Designing Normative Behaviour by the Use of Landmarks

Huib Aldewereld, Davide Grossi, Javier Vázquez-Salceda, and Frank Dignum

Institute of Information and Computing Sciences
Utrecht University, The Netherlands
{huib, davide, javier, dignum}@cs.uu.nl

Abstract. In highly regulated environments, where a set of norms defines accepted behaviour, protocols provide a way to reduce complexity by giving direct, step by step guidelines for behaviour, as long as the protocols comply with the norms. In this work we propose a formal framework to design a protocol from a normative specification. In order to be able to connect (descriptive) norms with (operational) protocols, an intermediate level is created by the use of landmarks.

1 Introduction

In last years there has been an explosion of new approaches, both theoretical and practical, focusing on the use of some kind of normative specification as a flexible way to structure, restrict and/or impose behaviour in Multiagent Systems. In particular, recent developments focus on norm languages, agent-mediated electronic institutions, contracts, protocols and policies. Our work focuses on a normative approach based on the use of Norms in Artificial Institutions. Norms are high-level specifications of acceptable behaviour within a given context. Definitions of norms range from very philosophical, in deontic logic, to precise specifications of protocols in agent-mediated electronic institutions.

One of the questions that arises is how to properly connect the norm specification with the behaviour of the agents. Norms are usually defined in some form of deontic logic [13], in order to express accepted (legal) behaviour through *obligations*, *permissions* and *prohibitions*. However, it is hard to directly connect this kind of norms with the practice as:

1. Norms in Law are formulated in a very abstract way, i.e., the norms are expressed in terms of concepts that are kept vague and ambiguous on purpose.
2. Norms expressed in deontic logic are *declarative*, i.e., they have no *operational* semantics (they express *what* is acceptable, but not *how* to achieve it).
3. As Wooldridge and Ciancarini explain in [17], in those formalisms and agent theories based in *possible worlds*, there is usually no precise connection between the abstract accessibility relations used to characterise an agent's state and any computational model. This makes it difficult to go directly from a formal specification to an implementation in a computational system.

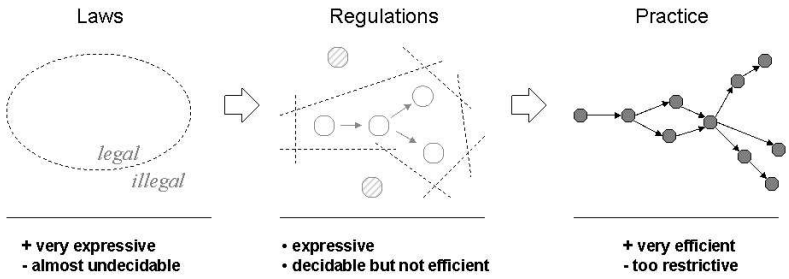

Fig. 1. Comparison between Laws, Regulations and Practice

All these three issues together create a gap between the normative dimension of agent-mediated institutions and their procedural one (first introduced in [4]). Some of our previous work has focused on reducing this gap from different perspectives. In [9] [10] formal tools have been proposed to link abstract normative specifications to more concrete ones (issue 1). In [5] [7] the expressiveness of norms (issue 2) is extended by means of some variations of deontic logic that include conditional and temporal aspects [3]. However, by introducing some sort of temporal or dynamic logic operators, the resulting specification becomes more expressive but computationally too expensive to be used at run-time by agents. We have also explored some of the operational aspects of norms, by focusing on how norms should be operationally implemented in MAS from an institutional perspective [15], including the ontological aspects of norm implementation [9]. Here we try to bring our previous work further, tackling in part issue 3 and proposing a formal approach to describe an explicit bridge between institutional norms and protocols.

Our approach is inspired by how the gap is bridged in human institutions. Human Laws express in a very abstract way wanted (legal) and unwanted (illegal) states of affairs. Although Laws are very expressive, they do not express how to achieve a given state of affairs, and therefore they are very hard to use in practice to, e.g., guide each decision point in a process. In practice more efficient representations are needed, such as protocols or guidelines. In rule-based legal systems (those based in Roman-Germanic Law), *regulations* add an intermediate level between laws and practice, by giving some high-level specifications on some constraints about how things can or cannot be done. These high-level descriptions are therefore interpretations of the law that add some operational constraints to be met by the practice (see figure 1). Using this idea, we introduce an intermediate level between institutional norm specifications and institutional protocols based on *landmarks*.

