Normative Framework for Normative System Change

Guido Boella
Dipartimento di Informatica
University of Turin
Italy
boella@di.unito.it

Gabriella Pigozzi
CSC
University of Luxembourg
Luxembourg
gabriella.pigozzi@uni.lu

Leendert van der Torre
CSC
University of Luxembourg
Luxembourg
leendert@vandertorre.com

ABSTRACT

Normative systems in a multiagent system must be able to evolve over time, for example due to actions creating or removing norms in the system. The only formal framework to evaluate and classify normative system change methods is the so-called AGM framework of theory change, which has originally been developed as a framework to describe and classify both belief and normative system change. However, it has been used for belief change only, since the beliefs or norms are represented as propositional formulas. We therefore propose, as a normative framework for normative system change, to replace propositional formulas in the AGM framework of theory change by pairs of propositional formulas, representing the rule based character of norms, and to add several principles from the input/output logic framework. In this new framework, we show that some of the AGM properties cannot be expressed, and other properties are consistent only for some logics, but not for others.

Categories and Subject Descriptors

I.2.11 [**Distributed Artificial Intelligence**]: Multiagent systems

General Terms

Theory

Keywords

Normative systems, norm change, input/output logic, belief revision

1. INTRODUCTION

The focus of this paper is the social/organizational structure of a multiagent system, and in particular norms and normative behavior. The consensus definition of the first workshop on normative multiagent systems in 2005 is that "Normative MultiAgent Systems are multiagent systems 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" [5]. In this definition, the explicit representation of norms and the possibility to violate norms were directly borrowed from the deontic logic in computer science tradition, and therefore uncontroversial. What was new in this definition is the common agreement in the community that normative systems specify how and to which extent the agents can modify the norms. It is therefore known as "the normchange definition" of normative multiagent systems. The central problem of changing norms lead to a workshop on norm change in 2007 [10], which highlighted that there are many distinct and complicated approaches, without a common framework or consensus on how to change norms. The only formal framework to evaluate and classify normative system change methods is the so-called AGM framework of theory change [2], which has originally been developed as a framework to describe and classify both belief and normative system change. AGM theory studies how a set of propositions should change in view of a possibly new conflicting information, by providing a set of postulates that the new belief or norm set should satisfy. Typically there are several sets that satisfy the conditions and no solution about which one to chose is provided. AGM theory respects a number of postulates which are useful in the setting of norm change, like success, the retracted proposition should not be obligatory anymore, or minimality. However, it has been used for belief change only, since the beliefs or norms are represented as propositional formulas. This leads to our central research question:

> How should we evaluate and classify norm change methods?

This breaks down into the following questions:

1. Which abstract model of normative system change?
2. How to classify norm contraction methods?
3. How to classify norm revision methods?

The description level of this paper is methodologies and languages. In our abstract model, we start from AGM theory of belief revision. However, we represent norms as pairs of propositional formulas, which are interpreted as rules that can be applied to a context to lead to a set of obligations. In our norm change approach we adopt principles from the input/output logic framework. As examples of normative systems, we consider three input/output logics. The properties for norm change are derived from properties of belief change like success, recovery, and so on.

The inspiration source of our work is from law, social sciences and philosophy. In law, norms represent the legal code, and change due to actions of legislators. In sociology, norms describe expectations about interactions among agents, and change due to changes in expectations. In philosophy and ethics, norms are studied in deontic logic.

The layout of this paper follows the research questions of this paper and is as follows. In Section 2 we introduce our abstract model of normative system change, in Section 3 we discuss contraction, and in Section 4 we consider the revision of normative systems.

Cite as: Normative Framework for Normative System Change, Guido Boella, Gabriella Pigozzi and Leendert van der Torre, *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. XXX-XXX.
Copyright © 2009, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

2. ABSTRACT MODEL NORM CHANGE

2.1 Norm change in multiagent systems

If the agents have an explicit representation of norms, and the normative system is endowed with mechanisms to change norms, then the multiagent system becomes more flexible and it can adapt itself to changes in the environment or in the goals of agents.

There are various kinds of norm modification in legal theories. The most common case is when the legal system is revised by introducing new norms that are incompatible with existing ones. A national law can derogate a regional law in a more restricted context, since obligations and permissions work as exceptions to previous norms, as, for example, we study in [4].

Besides this implicit modification mechanism, legal systems have explicit modifications. In this case, the law introduces norms whose peculiar objective is to change the system by specifying what and how other existing norms should be modified. Governatori and Rotolo [8] study such mechanisms, where norms can be annulled, eliminating also their effects in the past, abrogated, leaving their effects in the past hold, their temporal extension changed, etc.

Both implicit and explicit modifications may lead to more and more complex normative systems. Thus, in legal or normative multiagent systems, it is necessary at some point to redesign the system, for example by removing redundant norms. The problem of defining when a norm is redundant has been studied in [6].

2.2 Abstract model of normative system

In this paper we start from a general definition of an input/output logic framework introduced by Makinson and van der Torre [11], by discussing several of their principles.

The first input/output logic principle we adopt in this paper is that norms are not represented by propositional sentences, as in AGM framework for theory change [2], or as modal formulas, as in deontic logic, but as pairs of formulas of an arbitrary logic. In this paper, following most of the work on input/output logic, we use propositional logic for this arbitrary logic, and norms are thus represented by pairs of propositional formulas. The results in this paper hold also if we use a first-order, temporal or action logic, but the use of propositional logic makes the formal exposition simpler, and makes it also straightforward to relate our study to the AGM framework of theory change. In contrast to [11], we assume that the set of propositional atoms is finite, such that for each set of propositional formulas, there is a single propositional formula that is equivalent to it. The pair of propositional formulas represents a rule, and the two propositional formulas are called the antecedent and consequent of the rule. We restrict ourselves to the class of regulative norms that are used to generate obligations and prohibitions.

