

Algebras for Agent Norm-Regulation

Jan Odelstad

Division of Computer Science
Univ of Gävle, Sweden E-mail: jod@hig.se

Magnus Boman

Department of Computer and Systems Sciences
The Royal Institute of Technology
and
Swedish Institute of Computer Science
Kista, Sweden E-mail: mab@sics.se

November 5, 2018

Abstract

An abstract architecture for idealized multi-agent systems whose behaviour is regulated by normative systems is developed and discussed. Agent choices are determined partially by the preference ordering of possible states and partially by normative considerations: The agent chooses that act which leads to the best outcome of all permissible actions. If an action is non-permissible depends on if the result of performing that action leads to a state satisfying a condition which is forbidden, according to the norms regulating the multi-agent system. This idea is formalized by defining set-theoretic predicates characterizing multi-agent systems. The definition of the predicate uses decision theory, the Kanger-Lindahl theory of normative positions, and an algebraic representation of normative systems.

Keywords: Norm, Multi-agent system, Norm-regulated system, Agent architecture, Boolean algebra, Normative position

1 Introduction

Within economic theory the consumer's behaviour has traditionally been described as determined by a utility function. During the latest two decades there has been a growing interest among researchers in how norms (for example rules of law) give restrictions on the behaviour induced by the utility function. The behaviour of the consumers or other economic agents, according to this model, is the result of an interplay between optimization of the utility function and restrictions due to norms. We will here show how this model can be used for regulating the behaviour of artificial agents.

An important problem in the behavioral sciences is how the autonomy of the individual can be reconciled with collective rationality, i.e. the rationality of the group?¹ If the individual is a human agent and the collective in question is the state, we are approaching a profound problem for politics and social science. However, here we limit ourselves to some aspects of a more modest question: What is an appropriate formal framework for the unification of individual autonomy with collective rationality for *artificial* agent systems? To our technical problem, the solution could be an agent architecture with a large range of applications. But this technical problem also has some bearing on the more general one mentioned above, as it may make it possible to construct and test model systems which resemble human social systems in some basic sense.

The approach to unification of individual autonomy and collective rationality chosen here is, as has already been emphasized, to focus on norms. There are mainly two aspects of norms in relation to multi-agent systems that will be dealt with, viz.

(1) the formal representation of norms regulating multi-agent systems and

(2) the role of norms in architectures for multi-agent systems.

These two aspects of multi-agent systems (MAS) are not independent, on the contrary: they presuppose each other. The aim of the essay as far as norm representation is concerned is to test an algebraic representation of normative systems [19] [20] [21] [24] [25] utilizing the Kanger-Lindahl theory of normative positions [14] [17] [18], and its later developments. In this essay we will discuss some characteristic features of this kind of norm representation and investigate the possibility of using it in MAS architectures. We have strived for imposing no particular choice of MAS architecture; a related study for a Belief-Desire-Intention agent architecture, for instance, is [9].

Next a few words about (2). The role that norms will have in MAS architectures is to delimit the autonomy of the agents. Metaphorically one can say that the norms define the scope (*Spielraum*) for an agent. The ‘wish’ or ‘desire’ of an agent is represented as a preference structure over possible states or situations.² From the norms of the system will follow a deontic structure of the states (consisting for example of normative positions) implying that some states are permissible while the rest are non-permissible. The agent chooses an act which leads to the permissible state it prefers the most. In section 3, we will transform this idea to a theory or model of how norms can be used to regulate the behaviour of multi-agent systems. The model will use the kind of norm representation mentioned above and it will be expressed in terms of set-theoretical predicates. Since this is the first attempt to test the above described approach we will use a rather idealized kind of MAS. The extent and goal of the idealization will be discussed at the beginning of section 3.

The study of how the behaviour of agents can be regulated by norms

¹Related to this problem is the question whether the autonomy of one individual restricts the autonomy of other individuals.

²A preference structure consists of a preference relation and perhaps one or several other relations, for example a difference relation.

has been pursued within many fields, including biology, computer science, economics, law, sociology and philosophy. Formalisms have often been employed in cross-disciplinary fashion, as in the combination of mathematics and logic [24], speech act theory and computational linguistics [8], and computer science and anticipatory systems [4]. What constitutes a norm naturally differs between such approaches. In this essay, a norm is represented as an implicative sentence where the antecedent is a descriptive condition stating the circumstances of an agent, and the consequent is a condition expressing the normative position that the agent has with respect to a state of affairs. A normative system is a joining of two Boolean algebra based on such implications. Our approach to norms is within the tradition of algebraic logic.

It is possible to regard the expression ‘norms for artificial agents’ as metaphorical, but we do not. From a formal point of view, multi-agent systems are sufficiently similar independently of whether the agents are human or not. In this essay, we adopt a fairly formal notion of a norm: a norm is characterized by its form or structure and not, for example, by its function.

In section 2, norms and normative systems are examined from different perspectives. The Kanger-Lindahl theory of normative positions is presented, and the representation problem of normative systems is introduced and an algebraic approach to its solution is outlined. In section 3, an abstract architecture for MAS based on deontic-action-logic is introduced. The formal structure of norms regulating the behaviour of such systems is presented in section 4. In sections 2-4, some of the ideas are elucidated using the waste collectors example. In section 5, an important step in the theory construction is presented, namely a scheme for how normative positions will restrict the set of acts that the agents are permitted to choose from. Section 6 contains conclusions and a discussion of possible extensions of the study.

2 Representations of Norms

2.1 The Waste Collectors Example

We illustrate some of the ideas in this essay using an example, introduced in [3], of a team of robots collecting nuclear waste. Each robot is here represented as one agent, and is placed on a spatial grid. It is important to realize that this example is here exploited for elucidating the role of norms in regulating the behavior of an agent system. It is not what the norms say, the material content of the norms, that is of interest here but their logical form and their formal connection to other parts of the agent architecture. Thus it is not the reasonableness of the normative systems which interest us but the reasonableness of the form of their representation and their role in the architecture.

Think of a spatial grid of squares in rows and columns. Each square is assigned a coordinate consisting of an ordered pair of integers, where the first number in the pair represents the column and the second the row. On some squares there is a number representing an amount of waste. A

group of agents, in the sequel called collectors, tries to collect as much waste as possible. Each collector has a utility function, such that the utility depends on the amount of waste per unit time (or unit distance).

The agents can move one square at a time in four directions, up (north), down (south), left (west) and right (east); these are the possible actions of the agents. But there are restrictions on how they may move. It is especially important how an agent may move in the neighbourhood of another. The *protected sphere* around the agent ω consists of nine squares forming a large square with ω in the middle, i.e. the protected sphere is the von Neumann neighbourhood of ω . The protected spheres of ω_1 and ω_2 can overlap with 1, 2, 3, 4, 6 or 9 squares.

We will now present a set of norms which regulates how the agents may move. Suppose ω_1 is about to move.

(n_1) If the protected spheres of ω_1 and ω_2 do not overlap then ω_1 may not move so that ω_1 and ω_2 overlap with two or three squares.

(n_2) If the protected spheres of ω_1 and ω_2 do not overlap then ω_1 may move so that ω_1 and ω_2 overlap with zero or one square.³

(n_3) If the protected spheres of ω_1 and ω_2 overlap with one or two squares, then ω_1 may move so that the protected spheres of ω_1 and ω_2 overlap with any number of squares (even zero).

(n_4) ω_1 may move so that the protected spheres of ω_1 and ω_2 overlap with six squares only if the protected spheres of ω_1 and ω_2 overlap with at least four squares.

(n_5) If the protected spheres of ω_1 and ω_2 overlap with four squares then ω_1 shall move so that the protected spheres of ω_1 and ω_2 do not overlap with three squares.

(n_6) If the protected spheres of ω_1 and ω_2 overlap with six squares, then ω_1 must move so that the protected spheres of ω_1 and ω_2 overlap with at least four squares.

(n_7) ω_1 may never move so that the protected spheres of ω_1 and ω_2 overlap with nine squares.

(n_8) ω_1 may always move so that the protected spheres of ω_1 and ω_2 overlap with zero squares.

This is a very primitive normative system, but it is not so easy to see what it says. To grasp the content of the normative system a transparent way of structuring and representing the norms is necessary. It is easy to see that most of the norms, but not all of them, are implications, i.e. of the form *p implies q*. However, as we will see in a later section, all of them can easily be transformed to implications. Note also that the norms contain expressions like *may*, *may not* and *shall*. To simplify, let us introduce some predicates. The protected sphere around ω is denoted *Protec*(ω). *Lap_i*(ω_1, ω_2) means that the protected spheres around ω_1 and ω_2 overlap with *i* squares. As a first step in the formalization of norms (n_1) and (n_2) we can write them as follows:

(n_1) If *Lap₀*(ω_1, ω_2) then it may not be the case that ω_1 sees to it that *Lap₂*(ω_1, ω_2) or *Lap₃*(ω_1, ω_2).

³Note that if the protected spheres of ω_1 and ω_2 do not overlap then ω_1 cannot move so that the spheres of ω_1 and ω_2 overlap with more than three squares.

(n_2) If $Lap_0(\omega_1, \omega_2)$ then it may be the case that ω_1 sees to it that $Lap_0(\omega_1, \omega_2)$ and $Lap_1(\omega_1, \omega_2)$.

