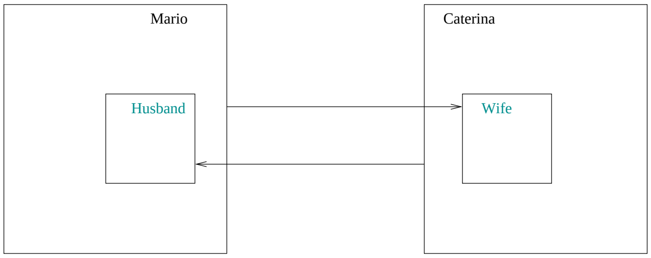
$$RO = (< husband, Woman >, < wife, Man >)$$

$$PL = (< Man, husband >, < Woman, wife >)$$

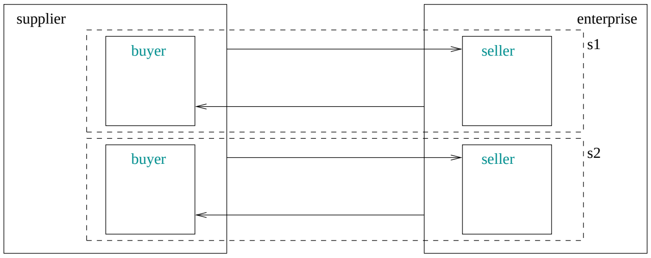
It would be sensible to impose that if *Mario::Man* plays the role husband, also *Caterina::Woman* plays the role wife with *Mario*, in other words:

$$husband(Caterina, Mario, x) \leftrightarrow wife(Mario, Caterina, y)$$

This relationship could be depicted in the following way:



Where we can see that *Mario* interacts with *Caterina* through the role instance *Husband* and complementary, *Caterina* interacts with *Mario* being his *Wife*. Another way to force the engagement in a two way relationship is through the context coherence, as already mentioned in Section 2.2.
With sessions we can explicitly link two role instances, in this way is also possible to model the following representation:



where a customer sells products to an enterprise, in one interaction the enterprise buys products for the IT department in *s1*, in the other for the HR division in *s2*. The customer has different accounts with the two departments, with the HR it sells discounted products, with the IT it sells at standard price. It is fundamental to notice that buyer in *s1* and buyer in *s2* are both instances of a common role class *Buyer*, the same happens between *seller* and a role class *Seller*. Thanks to sessions *s1* and *s2*, each one linking two role instances, it is possible to model this complex interaction.

3. *An object may play different roles simultaneously*: "Yes. An object may occur in as many different roles within the same or different associations as allowed by the relationships' specifications."
   In our model this situation could be easily expressed in the following:
   $$R(i,a,x) \wedge R'(j,a,y) \wedge x \neq y \wedge i \neq j$$
4. *An object may play the same role several time, simultaneously:* "Yes. An object may occur in the same role within different associations of the same or different relationships, as allowed by the relationship specifications."
   In our model the same role can be played several time in different institution so that:
   $$R(i,a,x) \wedge R(j,a,y) \wedge i \neq j$$
5. *An object may acquire and abandon roles dynamically*: " Yes. Roles are assumed by an object as associations with that object are added, and relinquised as associations are removed from the dynamic extensions of relationships."
   This is the same as in our model, for a complete discussion we refer to Section 2.3 where we define the role deletion.
6. *The sequence in which roles may be acquired and relinquished can be subject to restrictions*: "Possible. The specification of sequences lies in the responsability of the dynamic model."
   This is quite a subtle subject, but we can handle it exploiting the Requirements set. Suppose the we are in a Office and that the actor Leonard wants to become Director, one requirement could be that, in order to become Director you first need to be an employee, in our model suppose that $\pi_{Req}(t, Director)$ contains the following logical constraint:
   $$[director, bank \hookrightarrow leonard]\, \exists x\, director(bank, leonard, x)$$
   $$\rightarrow$$
   $$employee_t(bank, leonard, o)$$
7. *Objects of unrelated type can play the same role*: "Yes. This is one of the cornerstones of Lodwick's role formalizations; it follows from the definition of the role-filler relations linking the type and the role hierarchy."
   This point can also be easly expressed through the PL relation where we can put different universals in relation with the same role.
8. *Roles can play roles*: "No. This is not possible, since roles have no instances of their own."
   Albeit in our model we can express such a possibility, we can let Players $\cap$ Roles $= \emptyset$ in order to be consistent with Lodwick model.

9. *A role can be transferred from one object to another*: “Possible. This however would require the introduction of variables, which would be an extension to Lodwick.”

   Our model has its roots in roles’ foundation, in fact a (instance of) role must always be associated with an instance of the institution it belongs to, besides being associated with an instance of its player. So it is impossible to transfer a role from one object to another, what we can do is to let a different role instance x played by actor a in session s have the same state of another one z played by b in the same session, such as the state of x is copied into z , this could be interpreted as a dummy role transfer.

10. *The state of an object can be role-specific*: “Partly. The associations an object participates in contribute to its state. These associations can be extended to capture the state that is associated with the object as playing the role. For example, the different salaries of a person in different employee roles may be included in the *employ* relationship.”

    Our approach can model two substantially different situations, in the case that roles instances have not their own identity it is clear that the state of the actor is directly changed by the fact of playing a role R, because it acquires new operations and attributes. On the other side a role instance can come with its own identity, in this approach we can say that the state of the object in the interaction with other entities, is also composed by all the role instances it plays simultaneously (all roles instance share the same session). From this point of view, also in this case the state of an object can be role specific.

11. *Features of an object can be role-specific*: “Possible. Role are types and as such come with their own features. Role features are inherited to the types filling the roles, but a role-sensitive resolution mechanism (qualification) is needed if the same features are inherited from more than one role.”

    As we already said, is it possible to model that if an actor plays a role a it acquires attr or/and op of the role instance played.

12. *Roles restrict access*: “Not applicable. Lodwick does not have notions of accessibility or visibility.”

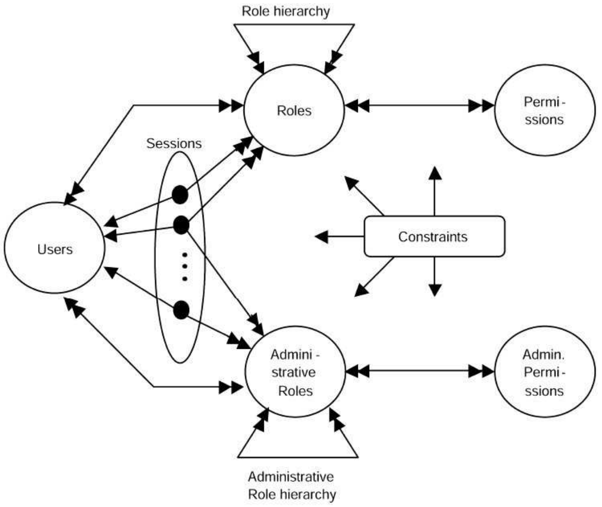
    If the role instance has its own identity it restrict access because it gives certain powers to the player playing it. These powers let the player access the private state of the Institution to which the role instance is linked. If we constraint the interaction with an object only through the roles it offers, we can model the situation in which roles restrict access.

13. *Different roles may share structure an behaviour*: “Partly. As noted under item 11, the features of role specifications are inherited down the role hierarchy to the types filling the roles. Vice versa, properties of the types filling roles are not inherited to these roles. For instance, if the type *Person* has a *placeOfBirth* attribute, this attribute is not shared by its role *Customer*. This however makes sense since not all customers are persons.”

    Exploiting role hierarchies we can model inheritance of role’s specifications, and through sessions we can let the behaviour of a role instance influenced by other roles.

14. *An object and its roles share identity*: “Yes. An object in a role is the object itself.”
    In our logical formalism: $R(i, a, o) \to a = o$
15. *An object and its roles have different identities*: “No. This follows from item 14.”
    In our logical formalism: $R(i, a, o) \to a \neq o$

3.3 RBAC model



—sets: U, R, AR, P, AP, S for sets of users, (regular) roles, administrative roles, (regular) permissions, administrative permissions, and sessions, respectively.
—$UA \subseteq U \times A$: user-role assignments
$AUA \subseteq U \times AR$: user-administrative role assignments
—$PA \subseteq P \times R$: permission-role assignments
$APA \subseteq AP \times AR$: permission-administrative role assignments
—$RH \subseteq R \times R$: role hierarchy
—$ARH \subseteq AR \times AR$: administrative role hierarchy
—$user : S \to U$, a function mapping a session to a single user
—$roles : S \to 2^{R \cup AR}$, a function mapping a session to a set of roles:
$roles(s) \subseteq \{r : R \mid (\exists r' \geq r) \cdot [(user(s), r') \in UA \cup AUA]\}$
—$permissions : R \to 2^{P \cup AP}$, a function mapping a role to a set of permissions:
$permissions(r) = \{p : P \mid (\exists r'' \leq r) \cdot [(p, r'') \in PA \cup APA]\}$
—collection of constraints

Fig. 1. Summary of RBAC96 model.

There are a few element which needs a deeper analysis to fit them in our role account.

- *Absence of an explicit context:* RBAC is a model which let a highly decentralized security administration thanks to a subtle role account, the model doesn't cope with contexts. In order to fit it with our approach we say that there is one dummy context which contains all system's roles.

- *Permissions:* In RBAC permissions are assigned to roles [2], a permission is an approval of a particular mode of access to one or more objects in the system. The terms *authorization*, *access right*, and *privilege* are also used in the literature to denote a permission. Permissions are always positive and confer on their holder the ability to perform an action in the system. A user who plays a role acquires all the system's permissions linked with the role played. One issue is how to fit the notion of permission in our model. In the literature in order to be able to define RBAC in a general and formal way, permissions are treated as uninterpreted symbols because permissions are implementation and system dependent. In fact each system has its own way to describe a permission and different accounts could dramatically differ one from another; from a formal point of view we are much more interested on *where* permissions are and not what they are. In RBAC permissions are assigned to role, so to fit with our model we decide to let permission be attributes so that $\mathsf{permissions} \subset \mathsf{Attr}$.
- *Sessions:* Users establish *sessions* during which they may activate a subset of the roles they belong to. Each session maps one user to possibly many roles. The double-headed arrow from session to R in Figure 1 indicates that multiple roles are simultaneously activated. The permissions available to the user are the union of permissions from all roles activated in that session. Each session is associated with a single user, as indicated by the single-headed arrow from the session to U in Figure 1. This association remains constant for a session's duration. A user might have multiple sessions open simultaneously,for example each in a different window on a workstation screen. Hence, each session is linked with a user and is always different from all other sessions, so we can say that a session is an instance of the user, if user x enters the system an instance y of x $(y :: x)$ is created, and this instance (session) can, for example play roles (activate roles), there can exists many instances of x which are all linked with it but everyone is different from each other, in other words:

  $$R(i, a, s, o) \rightarrow s = a$$

  With this in mind we can state that the instantiation of a player individual x::y in our model corresponds to a session's activation. And the creation of the role instance R(i,x,o) correspond to the activation of the role R by the user y in the session x (where i is a dummy context). Playing a role gives to the user in that session all the permissions the are linked with R:

  $$R(i, a, s, x) \rightarrow (\exists v : \mathsf{attr}(x) = v \leftrightarrow (\mathsf{attr}(R) \vee \mathsf{attr}(a) = v))$$

- *Administrative authority:* One of the most interesting points of RBAC is the possibility to use RBAC to manage itself. For this purpose the model introduces *administrative roles* AR and *administrative permissions* AP, the intent is for AR and AP to be respectively disjoint from regular role R and permissions P. The model shows that permissions can be assigned only to roles and that administrative permissions can be assigned only to administrative roles; this is a built-in constraint. Usually, each administrative role is mapped to the subset of the role hierarchy it manages. With the introduction of AR and AP, in RBAC is defined a structured way to change what in our model is the *Universal Level*, in the literature there are many ways to administrate RBAC but each one could be easily merged with our model simply introducing an *administrative* meta-level which discriminate who and how can change the universal level.