DEFINITION 1 (NORMS AS RULES [11]). *Let L be a propositional logic built on a finite set of propositional atoms A. A normative system $N \subseteq L \times L$ is a set of pairs of L, written as $N = \{(a_1, x_1), (a_2, x_2), \ldots, (a_n, x_n)\}$ and read as "if a_1, then it is obligatory that x_1", and so on. It may also be read as "if a_1, then it is forbidden that $\neg x_1$", etc.*

The second principle of the input/output logic framework we adopt in this paper is that the primary role of norms in a normative system is the derivation of obligations and prohibitions. Which obligations and prohibitions can be derived from a normative system depends on the factual situation, which we call the *context* or *input* and represent by a propositional formula. Makinson and van der Torre use a set of propositional formulas as input, but to simplify the third principle in the following section, we assume that the situation can be represented by a finite set of sentences, and thus by their conjunction. The function that associates with each context the set of obligations describes the meaning of the normative system, because it is a kind of 'operational semantics' of the normative system. We put the term 'operational semantics' in quotes, because this terminology is not used by Makinson and van der Torre. We use this terminology, because we hope it clarifies the notion of 'implied norm' in Definition 6 below. We represent the set of derived obligations and prohibitions as a set of propositional formulas, because that is the simplest representation and it facilitates the comparison with the AGM framework.

DEFINITION 2 ('OPERATIONAL SEMANTICS' [11]). *An input/output operation out : $(2^{L \times L}) \times L \rightarrow 2^L$ is a function from the set of normative systems and contexts, to a set of sentences of L. We say that x is obligatory in normative system N and context a if $x \in out(N, a)$.*

The following property expresses irrelevance of syntactic presentation.

DEFINITION 3 (IRRELEVANCE OF SYNTAX). *out satisfies irrelevance of syntax if the following two properties hold.*

IS1 *If $x \in out(N, a)$, a is logically equivalent to b in propositional logic, and x is logically equivalent to y, then $y \in out(N, b)$.*

IS2 *If $x \in out(N \cup \{(b, y)\}, a)$, b is logically equivalent to c in propositional logic, and y is logically equivalent to z, then $x \in out(N \cup \{(c, z)\}, a)$.*

The simplest input/output logic defined by Makinson and van der Torre is so-called simple-minded output.

DEFINITION 4 (SIMPLE-MINDED OUTPUT [11]). *x is in the simple-minded output of N in context a, written as $x \in out_1(N, a)$, if there is a set of norms $(a_1, x_1), \ldots, (a_n, x_n) \in N$ such that $a_i \in Cn(a)$ and $x \in Cn(x_1 \wedge \ldots \wedge x_n)$, where $Cn(S)$ is the consequence set of S in L.*

In this paper we use the following running example on the welfare policies of a virtual community on Second Life.

EXAMPLE 1. *Let the normative system consist of two norms $N = \{(poor, house), (old, healthins)\}$, stating that the community has to give a house with low rent (house) to low income agents (poor), and to provide free health insurance (healthins) to elderly agents (old). Therefore we have that the community has to provide a house to someone with no income if no-income implies poor, $house \in out_1(N, (no\text{-}income \rightarrow poor) \wedge no\text{-}income)$, because we have $poor \in Cn((no\text{-}income \rightarrow poor) \wedge no\text{-}income))$ and $house \in Cn(house)$. Moreover, the obligations of the community for low income elderly agents are all logical consequences of giving a house with low rent and providing a free health insurance, because we have $out_1(N, poor \wedge old) = Cn(house \wedge healthins)$.*

The operational semantics of a set of norms can be used to define the two fundamental concepts of equivalence of normative systems, and the redundancy of a norm in a normative system. For example, since normative systems tend to grow quickly and become difficult to understand, redundancy is a useful notion to simplify normative systems.

DEFINITION 5 (EQUIVALENCE AND REDUNDANCY). *Normative systems N and M are equivalent if and only if for all propositional formulas a, we have that $out(N, a) = out(M, a)$. A norm $(a, x) \in N$ is redundant in normative system N if and only if N is equivalent to $N \setminus \{(a, x)\}$.*

2.3 'Implication' among norms

The third principle we adopt from the input/output logic framework is that we define a notion of 'implication' among rules – again we write 'implication' in quotes, because Makinson and van der Torre do not use this terminology. It is not defined as a primitive notion, but derived from the 'operational semantics' of the set of norms, in the sense that a set of norms 'implies' another norm (a, x) if and only if x is obligatory in context a. Likewise, we say that $\langle a, x\rangle$ is implied if x is not obliged in context a. Note that this corresponds to a weak permission of $\neg x$, which is derived from regulative norms only, not from permissive norms (see [13] for a discussion on this distinction).

DEFINITION 6 (NORM 'IMPLICATION' [11, 13]). *Norm (a, x) is 'implied' by normative system N, written as $(a, x) \in out(N)$, if and only if $x \in out(N, a)$, and $\langle a, x\rangle$ is 'implied' by N, written as $\langle a, x\rangle \in out(N)$ if and only if $x \notin out(N, a)$. If it is clear from content which input/output logic out is used, then we write also $\overline{N}$ for $out(N)$.*

Makinson and van der Torre use this new representation of norm 'implication', because it shortens the notation in the definitions of the proof theory, as illustrated in the following example. In this paper we use this notation to enable the use of AGM revision postulates to input/output logics. For example, with a notion of implication defined among rules, we can define a success postulate on norm contraction by saying that the new set of norms does not 'imply' the contracted norm (see Section 3.4 for the formal details).