The expression ‘ ω_1 sees to it that’ states the fact that ω_1 is active and moves and thereby sees to it that the overlap in question is realized. Let us use $Do(\omega_1, F)$ as an abbreviation for ‘ ω_1 sees to it that F ’. Furthermore, let us use ‘May’ as an abbreviation for ‘it may be the case that’ and ‘Shall’ for ‘it shall be the case that. We can then represent norms (n_1) and (n_2) as follows.

$$(n_1) \quad Lap_0(\omega_1, \omega_2) \longrightarrow \neg \text{May } Do(\omega_1, Lap_2(\omega_1, \omega_2) \vee Lap_3(\omega_1, \omega_2))$$

$$(n_2) \quad Lap_0(\omega_1, \omega_2) \longrightarrow \text{May } Do(\omega_1, Lap_0(\omega_1, \omega_2)) \wedge \text{May } Do(\omega_1, Lap_1(\omega_1, \omega_2))$$

The other norms can be represented in a similar way.

$$(n_3) \quad \text{For all } i, 0 \leq i \leq 9, \quad Lap_1(\omega_1, \omega_2) \vee Lap_2(\omega_1, \omega_2) \longrightarrow \text{May } Do(\omega_1, Lap_i(\omega_1, \omega_2))$$

$$(n_4) \quad \neg Lap_4(\omega_1, \omega_2) \wedge \neg Lap_6(\omega_1, \omega_2) \wedge \neg Lap_9(\omega_1, \omega_2) \longrightarrow \neg \text{May } Do(\omega_1, Lap_6(\omega_1, \omega_2))$$

$$(n_5) \quad Lap_4(\omega_1, \omega_2) \longrightarrow \text{Shall } Do(\omega_1, \neg Lap_3(\omega_1, \omega_2))$$

$$(n_6) \quad Lap_6(\omega_1, \omega_2) \longrightarrow \text{Shall } Do(\omega_1, Lap_4(\omega_1, \omega_2) \vee Lap_6(\omega_1, \omega_2) \vee Lap_9(\omega_1, \omega_2))$$

$$(n_7) \quad \omega_1 \neq \omega_2 \longrightarrow \neg \text{May } Do(\omega_1, Lap_9(\omega_1, \omega_2))$$

$$(n_8) \quad \text{May } Do(\omega_1, Lap_0(\omega_1, \omega_2))$$

In the formulations of (n_1) – (n_8) we have as a simplification omitted the universal quantifiers. For example (n_1) ought to be written more completely as

$$(n'_1) \quad \forall \omega_2 : [Lap_0(\omega_1, \omega_2) \longrightarrow \neg \text{May } Do(\omega_1, Lap_2(\omega_1, \omega_2) \vee Lap_3(\omega_1, \omega_2))]$$

This is just the first step in the representation of norms using deontic and action logic. In the next section we will discuss this kind of representation in some detail and show how we can get a more effective representation. We will return to the waste collectors in sections 3 and 4.

Note that a typical norm asserts that if a condition on the position of ω_1 is fulfilled, then ω_1 may or may not see to it that another condition will or will not be the case. An idea elaborated on in section 5 is that it is not permissible for ω_1 to perform an act which leads to a situation such that a condition c will be fulfilled, if c is such that ω_1 may not see to it that c is fulfilled. In this way the norms formulated in terms of requirements on positions determine what is permissible or obligatory for the agent to do in a given situation.

2.2 Norms as Ordered Pairs

A typical norm is a universal implication and in predicate logic it can often be represented as a universal sentence of the following form:

$$(n_9) : \forall x_1, \dots, x_\nu : p(x_1, \dots, x_\nu) \longrightarrow q(x_1, \dots, x_\nu).$$

Syntactically it consists of three parts: the sequence of universal quantifiers, the antecedent formula and the consequent formula. Note that norm (n_9) correlates open sentences: $p(x_1, \dots, x_\nu)$ is correlated to $q(x_1, \dots, x_\nu)$. An alternative point of view is to regard p and q as *conditions* and a norm as a relational statement correlating them:

$$(n_{10}) : p \mathcal{R} q$$

It is important here that the free variables in $p(x_1, \dots, x_\nu)$ are the same and in the same order as the free variables in $q(x_1, \dots, x_\nu)$. (It is, however,

not necessary that p and q have the same arity, for details see [20].) \mathcal{R} is a binary relation, and $p\mathcal{R}q$ is a relational statement equivalent to $\langle p, q \rangle \in \mathcal{R}$. Thus, we can represent a norm as $p\mathcal{R}q$ or $\langle p, q \rangle \in \mathcal{R}$, and from the latter relational statement it is just a small step to the representation of (n_{10}) as the ordered pair $\langle p, q \rangle$ where p is called the *ground* and q the *consequence* of the norm. A ground is a descriptive and a consequence is a normative condition.

As is easy to see, we can form conjunctions, disjunctions and negations of conditions in the following way.

- $(p \wedge q)(\omega_1, \dots, \omega_\nu)$ if and only if $p(\omega_1, \dots, \omega_\nu)$ and $q(\omega_1, \dots, \omega_\nu)$.
- $(p \vee q)(\omega_1, \dots, \omega_\nu)$ if and only if $p(\omega_1, \dots, \omega_\nu)$ or $q(\omega_1, \dots, \omega_\nu)$.
- $(p')(\omega_1, \dots, \omega_\nu)$ if and only if $\neg p(\omega_1, \dots, \omega_\nu)$.

It is therefore possible to construct Boolean algebras of conditions and we will return to this in subsection 2.4.

If p is a ν -ary condition and i_1, \dots, i_ν are individuals, then $p(i_1, \dots, i_\nu)$ is a sentence. A norm $\langle p, q \rangle$ can be regarded as a mechanism of inference. We can distinguish two cases. Suppose that p and s are descriptive conditions and q and t are normative.

1. From $p(i_1, \dots, i_\nu)$ together with $\langle p, q \rangle$ follows $q(i_1, \dots, i_\nu)$.
2. From $s\mathcal{R}p$ together with $\langle p, q \rangle$ and $q\mathcal{R}t$ follows $s\mathcal{R}t$.⁴

In 1, $\langle p, q \rangle$ functions as a deductive mechanism correlating sentences by means of instantiation, while in 2, $\langle p, q \rangle$ correlates one condition to another condition. Therefore, in the terminology of [1], 1 corresponds to the correlation of individual cases to individual solutions, and 2 corresponds to the correlation of generic cases to generic solutions.

Let us return to the suggested norm (n_1) in the waste collector example above. The ground of the norm is the binary condition Lap_0 and the consequence is the condition Λ defined in the following way:

$$\Lambda(\omega_1, \omega_2) \text{ iff } \neg \text{May Do}(\omega_1, Lap_2(\omega_1, \omega_2) \vee Lap_3(\omega_1, \omega_2)).$$

Λ is a normative condition obtained by applying the operators May and Do to the disjunction of the descriptive conditions Lap_2 and Lap_3 .

2.3 Normative Positions

An important contribution to deontic logic was made by Stig Kanger, who combined the deontic operator Shall with the binary action operator Do [13]. To be specific, Shall Do(x, q) means that it shall be that x sees to it that q , while \neg Shall Do($y, \neg q$) means that it is not the case that it shall be that y sees to it that not q . The combination of the deontic operator Shall with the action operator Do and the negation operation \neg gives us a powerful language for expressing purely normative sentences. Kanger emphasized the possibilities of external and internal negation of sentences where these operators are combined. Using combinations of deontic and

⁴Note that $s\mathcal{R}p$ is relating two sentences of the same kind and the same holds for $q\mathcal{R}t$; s and p are descriptive but q and t are normative. A norm consists of sentences of different kinds.

action operators, we can formulate norms in a concise way. A conditional norm may for example have the form:

$$p(x, y) \rightarrow \text{Shall Do}(x, \neg q(y)).$$

Note that the sentence $\text{May Do}(x, q)$ can be defined in terms of the operators Shall and Do in the following way:

$$\text{May Do}(x, q) \text{ if and only if } \neg \text{Shall } \neg \text{Do}(x, q).$$

The logical postulates for Shall and Do assumed by Kanger are (where \implies is the relation of logical consequence and \iff of logical equivalence):

1. If $p \implies q$, then $\text{Shall } p \implies \text{Shall } q$.
2. $(\text{Shall } p \ \& \ \text{Shall } q) \implies \text{Shall}(p \ \& \ q)$.
3. $\text{Shall } p \implies \neg \text{Shall } \neg p$.
4. If $p \iff q$, then $\text{Do}(x, p) \iff \text{Do}(x, q)$.
5. $\text{Do}(x, p) \implies p$.

The five conditions are, according to Kanger, acceptable in most contexts.

Kanger used the deontic-action-language as a basis for a theory of normative positions and his theory, expressed as a theory of types of rights, was further developed by Lars Lindahl in his three systems of types of normative positions [17]. The simplest one is the system of one-agent types of normative position, and we will restrict ourselves to the utilization of this system here.

Let $\pm\alpha$ stand for either of α or $\neg\alpha$. Starting from the scheme $\pm\text{May}\pm\text{Do}(x, \pm q)$, where \pm stands for the two alternatives of affirmation or negation, a list is made of all maximal and consistent conjunctions—maxiconjunctions (see [22] p.405f.)—such that each conjunct satisfies the scheme. Note that the expression $\neg\text{Do}(x, q) \ \& \ \neg\text{Do}(x, \neg q)$ expresses x 's passivity with regard to q . Here this expression is abbreviated as $\text{Pass}(x, q)$. By this procedure the following list of seven maxiconjunctions is obtained, which are denoted $\mathbf{T}_1(x, q), \dots, \mathbf{T}_7(x, q)$ (see [17], p.92).

- $\mathbf{T}_1(x, q) : \text{MayDo}(x, q) \ \& \ \text{MayPass}(x, q) \ \& \ \text{MayDo}(x, \neg q)$.
- $\mathbf{T}_2(x, q) : \text{MayDo}(x, q) \ \& \ \text{MayPass}(x, q) \ \& \ \neg\text{MayDo}(x, \neg q)$.
- $\mathbf{T}_3(x, q) : \text{MayDo}(x, q) \ \& \ \neg\text{MayPass}(x, q) \ \& \ \text{MayDo}(x, \neg q)$.
- $\mathbf{T}_4(x, q) : \neg\text{MayDo}(x, q) \ \& \ \text{MayPass}(x, q) \ \& \ \text{MayDo}(x, \neg q)$.
- $\mathbf{T}_5(x, q) : \text{MayDo}(x, q) \ \& \ \neg\text{MayPass}(x, q) \ \& \ \neg\text{MayDo}(x, \neg q)$.
- $\mathbf{T}_6(x, q) : \neg\text{MayDo}(x, q) \ \& \ \text{MayPass}(x, q) \ \& \ \neg\text{MayDo}(x, \neg q)$.
- $\mathbf{T}_7(x, q) : \neg\text{MayDo}(x, q) \ \& \ \neg\text{MayPass}(x, q) \ \& \ \text{MayDo}(x, \neg q)$.