4 Conclusions

In this draft we try to give a general, and relatively simple, formalism through which we grasp different notions of role. The idea is to constrain the model to meet others approaches in order to cover a wider literature on the subject. The more we can describe with this framework, the more we are sure that terms like *player, session, context* and *role instance* are pivotal elements which can give a possible reply to the challenging research question: what are roles?.

References

1. Loebe, F.: Abstract vs. social roles - a refined top-level ontological analysis. In Procs. of AAAI Fall Symposium on Roles, an interdisciplinary perspective Series Technical Reports FS-05-08. pages 93-100 (2005)
2. S.Sandhu, R., J.Coyne, E.: Role-Based Access Control Models. IEEE Computer, Volume 29, Number 2 38-47 (1996)
3. van der Torre, L., Boella, G.: Oranizations as socially constructed agents in the agent oriented paradigm. In Procs. of ESAW'04 (2004)
4. Masolo, C., Vieu, L., Bottazzi, E., Catenacci, C., Ferrario, R., Gangemi, A., Guarino, N.: Social roles and their descriptions. In Procs. KR2004, Whistler, Canada, June 2-5, 2004, pp.267-277 (2004)
5. Steimann, F.: On the representation of roles in object-oriented and conceptual modelling. Data and Knowledge Engineering, 35:83-848 (2000)
6. Baldoni, M., van der Torre, L., Boella, G.: Interaction between objects in powerJava. Journal of Object Technology(JOT) (2006)
7. R., D., R., H., M.Kuster: Roles in agent-oriented modelling. International Journal of Software Engineering and Knowledge Engineering (2001) 281–302
8. Boella, G., van der Torre, L.: The ontological properties of social roles in multiagent systems: Definitional dependence, powers and roles playing roles. Artificial Intelligence and Law Journal (AILaw) (2007)
9. Wieringa, R., de Jonge, W., Spruit, P.: Roles and dynamic subclasses: a model logic approach. In Procs. of 8th European Conference on Object-Oriented Programming (1994)
10. Wieringa, R., de Jonge, W.: The identification of objects and roles. (1991)
11. Herrmann, S.: Programming with Roles in ObjectTeams/Java. In proc. AAAI Fall Symposium 2005: Roles, an iterdisciplinary perspective (2005)
12. Oh, S., Sandhu, R., Zhang, X.: An effective role administration model using organization structure. ACM Transactions on Information and System Security Vol.9,No.2,May 2006, Pages 113-137 (2006)

13. Masolo, C., Guizzardi, G., Vieu, L., Bottazzi, E., Ferrerio, R.: Relational roles and qua-individuals. In Procs. of AAAI Fall Symposium on Roles, an interdisciplinary perspective (2005)
14. Baldoni, M., Boella, G., van der Torre, L.: Modelling the interaction between objects: Roles as affordances. In Procs. Knowledhe Science, Engineering and Management, First International Conference, KSEM 2006, Guilin, China, August 5-8, volume 4092 of Lecture Notes in Computer Science, pages 42-54. Springer, 2006 (2006)

What an Agent Ought To Do
A Review of John Horty's 'Agency and Deontic Logic'

Jan Broersen[1] and Leendert van der Torre[2]

[1] Department of Information and Computing Sciences, Utrecht University
[2] University of Luxembourgh

1 Introduction

John Horty's book 'Agency and deontic logic' appeared at Oxford Press in 2001. It develops deontic logic against the background of a theory of agency in non-deterministic time. Several philosophical reviews of the book appeared since then [1–5]. Our goal is to present the book to a general AI audience that is familiar with action theories developed in AI, classical decision theory [6], or formalizations of temporal reasoning like Computation Tree Logic (CTL) [7, 8]. Therefore, in contrast to the philosophical reviews, we discuss and explain several key examples in the book. We do not explicitly discuss the relevance for AI and law, because the book itself is not concerned with the application of the theory to the legal domain. However, the relevance of deontic logic and normative reasoning for legal reasoning is well established by a number of publications on deontic logic in AI and law, see for example the special issue of this journal on agents and norms (volume 4, 1999).

Horty presents a formal account of what individuals and groups of agents ought to do under various conditions and over extended periods of time, using the 'Seeing To It That' or STIT framework [9]. He explicitly develops a utilitarian / consequentialist perspective, which means roughly that an act is obligatory if performing it results in an optimal state of affairs. However, the question whether a state of affairs is 'optimal' is not a question that is exclusively linked to the deontic point of view. And also, seeing deontic necessity only from the perspective of an agent's welfare (optimality), might not suffice to model all subtleties involved in the semantics of deontic notions. Therefore, it is easy to be confused by the examples; sometimes it is not immediately clear why they are especially relevant for deontic logic.

Horty focusses on the common assumption that what an agent ought to do can be identified with the notion of what it ought to be that the agent does, and argues that this assumption is wrong. The assumption is based on the well-known conceptual distinction in deontic logic that concerns the notions of *ought-to-be* and *ought-to-do*. Roughly, ought-to-be deontic statements express a norm about the satisfaction of certain *conditions* at certain moments. Ought-to-do deontic statements apply to *actions*, which have been argued to fall in a different ontological category than conditions [10, 11]. The distinction between ought-to-do and ought-to-be forms the starting point for Horty's journey.

Dagstuhl Seminar Proceedings 07122
Normative Multi-agent Systems
http://drops.dagstuhl.de/opus/volltexte/2007/905

The central problem addressed in the first three chapters is the question whether ought-to-do deontic statements can be formalized within a STIT-framework that is extended with a Standard Deontic Logic or SDL-style ought-operator [12]. In particular, it is investigated whether it is intuitive to model 'agent α ought to do A' as 'it ought to be that agent α sees to it that A' in the STIT-framework. Horty argues that the answer is negative, and proposes, in chapter 4, a deontic operator that does formalize ought-to-do statements within the STIT-framework. In the remaining three chapters of his book Horty generalizes this theory to the conditional case, the group case, and the strategic case. The emphasis throughout the book is conceptual rather than technical, and as such the book is more aimed at offering food for thought for developers of deontic logic than at providing deontic logics which can directly be used in applications. Questions concerning deontic logic and the logic of agency are considered in parallel with issues from moral philosophy and the philosophy of action. This allows for a number of recent issues from moral philosophy to be set out clearly and discussed from a uniform point of view.

The review consists of two parts. In the first part we relate STIT-theory to standard decision trees, and explain concepts and ideas by selecting and discussing examples that are central to Horty's work. The critical discussion in the second part concerns three aspects: the examples, the concepts modelled by Horty's logic, and logical and technical issues.

2 Examples

At first sight, Horty's examples may seem innocent and their formalization straightforward. However, a more detailed analysis reveals that each example highlights a basic choice, which also is bound to appear in more detailed and realistic examples. The examples thus play the same role as simple examples in reasoning about action and change, like the widely discussed Yale shooting problem and stolen car problem that illustrate the frame problem [13, 14].

2.1 From decision trees to STIT models

Horty uses STIT-models to discuss a variety of examples and concepts. However, STIT-theory is not well-known outside the area of philosophical logic. Therefore we first explain STIT-theory by relating STIT-models to standard decision trees, which are well known in artificial intelligence. Roughly, each STIT-model contains a classical decision tree with decision nodes that abstracts from the probabilities. Horty describes the relation with decision theory as follows: 'The new analysis is based on a loose parallel between action in non-deterministic time and choice under uncertainty, as it is studied in decision theory' [15, p.4].

Decision trees are widely used to formalize decision problems. The branches of decision trees represent courses of time. Nodes without branches are called 'terminal nodes', to distinguish them from other nodes where time may advance and branch. Branching is either due to a choice made by the decision-making agent, or due to the occurrence of events, where each possible event is associated with a probability. Branching nodes reflecting points of choice for the agent are called ‘decision nodes’. Branching nodes that correspond to moments where events occur are called ‘event nodes’. It is assumed that the decision-making agent knows what the probabilities are for each branch of an event node. The sum of the probabilities for the possible events at an event node is 1. Paths from the root to a terminal node correspond to sequences of choices and events in time. With each path a utility is associated, a kind of payoff under uncertainty. Rationality is defined as choosing an alternative that has the highest expected utility.

Of this decision theoretic setting, Horty adopts the utilities associated with series of events and choices in a decision tree. Paths are called ‘histories’. Roughly, STIT-models can be constructed from decision trees by dropping the event nodes, and consequently, the probabilities. The branches of a dropped event node are connected to the first decision node closer to the root in the tree, to form a non-deterministic action. Figure 1 visualizes such a transformation by showing a decision tree and the utilitarian STIT-model it reduces to. The decision tree should be read as follows. Boxes represent decision nodes and circles represent event nodes. Numbers at the end of paths through the tree represent utilities, and events are provided with probabilities. Expected utilities are represented at each node in italics.

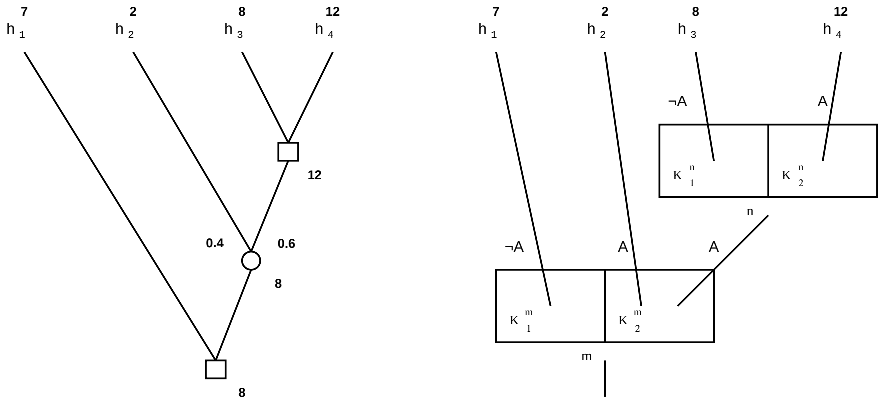


Fig 1. A decision tree and the corresponding utilitarian STIT-model

The two decision nodes of the decision tree correspond to the moments m and n in the STIT-models. Histories are series of events and choices from the root to the leaves of a STIT-model ($h_1 \dots h_4$ in figure 1). The event node in the decision tree on the left has turned into a non-deterministic action K_2^m, in the STIT-model on the right. Note that in the decision tree the choices in decision nodes are always deterministic. This is because in the decision tree, the non-determinism introduced by the events is ‘temporally separated’ from the decision nodes. Another distinction is that decision trees have no valuations of atomic propositions. In STIT-models there is a valuation of atomic propositions for every moment-history pair. Horty denotes the set of histories through a moment m for which the atomic proposition A holds by $|A|_m$, and the utility of a history h by $Value(h)$.