EXAMPLE 2. Strengthening of the Input *or SI is represented by $out(N, a) \subseteq out(N, a \wedge b)$, which is equivalent to $x \in out(N, a)$ implies $x \in out(N, a \wedge b)$, and which therefore can be represented by: if $(a, x) \in out(N)$, then $(a \wedge b, x) \in out(N)$. Visualized as proof rules, this property can thus be represented either by the following rule on the left using the 'operational semantics', or the rule with the norm 'implication' on the right. If the set of norms N does not change, then the two notations are equivalent.*

$$\frac{x \in out(N, a)}{x \in out(N, a \wedge b)} SI \qquad \frac{(a, x)}{(a \wedge b, x)} SI$$

Yet another way to represent this principle is the contrapositive: if $(a \wedge b, x) \notin out(N)$, then $(a, x) \notin out(N)$, which can be represented by the following two equivalent representations of the Weakening of the Input *or WI proof rule. It may be seen as the inverse of the Strengthening of the Input rule, and Makinson and van der Torre therefore write also SI^{-1}.*

$$\frac{x \notin out(N, a \wedge b)}{x \notin out(N, a)} SI^{-1} = WI \qquad \frac{\langle a \wedge b, x\rangle}{\langle a, x\rangle} SI^{-1} = WI$$

Other properties can be represented as proof rules too. For example, the first item of irrelevance of syntax or IS1 in Definition 3 can be represented as follows, where $\models$ stands for logical implication in propositional logic.

$$\frac{x \in out(N, a), \models a \leftrightarrow b, \models x \leftrightarrow y}{y \in out(N, b)} IS1 \quad \frac{(a, x), \models a \leftrightarrow b, \models x \leftrightarrow y}{(b, y)} IS1$$

Thus, to decide when a set of norms 'implies' another norm, we resort to the 'operational semantics' of the norms.

2.4 Tarskian closure properties on norms

Makinson and van der Torre observe that the relation between the 'implication' among rules $(a, x) \in out(N)$ and the 'operational semantics' $x \in out(N, a)$ has an analogy in classical logic, where the pair $a \models x$ is equivalent to the membership of x in the consequence set of a, written as $x \in Cn(a)$. However, it is important to see that the notion of 'implication' should not be identified with the notion of implication among conditionals in classical conditional logic. This is illustrated by the Tarskian properties on the closure operation on the input/output pairs. Whereas these properties hold by definition for a classical conditional logic, they represent non-trivial principles for a logic of rules. These additional principles may either be accepted or rejected.

Reflexivity expresses that if the input is precisely the antecedent of one of the norms, then the output contains the consequent of the norm. Monotony expresses that the application of one rule cannot block another rule, as in most rule based non-monotonic logics like, for example, Reiter's default logic. Idempotence expresses that if we have x in the output of a, then we can add (a, x) as a rule without changing the output.

DEFINITION 7 (CLOSURE). *out is a closure operation when the following three conditions hold.*

Reflexivity $x \in out(N \cup \{(a, x)\}, a)$ *(i.e., $N \subseteq out(N)$), if the context is precisely the antecedent of one of the rules, then the output contains the consequent of that rule.*

Monotony $x \in out(N_1, a)$ *implies* $x \in out(N_1 \cup N_2, a)$ *(i.e., $out(N_1) \subseteq out(N_1 \cup N_2)$), if the set of norms increases, then no conclusions are lost.*

Idempotence *if $x \in out(N, a)$, then for all b, we have $out(N, b) = out(N \cup \{(a, x)\}, b)$ (i.e., $out(N) = out(out((N)))$), if x is obligatory in context a, then (a, x) can be added to the normative system without any consequence.*

Makinson and van der Torre show that their seven input/output logics satisfy the Tarskian properties, and their notion of 'implication' among norms is therefore a Tarskian consequence relation. In this paper we consider only the first three of their logics.

DEFINITION 8 ([11]). *Let $N(a) = \{x \mid (a, x) \in N\}$, and a complete formula v is a conjunction of a maxiconsistent set of literals, or the conjunction of all literals of L. Simple-minded, basic and reusable output are defined as follows.*
$out_1(N, a) = Cn(N(Cn(a)))$
$out_2(N, a) = \cap\{out_1(v) \mid a \in Cn(v), v \text{ complete}\}$
$out_3(N, a) = \cap\{out_1(b) \mid a \in Cn(b), out_1(b) \subseteq Cn(b)\}$

The following property illustrates the major properties of the three logics, and implies that none of the logics satisfies identity ((a, a) for all a) or contraposition. Basic output handles reasoning by cases, and reusable output handles iterated detachment. See [11] for examples and additional properties.

PROPOSITION 1 ([11]). *$out_1(N)$ is the minimal set that contains $N \cup \{(\top, \top)\}$, is closed under replacement of logical equivalents in antecedent and consequent, and the following proof rules strengthening of the input SI, weakening of the output WO, and conjunction rule AND.*

$$\frac{(a, x)}{(a \wedge b, x)} SI \qquad \frac{(a, x \wedge y)}{(a, x)} WO \qquad \frac{(a, x), (a, y)}{(a, x \wedge y)} AND$$

out_2 satisfies in addition the proof rule disjunction OR, and out_3 the proof rule cumulative transitivity CT.

$$\frac{(a, x), (b, x)}{(a \vee b, x)} OR \qquad \frac{(a, x), (a \wedge x, y)}{(a, y)} CT$$

2.5 AGM theory and norm change

The AGM framework [2] for theory revision is the formal study of how a set of propositions should change in view of a new information that may cause an inconsistency with the existing ones. Expansion, revision and contraction are the three theory change operations that Alchourrón, Gärdenfors and Makinson identified in their AGM approach. The initial inspiration was a formal analysis of the dynamics of a legal code. How can a norm be removed or added to a code? What happens when the addition of a new law conflicts with some of the existing ones?