$\mathbf{T}_1, \dots, \mathbf{T}_7$ are called the types of one-agent positions. Note that $\neg\text{MayDo}(x, q) \ \& \ \neg\text{MayPass}(x, q) \ \& \ \neg\text{MayDo}(x, \neg q)$ is logically false, according to the logic of Shall and May . It is easy to see that the last three types can more concisely be described as follows:

- $\mathbf{T}_5(x, q) : \text{Shall Do}(x, q)$.
- $\mathbf{T}_6(x, q) : \text{Shall Pass}(x, q)$.

- $\mathbf{T}_7(x, q)$: Shall Do($x, \neg q$).

Note that the following symmetry principles hold:

- $\mathbf{T}_1(x, q)$ if and only if $\mathbf{T}_1(x, \neg q)$
- $\mathbf{T}_2(x, q)$ if and only if $\mathbf{T}_4(x, \neg q)$
- $\mathbf{T}_3(x, q)$ if and only if $\mathbf{T}_3(x, \neg q)$
- $\mathbf{T}_5(x, q)$ if and only if $\mathbf{T}_7(x, \neg q)$
- $\mathbf{T}_6(x, q)$ if and only if $\mathbf{T}_6(x, \neg q)$

The systems of normative positions can serve as a tool for describing the normative positions of different agents x, y, z, \dots with regard to states of affairs p, q, r, \dots . A set of such descriptions, however, is not a representation of a normative system. This is due to the fact that a normative system is not a description of the actual normative positions of individuals. Rather, the essential feature of a normative system consists in so-called normative correlations, i.e., as jurists might say, in correlating normative consequences to operative facts. The formal system of normative positions increases the expressive power of norm formulation as regards consequences.

The one-agent types in the Kanger-Lindahl theory of normative positions can be used as operators on descriptive conditions to get deontic conditions [21]. As a simple example, suppose that r is a unary condition. Then $T_i r$ (with $1 \leq i \leq 7$) is the binary condition such that $T_i r(y, x)$ iff $\mathbf{T}_i(x, r(y))$, where $\mathbf{T}_i(x, r(y))$ is the i th formula of one-agent normative positions. Note that for example $\mathbf{T}_3(x, r(y))$ means

$$\text{MayDo}(x, r(y)) \ \& \ \neg \text{MayPass}(x, r(y)) \ \& \ \text{MayDo}(x, \neg r(y)).$$

If $\langle p, T_i r \rangle$ is a norm, then from $p(x_1, x_2)$ we can, by using the norm, infer $T_i r(x_1, x_2)$ and thus also $\mathbf{T}_i(x_2, r(x_1))$, which means that, with regard to the state of affairs $r(x_1)$, x_2 has a normative position of type \mathbf{T}_i .

The theory of normative positions was developed during the 60s and 70s, primarily as an analytical tool to be used in jurisprudence and political science; Kanger's theory was originally expressed as a theory of types of rights. Some further refinement of the systems have more recently been made by Andrew J.I. Jones and Marek Sergot (see [11] [12] [27] [28]). A special feature of the work of Jones and Sergot, as of Herrestad and Krogh [15] [16], is that applications in computer science are in view. Even though Sergot has constructed a computer program, Norman-G, based on the theory of normative positions, we have chosen to base our exposition on Kanger/Lindahl in order to not complicate the model further.

2.4 Representations of Norms: The Algebraic Approach

An adequate representation of norms is important for the construction of norm-based architectures and we will here use an algebraic approach to the representation problem.⁵ One of the main tools in this endeavour is

⁵This subsection is based on earlier work by Lindahl and Odelstad, see [21] [24] [25].

the theory of a Boolean quasi-ordering, which is an extension of the theory of Boolean algebras. A norm is regarded as consisting of two objects, a ground and a consequence standing in a relation to each other. The ground belongs to one Boolean quasi-ordering and the consequence to another. Therefore, we can view a normative system as a joining of a Boolean quasi-ordering of grounds to a Boolean quasi-ordering of consequences.

Definition. The relational structure $\langle B, \wedge, ', R \rangle$ is a *Boolean quasi-ordering (Bqo)* if $\langle B, \wedge, ' \rangle$ is Boolean algebra and R is a binary, reflexive and transitive relation on B (i.e. R is a quasi-ordering) such that R satisfies the following conditions for all a, b and c in B :

- (1) aRb and aRc implies $aR(b \wedge c)$.
- (2) aRb implies $b'Ra'$.
- (3) $(a \wedge b)Ra$.
- (4) not $\top R \perp$. ■

Before we describe the use of Boolean quasi-orderings for representing norms, let us say a few words on some formal aspects of such structures. The indifference part of R is denoted Q and is defined by: aQb if and only if aRb and bRa . Similarly, the strict part of R is denoted S and is defined by: aSb if and only if aRb and not bRa .

Let \leq be the partial ordering determined by the Boolean algebra $\langle B, \wedge, ' \rangle$.⁶ From requirement (3) for Boolean quasi-orderings it follows that $a \leq b$ implies aRb . If $\langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering then we say that the Boolean algebra $\langle B, \wedge, ' \rangle$ is the *reduct* of $\langle B, \wedge, ', R \rangle$, denoted \mathcal{B}^{red} .

Suppose that $\mathcal{B} = \langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering and Q is the indifference part of R . The *quotient algebra* of \mathcal{B} with respect to Q is a structure $\langle B/Q, \cap, -, \preceq_Q \rangle$ such that $\langle B/Q, \cap, - \rangle$ is a Boolean algebra and \preceq_Q is the partial ordering determined by this algebra. The natural mapping of $\langle B, \wedge, ' \rangle$ onto $\langle B/Q, \cap, - \rangle$ is a homomorphism [24], and $\langle B/Q, \cap, - \rangle$ is called the *quotient reduction* of \mathcal{B} . Thus there are two Boolean algebras which should be kept apart, namely \mathcal{B}^{red} , i.e. the reduct of \mathcal{B} , and the quotient reduction of \mathcal{B} .

Although, by a transition to equivalence classes, from a Boolean quasi-ordering we get a new Boolean algebra, there is a point in remaining within the framework of Boolean quasi-orderings. In the models where the domain of a Boolean quasi-ordering is a set of conditions, we may want to distinguish two conditions a and b even though it holds that aQb (and therefore a and b belong to the same Q -equivalence class), because they may have different meaning.

An important class of models of the theory of Boolean quasi-orderings consists of models having a set of conditions as its domain.

Definition. A *condition implication structure (cis)* is a Boolean quasi-ordering $\mathcal{B} = \langle B, \wedge, ', R \rangle$ such that B is a domain of conditions, and R is such that aRb represents that a implies b . ■

This reading of R is justified since, if a and b are ν -ary conditions, aRb is the representation of

$$\forall x_1, \dots, x_\nu : a(x_1, \dots, x_\nu) \rightarrow b(x_1, \dots, x_\nu).$$

⁶As per usual, \leq is defined by $a \leq b$ if and only if $a \wedge b = a$.

The theory of Boolean quasi-orderings is of a very general character, and condition implication structures are not the only kind of models which are interesting as representations of normative structures. It is easy to see that we can construct a model of this theory out of a first order theory Σ . Consider the structure $\langle B, \wedge, ', R \rangle$ where $\langle B, \wedge, ' \rangle$ is the Lindenbaum algebra of the predicate calculus. Let R be the quasi-ordering on B determined by the Lindenbaum algebra of Σ . Then $\langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering (cf. [2], p.61 and [7], p.73).

From an algebraic point of view a norm is a kind of link or joining of one Boolean quasi-ordering to another. To make this idea precise we need some definitions.

Definition. The *subinterval* relation generated by the quasi-orderings $\langle B_1, R_1 \rangle$ and $\langle B_2, R_2 \rangle$ is the binary relation \leq on $B_1 \times B_2$ such that $\langle a_1, a_2 \rangle \leq \langle b_1, b_2 \rangle$ if and only if $b_1 R_1 a_1$ and $a_2 R_2 b_2$. ■

Note that \leq is a quasi-ordering, i.e. transitive and reflexive. Let \approx denote the equality part of \leq and \triangleleft the strict part of \leq . Then the following holds:

$\langle a_1, a_2 \rangle \approx \langle b_1, b_2 \rangle$ if and only if $b_1 Q_1 a_1$ and $a_2 Q_2 b_2$
 $\langle a_1, a_2 \rangle \triangleleft \langle b_1, b_2 \rangle$ if and only if $(b_1 S_1 a_1 \text{ and } a_2 R_2 b_2)$ or $(b_1 R_1 a_1 \text{ and } a_2 S_2 b_2)$

where Q_i is the equality-part of R_i and S_i is the strict part of R_i . $\langle a_1, a_2 \rangle$ is a *minimal element* in $X \subseteq B_1 \times B_2$ with respect to $\langle B_1, R_1 \rangle$ and $\langle B_2, R_2 \rangle$ if and only if there is no $\langle x_1, x_2 \rangle \in X$ such that $\langle x_1, x_2 \rangle \triangleleft \langle a_1, a_2 \rangle$.

A quasi-ordering is closely related to a partial ordering. If $\langle B, R \rangle$ is a quasi-ordering and Q is the equivalence part of R , then R generates a partial ordering on the set of Q -equivalence classes generated from B . The definitions of least upper bound (lub) and greatest lower bound (glb) for partial orderings are easily extended to quasi-orderings, but the lub or glb of a subset of a quasi-ordering is not necessarily unique but can consist of a set of elements.