Now we explain the semantics of some key concepts in Horty's STIT-formalism. A noticeable feature of the semantics is that formulas are evaluated with respect to moment-history pairs, and not with respect to moments, which is the viewpoint adopted in many temporal formalisms used in AI (e.g., CTL [7, 8]). This refinement of the unit of evaluation is induced by the basic assumption of the STIT-framework that actions constrain the possible future courses of time without actually 'taking' time. This means that we need to partition the histories of each moment according to the set of actions possible at it.

Some basic temporal formulas of Horty's utilitarian STIT-formalism are A and FA for 'the atomic proposition A' and 'some time in future A'. In particular, A is settled true at a moment-history pair m, h if and only if it is assigned the value 'true' in the STIT-model; FA is settled true at a moment-history pair m, h if and only if there is some future moment on the history, where A is settled true. On the STIT-model of figure 1 we have $\mathcal{M}, m, h_3 \models A$, which follows directly from the valuation of atomic propositions on moment-history pairs, and $\mathcal{M}, m, h_3 \models F\neg A$, which is due to the fact that the proposition $\neg A$ is true later on, at moment n, on the history h_3 through m.

An action formula is $[\alpha\ cstit : A]$, 'agent α Sees To It That A'. The 'c' in 'cstit' stands for 'Chellas', whose version of the STIT-operator [16] is predominant in Horty's work. $[\alpha\ cstit : A]$ is settled true at a moment history pair m, h if and only if A is settled true at all moment-history pairs through m that belong to the same *action* as the pair m, h, i.e., if $h \in K$ at m then $K \subseteq |A|_m$. Following Horty, we use a symbol like K both as a name of an action at a moment m and as a denotation for the set of 'admissible' histories determined by that action at moment m. In figure 1, we have $\mathcal{M}, m, h_3 \models [\alpha\ cstit : A]$, because A holds for all histories through m that belong to the action to which also h_3 belongs, that is, action K_2^m.

Finally a deontic formula $\bigcirc A$ stands for 'it ought to be that A'. $\bigcirc A$ is settled true at a moment history pair m, h if and only if there is some history h' through m such that A is settled true at all pairs m, h'' for which the history h'' has a utility at least as high as h', i.e., $\exists(m, h')$ such that $\forall(m, h'')$ for which $Value(h') \leq Value(h'')$ it holds that $h'' \in |A|_m$. In figure 1 we have $\mathcal{M}, m, h_3 \models \bigcirc A$ and $\mathcal{M}, m, h_3 \models \bigcirc[\alpha\ cstit : A]$. These two meta-propositions are true for the same reason: the history h_4 through m has the highest utility and satisfies both A and $[\alpha\ cstit : A]$ at m.

Note that this condition guarantees that on separate histories through a moment any ought formula evaluates to the same value, which is why ought-formulas are called 'moment determinate'. This semantic condition for the to-be ought is a utilitarian generalization of the standard deontic logic view (SDL [12]) that 'it ought to be that A' means that A holds in all deontically optimal worlds. Satisfaction of a formula A by a STIT-model can be defined as truth of A in all moment-history pairs of the model, and validity as satisfaction by all STIT-models. Horty does not give these definitions explicitly, but this is the general STIT-view on validity (see, e.g., [9]).

2.2 'Ought-to-do' and the gambling problem

The central thesis of the book is that ought-to-do statements cannot be formalized as ought-to-be statements about action. More precisely, Horty claims that 'agent α ought to see to it that A' cannot be modeled by the formula $\bigcirc[\alpha\ cstit : A]$, whose reading is 'it ought to be that agent α sees to it that A'. Justification of this claim is found in what Horty calls the 'gambling problem' [15, p.53-58]. This example concerns the situation where an agent faces the choice between gambling to double or lose five dollar (action K_1^m) and refraining from gambling (action K_2^m). This STIT model is visualized in figure 2.

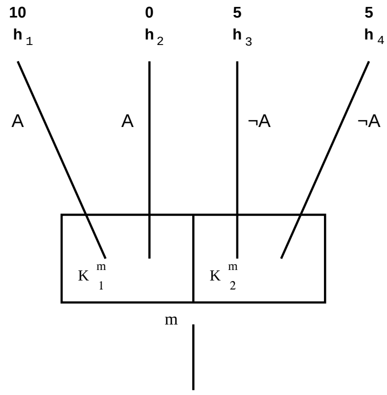


Fig 2. The gambling problem [15, Fig 3.8]

The two histories that are possible by choosing action K_1^m represent ending up with ten dollar by gaining five, and ending up with nothing by loosing all, respectively. Also for action K_2^m, the game event causes histories to branch. For this action the two branches have the same utility, because the agent is not taking part in the game, thereby preserving his five dollar. Note this points to redundancy in the model representation: the two branches are logically indistinguishable, because there is no formula whose truth value would change by dropping one of them.

The formula $\bigcirc[\alpha\ cstit : A]$ is settled true at m, because the formula $[\alpha\ cstit : A]$ is settled true for history h_1 and for all histories with a higher utility (of which there are none!). However, a reading of $\bigcirc[\alpha\ cstit : A]$ as 'agent α ought to perform action K_1^m' is counter-intuitive for this example. From the description of the gambling scenario it does not follow that one action is better than the other. In particular, without knowing the probabilities, we cannot say anything in favor of action K_1^m: by choosing it, we may either end up with more or with less money then by doing K_2^m. The only thing one may observe is that action

K_1^m will be preferred by more adventurous agents. But that is not something the logic is concerned with.

This demonstrates that 'agent α ought to see to it that A' cannot be modelled by $\bigcirc[\alpha\ cstit : A]$. The cause of the mismatch can be explained as follows. Adapting and generalizing the main idea behind SDL to the STIT-context, ought-to-be statements concern truth in a set of optimal histories. Optimality is directly determined by the utilities associated with the individual histories. If ought-to-be is about optimal histories, then ought-to-do is about optimal actions. But, since actions are assumed to be non-deterministic, actions do not correspond with individual histories, but with *sets* of histories. This means that to apply the idea of optimality to the definition of ought-to-do operators, we have to generalize the notion of optimality such that it applies to *sets* of histories, namely, the sets that make up the non-deterministic actions. More specifically, we have to obtain an ordering of non-deterministic actions that is based on the underlying ordering of histories. The ordering of actions suggested by Horty is very simple: an action is strictly better than another action if all of its histories are at least as good as any history of the other action, and not the other way around.

Having 'lifted' the ranking of histories to a ranking of actions, the utilitarian ought conditions can now be applied to actions. Thus, Horty defines the new operator 'agent α ought to see to it that A', written as $\bigodot[\alpha\ cstit : A]$, as the condition that for all actions not resulting in A there is a higher ranked action that does result in A, together with the condition that all actions that are ranked even higher also result in A. This 'solves' the gambling problem. We do not have $\bigodot[\alpha\ cstit : A]$ or $\bigodot[\alpha\ cstit : \neg A]$ in the gambling scenario, because in the ordering of actions, K_1^m is not any better or worse than K_2^m. So, it is not the case that the agent ought to gamble, nor is it the case that the agent ought to refrain from gambling.

2.3 The driving example

Horty generalizes the ordering on actions to the multi-agent context by imposing the so-called 'sure-thing principle' [6]. If there are only two agents, then at m for agent α action K_1^m is better than action K_2^m if for *each* action K_3^m by agent β it holds that $K_1^m \cap K_3^m$ is better than $K_2^m \cap K_3^m$. Here, an intersection like $K_1^m \cap K_3^m$ stands for a group action where agent α and agent β simultaneously perform K_1^m and K_3^m, respectively. The actions optimal for an agent α at a moment m are denoted $Optimal_\alpha^m$. The corresponding generalized operator $\bigodot[\alpha\ cstit : A]$ reflects what Horty calls 'dominance act utilitarianism'. The driving example [15, p.119-121] is used to illustrate the difference between dominance act utilitarianism and an orthodox perspective on the agent's ought. Dominance act utilitarianism says that α ought to see to it that A just in case the truth of A is guaranteed by each of the optimal actions available to the agent – formally, that $\bigodot[\alpha\ cstit : A]$ should be settled true at a moment m just in case $K^m \subseteq |A|_m$ for each $K^m \in Optimal_\alpha^m$. When we adopt the orthodox perspective, the truth or falsity of ought statements can vary from index to index. The orthodox perspective is that α should see to it that A at a certain index just in case the truth of A

is guaranteed by the available actions that are optimal given the circumstances in which he finds himself at this index. Horty uses the symbol $\bigoplus$ to denote the orthodox ought operator.

According to Horty, the driver example is due to Holly Goldman [17], and it is also discussed by Humberstone in [18], a paper that sets out in a different context some of the fundamental ideas underlying the orthodox ought defined by Horty.

> "In this example, two drivers are travelling toward each other on a one-lane road, with no time to stop or communicate, and with a single moment at which each must choose, independently, either to swerve or to continue along the road. There is only one direction in which the drivers might swerve, and so a collision can be avoided only of the drivers swerves and the other does not; if neither swerves, or both do, a collision occurs. This example is depicted in Figure 3, where α and β represent the two drivers, K_1^m and K_2^m represent the actions available to α of swerving or staying on the road, K_3^m and K_4^m likewise represent the swerving or continuing actions available to β, and m represents the moment at which α and β must make their choice. The histories h_1 and h_3 are the ideal outcomes, resulting when one driver swerves and the other one does not; collision is avoided. The histories h_2 and h_4, resulting either when both drivers swerve or both continue along the road, represent non-ideal outcomes; collision occurs. The statement A, true at h_1 and h_2, expresses the proposition that α swerves." [15, p.119]

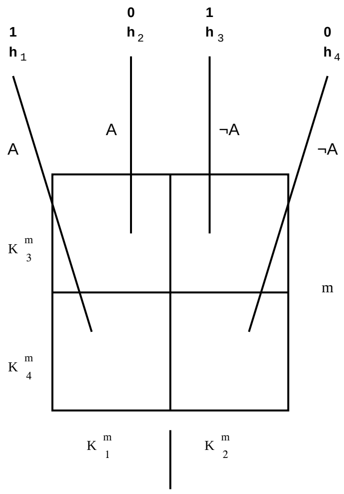


Fig 3. The driving example [15, Fig 5.6]

From the dominance point of view both actions available to α are classified as optimal, i.e. $Optimal_\alpha^m = \{K_1^m, K_2^m\}$, because the sure-thing principle does not favor one of the actions over the other. Thus, one of the optimal actions available to α guarantees the truth of A and the other guarantees the truth of $\neg A$. Consequently $M, m \not\models \bigodot[\alpha\ cstit : A]$ and $M, m \not\models \bigodot[\alpha\ cstit : \neg A]$. But from the orthodox point of view, we have for example $M, m, h_1 \models \bigoplus[\alpha\ cstit : A]$ and $M, m, h_2 \models \bigoplus[\alpha\ cstit : \neg A]$, because A and $\neg A$ hold for all optimal actions given that agent α does K_1^m or K_2^m, respectively. So, α ought to do A or $\neg A$, depending at the index.