They observed that these operations are more general and can be defined on any deductively closed set K of propositional formulas. In particular, the *expansion* of K by a formula x is a set that accepts the unproblematic x and it is denoted by $K + x$. A *contraction* of K by x results in a set $K - x$ from which x does not follow anymore. Finally, the *revision* of K by x yields to a set $K * x$ which contains x and from which parts of K that were conflicting with x have been removed.

AGM theory is a well established formal theory. Here, we want to take back its original inspiration and use AGM postulates for norm change. Therefore, we define norm expansion, norm revision, and norm contraction. To generalize the AGM postulates for our framework, we define a norm set as a set of norms closed under an input/output logic, denoted by $\overline{N}$. Since we do not want to restrict ourselves to one particular logic, we use *out* to refer to any input/output logic. We write $\overline{N} \oplus (a, x)$ to indicate the expansion of a normative system N by a new rule, $\overline{N} \ominus (a, x)$ for the contraction of a norm (a, x) from $\overline{N}$, and $\overline{N} \circledast (a, x)$ for the revision of $\overline{N}$ by the new norm (a, x).

Like AGM expansion, the definition of norm expansion is straightforward. The new norm that the legislator wants to enforce does not cause any conflict with the existing legal code. Hence, (a, x) is added to $\overline{N}$ together with all the norms that can be derived from the union of $\overline{N}$ and (a, x) (similarly to theory revision, we assume $\overline{N}$ to be closed under input/output logic): $\overline{N} \oplus (a, x) = \overline{N \cup \{(a, x)\}}$.

EXAMPLE 3. $\overline{\{(\mathit{poor}, \mathit{house})\}} \oplus (\mathit{old}, \mathit{healthins}) = \overline{\{(\mathit{poor}, \mathit{house}), (\mathit{old}, \mathit{healthins})\}}$

Theory contraction and revision are the more complex and interesting types of change. The Levy and Harper identities highlight that revision and contraction are interdefinable. More specifically, the Levy identity defines the revision of a set by x as the expansion of the contracted set by $\neg x$. Because of the simplicity of this definition, the contraction is often considered as the basic type of change in revision. For this reason, we will also start from norm contraction.

Does revision really offer a satisfactory framework for norm revision? Some of the AGM axioms seem to be rational requirements in a legal context, whereas they have been criticized when imposed on theory change operators. An example is the *success* postulate, requiring that a new input must always be accepted in the set. It seems to be reasonable to impose such a requirement when we wish to enforce a new norm or obligation. On the other hand, when we consider a legal code as a set of *conditional* norms, the AGM principles prove to be too general to deal with the revision of a normative system.

In order to have a closer look to these questions, we now need to turn to the postulates for AGM theory contraction and revision and to the corresponding norm contraction and revision.

3. NORM CONTRACTION

We start by reminding the AGM contraction postulates.

DEFINITION 9. *Let K be a deductively closed set of propositional formulas. An AGM contraction operation $-$ satisfies the following postulates.*

K-1: *$K - x$ is a deductively closed set (closure or type)*

K-2: *$K - x \subseteq K$ (inclusion or contraction)*

K-3: *If $x \notin K$ then $K = K - x$ (vacuity or min. action)*

K-4: *If $\nvdash x$ then $x \notin (K - x)$ (success)*

K-5: *If $x \in K$ then $K \subseteq (K - x) + x$ (recovery)*

K-6: *If $\vdash x \leftrightarrow y$ then $K - x = K - y$ (extensionality)*

K-7: *$((K - x) \cap (K - y)) \subseteq K - (x \wedge y)$ (min-conjunction)*

K-8: *If $x \notin (K - (x \wedge y))$ then $K - (x \wedge y) \subseteq K - x$ (max-conjunction)*

The first problem we encounter is that the last two postulates refer to conjunctions, which are not defined for norms. This is not a major problem, since the first six postulates of AGM theory are known as the basic postulates, whereas the latter two are optional. So we restrict ourselves to the basic postulates.

DEFINITION 10. *Let out be an input/output logic. A norm contraction operator $\ominus$ satisfies the following postulates.*

N-1: *$\overline{N} \ominus (a, x)$ is closed under out (closure or type)*

N-2: *$\overline{N} \ominus (a, x) \subseteq \overline{N}$ (inclusion or contraction)*

N-3: *If $(a, x) \notin \overline{N}$ then $\overline{N} = \overline{N} \ominus (a, x)$ (vacuity or min. action)*

N-4: *If $(a, x) \notin out(\emptyset)$ then $(a, x) \notin \overline{N} \ominus (a, x)$ (success)*

N-5: *If $(a, x) \in \overline{N}$ then $\overline{N} \subseteq (\overline{N} \ominus (a, x)) \oplus (a, x)$ (recovery)*

N-6: *If $out(\{(a, x)\}) = out(\{(b, y)\})$ then $\overline{N} \ominus (a, x) = \overline{N} \ominus (b, y)$ (extensionality)*

In the remainder of this section, we are going to study these postulates. We approach them by translating the 'implied norm' representation back to the 'operational semantics'.

3.1 N-1: Closure

K-1: $K - x$ is a deductively closed set (closure or type)

N-1: $\overline{N} \ominus (a, x)$ is closed under *out* (closure or type)

The first postulate for theory change requires that the result of the contraction is a deductively closed set. Expansion, contraction and revision operators are indeed defined as functions from a deductively closed set and a formula x to a new deductively closed set.

The corresponding *closure* (type) postulate for norm contraction says that $\overline{N} \ominus (a, x)$ is closed under *out*. In other words, if $(b, y) \in out(\overline{N} \ominus (a, x))$, then $(b, y) \in \overline{N} \ominus (a, x)$. Therefore, if $y \in out(\overline{N} \ominus (a, x), b)$, then $(b, y) \in \overline{N} \ominus (a, x)$.