Definition. A *joining-system* is an ordered triple $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ such that $\mathcal{B}_1 = \langle B_1, R_1 \rangle$ and $\mathcal{B}_2 = \langle B_2, R_2 \rangle$ are quasi-orderings and $\mathcal{J} \subseteq B_1 \times B_2$ and the following conditions are satisfied where \leq is the subinterval relation generated by \mathcal{B}_1 and \mathcal{B}_2 :

- (1) for all $b_1, c_1 \in B_1$ and $b_2, c_2 \in B_2$, $\langle b_1, b_2 \rangle \in \mathcal{J}$ and $\langle b_1, b_2 \rangle \leq \langle c_1, c_2 \rangle$ implies $\langle c_1, c_2 \rangle \in \mathcal{J}$,
- (2) if for all $c_1 \in C_1 \subseteq B_1$, $\langle c_1, b_2 \rangle \in \mathcal{J}$ and $\text{lub } C_1 \neq \emptyset$, then $\langle a_1, b_2 \rangle \in \mathcal{J}$ for all $a_1 \in \text{lub } C_1$,
- (3) if for all $c_2 \in C_2 \subseteq B_2$, $\langle b_1, c_2 \rangle \in \mathcal{J}$ and $\text{glb } C_2 \neq \emptyset$, then $\langle b_1, a_2 \rangle \in \mathcal{J}$ for all $a_2 \in \text{glb } C_2$. ■

Definition. Suppose that $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ is a joining system. A *minimal element* in $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ is a minimal element $\langle a_1, a_2 \rangle$ in \mathcal{J} with respect to \mathcal{B}_1 and \mathcal{B}_2 . The set of minimal elements in $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ is denoted $\text{min } \langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ or just $\text{min } \mathcal{J}$. ■

Definition. A *Boolean joining-system* $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ is a joining-system such that \mathcal{B}_1 and \mathcal{B}_2 are Boolean quasi-orderings. A *ground-consequence*

system (*gc-system*) is a Boolean joining-system $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ such that \mathcal{B}_1 and \mathcal{B}_2 are condition implication structures. ■

Definition. A Boolean joining-system $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ satisfies *connectivity* if whenever $\langle c_1, c_2 \rangle \in \mathcal{J}$ there is $\langle b_1, b_2 \rangle \in \mathcal{J}$ such that $\langle b_1, b_2 \rangle$ is a minimal element in $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ and $\langle b_1, b_2 \rangle \trianglelefteq \langle c_1, c_2 \rangle$. ■

It is easy to see that if a Boolean joining-system satisfies connectivity, then the set of minimal joinings determines the set of joinings. If in a Boolean joining-system $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$, \mathcal{B}_1 and \mathcal{B}_2 are complete in a sense which is a straightforward generalization of the notion of completeness applied to Boolean algebras, then $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ satisfies connectivity. For a proof, see [23].

A normative system \mathcal{N} can be represented as a *gc-system* $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$. The elements in \mathcal{J} are then the norms of the system and in a norm $\langle a_1, a_2 \rangle$, a_1 is the ground and a_2 the consequence. The elements in $\text{min}\mathcal{J}$ constitute the set of minimal norms of \mathcal{N} . If \mathcal{N} satisfies connectivity, then it is characterized by its set of minimal norms, a fact which will be of interest in the sequel.

The set of minimal elements of a joining system exhibits an interesting structure and it is possible to distinguish between different types of joining systems depending on the properties of the set of minimal elements (see [24] and [21]). It is therefore also possible to distinguish between different kinds of normative system depending on the structural properties of the set of minimal norms. Furthermore, applications of the set of minimal norms seem to make changes and extensions of normative systems easy to describe and make it possible to divide the normative system into different parts which can be changed independently. However, these lines of thought will not be pursued here.

3 An Architecture for Norm-Regulation of Multi-Agent Systems

3.1 Deontic-Action-Logic Multi-Agent Systems

In this section we will give a definition of a deontic-action-logic based multi-agent system, abbreviated DALMAS. DALMAS is an abstract architecture (cf. [30], p.31.) for idealized multi-agent systems using normative systems. The idea behind the architecture is roughly the following. When it is agent ω 's turn to move it chooses an act out of a set of feasible alternatives and the result will be that the system enters a new state; which state depends on the actual state of the system when the act is performed. The agent's choice is determined partially by the preference ordering of the possible states and partially by the deontic structure: the agent chooses that act which leads to the best outcome of all permissible actions. If an action is permissible or not depends on whether the result of performing the action leads to a state which satisfies a condition which is forbidden according to the normative system regulating the multi-agent system. In this section we shall see how this idea can be formalized.

DALMAS is a global clock (synchronous update), global state, global dynamics system. It can be viewed as a simplification constructed for conceptual and computer simulation purposes. In particular, we use the system as a model system for studying the interplay between preferences and norms in MAS architectures. We hope that it will be possible to transform the definition of a DALMAS into a system that allows for study through microsimulation (cf. [5]).

Definition. A DALMAS is an ordered 7-tuple $\langle \Omega, S, A, \mathcal{A}, \Delta, \Pi, \Gamma \rangle$ containing

- an agent set Ω (ω, ω_1, \dots elements in Ω),
- a state or phase space S (r, s, s_1, \dots elements in S),
- an action set A such that for all $a \in A$, $a : \Omega \times S \rightarrow S$ such that $a(\omega, r) = s$ means that if the agent ω performs the act a in state r , then the result will be state s (a, b, a_1, \dots elements in A),⁷
- a function $\mathcal{A} : \Omega \times S \rightarrow \wp(A)$ where $\wp(A)$ is the power set of A ; $\mathcal{A}(\omega, s)$ is the set of acts accessible (feasible) for agent ω in state s ,
- a deontic structure-operator $\Delta : \Omega \times S \rightarrow \mathcal{D}$ where \mathcal{D} is a set of deontic structures of the same type with subsets of A as domains and $\Delta(\omega, s)$ is ω 's deontic structure on $\mathcal{A}(\omega, s)$ in state s ,⁸
- a preference structure-operator $\Pi : \Omega \times S \rightarrow \mathcal{P}$ where \mathcal{P} is a set of preference structures of the same type with subsets of A as domains and $\Pi(\omega, s)$ is ω 's preference structure on $\mathcal{A}(\omega, s)$ in state s ,
- a choice-set function $\Gamma : \Omega \times S \rightarrow \wp(A)$ where $\Gamma(\omega, s)$ is the set of actions for ω to choose from in state s .

■

A *situation* for the DALMAS \mathfrak{D} is determined by the agent to move, ω , and the state s . A situation is represented by an ordered pair $\langle \omega, s \rangle$. The set of situations for \mathfrak{D} is thus $\Omega \times S$.

In a DALMAS, all the agents have the same initial set of actions. The set of actions to choose from (the choice-set) in a situation $\langle \omega, s \rangle$ is determined by the agent's deontic structure $\Delta(\omega, s)$ and preference structure $\Pi(\omega, s)$. If $\Gamma(\omega, s)$ consists of one action, then this action applied in the situation $\langle \omega, s \rangle$ is the resulting state when ω acts in state s .

A *simple* DALMAS is a DALMAS containing the following simple versions of Δ , Π , and Γ .

1. $\Delta(\omega, s) \subseteq \mathcal{A}(\omega, s)$ and $\Delta(\omega, s)$ is the set of permissible actions for ω in the state s ,
2. $\Pi(\omega, s) = \langle \mathcal{A}(\omega, s), \succsim \rangle$ where \succsim is a weak ordering,⁹
3. $\Gamma(\omega, s) = \{x \in \Delta(\omega, s) : \text{for all } y \in \Delta(\omega, s), x \succsim y\}$.

⁷According to Savage [26] p.13: "an act is a function attaching a consequence to each state of the world".

⁸Two structures are of the same type if they have the same number of relations and corresponding relations in both structures have the same arity.

⁹ $\langle A, \succsim \rangle$ is a *weak ordering* if for all a, b, c in A , the following axioms are satisfied: $a \succsim b$ or $b \succsim a$; if $a \succsim b$ and $b \succsim c$, then $a \succsim c$.

Hence, in a simple DALMAS the choice-set consists of the best actions which are permissible. Among the elements in A there can be a pass action, which means the agent does nothing. If we combine the existence of such an action with very short clock cycles, we obtain systems with close to asynchronous behaviour (cf. [5] [10]).

A DALMAS is not deterministic, since it does not determine in which order the agents are going to move, and the choice-set may contain more than one action in every situation. Let us therefore make the following definition.

Definition. A *deterministic* DALMAS is an ordered 9-tuple

$\langle \Omega, A, S, \mathcal{A}, \Delta, \Pi, \Gamma, \tau, \gamma \rangle$ such that $\langle \Omega, A, S, \mathcal{A}, \Delta, \Pi, \Gamma \rangle$ is a DALMAS and

- $\tau : \Omega \rightarrow \Omega$ is a turn-operator such that $\tau(\omega) = \varkappa$ means that it is \varkappa 's turn after ω ; τ determines a simple agent priority,
- $\gamma : \wp(A) \rightarrow A$ is a tie-breaking function, determining which of several permissible and equally preferred actions to choose.

■

Note that $\gamma(\Gamma(\omega, s)) \in A$ and thus $[\gamma(\Gamma(\omega, s))](\omega, s) \in S$. Define $f : \Omega \times S \rightarrow \Omega \times S$ in the following way:

$f(\omega, s) = \langle \tau(\omega), [\gamma(\Gamma(\omega, s))](\omega, s) \rangle$. Note further that it is possible to iterate f . We define f^n as:

$$\begin{aligned} f^1(\omega, s) &= f(\omega, s) \\ f^n(\omega, s) &= f(f^{(n-1)}(\omega, s)). \end{aligned}$$

Definition. The k -event run of a deterministic DALMAS \mathfrak{D} determined by the *initial situation* $\langle \omega_0, s_0 \rangle$ is the sequence $\langle \langle \omega_0, s_0 \rangle, f^1(\omega_0, s_0), \dots, f^k(\omega_0, s_0) \rangle$.

■

Suppose that $\langle \Omega, A, S, \mathcal{A}, \Delta, \Pi, \Gamma, \tau, \gamma \rangle$ is a deterministic DALMAS. Then $\langle \Omega \times S, T, \phi \rangle$ is a dynamical system, where T is the set of natural numbers and ϕ is the evolution operator defined by $\phi : \Omega \times S \times T \rightarrow \Omega \times S$ such that

$$\begin{aligned} \phi(\omega, s, 0) &= \langle \omega, s \rangle \\ \phi(\omega, s, t) &= f^t(\omega, s) \text{ if } t \geq 1. \end{aligned} \text{ }^{10}$$

Note that $\langle \phi(\omega_0, s_0, 0), \phi(\omega_0, s_0, 1), \dots, \phi(\omega_0, s_0, k) \rangle$ is the run of \mathfrak{D} determined by $\langle \omega_0, s_0 \rangle$ and consisting of k moves.