Horty also discusses the so-called Whiff and Poof example, an example with the same logical structure, introduced for example in [19–21]. In this example, there are two agents in the moral universe, who can each push a button or not. If both push the button the overall utility is 10, if neither push their button the utility is 6, and otherwise 0. Both the driver example and the Whiff and Poof example are instances of classical coordination games studied in game theory.

Horty concludes that from the standpoint of intuitive adequacy, the contrast between the orthodox and dominance deontic operators provides us with another perspective on the issue of moral luck, the role of external factors in our moral evaluations [15, p.121]. The orthodox ought can suitably be applied when an agent looks back in time and considers what he ought to have done in a certain situation. For example, when there has been a collision then α might say – perhaps while recovering from the hospital bed – that he ought to have swerved. The dominance ought is looking forward. Though the agent may legitimately regret his choice, it is not one for which he can be blamed, since either choice, at the time, could have led to a collision. The distinction corresponds to what has been called the diagnostic and the decision-theoretic perspective in [22], and can be related to Thomason's distinction between evaluative and judgmental oughts [23].

2.4 Procrastinate's choice

The example of Procrastinate's choices [15, p. 162] illustrates the notion of strategic oughts. A strategy is a generalized action involving a series of actions. Like an action, a strategy determines a subset of histories. The set of admissible histories for a strategy σ is denoted $Adh(\sigma)$. If a strategy σ is not more than a single action K^m at moment m, i.e. $\sigma = \{\langle m, K^m \rangle\}$, Horty simply writes K (assuming m is clear from the context) for $Adh(\{\langle m, K^m \rangle\})$.

A crucial new concept here is the concept of a *field*, which is basically a subtree of the STIT-model which denotes that the agent's reasoning is limited to this range. A strategic ought is defined analogous to dominance act utilitarianism, by replacing actions by strategies in a field. α ought to see to it that A just in case the truth of A is guaranteed by each of the optimal strategies available to the agent in the field – formally, that $\bigodot[\alpha\ cstit : A]$ should be settled true at a moment m just in case $Adh(\sigma) \subseteq |A|_m$ for each $\sigma \in Optimal_\alpha^m$. Horty observes some complications, and says that a 'proper treatment of these issues might well push us beyond the borders of the current representational formalism' [15, p.150].

Horty also uses the example of Procrastinate's choices to distinguish between actualism and possibilism, for which he uses the strategic oughts, and in particular the notion of a field. Roughly, actualism is the view that an agent's current actions are to be evaluated against the background of the actions he is actually going to perform in the future. Possibilism is the view that an agent's current actions are to be evaluated against the background of the actions that he might perform in the future; the available future actions. The example is due to Jackson and Pargetter [24].

> "Professor Procrastinate receives an invitation to review a book. He is the best person to do the review, has the time, and so on. The best thing that can happen is that he says yes, and then writes the review when the book arrives. However, suppose it is further the case that were he to say yes, he would not in fact get around to writing the review. Not because of incapacity or outside interference or anything like that, but because he would keep on putting the task off. (This has been known to happen.) Thus although the best thing that can happen is for Procrastinate to say yes and then write, and he *can* do exactly this, what *would* happen in fact were he to say yes is that he would not write the review. Moreover, we may suppose, this latter is the worst thing which may happen.
> [...]
> According to possibilism, the fact that Procrastinate would not write the review were he to say yes is irrelevant. What matters is simply what is possible for Procrastinate. He can say yes and then write; that is best; that requires *inter alia* that he says yes; therefore, he ought to say yes. According to actualism, the fact that Procrastinate would not actually write the review were he to say yes is crucial. It means that to say yes would be in fact to realize the worst. Therefore, Procrastinate ought to say no." [24, p.235]

Horty represents the example by the STIT-model in Figure 4. Here, m_1 is the moment at which Procrastinate, represented as the agent α, chooses whether or not to accept the invitation: K_1 represents the choice of accepting, K_2 the choice of declining. If Procrastinate accepts the invitation, he then faces at m_2 the later choice of writing the review or not: K_3 represents the choice of writing the review, K_4 another choice that results in the review not being written. For convenience, Horty also supposes that at m_3 Procrastinate has a similar choice whether or not to write the review: K_5 represents the choice of writing, K_6 the choice of not writing. The history h_1, in which Procrastinate accepts the invitation and then writes the review, carries the greatest value of 10; the history h_2, in which Procrastinate accepts the invitation and then neglects the task, the least value of 0; the history h_4, in which he declines, such that a less competent authority reviews the book, carries an intermediate value of 5; and the peculiar h_3, in which he declines the invitation but then reviews the book anyway, carries a slightly lower value of 4, since it wastes his time, apart from doing no one else any good. The statement A represents the proposition that he accepts the invitation; the statement B represents the proposition that Procrastinate will write the review.

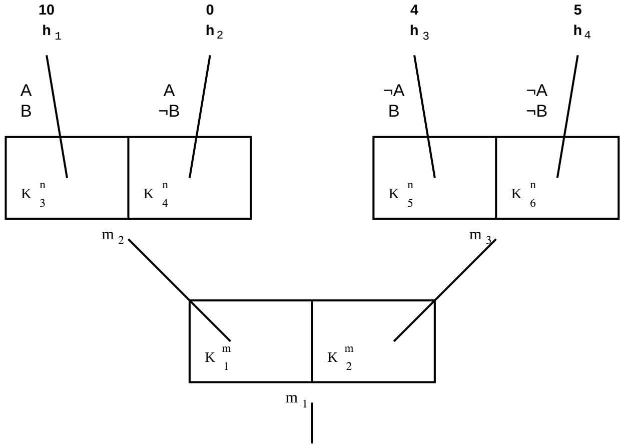


Fig 4. Procrastinate's choices [15, Fig 7.6]

Now, in the possibilist interpretation, $M = \{m_1, m_2, m_3\}$ is the background field. In this interpretation, Procrastinate ought to accept the invitation because this is the action determined by the best available strategy – first accepting the invitation, and then writing the review. Formally, $Optimal^M_\alpha = \{\sigma_6\}$ with $\sigma_6 = \{\langle m_1, K_1\rangle, \langle m_2, K_3\rangle\}$. And since $Adh(\sigma_6) \subseteq |A|_m$, the strategic ought statement $\bigodot[\alpha\ cstit : A]$ is settled true in the field M. In the actualist interpretation, the background field may be narrowed to the set $M' = \{m_1\}$, which shifts from the strategic to the momentary theory of oughts. but In this case, we have $\bigodot[\alpha\ cstit : A]$ is settled false. It is as if we choose to view Procrastinate as gambling on his own later choice in deciding whether to accept the invitation. However, from this perspective, this should not be viewed as a gamble; an important background assumption – and the reason that he should decline the invitation – is that he will not, in fact, write the review.

3 Discussion

3.1 The examples

The examples in Horty's book are meant to provoke discussion. In this section we raise some issues ourselves.

According to Horty [15, p.57], the gambling example "seems to reflect a real difficulty with the strategy of identifying even a purely utilitarian notion of what an agent ought to do with the notion of what it ought to be that the agent does – at least on the basis of any theory conforming to the standard deontic idea that whatever holds in all the best outcomes is what ought to be. Any such theory would have the result that, in this situation, it ought to be that the agent gambles; after all, gambling is a necessary condition for achieving the best outcome, the outcome with the greatest utility." This observation is the basis of the formal distinction between a logic for ought-to-be and a logic for ought-to-do. However, the quote also indicates a way in which the two may be identified anyway, namely by leaving the idea that whatever holds in all the best outcomes is what ought to be! This idea of so-called standard deontic logic, a modal logic proposed by von Wright in 1951, has been criticized during the last five decades by many authors for various reasons, and many alternative deontic logics have been proposed. Horty does not discuss the question whether his example is also a problem for these logics. For example, in preference-based deontic logics [25] an obligation for p is formalized by a preference of p over $\neg p$, i.e. $O(p) = p \succ \neg p$. For most preference orderings we do not seem to have the result that, in this situation, it ought to be that the agent gambles. This suggests that the gambling problem may not occur in such settings.

Moreover, Danielsson [3] observes that situations with the same structure as the gambling problem also appear in examples with no actions involved. He discusses an example in which a window may be open or not, and the wind may bring something good or bad if the window is open. Finally, McNamara [**?**] observes that the gambling problem is closely related to an example discussed by Feldman [26]. Feldman imagines that it will rain tomorrow, and that given that it rains it is best for the reporter to predict that fact. However, there is no good reason now to think it will rain rather than that it won't. It is in fact indeterminate without probabilities. So although it is ideal for the reporter to report that it will rain, it does not folow that he should do so.

The driver example is, as Horty observes, a classical coordination game as studied in classical game theory. This raises the question whether the techniques used in game theory are relevant for the analysis of this example. For example, what is the role of Nash equilibria is the analysis of the example? Moreover, is Horty's philosophical study relevant for game theory, and if so, why?

Procrastinate's choices also raises questions. For example, is the notion of a field related to the notion of bounded or limited reasoning as studied in, amongst others, artificial intelligence? Moreover, Horty does not discuss that the notion of strategic ought can be applied to the most famous of all deontic examples, Chisholm's contrary-to-duty paradox, as has been suggested by van der Torre and Tan [27]. Horty observes [15, p.40] that STIT-models can deal with reperational oughts (contrary-to-duty oughts), but he does not discuss the paradox. If in Figure 4 we read A as 'the man tells his neighbors that he will come' and B as 'the man goes to the assistance of his neighbors', and the utility of history h_2 is raised to for example 8, then the STIT-model seems to reflect a variant of Chisholm's paradox:

1. A certain man is obliged to go to his neighbour's assistance;
2. If he goes, he should tell them he will come;
3. If he does not go, then he should tell them that he does not come;
4. He does not go.

3.2 The relation with other motivational concepts

As we mentioned in the introduction, Horty explicitly develops a utilitarian perspective, which means roughly that an act is right or obligatory if it is a best promoter of (social) welfare. Danielsson [3] emphasizes that it is also a consequentialist perspective, which means that an act is right or obligatory if it is a best act for achieving a highest ranked state of affairs. Danielsson also observes that Horty apparently sees no need to discuss possible important differences between rules of rational behaviour, moral rules, and rules of semantics, which makes the whole project somewhat unclear.

The decision-theoretic setting used by Horty to define obligations has also been used to define goals in knowledge-based systems in artificial intelligence, and to define desires in belief-desire-intention (BDI) systems in agent theory, see e.g. [28]. In such settings, the basic distinction between obligations on the one hand and goals and desires on the other hand is that the former are external motivations, whereas the latter are internal motivations of the agent.

Now, some of Horty's examples can also be interpreted in terms of goals and desires. For example, in another example [15, p.49] an agent is discussed who wishes to buy a horse which costs \$15,000 whereas the agent only has \$10,000. The problem in this example is whether the agent should bid \$10,000 for the horse or not. In this example, it seems that we might as well say that the agent desires to buy the horse for \$10,000. Horty mentions that his "characterization of values, or utilities, as abstract, and intended to accommodate a variety of different approaches. It says nothing about what is ultimately taken as a measure of the individual agent's utility – pleasure, mental states of intrinsic worth, happiness, money, an ndex of basic goods" [15, p.38]. These measures seem related to goals and desires.