	$\overline{N}=\overline{N_1}$ {(*poor*, *house* ∧ *healthins*)}	$\overline{N_2}$ {(*poor*, *healthins*)}	$\overline{N_3}$ {(*poor*, *healthins*), (*poor* ∧ ¬*old*, *house* ∧ *healthins*)}	$\overline{N_4}$ {(*poor*, *healthins*), (*poor* ∧ ¬*old*, *house* ∧ *healthins*), (*poor* ∧ *old* ∧ *young*, *house* ∧ *healthins*)}
$\top$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
poor	$Cn(house \wedge healthins)$	$Cn(healthins)$	$Cn(healthins)$	$Cn(healthins)$
¬*poor*	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
poor ∨ *old*	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
poor ∧ *old*	$Cn(house \wedge healthins)$	$Cn(healthins)$	$Cn(healthins)$	$Cn(healthins)$
poor ∧ ¬*old*	$Cn(house \wedge healthins)$	$Cn(healthins)$	$Cn(house \wedge healthins)$	$Cn(house \wedge healthins)$
poor ∧ *old* ∧ *young*	$Cn(house \wedge healthins)$	$Cn(healthins)$	$Cn(healthins)$	$Cn(house \wedge healthins)$
...				

Figure 1: Four alternatives for $\overline{N} \ominus (poor \wedge old, house)$, $\overline{N_2}$, $\overline{N_3}$ and $\overline{N_4}$ satisfy the success postulate.

3.2 N-2: Inclusion

K-2: $K - x \subseteq K$ (inclusion or contraction)

N-2: $\overline{N} \ominus (a, x) \subseteq \overline{N}$ (inclusion or contraction)

K-2 states that, because $K - x$ is obtained from K by giving up x, it is required that no new formulas occur in $K - x$.

The corresponding *inclusion* (contraction) postulate for norm contraction says that $\overline{N} \ominus (a, x) \subseteq \overline{N}$. Together with N-1, this property says that if $y \in out(\overline{N} \ominus (a, x), b)$, then $y \in out(\overline{N}, b)$. In other words, for any context b, if something is obligatory after norm contraction, then it was already obligatory before the contraction.

3.3 N-3: Vacuity

K-3: If $x \notin K$ then $K = K - x$ (vacuity or min. action)

N-3: If $(a, x) \notin \overline{N}$ then $\overline{N} = \overline{N} \ominus (a, x)$ (vacuity or min. action)

If $x \notin K$, the criterion of minimal change requires that nothing is retracted from K.

The corresponding *vacuity* (min. action) postulate for norm contraction says that if $(a, x) \notin \overline{N}$ then $\overline{N} = \overline{N} \ominus (a, x)$. Together with N-1, this property says that if $x \notin out(\overline{N}, a)$, then $\overline{N} = \overline{N} \ominus (a, x)$. In other words, if we contract a norm (a, x), but x is not obligatory in context a, then the contraction does not have any effect.

3.4 N-4: Success

K-4: If $\not\vdash x$ then $x \notin (K - x)$ (success)

N-4: If $(a, x) \notin out(\emptyset)$ then $(a, x) \notin \overline{N} \ominus (a, x)$ (success)

K-4 says that, unless x is logically valid (in which case it can never be retracted), if we remove x from K, the resulting set will not contain x.

The corresponding *success* postulate for norm contraction says that if $(a, x) \notin out(\emptyset)$ then $(a, x) \notin \overline{N} \ominus (a, x)$. In other words, if $x \notin out(\emptyset, a)$, then $x \notin out(N \ominus (a, x), a)$. The following example illustrates the success postulate.

EXAMPLE 4. *Assume out_1, $N = \{(poor, house \wedge healthins)\}$ and four possible normative systems $\overline{N} \ominus (poor \wedge old, house)$ visualized in Figure 1. This figure must be read as follows. Each column represent one possible way in which $(poor \wedge old, house)$ can be contracted from $\overline{N}$. The short notation in on top, and the output of the set of norms for each input on the left is written below it.*

The success condition says that house $\notin out(\overline{N}', poor \wedge old)$, which is not satisfied by $\overline{N_1}$, but it is satisfied by $\overline{N_2}$, $\overline{N_3}$ and $\overline{N_4}$. Thus, there are several ways in which a set of norms can be contracted. The purpose of the postulates is precisely to distinguish the admissible solutions from the inadmissible ones.

However, unlike in AGM theory revision, the question here is not only what and how much to contract, but also for which inputs *to contract. This is what distinguishes the different solutions in the example. All the possible $\overline{N_i}$ prescribe that a free health insurance must be given to the poor. Independently of whether the input is the poor people only, or the poor and old or, again, the poor and the not old, we obtain the same $\overline{N_i}$. Hence, in order to obtain a normative system that performs a contraction for all poor people (also the not old ones), $\overline{N_i}$ should contain an additional norm as in N_3 and N_4.*

PROPOSITION 2. *Logical consequences of the success criterion for out_1, out_2 and out_3 are the inverses of proof rules given in Definition 8.*

PROOF. *Follows from [13, Observation 3], which says that the non-repetition property holds for:*

- *out_1 with TAUT, SI, WO, AND*
- *out_2 with TAUT, SI, WO, AND, OR*
- *out_3 with TAUT, SI, WO, CTA*

where CTA derives $(a, x \wedge y)$ from (a, x) and $(a \wedge x, y)$.