In this paper we consider norms as specifying deontic constraints at a level that *abstracts* from the procedural aspects of institutions which are instead involved in the design of the protocols of the institution [4]. Additionally, we view norms as specifying (abstract) constraints which have an intrinsic *temporal*

flavour [6]. In particular, we are interested in two types of norms: 1) Norms of the form "*it ought to be the case that ρ is the case before δ happens*", which will be represented by formulas such as $O(\rho \leq \delta)$; and 2) Norms of the form "*it ought never to be the case that ρ*", which will be represented by formulas $F\rho$.

Throughout this paper we will use as an example a simplification of the information sharing problem between Police forces that belong to either a) different geographical regions, or to b) different levels of national security (standard police, secret services, military forces), with national and/or international regulations that highly constrain the amount of information that can be shared between the forces. In our simplified version of the problem, let us suppose that police officers from two different regions have an individual investigation towards a suspect. However, both regions are forced by law to protect their investigation and, therefore, they cannot always ask the other about this suspect because that could compromise their investigation. The problem can be summarised in the following norm:

> "Police regions are obliged to confirm the knowledge of other police regions about suspects (without leaking that information) before exchanging information on this suspect."

From this norm the following issues arise: 1) How can such a norm be linked to a norm-abiding protocol? 2) Can this link be formally described? These are, in a nutshell, the motivating questions of the present paper.

We claim that landmarks can provide a viable bridge between norms and protocols. If norms specify abstract constraints on a temporal structure, then from this normative/temporal specification a landmark pattern can be extracted which can be used as a *yardstick* to evaluate the norm compliance of concrete protocols. In order to tackle the problem, our approach consists of three steps: 1) formalising a conception of institutional norms (tuned on the ideas just presented); 2) extracting landmark patterns (from such a formalisation); and 3) relating landmark patterns to protocols.

The remainder of this paper is organised as follows. In the next section we discuss the framework for using norms, expressed in CTL, to obtain the landmarks which we use to design a protocol. Then in section 3 we show a concrete example using this formal framework. We end the paper with some conclusions.

2 From norms to protocols via landmarks: a framework

2.1 Landmarks

The notion of landmarks has obtained various attention in recent literature about multiagent systems. In [12] landmarks are used in order to specify conversation protocols between agents at an abstract level. They are represented as states and they are structured in a partial order describing, essentially, the respective order in which each landmark should be reached. In [7] and [16] landmarks are used with similar purposes in order to provide abstract specifications of organisational interaction in general. In that work, landmarks are formalised as state

descriptions, and therefore as sets of states (in a modal logic setting). Analogously, these state descriptions are then partially ordered in directed graphs to form landmark structures which are called *landmark patterns*.

No matter how landmarks are represented -as states, or sets of states- their relevance in protocol specification is dictated by the simple observation that several different agents' actions can bring about the same outcome. Once the outcomes of actions are organised in a structured description (i.e. a landmark pattern), it becomes possible to represent families of protocols abstracting from the actual transitions by which each protocol is constituted. Intuitively, a landmark pattern then represents the important steps that any protocol should contain, and the order in which those steps should be performed: "which steps should be performed and in which order".

In this work, we intend to borrow the notion of landmarks and apply it to the domain of eInstitutions. However, to apply the landmark approach to eInstitutions a key refinement is necessary. In domains such as the one concerning information exchange between Police regions, such positive constraints are not always enough. In fact, institutional regulations also express explicit limitation aspects by means of norms of a prohibitive type. Therefore, in the present work we also introduce a notion of *negative landmarks*. Intuitively, negative landmarks mark the states that should not be reached by any protocol. By means of them, it becomes then possible to extend a landmark pattern description to incorporate a reference to "which steps should not be performed".

The formal definition of a landmark pattern we propose is the following one.

Definition 1. (Landmark pattern)
A landmark pattern is a structure $\mathfrak{L} = \langle L^+, L^-, \leq \rangle$ *where* L^+ *and* L^- *are finite sets of landmarks and* $\leq$ *is a partial order on* L^+.