3.2 The Waste Collector System as a Dalmas

We elucidate some of the aspects of the definition of a DALMAS using the waste collectors example introduced in section 2.1.

3.2.1 Agents and States

Suppose that we have a set Ω of k agents, $\Omega = \{\omega_1, \dots, \omega_k\}$. A state for the system is characterized by the position of each of the agents, the location

¹⁰See, for example, [6] p.8 for a definition of dynamical system.

of the waste, and the amount of waste each agent has collected. A state s is therefore characterized by three functions,

$$\begin{aligned}\pi &: \Omega \longrightarrow \mathbb{N}^2 \text{ such that } \pi(\omega_1) \neq \pi(\omega_2) \text{ if } \omega_1 \neq \omega_2. \\ \gamma &: \mathbb{N}^2 \longrightarrow \text{Re}. \\ \sigma &: \Omega \longrightarrow \text{Re}.\end{aligned}$$

π assigns a position to each of the agents, γ assigns the amount of waste to each position, and σ assigns the amount of waste each agent has collected. To denote points in \mathbb{N}^2 , bold face letters $\mathbf{x}, \mathbf{y}, \dots$ will be used. If $\mathbf{x} = \langle x_1, x_2 \rangle \in \mathbb{N}^2$ then $\mathbf{x}^{(1)} = x_1$ and $\mathbf{x}^{(2)} = x_2$. If the state s is represented by the ordered triple of π, γ and σ , i.e. $s = \langle \pi, \gamma, \sigma \rangle$, then $s^{(1)} = \pi$, $s^{(2)} = \gamma$, and $s^{(3)} = \sigma$. Let S be the state space of the system. A state s can be an initial state if $s^{(3)}(\omega) = 0$, for all $\omega \in \Omega$.

3.2.2 Actions and Utility

Each agent has a repertoire of four different actions, viz. *going east*, *going south*, *going west*, and *going north*. For simplicity, we denote the actions with their point of compass: *east*, *south*, *west*, and *north*. Now, *east* is defined as: $east(\omega, s) = \langle \pi, \gamma, \sigma \rangle$ where

1. $\pi(\omega) = \langle (s^{(1)}(\omega))^{(1)} + 1, (s^{(2)}(\omega))^{(2)} \rangle$ and $\pi(\varkappa) = s^{(1)}(\varkappa)$, if $\varkappa \neq \omega$,
2. $\gamma(x, y) = 0$ if $\pi(\omega) = \langle x, y \rangle$ and $\gamma(x, y) = s^{(2)}(x, y)$, if $\pi(\omega) \neq \langle x, y \rangle$,
3. $\sigma(\omega) = s^{(3)}(\omega) + s^{(2)}(\pi(\omega))$.

The other actions are defined analogously. The ‘feasibility function’ \mathcal{A} is such that an action is feasible as long as the performing of the action leads to a new state within the boundaries of the grid.

The preference structure is determined by a utility function

$U_\omega : S \longrightarrow \text{Re}$ for every agent ω : $\Pi(\omega, s) = \langle \mathcal{A}(\omega, s), \succsim_{\omega, s} \rangle$, such that $a \succsim_{\omega, s} b$ iff $U_\omega(a(\omega, s)) \geq U_\omega(b(\omega, s))$.

The utility function for an agent can be defined in many ways and we will leave open the exact definition of it. The norm-regulated DALMAS can have its choice-set function Γ defined as:

$$\Gamma(\omega, s) = \{x \in \Delta(\omega, s) : \text{for all } y \in \Delta(\omega, s), x \succsim_{\omega, s} y\}.$$

4 Conditions and Normative Systems for a Dalmas

4.1 Preamble

In this section, the building blocks of norms which can regulate a DALMAS will be described. The grounds and consequences consist of conditions on agents, true or false in a situation. This kind of conditions are called sit-conditions. As its core, a sit-condition has a state-condition, i.e. a condition true or false in a state. We obtain sit-conditions from state-conditions by the application of the operators M, T_1, \dots, T_7 . Suppose c is a ν -ary state-condition. Then $c(\omega_1, \dots, \omega_k)$ is true or false in a state s and Mc and $T_i c$ are $\nu+1$ -ary sit-conditions. $Mc(\omega_1, \dots, \omega_\nu, \omega_{\nu+1})$ is defined as being true in situation $\langle \omega, s \rangle$ if and only if $c(\omega_1, \dots, \omega_k)$ is true in s

and it is $\omega_{\nu+1}$'s turn to draw, i.e. $\omega_{\nu+1} = \omega$. $T_i c(\omega_1, \dots, \omega_\nu, \omega_{\nu+1})$ is true in $\langle \omega, s \rangle$ if and only if \mathbf{T}_i is the normative position for $\omega_{\nu+1}$ with respect to $c(\omega_1, \dots, \omega_\nu)$ being true in the state that will be the result of the action taken in $\langle \omega, s \rangle$, where \mathbf{T}_i is the i th type of the one agent positions. Elementary norms for a DALMAS are ordered pairs of the form $\langle Mc, T_i d \rangle$ where c and d are state-conditions, Mc is the ground, and $T_i d$ the consequence. In this essay, we will focus on elementary norms.

The rest of this section is devoted to a detailed development of the ideas just presented.

4.2 State-Conditions and Sit-Conditions

The idea behind the definition of a DALMAS is that its behaviour will be regulated by a normative system and that the normative system will be represented by a ground-consequence system $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ where \mathcal{B}_1 and \mathcal{B}_2 are condition implication structures. The condition of interests in connection with a DALMAS is a little more complicated than those discussed in section 2 and it is pertinent to distinguish between two different kinds of conditions, viz.

1. condition on agents in a state, abbreviated just 'state-condition'
2. condition on agents in a situation, abbreviated just 'sit-condition'.

A ν -ary state-condition c is true or false of ν agents $\omega_1, \dots, \omega_\nu$ in a state s , which will be written $c(\omega_1, \dots, \omega_\nu; s)$. Note the use of the semicolon to separate the state from the agents. A ν -ary sit-condition d is true or false of ν agents $\omega_1, \dots, \omega_\nu$ in a state $\langle \omega, s \rangle$, which will be written $d(\omega_1, \dots, \omega_\nu; \omega, s)$.

A normative system regulating a DALMAS is a ground-consequence system $\langle \mathcal{B}_1, \mathcal{B}_2, \mathcal{J} \rangle$ where \mathcal{B}_1 and \mathcal{B}_2 are condition implication structures consisting of certain kinds of sit-conditions. These sit-conditions are the result of applying certain operations to state-conditions. We discuss this in next subsection.

4.3 The Move-Operator

Let us now introduce the move-operator M which transforms a ν -ary state-condition to a ν -ary sit-condition.

Definition. M is an operator on state-conditions such that if c is a ν -ary state-condition then Mc is a ν -ary sit-condition and $Mc(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $\omega_{\nu+1} = \omega$ and $c(\omega_1, \dots, \omega_\nu; s)$. ■

Note that $Mc(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ means that $\omega_{\nu+1}$ is to move in state $\langle \omega, s \rangle$ (since $\omega_{\nu+1} = \omega$), and $c(\omega_1, \dots, \omega_\nu; s)$.

If $\langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering where B consists of state-conditions. Define $B_M = \{Mb : b \in B\}$. We can now define a unary operation $'_M$ on B_M in the following way: $(Mc)'_M = M(c')$.

Note that $(Mc)'_M(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $M(c')(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $\omega_{\nu+1} = \omega$ and $c'(\omega_1, \dots, \omega_\nu; s)$. Define a binary relation \wedge_M on B_M in the following way: $Mb \wedge_M Mc = M(b \wedge c)$.

Note that $(Mb \wedge_M Mc)(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff
 $M(b \wedge c)(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff
 $[\omega_{\nu+1} = \omega \text{ and } (b \wedge c)(\omega_1, \dots, \omega_\nu; s)]$ iff
 $[\omega_{\nu+1} = \omega \text{ and } b(\omega_1, \dots, \omega_\nu; s) \text{ and } c(\omega_1, \dots, \omega_\nu; s)]$ iff
 $[(Mb)(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s) \text{ and } (Mc)(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)]$.

Let us further define a binary relation R_M on B in the following way: bRc . It is easy to see that M is an isomorphism from $\langle B, \wedge, ', R \rangle$ to $\langle B_M, \wedge_M, 'M, R_M \rangle$ which thus also is a Boolean quasiordering. We say that $\langle B_M, \wedge_M, 'M, R_M \rangle$ is the m -cis over $\langle B, \wedge, ', R \rangle$.

4.4 The Type-Operators

Type-operators (see [21]) can be applied to state-conditions, resulting in sit-conditions.

Definition. For $i, 1 \leq i \leq 7$, T_i is an operator on state-conditions such that if c is a ν -ary state-condition then $T_i c$ is a ν -ary sit-condition and $T_i c(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $\mathbf{T}_i(\omega_{\nu+1}, c(\omega_1, \dots, \omega_\nu; s^+))$ where \mathbf{T}_i is the i th type of the one agent positions and s^+ is the state which will be the result of the action taken by ω in state s . ■

The meaning of \mathbf{T}_i is discussed in more detail in section 5.

Suppose that $\langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering where B consists of state-conditions, define $B_T = \{T_i b : b \in B \text{ and } 1 \leq i \leq 7\}$. B_T is the set of *normative positions* over B . If $T_i b, T_j c \in B_T$, then define

- (1) $(T_i b \wedge_T T_j c)(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $T_i b(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ and $T_j c(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$,
- (2) $(T_i b)'_T(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$ iff $\neg T_i b(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$.

Define B_T^* recursively as follows:

- (1) $B_T \subseteq B_T^*$
- (2) If $p, q \in B_T^*$ then $p'_T \in B_T^*$ and $p \wedge_T q \in B_T^*$
- (3) The only members of B_T^* are those resulting from a finite number of applications of (1) and (2).