3.5 N-5: Recovery

K-5: If $x \in K$ then $K \subseteq (K - x) + x$ (recovery)

N-5: If $(a, x) \in \overline{N}$ then $\overline{N} \subseteq (\overline{N} \ominus (a, x)) \oplus (a, x)$ (recovery)

K-5 requires that expanding $K - x$ by x should give the same set as before the contraction, that is K.

The corresponding *recovery* postulate for norm contraction says that, if $(a, x) \in \overline{N}$, then $\overline{N} \subseteq (\overline{N} \ominus (a, x)) \oplus (a, x)$. In other words, contracting a normative system by (a, x) and then expanding by the same (a, x) should leave $\overline{N}$ unchanged.

EXAMPLE 5. *Assume $N = \{(poor, house \wedge healthins), (old, house \wedge healthins)\}$ and $\overline{N}' = \overline{N} \ominus (poor, house)$. Suppose that $\overline{N}'$ is calculated by the obviously too strong principle that house is contracted from the consequent of each rule, such that $\overline{N}' = out(\{(poor, healthins), (old, healthins)\})$. This does not satisfy recovery since for all reasonable input/output logics, adding $(poor, house)$ to $\overline{N}'$ does not give us $\overline{N}$ again.*

	$\overline{N}$ = $\{(poor, house)\}$	$\overline{N_1} = \overline{N} \ominus (poor \wedge old, house)$ = $\{(poor \wedge \neg old, house), (poor \wedge old \wedge young, house)\}$	$\overline{N_2} = \overline{N_1} \oplus (poor \wedge old, house)$ = $\{(poor \wedge \neg old, house), (poor \wedge old \wedge young, house), (poor \wedge old, house)\}$
$\top$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
$poor$	$Cn(house)$	$Cn(\top)$	$Cn(\top)$
$\neg poor$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
$poor \vee old$	$Cn(\top)$	$Cn(\top)$	$Cn(\top)$
$poor \wedge old$	$Cn(house)$	$Cn(\top)$	$Cn(house)$
$poor \wedge \neg old$	$Cn(house)$	$Cn(house)$	$Cn(house)$
$poor \wedge old \wedge young$	$Cn(house)$	$Cn(house)$	$Cn(house)$
...			

Figure 2: Recovery. $(N \ominus (poor \wedge old, house)) \oplus (poor \wedge old, house)$**.**

The following result shows that, surprisingly, the five postulates considered thus far are consistent only for some input/output logics, but not for others. In particular, if we adopt out_1 or out_3 then there is no single norm contraction operator satisfying the postulates.

PROPOSITION 3. *N-1 until N-5 cannot hold together for out_1 or out_3, but they can hold together for out_2.*

PROOF. *(sketch) A counterexample for out_1 and out_3 is $(poor, house) \notin (\{(poor, house)\} \ominus (poor \wedge old, house)) \oplus (poor \wedge old, house)$. $(poor, house)$ is not part of $\{(poor, house)\} \ominus (poor \wedge old, house)$, and there is no way to derive it when $(poor \wedge old, house)$ is added again. The same counterexample also works for out_3. It is visualized in Figure 2.*

An operator for out_2 can be defined as follows. From Definition 8 it follows directly that the output is determined by complete formulas. Assume we contract (a, x). Let V be a complete formula implying a, and contract x from $out(N, V)$ using a belief revision contraction.

3.6 N-6: Extensionality

K-6: If $\vdash x \leftrightarrow y$ then $K - x = K - y$ (extensionality)

N-6: If $out(\{(a, x)\}) = out(\{(b, y)\})$ then $\overline{N} \ominus (a, x) = \overline{N} \ominus (b, y)$ (extensionality)

It is the content of x rather than its particular linguistic formulation that determines the contraction of K by x. This means that logically equivalent sentences should lead to identical revisions.

The *extensionality* postulate for norm contraction says that if $out(\{(a, x)\}) = out(\{(b, y)\})$ then $\overline{N} \ominus (a, x) = \overline{N} \ominus (b, y)$. In other words, if for all c, we have $out(\{(a, x)\}, c) = out(\{(b, y)\}, c)$, then we have for all c: $out(\{(a, x)\} \ominus (b, y), c) = out(\{(b, y)\} \ominus (b, y), c)$.

4. NORM REVISION

In this section we discuss norm revision. We start by recalling the AGM postulates for revision, and we then introduce the corresponding postulates for norm revision. Possible definitions of a coherent normative system conclude the section.

4.1 Postulates for norm revision

DEFINITION 11. *Let K be a deductively closed set. An AGM revision operation $*$ satisfies the following postulates.*

K $*$ 1: *$K * x$ is a deductively closed set (closure or type)*

K $*$ 2: *$x \in (K * x)$ (success)*

K $*$ 3: *$K * x \subseteq K + x$ (inclusion)*

K $*$ 4: *If $\neg x \notin K$ then $K + x = K * x$ (vacuity)*

K $*$ 5: *$K * x = K_\perp$ iff $\vdash \neg x$ (triviality)*

K $*$ 6: *If $\vdash x \leftrightarrow y$ then $K * x = K * y$ (extensionality)*

K $*$ 7: *$K * (x \wedge y) \subseteq (K * x) + y$ (iterated $K*$ 3)*

K $*$ 8: *If $\neg y \notin K * x$ then $(K * x) + y \subseteq K * (x \wedge y)$ (iterated $K * 4$)*

The same argument as for contraction applies to the last two revision postulates. Hence, we restrict ourselves to the first six postulates. The corresponding postulates for norm revision are the following.