Protocols are treated as state-transition systems, that is, structures composed of states and labelled transitions expressing how one can change between states. This means that actions in protocols are expressed as state-transitions, changing the state of the world/protocol.

Definition 2. (Protocol)
A protocol is a structure $\mathfrak{P} = \langle S, \{R_\alpha\}_{\alpha \in \mathcal{A}} \rangle$ *where:* S *is a non-empty finite set of states containing* s_0 *(the starting state of the protocol) and such that* $S_f \subseteq S$ *with* S_f *is a finite non-empty set (the set of final states of the protocol), and* $\{R_\alpha\}_{\alpha \in \mathcal{A}}$ *is a family of relations indexed by a set of transition labels* $\mathcal{A}$.

The set $\mathcal{A}$ *is inductively defined from a set* A *of atomic labels as follows: 1)* $A \in \mathcal{A}$*; 2) if* $\alpha, \beta \in \mathcal{A}$ *then* $\alpha; \beta$ *and* $\alpha \cup \beta \in \mathcal{A}$*. The family* $\{R_\alpha\}_{\alpha \in \mathcal{A}}$ *is assumed to be closed under the relational algebra operations of sequencing and choice. That is, if* $(s_1, s_2) \in R_\alpha$ *and* $(s_2, s_3) \in R_\beta$ *then* $\alpha; \beta \in \mathcal{A}$*. Analogously, if* $(s_1, s_2) \in R_\alpha$ *and* $(s_1, s_2) \in R_\beta$ *then* $\alpha \cup \beta \in \mathcal{A}$ *and* $(s_1, s_2) \in R_{\alpha \cup \beta}$[1].

We will show how to connect these two definitions and how to exploit the notion of landmark patterns as a useful tool in order to build an intermediate step between the norms specifying the deontic constraints ranging on the institutions and the actual protocols operating the institution itself.

[1] Notice that $\mathfrak{P}$ is then nothing but a frame for propositional dynamic logic [11].

2.2 Computational tree logic

In this section we provide a brief sketch of computational tree logic (CTL), referring to [8] for more detailed discussions.

Well-formed formulas of the language $\mathcal{L}_{\mathrm{CTL}}$ consist of propositional elements combined with $\neg$, $\wedge$ and the temporal operators $E(\varphi U \psi)$ and $A(\varphi U \psi)$, with the following informal reading: $E(\varphi U \psi)$ means that there is a future for which eventually, at some point m the condition ψ will hold, while φ holds from now until then; $A(\varphi U \psi)$ means that for all futures, eventually, at some point m the condition ψ will hold, while φ holds from now until then. Other CTL-operators we use are introduced as abbreviations: $EF\varphi \equiv_{def} E(\top U \varphi)$ and $AG\varphi \equiv_{def} \neg EF\neg\varphi$. With the following informal meaning: $EF\varphi$ means that there exists a future in which φ will eventually hold; $AG\varphi$ means instead that for all possible futures φ holds globally. Standard propositional abbreviations are also assumed.

A CTL model $\mathcal{M} = (S, \mathcal{R}, \pi)$, consists of a non-empty set S of states, an accessibility relation $\mathcal{R}$, and an interpretation function π for propositional atoms. A full path σ in $\mathcal{M}$ is a sequence $\sigma = s_0, s_1, s_2, \ldots$ such that for every $i \geq 0$, s_i is an element of S and $s_i \mathcal{R} s_{i+1}$, and if σ is finite with s_n its final situation, then there is no situation s_{n+1} in S such that $s_n \mathcal{R} s_{n+1}$. We say that the full path σ starts at s if and only if $s_0 = s$. We denote the state s_i of a full path $\sigma = s_0, s_1, s_2, \ldots$ in $\mathcal{M}$ by σ_i. The validity, $\mathcal{M}, s \models \varphi$, of a CTL-formula φ in a world s of a model $\mathcal{M} = (S, \mathcal{R}, \pi)$ is defined as (the propositional connectives are interpreted as usual):