Horty acknowledges this problem, when he observes that his notion of ought is completely utilitarian, whereas our intuitive idea that an agent α ought to see to it that A often seems to be sensitive to non-utilitarian considerations. Our conception of what we ought to do is often influenced, not only by the utility of the outcomes that might result from our actions, but also by considerations involving a number of additional concepts, such as rights or personal integrity. If Smith makes a promise to Jones, for example, Jones has a right, a claim against Smith, that Smith should keep the promise, even if the outcome that would result from Smith's keeping the promise carries less utility than the outcome that would result if the promise were broken.

Horty's answer to such objections is pragmatic. Such objections, he says, are perhaps too broad to be illuminating. The objection is directed not so much against the analysis itself as against the utilitarian framework within which the analysis is developed. Rather than attempting to model our ordinary, common sense notion of what an agent ought to do, governed as it is by a variety of considerations, he instead restricts his attention only to those oughts generated by considerations of utility. His goal, then is to model only a more limited, utilitarian notion of what an agent ought to do, a notion of what the agent ought to do on the basis of utilitarian considerations alone [15, p.54].

3.3 Logical and technical issues

Since Horty's book is about *logic*, one may expect that the logical repercussions of the semantic definitions in the book are studied in depth. However, the book mentions most logical considerations only briefly.

For instance, it is mentioned that the logic of the composed operator $\bigcirc[\alpha\ \mathit{cstit} : A]$ is similar to the logic of $\odot[\alpha\ \mathit{cstit} : A]$. Horty [15, p.79]: 'Although perhaps already apparent, it is worth noting explicitly that the notion carried by the new operator of what an agent ought to do is logically neither weaker nor stronger than the notion of what it ought to be that the agent does, but incomparable.' Horty demonstrates this incomparability in various ways, since it is directly related to his central thesis about the irreducibility of the ought-to-do operator. But in our opinion, the other part of the claim, i.e., that the first operator is neither weaker or stronger than the second, requires a proof. It is not enough just to observe and prove that the operators both satisfy some properties that are typical for normal modal logics.

The second issue we raise in this section concerns the 'intuitiveness' of the orderings used for actions. This concerns Horty's choice for the definition of an ordering of actions in terms of the ordering of the underlying sets of histories. We argue that this 'lifting' of the ordering of histories to an ordering of actions can also be defined intuitively in another way.

Notice first that in Horty's formalism, the utilities associated with the histories are relevant in as far as they determine relative strengths. So, the absolute values of the utilities have no meaning. In particular the following two models are indistinguishable for Horty's logic:

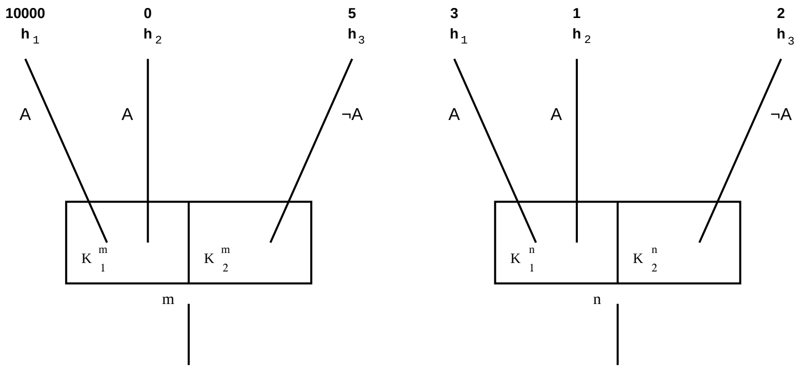


Fig 5. Two models that cannot be distinguished in Horty's logic.

The value of the numbers is only used to decide whether a history is better or worse than another history, which means that any linear order will do. We emphasize this point, because when being presented such example models one is inclined to attach meaning to the absolute values. In particular, when one is used to work in a classical decision theoretic setting, one could easily reason that the high value in the left model will inevitably influence decisions, culminating in some formulas being evaluated differently. But, for Horty's theory the two choice situations are identical.

The above observation is important for our discussion on the lifting of the ordering of histories to an ordering of actions. Consider the two choice situations sketched in figure 6.

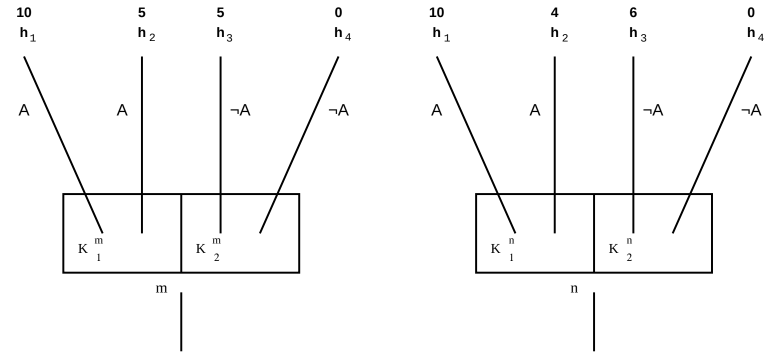


Fig 6. Two more choice situations

In the situation on the left, Horty's ordering on actions gives that action K_1^m is better than action K_2^m, resulting in satisfaction of $\bigodot[\alpha\ cstit : A]$ at m, i.e., the agent ought to perform K_1^m. This is intuitive, since any possible outcome of performing K_1^m is at least as good as any outcome of K_2^m. But in the choice situation n on the right, Horty's ordering gives no decision: there is a possible outcome of K_1^n, namely history h_2, for which there is an outcome of K_2^n, namely h_3, that is better. So, it is not the case that the agent ought to do K_1^n, nor is it the case that he ought to refrain from K_1^n (i.e. do K_2^n). However, we think that in the utilitarian setting put forward by Horty, it is very well possible to defend that action K_1^n is actually *better* than action K_2^n. Let us analyze the information contained in the model. As argued above, we should not attach any meaning to the absolute values of the utilities. Then, all the information that is available is that the highest utility can be reached by doing K_1^n and the lowest by doing K_2^n, and what's more, the highest utility *cannot* be reached by doing K_2^n, and the lowest cannot be reached by doing K_1^n. If an agent is presented with such a choice, he should choose K_1^n, for two good reasons:

1. it is the *only* choice that *might* result in the best possible history, and
2. it is the *only* choice by which he can be *sure* to avoid the worst possible history.

This line of reasoning cannot be countered by claiming that such arguments should account for probabilities concerning the occurrence of separate histories. As said, Horty simply does not consider a logic for situations where the probabilities are known; the logic is only about choices, non-determinism and utilities. It can also not be countered by claiming that there can be (causal) dependencies between the histories of separate actions. Such information is not represented in the models, meaning that we cannot account for it in the logic.

We do not suggest that the above two conditions are each individually sufficient for concluding that an action is better. But following the line of reasoning, we can define a more subtle way in which an ordering of actions is derived from an underlying ordering of histories. In [29] we show how to define such an ordering, and apply it to the semantics of deontic modalities in a dynamic logic setting. If we apply this ordering to the present STIT-theory, we get a weaker utilitarian ought-to-do-operator (weaker in the sense that it allows more models) that also solves the gambling problem of fig. 2.

4 Conclusion

John Horty's book 'Agency and deontic logic' develops deontic logic against the background of a theory of agency in non-deterministic time. Horty tells a self-contained story without loosing momentum by diving into the conceptual and technical details that are met along the way. He formulates precise and clear, and takes his time to put forward a wealth of concepts and ideas. The book itself is not concerned with the application of the theory to the legal domain, but the relevance of deontic logic and normative reasoning for legal reasoning is well established.

We presented the book to a general AI audience that is familiar with action theories developed in AI, classical decision theory, or formalizations of temporal reasoning. We discussed three representative examples: the gambling paradox, the driving example and Procrastinate's choice. The first illustrates the distinction between ought-to-do and ought-to-be, the second illustrates the distinction between dominance act utilitarianism and an orthodox perspective on the agent's ought. The third example illustrates the distinction between actualism and possibilism. The reader who is intrigued by one of the examples, or the distinctions they illustrate, should read Horty's book for the full story, and for other instructive examples and distinctions.

The book does not study the developed logics in any depth, and there are no axiomatizations. Moreover, Horty does not discuss why utilities should be used for obligations, in contrast to for example goals and desires. Finally, the relation between his logic and related work in for example logics of action in AI, classical decision theory, and temporal logic is not studied. This may be judged as an omission, but also as an opportunity.

In this review we indicated how classical decision trees can be related to STIT models, and we have given an alternative way to lift the ordering on histories to a dominance relation on actions. We believe that the book is a good starting point for other comparisons that relate philosophical logic to theories developed in AI. We strongly recommend anyone interested in the philosophical and logical aspects of reasoning about oughts, agency and action to get hold of a copy of this book.

Acknowledgements

Thanks to Paul McNamara for very useful comments on an earlier version of this review.

References

1. Bartha, P.: A review of john horty's 'agency and deontic logic'. Notre Dame Philosophical Reviews (2002.02.01) (2002) ndpr.icaap.org.
2. Czelakowski, J.: John f. horthy, agency and deontic logic. Erkenntnis **58**(1) (2003) 116–126
3. Danielsson, S.: A review of john horty's 'agency and deontic logic'. The Philosophical Quarterly (2002) 408–410
4. McNamara, P.: Agency and deontic logic by john horty. Mind **112**(448) (2003)
5. Wansing, H.: A review of john horty's 'agency and deontic logic'. Journal of Logic, Language and Information (2003) to appear.
6. Savage, L.: The Foundations of Statistics. John Wiley and Sons (1954)
7. Clarke, E., Emerson, E., Sistla, A.: Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Transactions on Programming Languages and Systems **8**(2) (1986)
8. Emerson, E.: Temporal and modal logic. In Leeuwen, J.v., ed.: Handbook of Theoretical Computer Science, volume B: Formal Models and Semantics. Elsevier Science (1990) 996–1072
9. Belnap, N., Perloff, M., Xu, M.: Facing the future. Oxford University Press (2001)
10. Castañeda, H.N.: The paradoxes of deontic logic: the simplest solution to all of them in one fell swoop. In Hilpinen, R., ed.: New Studies in Deontic Logic: Norms, Actions and the Foundations of Ethics. D. Reidel Publishing Company (1981) 37–85
11. Castañeda, H.N.: Aspectual actions and davidson's theory of events. In E. LePore, B.M., ed.: Actions and Events: Perspectives on the Pholosophy of Donald Davidson. Basil-Blackwell (1985) 294–310
12. Wright, G.v.: Deontic logic. Mind **60** (1951) 1–15
13. Hanks, S., Dermott, D.M.: Default reasoning, nonmonotonic logics, and the frame problem. In: Proceedings of the National Conference on Artificial Intelligence (AAAI86), Morgan Kaufmann Publishers (1986) 328–333
14. Kautz, H.: The logic of persistence. In: Proceedings of the National Conference on Artificial Intelligence (AAAI86), Morgan Kaufmann Publishers (1986) 401–405
15. Horty, J.: Agency and Deontic Logic. Oxford University Press (2001)
16. Chellas, B.: The Logical Form of Imperatives. PhD thesis, Philosophy Department, Stanford University (1969)
17. Goldman, H.: Dated rightness and moral imperfection. The Philosophical Review **85** (1976) 449–487
18. Humberstone, I.: The background of circumstances. Pacific Philosophic Quarterly **64** (1983) 19–34
19. Gibbard, A.: Rule-utilitarianism: merely an illusionary alternative? Australasian Journal of Philosophy **43** (1965) 211–220
20. Sobel, J.: Rule-utilitarianism. Australasian Journal of Philosophy **46** (1968) 146–165
21. Regan, D.: Utilitarianism and Co-operation. Clarendon Press (1980)