DEFINITION 12. *Let out be an input/output logic, and $\overline{N}$ a set of norms closed under out: A norm revision operator $\circledast$ satisfies the following postulates.*

N $\circledast$ 1: *$\overline{N} \circledast (a, x)$ is closed under out (closure or type)*

N $\circledast$ 2: *$(a, x) \in (\overline{N} \circledast (a, x))$ (success)*

N $\circledast$ 3: *$\overline{N} \circledast (a, x) \subseteq \overline{N} \oplus (a, x)$ (inclusion)*

N $\circledast$ 4: *If $\neg(a, x) \notin \overline{N}$ then $\overline{N} \oplus (a, x) = \overline{N} \circledast (a, x)$ (vacuity)*

N $\circledast$ 5: *$\overline{N} \circledast (a, x) = \overline{N}_\perp$ iff $\neg(a, x) \in out(\emptyset)$ (triviality)*

N $\circledast$ 6: *If $out(\{(a, x)\}) = out(\{(b, y)\})$ then $\overline{N} \circledast (a, x) = \overline{N} \circledast (b, y)$ (extensionality)*

Closure, success and extensionality state for revision the homonymous conditions we have seen for theory and norm contraction. N $\circledast$ 3 and N $\circledast$ 4 define the relation between revision and expansion: When $\neg(a, x) \notin \overline{N}$, revision and expansion coincide, i.e. $\overline{N} \circledast (a, x) = \overline{N} \oplus (a, x)$. More generally, the result of revising $\overline{N}$ by (a, x) should contain the new norm together with all those norms that have not been excluded from $\overline{N}$ in order to accommodate (a, x), hence $\overline{N} \circledast (a, x)$ should be a subset of $\overline{N} \oplus (a, x)$, as the inclusion condition states. Finally, N $\circledast$ 5 says that the revision of a normative system should be coherent if the new norm is not contradictory.

However, $\neg(a, x)$ and $\overline{N}_\perp$ are not defined in input/output logic thus far. Since revision is the type of change that occurs when a new information conflicts with the existing ones, we now have to turn to investigate what it means that a norm 'conflicts' with a set of existing ones.

4.2 Coherent normative systems

Theory revision postulates prescribe how a deductively closed set of formulas should change when the addition of a new formula causes an inconsistency to the initial set. Similarly, a legislator who wishes to add a new norm to a normative system N, and the addition conflicts with N, needs to make some changes to N in order to accommodate the new norm.

As norms do not bear truth values, we cannot, in any usual sense, say that a normative system is inconsistent. All we can consider is the consistency of the output of a set of norms. As in [9], we like to use the term *coherence* with respect to a set of norms with consistent output. However, when a normative system can be said to be coherent is not a trivial question, as the following tentative definitions show. The first proposal is to say that N is coherent if no inconsistent output can be derived:

(1) A normative system N is coherent iff $\bot \notin out(N, a)$.

However, one might argue that one should be able to determine whether a set of norms N is coherent or not regardless of what arbitrary facts a might be assumed. A better definition would be:

(1a) A normative system N is coherent iff there exists a context a such that $\bot \notin out(N, a)$.

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 $N = \{(poor, house), (poor, \neg house)\}$ that requires both $house$ and $\neg house$ to be realized in condition $poor$. If we denote by out any of the output operations out_1, out_2, out_3, we can say that $\bot \notin out(N, \neg poor)$: no conflicting demands arise when $\neg poor$ is factually assumed. Yet something seems wrong with a normative system that explicitly considers a fact only to tie to it conflicting normative consequences. The dual of $(1a)$ would be:

(1b) A normative system N is coherent iff for all contexts a, $\bot \notin out(N, a)$.

Now a set $N = \{(poor, house), (poor, \neg house)\}$ would no longer be termed coherent. However, we cannot be satisfied with this definition as $(1b)$ makes the claim that for *no* situation a, two norms $(poor, house)$, $(old, house)$ would ever come into conflict, which might seem too strong. We may wish to restrict a to contexts 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) A normative system N is coherent iff for all a with $(a, x) \in N$, $\bot \notin out(N, a)$.

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 $poor$ is a foreseen situation, and so is $house$, then also $poor \vee house$ or $poor \wedge house$ might be counted as foreseen situations for which the norms should be coherent.

However, there is a further difficulty: let N contain a norm $(poor, \neg poor)$ that, for conditions in which $poor$ is unalterably true, demands that $\neg poor$ be realized. For the principal output operations out, we then have $\neg poor \in out(N, poor)$, but not $\bot \in out(N, poor)$. 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 the input is not necessarily included in the output for all output operations, 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 normative system N is coherent iff $\bot \notin out(N, a) \cup \{a\}$.

But with definition (2) we obtain the questionable result that for any case of norm-violation, i.e. for any case in which $(poor, house) \in N$ and $(poor \wedge \neg house) \in Cn(a)$, N 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 $out(N, a)$ take $out(\{(poor, house) \in N \mid (poor \wedge \neg house) \notin Cn(a)\}, a)$ – but then a set N such that $(poor, \neg poor) \in N$ would not be incoherent.

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' [12]:

Definition (Output under constraints) *Let N be a set of conditional norms and C a set of propositional formulas. Then N is coherent in a under constraints C when out$(N, a) \cup C$ is consistent.*

Future study must define an output operation, determine the relevant contexts a, and find the constraints C, such that any set of norms N would be appropriately termed coherent or incoherent by this definition.

5. RELATED WORK

The use of norms in multiagent systems allows to regulate the behavior of the agents. However, agents may possess conflicting or incoherent norms. A way to detect and resolve conflicts is proposed in [16], where norms are specific to the agents and agents can adopt roles as in [15].

In this paper we take AGM theory change as a framework to evaluate the change of a normative system. The first work on the logic of the revision of a set of norms goes back to Bulygin and Alchourrón [1], and to Alchourrón and Makinson [3]. There, a legal code is taken to be a set of propositions together with their logical consequences. In other words, a norm x is taken to be simply a formula in propositional logic.