Definition. A *normative-position-cis* (*np-cis*) over a cis $\langle B, \wedge, ', R \rangle$ is a Boolean quasi-ordering $\langle B_T^*, \wedge_T, 'T, R_T \rangle$ with \top_T as unit element, \perp_T as zero element and where Q_T is the indifference part of R_T such that the following requirements are satisfied. For any $c, d \in B$ it holds that:

- (1) if $i \neq j$, then $(T_i d \wedge_T T_j d)R_T \perp_T$ (for $i, j \in \{1, \dots, 7\}$),
- (2) $\top_T R_T (T_1 d \vee_T \dots \vee_T T_7 d)$,
- (3) $T_1 d Q_T T_1(d')$, $T_3 d Q_T T_3(d')$, $T_6 d Q_T T_6(d')$, $T_2 d Q_T T_4(d')$, $T_5 d Q_T T_7(d')$
- (4) if $c = d$ then $T_i c Q_T T_i d$ (for $i, j \in \{1, \dots, 7\}$),
- (5) $T_i(\top)Q_T \perp_T$ if $i = 1, 3, 4, 7$,
- (6) $T_i(\perp)Q_T \perp_T$ if $i = 1, 2, 3, 5$. ■

The reason for (3) is a kind of 'symmetry principle' (see section 2.3), and the justification for (4) is a logical postulate. The notion of an np-cis was introduced in [21].

4.5 A Norm-Regulated Dalmas

Definition. A *normative system* \mathcal{N} for a DALMAS \mathfrak{D} is a gc-system $\langle \mathcal{C}_M, \mathcal{D}_T, \mathcal{J} \rangle$ such that $\mathcal{C}_M = \langle C_M, \wedge_{M,M}, R_M \rangle$ is the *m-cis* over $\langle C, \wedge, R_C \rangle$ and $\mathcal{D}_T = \langle D_T^*, \wedge_{T,T}, R_T \rangle$ is an *np-cis* over $\langle D, \wedge, R_D \rangle$ where C and D are sets of state-conditions for \mathfrak{D} . ■

In the sequel, we will omit the M and T subscripts in \wedge_M and \wedge'_M , and in \wedge_T and \wedge'_T .

If $\langle \mathcal{C}_M, \mathcal{D}_T, \mathcal{J} \rangle$ is a normative system, then it is joining \mathcal{C}_M and \mathcal{D}_T . If $\langle x, y \rangle \in \mathcal{J}$ then $\langle x, y \rangle$ is a *norm* in \mathcal{N} and x is *the ground* and y is *the consequence* of that norm. Note that a norm in \mathcal{N} is a correlation of normative consequences to descriptive conditions. If $\langle x, y \rangle$ is a norm in \mathcal{N} and $y \in D_T$ then $\langle x, y \rangle$ is an *elementary norm*. An elementary norm in \mathcal{N} is an ordered pair $\langle Mc, T_id \rangle$ where $c \in C$ and $d \in D$ and the intended interpretation of it is a sentence of the following form:

$$\forall \omega_1, \dots, \omega_\nu, \omega \in \Omega : \forall s \in S : Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \rightarrow T_id(\omega_1, \dots, \omega_\nu, \omega; \omega, s^+)$$

or somewhat more generally:

$$\forall \omega_1, \dots, \omega_\varphi, \omega \in \Omega : \forall s \in S : Mc(\omega_1, \dots, \omega_\mu, \omega; \omega, s) \rightarrow T_id(\omega_1, \dots, \omega_\nu, \omega; \omega, s^+)$$

where $\varphi = \max\{\mu, \nu\}$ and s^+ is the state which will be the result of the action taken by ω in state s . In both cases the norm above is represented as the ordered pair $\langle Mc, T_id \rangle$.

Definition. A *norm-regulated DALMAS* is a system $[\mathfrak{D}, \mathcal{N}]$ where \mathfrak{D} is a DALMAS and \mathcal{N} is a normative system for \mathfrak{D} . ■

The idea behind a norm-regulated DALMAS is that the deontic structure operator Δ is defined in terms of \mathcal{N} in the sense that what is permissible to do in a situation is determined by the normative system. We will return to this in section 5.

4.6 Norms for the Waste Collectors

Essential to the behaviour of the collectors is how close they are to each other. To be able to talk about the nearness of the collectors it is convenient to infer the notion of an n -surrounding.

Definition. The *n-surrounding* of the point $\langle x, y \rangle \in \mathbb{N}^2$ is the set $\{\langle z, u \rangle : |x - z| \leq n \text{ and } |y - u| \leq n\}$, and denoted by $Surr_n(x, y)$. ■

We can now define a family of state-conditions expressing overlapping surroundings:

Definition. Lap_j is the state-condition such that $Lap_j(\omega_1, \omega_2; s)$ iff $Surr_j(s^{(1)}(\omega_1)) \cap Surr_j(s^{(1)}(\omega_2))$ contains exactly j elements. ■

If we apply the move operator and the type operators, we obtain sit-conditions from the state-conditions Lap_j , i.e. we get $MLap_j$ and T_iLap_j . We also need the non-identity condition d , defined by $d(\omega_1, \omega_2)$ iff $\omega_1 \neq \omega_2$, and the sit-condition Md .

We can use the introduced terminology to express the norms for the waste collectors presented in subsection 2.1. The result is the following set of ordered pairs of sit-conditions. For legibility reasons, we use $-$ instead of \neg for negation.

1. $\langle MLap_0, T_4Lap_2 \vee T_6Lap_2 \vee T_7Lap_2 \rangle$
2. $\langle MLap_0, T_4Lap_3 \vee T_6Lap_3 \vee T_7Lap_3 \rangle$
3. $\langle MLap_0, T_1Lap_0 \vee T_2Lap_0 \vee T_3Lap_0 \vee T_5Lap_0 \rangle$
4. $\langle MLap_0, T_1Lap_1 \vee T_2Lap_1 \vee T_3Lap_1 \vee T_5Lap_1 \rangle$
5. For all $i, 0 \leq j \leq 9$, $\langle MLap_1, T_1Lap_j \vee T_2Lap_j \vee T_3Lap_j \vee T_5Lap_j \rangle$
6. For all $i, 0 \leq j \leq 9$, $\langle MLap_2, T_1Lap_j \vee T_2Lap_j \vee T_3Lap_j \vee T_5Lap_j \rangle$
7. $\langle M(-Lap_4 \wedge -Lap_6 \wedge -Lap_9), T_7Lap_6 \rangle$
8. $\langle MLap_4, T_7Lap_3 \rangle$
9. $\langle MLap_6, T_5(Lap_4 \vee Lap_6 \vee Lap_9) \rangle$
10. $\langle Md, T_7Lap_9 \rangle$
11. $\langle M\top, T_1Lap_0 \vee T_2Lap_0 \vee T_3Lap_0 \vee T_5Lap_0 \rangle$

The correspondence between the norms $(n_1) - (n_8)$ in section 2 and the norms above is the following:

- (n_1) is represented as (1) and (2),
- (n_2) is represented as (3) and (4),
- (n_3) is represented as (5) and (6),
- (n_4) is represented as (7),
- (n_5) is represented as (8),
- (n_6) is represented as (9),
- (n_7) is represented as (10),
- (n_8) is represented as (11).

Note that among the norms (1)-(11), only (7)-(10) are elementary.

5 From Norms to Actions

In this section, a scheme for how normative positions will restrict the set of acts that the agents are permitted to choose from is presented. The idea behind a norm-regulated DALMAS is that the deontic structure operator Δ is defined in terms of \mathcal{N} in the sense that what is permissible to do in a situation is determined by the normative system. More formally, let us suppose that the system is in situation $\langle \omega, s \rangle$ and that $\langle Mc, T_i d \rangle$ is a norm in \mathcal{N} . Suppose further that c and d are ν -ary and $c(\omega_1, \dots, \omega_\nu; s)$. Hence, $Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$. From $Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ together with the norm $\langle Mc, T_i d \rangle$ follows that $T_i d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$. An important question now is what restrictions on the set of feasible acts follow from $T_i d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$, i.e. what is prohibited by $T_i d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$? Let us see what holds when $i = 2$. The intended interpretation of T_2 is given by

$$\begin{aligned}
& T_2d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \text{ iff } \mathbf{T}_2(\omega, d(\omega_1, \dots, \omega_\nu; s^+)) \text{ iff} \\
& \text{MayDo}(\omega, d(\omega_1, \dots, \omega_\nu; s^+)) \ \& \ \text{MayPass}(\omega, d(\omega_1, \dots, \omega_\nu; s^+)) \ \& \\
& \neg\text{MayDo}(\omega, \neg d(\omega_1, \dots, \omega_\nu; s^+)).
\end{aligned}$$

Only the third conjunction above results in a prohibition: in the situation $\langle \omega, s \rangle$, ω may not see to it that not $d(\omega_1, \dots, \omega_\nu)$ will be the case in the next state. What does this mean? If in state s $d(\omega_1, \dots, \omega_\nu)$ is true but in state s^+ , which is the result of ω 's performing action a in situation s , $d(\omega_1, \dots, \omega_\nu)$ is not true, then ω has seen to it that not $d(\omega_1, \dots, \omega_\nu)$, by doing a . Since ω may not see to it that $d(\omega_1, \dots, \omega_\nu)$, it follows that a is prohibited.

Let us now consider the case $i = 3$. According to the intended interpretation of T_3d it holds that

$$\begin{aligned}
& T_3d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \text{ iff} \\
& \text{MayDo}(\omega, d(\omega_1, \dots, \omega_\nu; s^+)) \ \& \ \neg \text{MayPass}(\omega, d(\omega_1, \dots, \omega_\nu; s^+)) \ \& \\
& \text{MayDo}(\omega, \neg d(\omega_1, \dots, \omega_\nu; s^+)).
\end{aligned}$$

If $d(\omega_1, \dots, \omega_\nu)$ is true in s and $d(\omega_1, \dots, \omega_\nu)$ is true in s^+ , then $\text{Pass}(\omega, d(\omega_1, \dots, \omega_\nu; s^+))$. If $\neg d(\omega_1, \dots, \omega_\nu)$ is true in s and $\neg d(\omega_1, \dots, \omega_\nu)$ is true in s^+ then $\text{Pass}(\omega, d(\omega_1, \dots, \omega_\nu; s^+))$. Therefore,

$\text{Pass}(\omega, d(\omega_1, \dots, \omega_\nu; s))$ iff $[d(\omega_1, \dots, \omega_\nu)$ is true in s iff $d(\omega_1, \dots, \omega_\nu)$ is true in $s^+]$. Hence, if ω may not be passive with regard to $d(\omega_1, \dots, \omega_\nu)$ in $\langle \omega, s \rangle$, then $[d(\omega_1, \dots, \omega_\nu; s) \text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s))]$ implies that a is prohibited.