22. Torre, L.v.d., Tan, Y.: Diagnosis and decision making in normative reasoning. Artificial Intelligence and Law **7** (1999) 51–67
23. Thomason, R.: Deontic logic as founded on tense logic. In Hilpinen, R., ed.: New Studies in Deontic Logic. D. Reidel Publishing Company (1981) 165–176
24. Jackson, F., Pargetter, R.: Oughts, options and actualism. Philosophical Review **99** (1986) 233–255
25. Torre, L.v.d., Tan, Y.: Contrary-to-duty reasoning with preference-based dyadic obligations. Annals of Mathematics and Artificial Intelligence **27** (1999) 49–78
26. Feldman, F.: Doing the Best We Can. D. Reidel Publishing Company (1986)
27. Torre, L.v.d., Tan, Y.: The temporal analysis of chisholm's paradox. In: Proceedings of 15th National Conference on Artificial Intelligence (AAAI'98). (1998) 650–655
28. Lang, J., Torre, L.v.d., Weydert, E.: Utilitarian desires. Autonomous Agents and Multi-Agent Systems **5**(3) (2002) 329–363
29. Broersen, J., Dastani, M., Huang, Z., Torre, L.v.d.: Trust and commitment in dynamic logic. In Shafazand, H., Tjoa, A.M., eds.: Eurasia-ICT 2002: Information and Communication Technology. Volume 2510 of Lecture Notes in Computer Science., Springer (2002) 677–684

What is Input/Output Logic? Input/Output Logic, Constraints, Permissions*

David Makinson[1] and Leendert van der Torre[2]

[1] david.makinson@googlemail.com
[2] University of Luxembourg, Computer Science and Communications (CSC)
1359, Luxembourg, 6 rue Richard Coudenhove Kalergi, Luxembourg
leendert@vandertorre.com

Abstract. We explain the *raison d'être* and basic ideas of input/output logic, sketching the central elements with pointers to other publications for detailed developments. The motivation comes from the logic of norms. Unconstrained input/output operations are straightforward to define, with relatively simple behaviour, but ignore the subtleties of contrary-to-duty norms. To deal with these more sensitively, we constrain input/output operations by means of consistency conditions, expressed via the concept of an outfamily. They also provide a convenient platform for distinguishing and analysing several different kinds of permission.

Keywords. Deontic logic, input/output logic, constraints, permissions

1 Motivation

Input/output logic takes its origin in the study of conditional norms. These may express desired features of a situation, obligations under some legal, moral or practical code, goals, contingency plans, advice, etc. Typically they may be expressed in terms like: *In such-and-such a situation, so-and-so should be the case*, or *...should be brought about*, or *...should be worked towards*, or *...should be followed* – these locutions corresponding roughly to the kinds of norm mentioned.

To be more accurate, input/output logic has its source in a tension between the philosophy of norms and formal work of deontic logicians.

Philosophically, it is widely accepted that a distinction may be drawn between norms on the one hand, and declarative statements on the other. Declarative statements may bear truth-values, in other words are capable of being true or false; but norms are items of another kind. They may be respected (or not), and may also be assessed from the standpoint of other norms, for example when a legal norm is judged from a moral point of view (or vice versa). But it makes no sense to describe norms as true or as false.

However the formal work of deontic logicians often goes on as if such a distinction had never been heard of. The usual presentations of deontic logic, whether

* This paper extends [11] with Section 6 on permissions.

axiomatic or semantic, treat norms as if they could bear truth-values. In particular, the truth-functional connectives *and*, *or* and most spectacularly *not* are routinely applied to norms, forming compound norms out of elementary ones. Semantic constructions using possible worlds go further by offering rules to determine, in a model, the truth-value of a norm.

This anomaly was noticed more than half a century ago, by Dubislav [4] and Jørgensen [5], but little was done about it. Indeed, from the 1960s onwards, the semantic approach in terms of possible worlds deepened the gap. The first serious attempt by a logician to face the problem appears to be due to Stenius [15], followed by Alchourrón and Bulygin [2] for unconditional norms, then Alchourrón [1] and Makinson [7] for conditional ones. Input/output logic may be seen as an attempt to extract the essential mathematical structure behind these reconstructions of deontic logic.

Like every other approach to deontic logic, input/output logic must face the problem of accounting adequately for the behaviour of what are called 'contrary-to-duty' norms. The problem may be stated thus: given a set of norms to be applied, how should we determine which obligations are operative in a situation that already violates some among them? It appears that input/output logic provides a convenient platform for dealing with this problem by imposing consistency constraints on the generation of output.

We begin by outlining the central ideas and constructions of unconstrained input/output logic. These are quite straightforward, and provide the basic framework of the theory. We then sketch a strategy for constraining those operations so as to deal more sensitively with contrary-to-duty situations. Finally, we explain how the same operations may be deployed in the analysis of permission.

For further details, the reader is invited to refer to Makinson and van der Torre [8,9].

2 Unconstrained Input/Output Operations

We avoid assuming that conditional norms bear truth-values. They are not embedded in compound formulae using truth-functional connectives. To avoid all confusion, they are not even treated as formulae, but simply as ordered pairs (a, x) of purely boolean (or eventually first-order) formulae.

Technically, a normative code is seen as a set G of conditional norms, *i.e.*, a set of such ordered pairs (a, x). For each such pair, the body a is thought of as an *input*, representing some condition or situation, and the head x is thought of as an *output*, representing what the norm tells us to be desirable, obligatory or whatever in that situation. The task of logic is seen as a modest one. It is not to create or determine a distinguished set of norms, but rather to prepare information before it goes in as input to such a set G, to unpack output as it emerges and, if needed, coordinate the two in certain ways. A set G of conditional norms is thus seen as a transformation device, and the task of logic is to act as its 'secretarial assistant'.

The simplest kind of unconstrained input/output operation is depicted in Figure 1. A set A of propositions serves as explicit input, which is prepared by being expanded to its classical closure $Cn(A)$. This is then passed into the 'black box' or 'transformer' G, which delivers the corresponding immediate output

$$G(Cn(A)) = \{x \mid \text{ for some } a \in Cn(A), (a, x) \in G\}.$$

Finally, this is expanded by classical closure again into the full output $out_1(G, A) = Cn(G(Cn(A)))$. We call this *simple-minded output.*

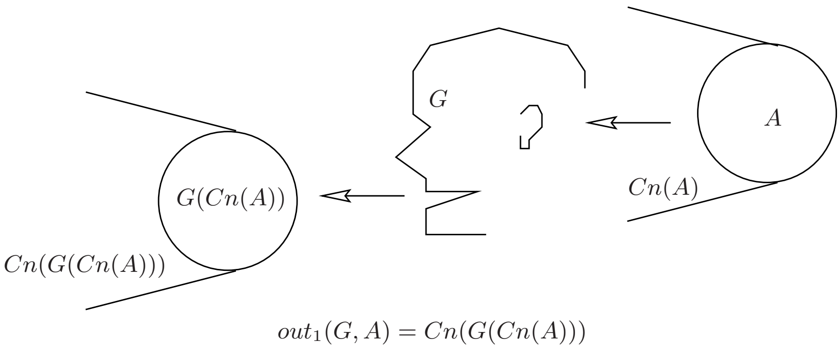


Fig. 1. Simple-minded Output

This is already an interesting operation. As desired, it does not satisfy the principle of identity, which in this context we call *throughput*, *i.e.*, in general we do not have $a \in out_1(G, \{a\})$ – which we write briefly, dropping the parentheses, as $out_1(G, a)$. It is characterized by three rules. Writing $x \in out_1(G, a)$ as $(a, x) \in out_1(G)$ and dropping the right hand side as G is held constant, these rules are:

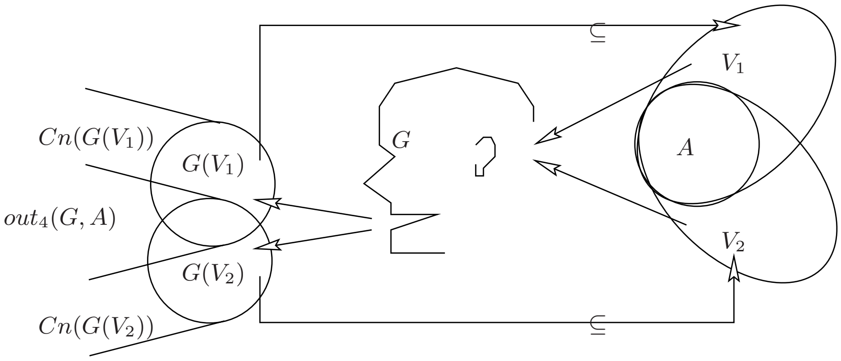
Strengthening Input (SI): From (a, x) to (b, x) whenever $a \in Cn(b)$
Conjoining Output (AND): From (a, x), (a, y) to $(a, x \wedge y)$
Weakening Output (WO): From (a, x) to (a, y) whenever $y \in Cn(x)$

But simple-minded output lacks certain features that may be desirable in some contexts. In the first place, the preparation of inputs is not very sophisticated. Consider two inputs a and b. By classical logic, if $x \in Cn(a)$ and $x \in Cn(b)$ then $x \in Cn(a \vee b)$. But there is nothing to tell us that if $x \in out_1(G, a) = Cn(G(Cn(a)))$ and $x \in out_1(G, b) = Cn(G(Cn(b)))$ then $x \in out_1(G, a \vee b) = Cn(G(Cn(a \vee b)))$.

In the second place, even when we do not want inputs to be automatically carried through as outputs, we may still want outputs to be reusable as inputs – which is quite a different matter.

Operations satisfying each of these two features can be provided with explicit definitions, pictured by diagrams in the same spirit as that for simple-minded output, and characterized by straightforward rules. We thus have four very natural systems of input/output, which are labelled as follows: *simple-minded* alias out_1 (as above), *basic* (simple-minded plus input disjunction: out_2), *reusable* (simple-minded plus reusability: out_3), and *reusable basic* (all together: out_4).