Although the amendment and the derogation of a legal code were the initial motivation in Alchourrón and Makinson' research on theory change, they considered a very general framework. Here we claimed that an appropriate account of norm change should be able to represent conditional norms, as in the input/output framework. So far, however, little attention has been paid on how the formalization of changes occurring in legal systems.

In [14] Maranhão focuses on the relation between belief revision and the jurisdictional activity. The AGM success postulate is criticized. Instead of accepting the new formula and removing some of the existing beliefs in the set, he defines a new operation, called *refinement*. According to refinement, the agent modifies his belief set by accepting the new sentence under certain condition. Formally, the refinement of K by a is the partial meet revision $K * (b \rightarrow a)$. The refinement operator is an attempt to capture the process in the legal practice in which a judge has to assess a case that presents a condition that was not foreseen by the legislator.

Governatori and Di Gusto [7] consider finite base revision (the revision of not deductively closed sets) where facts are distinguished from rules. They study a procedure which defines the revision of a base by a new fact as a change of the rules part of the base: new rules are found in order to accommodate the new information.

6. SUMMARY AND OUTLOOK

Norm change allows not only to adapt multiagent systems to the new situation, but also agents to simplify normative systems by recognizing and removing redundant ones. Given the current development of norm change methods in multiagent systems, we need a normative framework to evaluate and classify normative system change. Since the AGM theory has been developed as such a framework, we start from there, and to represent the rule based character of norms, we introduce various ideas from the input/output logic approach. Within this framework, we identify the principle to represent norms by pairs of formulas, the 'operational semantics' of a set of norms to derive obligations, prohibitions and weak permissions, and the notion of 'implication' between norms, indirectly defined in terms of the operational semantics.

In the tradition of theory change, we taken contraction to be the more elementary kind of change. We show that the basic postulates can easily carry over to the norm case, but the two additional postulates do not seem to have a straightforward counterpart in norm change. We illustrate that the success postulate may be interpreted as a set of success conditions on outputs. Then we show the surprising result that for two common systems for norms, called simple-minded and reusable output, the postulates do not allow for any norm change operation. On the positive side, we show that norm change operations exist for basic output. We also show that the proof theory of norm change is closely related to the proof theory of permissions from an input/output perspective.

Finally we consider norm revision, where we only give a partial answer to the question how norm revision methods can be classified. The translation of the AGM postulates is much more difficult than for the contraction case, because we have to define when a set of norms is "consistent", which in the area of normative systems is usually called coherent. We therefore introduce and discuss various notions of coherence in this setting. We leave to future work to investigate whether norm contraction and norm revision can be inter-defined as in the theory revision and contraction case.

Other topics of further research are an extension of the proposed framework with permissive and constitutive norms, and the study of norms under constraints, for example to represent contrary-to-duty reasoning, dilemmas and hierarchical normative systems. Existing developments of the input/ouput logic framework can be applied here directly. Moreover, when proposals of norm change methods are presented in a detailed way, the framework will be applied to study these methods.

Acknowledgment

Thanks to David Makinson and Jörg Hansen for discussions on the issues raised in this paper.

7. REFERENCES

[1] C. E. Alchourrón and E. Bulygin. The expressive conception of norms. In R. Hilpinen, editor, *New Studies in Deontic Logic*, pages 95–124. Reidel, 1981.

[2] C. E. Alchourrón, P. Gärdenfors, and D. Makinson. On the logic of theory change: Partial meet contraction and revision functions. *J. Symb. Log.*, 50(2):510–530, 1985.

[3] C. E. Alchourrón and D. Makinson. On the logic of theory change: Contraction functions and their associated revision functions. *Theoria*, 48:14–37, 1982.

[4] G. Boella and L. van der Torre. Institutions with a hierarchy of authorities in distributed dynamic environments. *Artificial Intelligence and Law Journal*, 16(1):53–71, 2008.

[5] G. Boella, L. van der Torre, and H. Verhagen. Introduction to normative multiagent systems. *Computation and Mathematical Organizational Theory*, 12(2-3):71–79, 2006.

[6] J. Broersen and L. van der Torre. Reasoning about norms, obligations, time and agents. In *Proc. of PRIMA 2007*, LNCS, 2008.

[7] G. Governatori and P. DiGiusto. Modifying is better than deleting: A new approach to base revision. In E. Lamma and P. Mello, editors, *AI*IA 99*, pages 145–154. Pitagora, 1999.

[8] G. Governatori and A. Rotolo. Changing legal systems: Abrogation and annulment part I: Revision of defeasible theories. In *Proc. of DEON 2008*, LNCS, pages 3–18, 2008.

[9] J. Hansen, G. Pigozzi, and L. van der Torre. Ten philosophical problems in deontic logic. In G. Boella, L. van der Torre, and H. Verhagen, editors, *Normative Multi-Agent Systems. Dagstuhl Seminar Proc. 07122*, 2007.

[10] http://icr.uni.lu/normchange07/.

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

[12] D. Makinson and L. van der Torre. Constraints for input-output logics. *Journal of Philosophical Logic*, 30(2):155–185, 2001.

[13] D. Makinson and L. van der Torre. Permissions from an input-output perspective. *Journal of Philosophical Logic*, 32(4):391–416, 2003.

[14] J. Maranhão. Refinement. A tool to deal with inconsistencies. In *Proc. of the 8th ICAIL*, pages 52–59, 2001.

[15] O. Pacheco and J. Carmo. A role based model for the normative speciÞcation of organized collective agency and agents interaction. In *Proc. of AAMAS 2003*, pages 145–184.

[16] W. W. Vasconcelos, M. J. Kollingbaum, and T. J. Norman. Resolving conflict and inconsistency in norm-regulated virtual organizations. In *Proc. of AAMAS 2007*, pages 632–639, 2007.