The other type-operators T_i can be analyzed in an analogous way. As the result of such an analysis, we suggest the following stipulations.¹¹

1. From $T_1d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows no restriction on the acts.
2. From $T_2d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if $d(\omega_1, \dots, \omega_\nu; s)$ and $\neg d(\omega_1, \dots, \omega_\nu; a(\omega, s))$ then $\text{Prohibited}_{\omega, s}(a)$.
3. From $T_3d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if $[d(\omega_1, \dots, \omega_\nu; s) \text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s))]$ then $\text{Prohibited}_{\omega, s}(a)$.
4. From $T_4d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if $\neg d(\omega_1, \dots, \omega_\nu; s)$ and $d(\omega_1, \dots, \omega_\nu; a(\omega, s))$ then $\text{Prohibited}_{\omega, s}(a)$.
5. From $T_5d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if $\neg d(\omega_1, \dots, \omega_\nu; a(\omega, s))$ then $\text{Prohibited}_{\omega, s}(a)$.
6. From $T_6d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if not $[d(\omega_1, \dots, \omega_\nu; s) \text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s))]$ then $\text{Prohibited}_{\omega, s}(a)$.
7. From $T_7d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ follows that if $d(\omega_1, \dots, \omega_\nu; a(\omega, s))$ then $\text{Prohibited}_{\omega, s}(a)$.

These stipulations can now be used to define the deontic structure-operator Δ . One possibility is to let $\Delta(\omega, s)$ be the set of feasible acts a that are not eliminated as $\text{Prohibited}_{\omega, s}(a)$ according to the rules 1-7 above, where $\text{Prohibited}_{\omega, s}(a)$ is equivalent to $\neg\text{Permissible}_{\omega, s}(a)$.¹² Hence, $\Delta(\omega, s) = \{\text{Permissible}_{\omega, s}(a) : a \in A\}$.

¹¹Note that this explication of $T_1 - T_7$ can easily be transformed to an explication of the one agent types \mathbf{T}_1 - \mathbf{T}_7 within the framework of predicate logic.

¹²The mentioned connection between Prohibited and Permissible is naturally not the only possible.

Note that in the outset all feasible acts are permissible, i.e. for all $a \in A$, $\text{Permissible}_{\omega,s}(a)$. The basic idea is now that we eliminate elements from the set of permissible acts for ω in s using the norms and sentences expressing what holds for the agents with respect to grounds in the norms. To facilitate the presentation it is convenient to introduce the following six operators on state-conditions:

$$\begin{aligned}
E_2^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff } [d(\omega_1, \dots, \omega_\nu; s) \text{ and } \neg d(\omega_1, \dots, \omega_\nu; a(\omega, s))] \\
E_3^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff } [d(\omega_1, \dots, \omega_\nu; s) \text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s))] \\
E_4^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff } [\neg d(\omega_1, \dots, \omega_\nu; s) \text{ and } d(\omega_1, \dots, \omega_\nu; a(\omega, s))] \\
E_5^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff } [\neg d(\omega_1, \dots, \omega_\nu; a(\omega, s))] \\
E_6^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff not } [d(\omega_1, \dots, \omega_\nu; s) \text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s))] \\
E_7^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\text{ iff } d(\omega_1, \dots, \omega_\nu; a(\omega, s)).
\end{aligned}$$

Note that for all i , $2 \leq i \leq 7$, $(T_i d \wedge E_i^a d)(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ implies that $\text{Prohibited}_{\omega,s}(a)$.

In situation $\langle \omega, s \rangle$, the action a is prohibited if there are $c \in C$ and $d \in D$ such that $\langle Mc, T_i d \rangle$ is a norm for some i , $2 \leq i \leq 7$, and there are $\omega_1, \omega_2, \dots, \omega_\nu \in \Omega$ such that $Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ and $E_i^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$. The essential part of the procedure is the following inference:

$$\frac{\frac{Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \ \& \ \langle Mc, T_i d \rangle}{T_i d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)} \quad E_i^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)}{\text{Prohibited}_{\omega,s}(a)}$$

Now we can for the normative system $\mathcal{N} = \langle C_M, \mathcal{D}_T, \mathcal{J} \rangle$ define $\text{Prohibited}_{\omega,s}$ more explicitly as:

$$\begin{aligned}
\text{Prohibited}_{\omega,s} &= \{a \in \mathcal{A}(\omega, s) \mid \exists c \in C : \exists d \in D : \exists i, 2 \leq i \leq 7 : \\
&\quad \langle Mc, T_i d \rangle \in \mathcal{J} \ \& \ \exists \omega_1, \dots, \omega_\nu \in \Omega : Mc(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \ \& \\
&\quad E_i^a d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)\}.
\end{aligned}$$

Thus, $\Delta(\omega, s)$ can be defined as $\Delta(\omega, s) = \mathcal{A}(\omega, s) \setminus \text{Prohibited}_{\omega,s}$, i.e. $\Delta(\omega, s) = \{a \in \mathcal{A}(\omega, s) \mid a \notin \text{Prohibited}_{\omega,s}\}$.

For a simple DALMAS it is a small step from $\Delta(\omega, s)$ to $\Gamma(\omega, s)$. If \mathfrak{D} is a simple DALMAS, then $\Gamma(\omega, s) = \Pi(\omega, s) / \Delta(\omega, s)$, i.e. $\Gamma(\omega, s) = \langle \Delta(\omega, s), \succsim' \rangle$, where \succsim' is the restriction of \succsim to $\Delta(\omega, s)$.

In the definition of $\text{Prohibited}_{\omega,s}$ above, we use only norms that are elementary. Whether this implies substantial limitations will here be left as an open problem. Consider for example the norm $\langle Mc, t \rangle$ where $t \in D_T^*$. The problem is under what state condition e the consequence t implies a prohibition of an act. We do not deal with this problem, but just point out that

$$\begin{aligned}
(T_{i_1} d_1 \vee T_{i_2} d_2)(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\longrightarrow \\
&[(E_{i_1}^a d_1 \wedge E_{i_2}^a d_2)(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \longrightarrow \text{Prohibited}_{\omega,s}(a)]. \\
(T_{i_1} d_1 \wedge T_{i_2} d_2)(\omega_1, \dots, \omega_\nu, \omega; \omega, s) &\longrightarrow \\
&[(E_{i_1}^a d_1 \vee E_{i_2}^a d_2)(\omega_1, \dots, \omega_\nu, \omega; \omega, s) \longrightarrow \text{Prohibited}_{\omega,s}(a)].
\end{aligned}$$

Another way of dealing with $\langle Mc, t \rangle$ is to try and show that all prohibitions following from it also follow from elementary (and eventually minimal) norms.

6 Conclusion and Discussion

The aim of this study is to present a theory of how norms can be used to regulate the behaviour of multi-agent systems on the assumption that the role of norms is to define the *Spielraum* for an agent. The theory can be summarized as follows. The norms for a MAS are regarded as belonging to a normative system and such a system is represented algebraically as a ground-consequence system containing a Boolean quasi-ordering of grounds and a Boolean quasi-ordering of consequences. The norms are joinings from grounds to consequences, and the specific normative content of a normative system is given by the set of minimal norms. The consequences are expressed using operators on conditions corresponding to the Kanger-Lindahl types of one-agent positions. An important step in the theory construction is the specification under what circumstances the sentence $T_i d(\omega_1, \dots, \omega_\nu, \omega; \omega, s)$ implies that an action a is prohibited for the agent ω in the state s (see section 5). An action a is regarded as a function, which is the usual way of representing an action in decision theory, and d is a ν -ary condition on agents, true or false in the situation s . An abstract architecture based on the theory of norm-regulation of behaviour is defined, and a MAS having this architecture is called a norm-regulated DALMAS. The system of waste collectors is an example of a such a norm-regulated DALMAS.

An important tool in the present study is the characterization of abstract architectures by the definitions of set-theoretical predicates. Among the abstract architectures defined in this way, the most important one is a norm-regulated DALMAS, as exemplified in the previous section. This is just the first step towards a theory of architectures for MAS that restricts the behaviour of the system using norms. The theory can be developed by defining a number of set-theoretical predicates that are specifications of the predicate DALMAS, and we can obtain a hierarchy of predicates with DALMAS as its root.¹³

There are many refinements and extensions to be made before it is possible to state the significance of the theory. For instance, the minimal norms represent the specific normative content of a normative system. It is therefore a reasonable question, whether, in the definition of $\text{Prohibited}_{\omega,s}$, it is sufficient to regard only minimal norms. Note that the stipulations expressing under what conditions $T_i d$ implies that the act a is prohibited are independent of the structure over the consequences. The latter, however, is relevant to determining the minimal norms.

Moreover, in the definition of $\text{Prohibited}_{\omega,s}$ above, we use only norms that are elementary. Whether this implies substantial limitations ought to be investigated. Consider the norm $\langle Mc, t \rangle$ where $t \in D_T^*$. Under what state condition e does the consequence t imply a prohibition of an act? Is it possible to prove that all prohibitions following from it also follow from elementary (and minimal) norms?

We have only considered norms of a rather special kind. The descriptive conditions are conditions on agents. Let us call them ‘agent norms’.

¹³One set-theoretical predicate \mathfrak{P}_2 is a specification of another \mathfrak{P}_1 if the following holds: if \mathfrak{S}_2 is a \mathfrak{P}_2 there is a sub-sequence \mathfrak{S}_1 of \mathfrak{S}_2 that is a \mathfrak{P}_1 . The relation ‘to be a specification of’ is a partial order.

An example of an elementary agent norm is $\langle Mc, T_i d \rangle$, where c and d are conditions on agents, true or false in a state. However, it is easy to generalize so that c and d can be conditions not just on agents but on a set U containing agents and objects of other kinds, for example physical objects. It seems to be straightforward to change our definitions so that this alternative will be accommodated.