For example, reusable basic output may be given a diagram and definition as in Figure 2. In the definition, a complete set is one that is either maximally consistent or equal to the set of all formulae.



$$out_4(G, A) = \cap\{Cn(G(V)) \mid A \subseteq V \supseteq G(V), V \text{ complete}\}$$

Fig. 2. Basic Reusable Output

The three stronger systems may also be characterized by adding one or both of the following rules to those for simple-minded output:

Disjoining input (OR):	From (a, x), (b, x) to $(a \vee b, x)$
Cumulative transitivity (CT):	From (a, x), $(a \wedge x, y)$ to (a, y)

These four operations have four counterparts that also allow *throughput*. Intuitively, this amounts to requiring $A \subseteq G(A)$. In terms of the definitions, it is to require that G is expanded to contain the diagonal, *i.e.*, all pairs (a, a). Diagrammatically it is to add arrows from G's ear to mouth. Derivationally, it is to allow arbitrary pairs of the form (a, a) to appear as leaves of a derivation; this is called the zero-premise identity rule ID.

All eight systems are distinct, with one exception: basic throughput, which we write as out_2^+, authorizes reusability, so that $out_2^+ = out_4^+$. This may be shown directly in terms of the definitions, or using the following simple derivation of CT from the other rules.

$$\dfrac{\dfrac{\dfrac{\dfrac{(a,x)}{(a\wedge\neg x,x)}\,\text{SI}\quad \dfrac{-}{(a\wedge\neg x,a\wedge\neg x)}\,\text{ID}}{(a\wedge\neg x,x\wedge(a\wedge\neg x))}\,\text{AND}}{(a\wedge\neg x,y)}\,\text{WO}\qquad\qquad (a\wedge x,y)}{(a,y)}\,\text{OR}$$

The application of WO here is justified by the fact that we have $y \in Cn(x \wedge (a \wedge \neg x))$ since the right hand formula is a contradiction. Note that all rules available in basic throughput (including, in particular, identity) are needed in the derivation, reflecting the fact that CT is not derivable in the weaker systems.

This strong system indeed collapses into classical consequence, in the sense that $out_4^+(G,A) = Cn(m(G) \cup A)$ where $m(G)$ is the materialization of G, *i.e.*, the set of all formulae $a \rightarrow x$ where $(a,x) \in G$.

The authors' papers [8] and [9, section 1] investigate these systems in detail – semantically, in terms of their explicit definitions, derivationally, in terms of the rules determining them, both separately and in relation to each other. We do not attempt to summarize the results here, but hope that the reader is tempted to follow further.

3 Why constrain?

As mentioned in section 1, all approaches to deontic logic must face the problem of dealing with contrary-to-duty norms. In general terms, we recall, the problem is: given a set of norms, how should we determine which obligations are operative in a situation that already violates some among them.

The following simple example is adapted from Prakken and Sergot [13].[1] Suppose we have the following two norms: *The cottage should not have a fence or a dog; if it has a dog it must have both a fence and a warning sign.*

In the usual deontic notation: $O(\neg(f \vee d)/t)$, $O(f \wedge w/d)$, where t stands for a tautology; in the notation of input/output logic: $(t, \neg(f \vee d))$, $(d, f \wedge w)$. Suppose further that we are in the situation that the cottage has a dog, thus violating the first norm. What are our current obligations?

Unrestricted input/output logic gives f: *the cottage has a fence* and w: *the cottage has a warning sign*. Less convincingly, because unhelpful if the presence of a dog is regarded as unalterable, it also gives $\neg d$: *the cottage does not have a*

[1] There are many examples in the literature. Most of them involve ingredients that, while perfectly natural in ordinary discourse, are extraneous to the essential problem and thus invite false analyses. These ingredients include defeasibility, causality, the passage of time, and the use of questionable rules such as CT and OR in deriving output. We have chosen a very simple example that avoids all those elements. There is one respect in which it could perhaps be further purified: under input d, the output is not only inconsistent with the input, but also itself inconsistent. This matter is discussed at the end of section 5.

dog. Even less convincingly, it gives $\neg f$: *the cottage does not have a fence*, which is the opposite of what we want.

These results hold even for simple-minded output, without reusability or disjunction of inputs. The only rules needed are SI and WO, as shown by the following derivation of $\neg f$.

$$\dfrac{\dfrac{(t, \neg(f \vee d))}{(t, \neg f)}\ \text{WO}}{(d, \neg f)}\ \text{SI}$$

A common reaction to examples such as these is to ask: why not just drop the rule SI of strengthening the input? In semantic terms, why not cut back the definition of simple-minded output from $Cn(G(Cn(A)))$ to $Cn(G(A))$, and in similar (but more complex) fashion with the others? Indeed, this is a possible option, and the strategy that we will describe below does have the effect of disallowing certain applications of SI. But simply to drop SI is, in the view of the authors, too heavy-handed. We need to know *why* SI is not always appropriate and, especially, *when* it remains justified.

4 A Strategy for Constraint: Maxfamilies and their Outfamilies

Our strategy is to adapt a technique that is well known in the logic of belief change – cut back the set of norms to just below the threshold of making the current situation contrary-to-duty. In effect, we carry out a contraction on the set G of given norms.

Specifically, we look at the maximal subsets $G' \subseteq G$ such that$out(G', A)$ is consistent with input A. In [8], the family of such G' is called the *maxfamily* of (G, A), and the family of outputs $out(G', A)$ for G' in the maxfamily, is called the *outfamily* of (G, A).[2]

To illustrate this, consider $G = \{(t, \neg(f \vee d)), (d, f \wedge w)\}$, with the contrary-to-duty input d. Using simple-minded output, *maxfamily*(G, d) has just one element $\{(d, f \wedge w)\}$, and so *outfamily*(G, d) has one element, namely $Cn(f \wedge w)$.

[2] So defined, the outfamily is not in general the same as the family of all maximal values of $out(G', A)$ consistent with A, for G' ranging over subsets of G. Every maximal value of $out(G', A)$ is in the outfamily, but not always conversely. For certain of our output operations, the two families do coincide, but not for others.

This can be shown by simple examples, such as the Möbius strip of Makinson [6,7]. Put $G = \{(a, x), (x, y), (y, \neg a)\}$. Then, for $out = out_3$ or $out = out_4$, *maxfamily*(G, a) has three elements, namely the three two-element subsets of G. As a result, *outfamily*(G, a) also has three elements – $Cn(\emptyset)$, $Cn(x)$, and $Cn(\{x, y\})$. Of these, only the last is a maximal value of $out(G', A)$ consistent with A for G' ranging over subsets of G.

We add that in this example, not even $Cn(\{x, y\})$ is a maximal subset of $out(G, a)$ that is consistent with a, for clearly $Cn(\{x, y\}) \subset Cn(\{x, y, \neg a \vee z\}) \subset out(G, a)$. Care is thus needed to avoid confusing maxfamilies with related maximal sets.

Although the outfamily strategy is designed to deal with contrary-to-duty norms, its application turns out to be closely related to belief revision and nonmonotonic reasoning when the underlying input/output operation authorizes throughput.

When all elements of G are of the form (t, x), then for the degenerate input/output operation $out_2^+(G, a) = out_4^+(G, a) = Cn(m(G) \cup \{a\})$, the elements of $outfamily(G, a)$ are just the maxichoice revisions of $m(G)$ by a, in the sense of Alchourrón, Gärdenfors and Makinson [3]. These coincide, in turn, with the extensions of the default system $(m(G), a, \emptyset)$ of Poole [12].

More surprisingly, there are close connections with the default logic of Reiter, falling a little short of identity. Read elements (a, x) of G as normal default rules $a; x/x$ in the sense of Reiter [14], and write $extfamily(G, A)$ for the set of extensions of (G, A). Then, for reusable simple-minded throughput out_3^+, it can be shown that $extfamily(G, A) \subseteq outfamily(G, A)$ and indeed that $extfamily(G, A)$ consists of precisely the maximal elements (under set inclusion) of $outfamily(G, A)$.

These results and related ones are proven in Makinson and van der Torre [9]. But in accord with the motivation from the logic of norms, the main focus in that paper is on input/output logics *without* throughput. Two kinds of question are investigated in detail there.

4.1 The search for truth-functional reductions of the consistency constraint

From the point of view of computation, it is convenient to make consistency checks as simple as possible, and executable using no more than already existing programs. For this reason, it is of interest to ask: under what conditions is the consistency of A with $out(G, A)$ reducible to the consistency of A with the materialization $m(G)$ of G, *i.e.*, with the set of all formulae $a \to x$ where $(a, x) \in G$?

It is easy to check that the latter consistency implies the former for all seven of our input/output operations. It turns out that we have equivalence for just two of them (reusable basic with and without identity).

On the level of derivations, the question can take a rather different form, with different answers. Given a derivation of (a, x) with leaves L, under what conditions is the consistency of a with $out(L, a)$ equivalent to its consistency with $m(L)$? Curiously, this holds for a wider selection of our input/output operations – in fact, for all of them except basic output. Even more surprisingly, for some of the operations (those without OR), the same reduction also holds with respect to the set $h(L)$ of heads x, and the set $f(L)$ of fulfilments $a \wedge x$, of elements (a, x) of L.

From this result on derivations, we can go back and sharpen the semantic one. When G is a *minimal* set with $x \in out(G, a)$ then, for each of our input/output operations other than basic output, a is consistent with $out(G, a)$ iff it is consistent with $m(G)$ – and for the operations without OR, with $h(G)$, $f(G)$.

4.2 More severe applications of the consistency check

From a practical point of view, whenever we constrain an operation to avoid excess production, the question arises: how cautious (timid) or brave (foolhardy) do we want to be? For input/output operations, this issue arises in different ways on the semantic and derivational levels. On the semantic level, once we have formed an outfamily we may ask: should we intersect, join, or choose from its elements to obtain a unique restrained output? On the level of derivations, it is natural to ask: do we want to apply the consistency check only at the root of a derivation, or at every step within it?

The policy of checking only at the root corresponds to the option, on the semantic level, of forming the join of the outfamily; while the stricter policy of checking at every step is an essentially derivational requirement. But whichever of the two we choose, it is of interest to know under what conditions they coincide. In other words, given a derivation of (a, x) with leaves L such that a is consistent with $out(L, a)$, under what conditions does it follow that for every node (b, y) in the derivation, b is consistent with $out(L, b)$? It turns out that for certain of the seven input/output operations (again, those without the OR rule) this result holds. For operations with OR but without the rule CT, a rather subtler result may be obtained.

One lesson of these rather intricate investigations is that the behaviour of the consistency constraint depends very much on the choice of input/output operation; in particular, the presence of the rule OR destroys some properties. Another lesson is that questions can take different forms, with different answers, on the semantic and derivational levels. Thirdly, a detour through derivations can sometimes sharpen semantic results.

5 Doubts and Queries

The investigation of constrained output is a much more complex matter than that of unconstrained output. It is also more open to doubts and queries. We put the main ones on the table.

5.1 Dependence on the formulation of G

The outfamily construction, at least in its present form, depends heavily on the formulation of the generating set G. To illustrate this, we go back to the cottage example of Prakken and Sergot [13] considered in sections 3 and 4. Here $G = \{(t, \neg(f \vee d)), (d, f \wedge w)\}$, and we consider the contrary-to-duty input d. As we have seen, using simple-minded output, $maxfamily(G, d)$ has unique element $\{(d, f \wedge w)\}$ and $outfamily(G, d)$ has unique element $Cn(f \wedge w)$. But if we split the first element of G into $(t, \neg f), (t, \neg d)$ then we get a different result. The maxfamily has two elements $\{(t, \neg f)\}$, $\{(d, f \wedge w)\}$ and the outfamily has two elements $Cn(\neg f)$ and $Cn(f \wedge w)$. Is this dependence on formulation of G a virtue, or a vice?

5.2 Are we cutting too deeply?

This problem is related to the first one. In some cases, the outfamily construction cuts deeply, perhaps too much. Consider again the cottage example, but this time with just one rule $(t, \neg(f \vee d))$ in G. Consider the same contrary-to-duty input d. Then the maxfamily has the empty set as its unique element, and so the outfamily has $Cn(\emptyset)$ as its unique element. Is this cutting too deeply? Shouldn't $Cn(\neg f)$ be retained?

5.3 Should we pre-process G?

If we wish to cut less deeply, then a possible procedure might be to 'pre-process' G. In the last example, when we decompose the sole element $(t, \neg(f \vee d))$ of G into $(t, \neg f)$, $(t, \neg d)$ then $Cn(\neg f)$ becomes the unique element of outfamily in the contrary-to-duty situation d. In general, for each element (a, x) of G, we could rewrite the head x in conjunctive normal form $x_1 \wedge \ldots \wedge x_n$, and then split (a, x) into $(a, x_1), \ldots, (a, x_n)$. This manoeuvre certainly meets the particular example. But is it appropriate for other examples of the same form with different content? And does it suffice for more complex examples? It looks suspiciously like hacking.

5.4 Avoid inconsistency with what?

On our definition, $maxfamily(G, A)$ is the family of maximal subsets $G' \subseteq G$ such that $out(G', A)$ is consistent with input A. It may be suggested that this is too radical – so long as $out(G, A)$ is consistent we should apply it without constraint.

To illustrate this, take another variation on the cottage example. Put $G = \{(t, \neg(f \vee d)), (d, w)\}$. The second norm no longer requires a fence when there is a dog, only a warning sign. Consider again the contrary-to-duty input d. Now $out(G, d) = Cn(\{(\neg f, \neg d, w\})$ which is inconsistent with the input d, but itself perfectly consistent. Should we cut it at all? Perhaps 'yes' if the input d is considered as unalterably true, but 'no' if it is presented as true but changeable.

6 Conditional Permission from an Input/output Perspective

In philosophical discussion of norms it is common to distinguish between two kinds of permission, negative and positive. Negative permission is easy to describe: something is permitted by a code iff it is not prohibited by that code, i.e. iff *nihil obstat*. In other words, taking prohibition in the usual way, something is negatively permitted by a code iff there is no obligation to the contrary.

Positive permission is more elusive. As a first approximation, one may say that something is positively permitted by a code iff the code explicitly presents it as such. But this leaves the central logical question unanswered. As well as the items that a code explicitly pronounces to be permitted, there are presumably others that in some sense follow from the explicit ones. The problem is to make it clear what kind of 'following' this is.

From the point of view of input/output logic, negative permission is straightforward to define: we simply put $(a, x) \in \mathit{negperm}(G)$ iff $(a, \neg x) \notin \mathit{out}(G)$, where *out* is any one of the four input/output operations that we have already discussed.

Because of its negative character, *negperm* fails the rule SI (strengthening the input). In other words, we don't have: $(a, x) \in \mathit{negperm}(G) \& a \in Cn(b) \Rightarrow (b, x) \in \mathit{negperm}(G)$. Indeed, it satisfies the opposite rule WI (weakening the input): $(a, x) \in \mathit{negperm}(G) \& b \in Cn(a) \Rightarrow (b, x) \in \mathit{negperm}(G)$. For if $(a, \neg x) \notin \mathit{out}(G)$ and $b \in Cn(a)$ then by SI for the underlying output operation, $(b, \neg x) \notin \mathit{out}(G)$ so $(b, x) \in \mathit{negperm}(G)$. This is a particular instance of a quite general pattern: whenever out satisfies a Horn rule (HR) then the corresponding *negperm* operation satisfies an 'inverse' Horn rule $(\mathrm{HR})^{-1}$.

How should we define positive permission for conditional norms? Let G, P be sets of ordered pairs of propositions, where G represents the explicitly given conditional obligations of a code and P its explicitly given conditional permissions. The operation of *forward positive permission* is defined by putting:

$(a, x) \in \mathit{forperm}(P, G)$ iff $(a, x) \in \mathit{out}(G \cup Q)$ for some singleton or empty $Q \subseteq P$

i.e. in the principal case that P is not itself empty,

$(a, x) \in \mathit{forperm}(P, G)$ iff $(a, x) \in \mathit{out}(G(c, z))$

for some pair $(c, z) \in P$. This tells us that (a, x) is permitted whenever there is some explicitly given permission (c, z) such that when we treat it as if it were an obligation, joining it with G and applying the output operation to the union, then we get (a, x). Permissions are thus treated like weak obligations, the only difference being that while the latter may be used jointly, the former may only be applied one by one.

On the other hand, the operation of *backward positive permission* is defined by setting:

$(a, x) \in \mathit{backperm}(P, G)$ iff $(c, \neg z) \notin \mathit{out}(G \cup \{(a, x)\})$ for some pair $(c, z) \in P$ with c consistent.

This tells us that (a, x) is permitted whenever, given the obligations already present in G, we can't forbid x under the condition a without thereby committing ourselves to forbid something that has been explicitly permitted. With this in mind, one could also speak of the operation as one of *prohibition immunity.*

What do these two notions mean in ordinary life? Forward permission answers to the needs of the citizen, who needs to know whether an action that he is entertaining is permitted in the current situation. It also corresponds to the needs of authorities assessing the action once it is performed. If there is some explicit permission that 'covers' the action in question, then it is itself implicitly permitted.

On the other hand, backward permission fits the needs of the legislator, who needs to anticipate the effect of adding a prohibition to an existing corpus of norms. If prohibiting x in condition a would commit us to forbid something that has been explicitly permitted, then adding the prohibition is inadmissible under pain of incoherence, and the pair (a, x) is to that extent protected from prohibition.

Forperm and *backperm* are very different operations. Whereas *forperm* satisfies SI, *backperm* satisfies WI. Like negative permission, *backperm* satisfies the 'inverse' rule $(\mathrm{HR})^{-1}$ of any Horn rule (HR) satisfied by out; but *forperm* satisfies instead a 'subverse' rule $(\mathrm{HR})^{\downarrow}$.

Backperm may be characterized in a rather different way, using an idea of Makinson, [7]. Let us say that G is cross-coherent with P iff there is no $(c, z) \in P$ with c consistent, such that $(c, \neg z) \in \mathit{out}(G)$. Then it is easy to check that $(a, x) \in \mathit{backperm}(P, G)$ iff $(a, x) \in \mathit{negperm}(H)$ for every $H \supset G$ that is cross-coherent with P. From this it follows, in particular, that when G is cross-coherent with P then $\mathit{backperm}(P, G) \subseteq \mathit{negperm}(G)$. In this sense, we can say that under 'normal conditions' backward permission is a strengthened negative permission.

Further details of the behaviour of these operations may be found in Makinson and van der Torre [10].

7 Conclusions

Drawing together the threads of this paper, we emphasize the main points.

- Input/output logic seeks to extract the essential mathematical structure behind recent attempts to reconstruct deontic logic that avoid treating norms as if they had truth-values.
- Unconstrained input/output provides us with a simple and elegant construction with straightforward behaviour, but whose application to norms totally ignores the subtleties of contrary-to-duty obligations.
- On the other hand, output constrained using the outfamily strategy provides a way of dealing with contrary-to-duty obligations. Its behaviour is quite subtle, and depends considerably on the choice of background input/output operation, in particular on whether or not it authorizes the rule of disjunction of inputs.
- However, our definition of an outfamily has features that might be regarded as shortcomings. Its effect depends on the formulation of the generating set of norms; in some examples it gives what may be regarded as a wrong result unless some pre-processing as carried out on the generating set; and in some contexts the requirement of consistency of output with input may be too strong. These are delicate issues, and it remains possible that they have no unique solution definable in purely formal terms.

– Input/output operations also enable us to give a clear formal articulation of the well-known distinction between negative and positive permission. They also enable us, for the first time, to distinguish two very different kinds of positive permission, with quite different uses in practical life.

A topic of further research is the analysis of structured assemblies of input/output operations. Such structures, called logical input/output nets, or lions for short, are graphs, with the nodes labelled by pairs (G, out) where G is a normative code and out is an input/output operations (or recursively, by other lions). The relation of the graph indicates which nodes have access to others, providing passage for the transmission of local outputs as local inputs. The graph is further equipped with an entry point and an exit point, for global input and output.

References

1. Alchourrón, C., "Philosophical foundations of deontic logic and the logic of defeasible conditionals", in: Meyer, J. and Wieringa, R. (eds.), *Deontic Logic in Computer Science*, New York: Wiley, 1993, 43–84.
2. Alchourrón, C. and Bulygin, E., "The expressive conception of norms", in: Hilpinen, R. (ed.), *New Essays in Deontic Logic*, Dordrecht: Reidel, 1981, 95–124.
3. Alchourrón, C., Grdenfors, P. and Makinson, D., "On the logic of theory change: partial meet contraction and revision functions", *The Journal of Symbolic Logic*, **50**, 1985, 510–530.
4. Dubislav, W., "Zur Unbegrndbarkeit der Forderungstze", *Theoria*, **3**, 1937, 330–342.
5. Jørgensen, J., "Imperatives and logic", *Erkenntnis*, **7**, 1937-8, 288–296.
6. Makinson, D., "General Patterns in Nonmonotonic Reasoning", in: Gabbay, H. and Robinson (eds.), *Handbook of Logic in Artificial Intelligence and Logic Programming*, vol. 3, Oxford University Press, 1994, 35–110.
7. Makinson, D., "On a fundamental problem of deontic logic", in: McNamara, P. and Prakken, H. (eds.), *Norms, Logics and Information Systems. New Studies in Deontic Logic and Computer Science*, vol. 49 of *Frontiers in Artificial Intelligence and Applications*, Amsterdam: IOS Press, 1999, 29–53.
8. Makinson, D. and van der Torre, L., "Input/output logics", *J. Philosophical Logic*, **29**, 2000, 383–408.
9. Makinson, D. and van der Torre, L., "Constraints for input/output logics", *J. Philosophical Logic*, **30(2)**, 2001, 155–185.
10. Makinson, D. and van der Torre, L., "Permission from an input/output perspective", *J. Philosophical Logic*, **32(4)**, 2003, 391–416.
11. Makinson, D. and van der Torre, L., "What is Input/Output Logic?", in: *Foundations of the Formal Sciences II: Applications of Mathematical Logic in Philosophy and Linguistics*, vol. 17 of *Trends in Logic*, Kluwer, 2003.
12. Poole, D., "A logical framework for default reasoning", *Artificial Intelligence*, **36**, 1988, 27–47.
13. Prakken, H. and Sergot, M., "Contrary-to-duty obligations", *Studia Logica*, **57**, 1996, 91–115.
14. Reiter, R., "A logic for default reasoning", *Artificial Intelligence*, **13**, 1980, 81–132.
15. Stenius, E., "Principles of a logic of normative systems", *Acta Philosophica Fennica*, **16**, 1963, 247–260.