Norms of a radically different character than agent norms are norms determining which are the agent norms for a MAS. Such norms can for example give competence to some agents to functions as a kind of ‘legislators’ and change the agent norms [29].

In this essay, we have tried to show how theories of normative positions and normative systems can be used in constructing an architecture for norm-regulated MAS. However, we have only used a small portion of the Kanger-Lindahl theory of normative positions, namely the theory of one-agent types. According to the intended interpretation of the type operators in section 4,

$$T_i c(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}; \omega, s)$$

means that $\omega_{\nu+1}$ has the normative positions of type T_i with respect to the state of affairs that $c(\omega_1, \dots, \omega_\nu; s^+)$. Here T_i is a one-agent type-operator. A two-agent type-operator W_i applied to c will give a condition $W_i c$ such that

$$W_i c(\omega_1, \dots, \omega_\nu, \omega_{\nu+1}, \omega_{\nu+2}; \omega, s)$$

means that the party $\omega_{\nu+1}$ has the normative positions of type W_i versus the counter party $\omega_{\nu+2}$ with respect to the state of affairs that $c(\omega_1, \dots, \omega_\nu; \omega, s^+)$.

¹⁴The use of two-agent type-operators in norm-regulated DALMAS will increase the strength and flexibility of the DALMAS hierarchy of abstract architectures.

In a norm-regulated DALMAS, the normative system plays the role of constitutive rules, and it is not possible for the agents to break the norms. In spite of this, it is reasonable to distinguish between ‘may’ and ‘can’. What an agent ω *can* do in a situation s is given by $\mathcal{A}(\omega, s)$ and what it *may* do is determined by $\Delta(\omega, s)$. Note also that the formal framework for representing norms can be used for different kinds of norms, even regulative norms. For example, an agent can choose between obeying a regulative law or breaking it and thereby giving one or several other agents the right (but possibly not the duty) to punish it, either immediately or when it is its turn to move. This can easily be expressed using normative positions.

DALMAS and deterministic DALMAS are intended to function as wide frames for studying norm-regulations of MAS and thus to be weak abstract architectures. Are they weak enough or are they excluding something essential? As has been pointed out earlier, for instance, a DALMAS has global states and global dynamics. This fact notwithstanding, the formal definition of a DALMAS is flexible enough to incorporate a kind of history dependence. One way of doing this is to introduce a set P of primary states (phases) such that $s \in S$ iff $s = \langle p, \langle p_1, \dots, p_\nu \rangle \rangle$, where $p, p_1, \dots, p_\nu \in P$.

¹⁴See [17], chapter 4.

The intended interpretation is that when \mathfrak{D} is in s then \mathfrak{D} is in the primary state p and its history is $\langle p_1, \dots, p_\nu \rangle$, i.e. the passed primary states of \mathfrak{D} is given by $\langle p_1, \dots, p_\nu \rangle$. Note that if $s_1 = \langle p, \sigma_1 \rangle$ and $s_2 = \langle p, \sigma_2 \rangle$ where $\sigma_1 \neq \sigma_2$, then it is possible that $a(\omega, s_1) \neq a(\omega, s_2)$, so the result of ω performing act a in the primary state p is dependent on the passed primary states of \mathfrak{D} .

Another way to obtain history dependence is to use the definition of a run (see section 3.1). With some modifications a run can be regarded as a state consisting of situations based on primary states. In this case we have dependence not just on passed primary states but on passed primary situations. If the DALMAS in question is deterministic, passed actions also determine passed primary situations, but the opposite does not hold. Therefore, we can distinguish between dependence on previous primary states, dependence on previous primary situations, and dependence on previous actions.

Suppose that \mathfrak{D} is a DALMAS, in which the agents cooperate to solve a problem. Which normative system will lead to the most effective behavior of the system? It is desirable that \mathfrak{D} itself could determine the optimal normative system for the task in question. Given a set of grounds and a set of consequences, which together constitute the vocabulary of the system, \mathfrak{D} can test all possible sets of minimal norms. If there is a function for evaluating the result of a run of \mathfrak{D} , then different normative systems can be compared and the best system can be chosen. A change of vocabulary corresponds to a mutation in the normative system, which can lead to dramatic changes of efficiency. Note that, in principle, the evaluation function can be very complicated, for example it can be multi-dimensional and based on ideas of fairness. This seems to imply that we can treat multi-agent systems as consisting of cooperative agents even if they do not have what is usually called ‘a common interest’ and cooperate only in the sense that they want to avoid destructive conflicts.

Acknowledgements

For this research, Jan Odelstad was funded by HSNR (project F1113/97) and the KK-foundation. Magnus Boman gratefully acknowledges Vinnova (Accessible Autonomous Software). The authors would like to express their gratitude to Lars Lindahl for his important input. Finally, the editors and the anonymous referees showed great patience and insight, for which we thank them.

References

- [1] Alchourron, C. E. and E. Bulygin: 1971, *Normative Systems*. Springer-Verlag.
- [2] Bell, J. L. and A. B. Slomson: 1969, *Models and Ultraproducts: An Introduction*. Amsterdam: North-Holland.
- [3] Boman, M.: 1999, ‘Norms in Artificial Decision Making’. *Artificial Intelligence and Law* **7**(1), 17–35.

- [4] Boman, M., P. Davidsson, and H. Younes: 1999, ‘Artificial Decision Making under Uncertainty in Intelligent Buildings’. In: E. Pagello (ed.): *Intelligent Autonomous Systems 6*. IOS Press, pp. 744–750.
- [5] Boman, M. and E. Holm: 2003, ‘Multi-Agent Systems, Time Geography, and Microsimulations’. In: M.-O. Olsson and G. Sjöstedt (eds.): *Systems Approaches and their Application*. Kluwer Academic, Chapt. 4. In press.
- [6] Broer, H.: 1991, ‘Introduction to Dynamical Systems’. In: H. Broer, F. Dumortier, S. J. van Strien, and F. Takens (eds.): *Structures in Dynamics—Finite Dimensional Deterministic Studies*, Vol. 2 of *Studies in Mathematical Physics*. North-Holland.
- [7] Chang, C. C. and H. J. Keisler: 1973, *Model Theory*. Amsterdam: North-Holland.
- [8] Dignum, F.: 1999, ‘Autonomous Agents with Norms’. *Artificial Intelligence and Law* **7**(1), 69–79.
- [9] Dignum, F., D. Morley, E. A. Sonenberg, and L. Cavedon: 2000, ‘Towards Socially Sophisticated BDI Agents’. In: *ICMAS’2000 Proceedings*. IEEE, pp. 111–118.
- [10] Huberman, B. and N. Glance: 1993, ‘Evolutionary Games and Computer Simulations’. *PNAS* **90**, 7716–7718.
- [11] Jones, A. and M. Sergot: 1993, ‘On the Characterisation of Law and Computer Systems—The Normative Systems Perspective’. In: J.-H. C. Meyer and R. J. Wieringa (eds.): *Deontic Logic in Computer Science—Normative System Specification*. Wiley.
- [12] Jones, A. and M. Sergot: 1996, ‘A Formal Characterisation of Institutionalised Power’. *Journal of the IGPL* **4**(3), 429–445.
- [13] Kanger, S.: 1957, *New Foundations for Ethical Theory*, Part 1. Stockholm University. Technical Report.
- [14] Kanger, S. and H. Kanger: 1966, ‘Rights and Parliamentarism’. *Theoria* **32**, 85–115.
- [15] Krogh, C.: 1995, ‘The Rights of Agents’. In: *Intelligent Agents II*. Springer-Verlag.
- [16] Krogh, C. and H. Herrestad: 1999, ‘Hohfeld in Cyberspace and Other Applications of Normative Reasoning in Agent Technology’. *Artificial Intelligence and Law* **7**(1), 81–96.
- [17] Lindahl, L.: 1977, *Position and Change—A Study in Law and Logic*. Reidel, Dordrecht.
- [18] Lindahl, L.: 1994, ‘Stig Kanger’s Theory of Rights’. In: D. Prawitz, B. Skyrms, and D. Westerståhl (eds.): *Logic, Methodology and Philosophy of Science IX*. Elsevier Science.
- [19] Lindahl, L. and J. Odelstad: 1999, ‘Intermediate Concepts as Couplings of Conceptual Structures’. In: H. Prakken and P. McNamara (eds.): *Norms, Logics and Information Systems—New Studies on Deontic Logic and Computer Science*. IOS Press.

- [20] Lindahl, L. and J. Odelstad: 2000, ‘An Algebraic Analysis of Normative Systems’. *Ratio Juris* **13**, 261–278.
- [21] Lindahl, L. and J. Odelstad: 2004, ‘Normative Positions within an Algebraic Approach to Normative Systems’. *Journal of Applied Logic*. In press.
- [22] Makinson, D.: 1986, ‘On the Formal Representation of Right Relations’. *Journal of Philosophical Logic* **15**, 403–425.
- [23] Odelstad, J.: 2003, ‘An Abstract Architecture for Norm-Regulated Agents’. Paper presented at Promote IT 2003, Visby, Sweden.
- [24] Odelstad, J. and L. Lindahl: 2000, ‘Normative Systems Represented by Boolean Quasi-Orderings’. *Nordic Journal of Philosophical Logic* **5**, 161–174.
- [25] Odelstad, J. and L. Lindahl: 2002, ‘The Role of Connections as Minimal Norms in Normative Systems’. In: T. Bench-Capon, A. Daskalopulu, and R. Winkels (eds.): *Legal Knowledge and Information Systems*. Amsterdam: IOS Press.
- [26] Savage, L. J.: 1972, *The Foundations of Statistics*. Dover, 2nd edition.
- [27] Sergot, M.: 1999, ‘Normative Positions’. In: P. McNamara and H. Prakken (eds.): *Norms, Logics and Information Systems*. Amsterdam: IOS Press, pp. 289–308.
- [28] Sergot, M.: 2001, ‘A Computational Theory of Normative Positions’. *ACM Transactions on Computational Logic* **2**(4), 581–622.
- [29] Spaak, T.: 2003, ‘Norms that Confer Competence’. *Ratio Juris* **16**(1), 89–104.
- [30] Wooldridge, M.: 2002, *An Introduction to Multi-Agent Systems*. Wiley.