$$
\begin{aligned}
\mathcal{M}, s \models E(\varphi U \psi) \Leftrightarrow\ & \exists \sigma \text{ in } \mathcal{M} \text{ with } \sigma_0 = s, \text{ and } \exists n > 0 \text{ such that:} \\
& (1)\ \mathcal{M}, \sigma_n \models \psi \text{ and} \\
& (2)\ \forall i \text{ with } 0 \leq i \leq n \text{ it holds that } \mathcal{M}, \sigma_i \models \varphi \\
\mathcal{M}, s \models A(\varphi U \psi) \Leftrightarrow\ & \forall \sigma \text{ in } \mathcal{M} \text{ such that } \sigma_0 = s, \text{ it holds that } \exists n > 0 \text{ such that} \\
& (1)\ \mathcal{M}, \sigma_n \models \psi \text{ and} \\
& (2)\ \forall i \text{ with } 0 \leq i \leq n \text{ it holds that } \mathcal{M}, \sigma_i \models \varphi
\end{aligned}
$$

Validity on a CTL model $\mathcal{M}$ is defined as validity in all states of the model. If φ is valid on a CTL model $\mathcal{M}$, we say that $\mathcal{M}$ is a model for φ. General validity of a formula φ is defined as validity on all CTL models. The logic CTL is the set of all general validities of $\mathcal{L}_{\mathrm{CTL}}$ over the class of CTL models.

2.3 A CTL reduction of deontic logic

In this work, we represent norms making use of the CTL reduction approach of deontic logic investigated in [6] [3]. The language $\mathcal{L}_{\mathrm{CTL}}$ is expanded by adding a violation constant of the form $Viol$[2] to the set of propositional atoms. Semantically, the atom $Viol$ works like all other atomic propositions and it intuitively denotes the fact that "a violation (of the relevant regulation) occurs".

[2] For reasoning in a multiagent context we may provide violation constants of the form $Viol(a)$ where $a \in Ag$, and Ag a finite set of agent identifiers.

Definition 3. (Semantics of $O(\rho \leq \delta)$)
Let $\mathcal{M}$ be a CTL model, s a state, and σ a full path starting at s. The modal semantics for formulas $O(\rho \leq \delta)$ is then defined as follows:

$$\begin{aligned}\mathcal{M}, s \models O(\rho \leq \delta) \Leftrightarrow \forall \sigma \text{ with } \sigma_0 = s, \forall j : \\ \text{if } \mathcal{M}, \sigma_j \models \delta \text{ and } \forall i, 0 \leq i \leq j : \mathcal{M}, \sigma_i \models \neg\rho \\ \text{then } \mathcal{M}, \sigma_j \models Viol.\end{aligned}$$

This captures the following intuitive reading: if at some future point δ occurs, and until then ρ has not yet been achieved, a violation occurs at that point. Another way to express this is that what norms do is specify which temporal substructures (i.e. which CTL paths) are norm abiding, i.e., do not contain a violation state. This can be characterised in CTL as:

$$O(\rho \leq \delta) \equiv_{def} \neg E(\neg\rho U(\delta \wedge \neg Viol)).$$

More complex variants of this definition are extensively discussed in [3].

With respect to prohibitive norms we define the following CTL reduction.

Definition 4. (Semantics of $F\rho$)
Let $\mathcal{M}$ be a CTL model, s a state, and σ a full path starting at s. The modal semantics for formulas $F\rho$ is then defined as follows:

$$\mathcal{M}, s \models F\rho \Leftrightarrow \forall \sigma \text{ with } \sigma_0 = s, \forall i : \mathcal{M}, \sigma_i \models \rho \rightarrow Viol.$$

Intuitively, the semantics just says that in all future paths it is globally true that ρ implies a violation. Readers acquainted with deontic logic will recognise that this semantics reflects a straightforward transposition of the Andersonian reduction of deontic logic [2] in a CTL modal setting[3]. Indeed, a CTL characterisation of this reduction is the following one:

$$F\rho \equiv_{def} AG(\rho \rightarrow Viol).$$

2.4 From norms to landmark patterns

Given the semantics of norms presented in the previous section, the operation of extracting landmark patterns from normative specifications amounts to consider the temporal structure characterising the CTL paths in which no violation ever occurs. Technically, this means to explore the CTL models which satisfy the set of norms at issue together with the assertion $AG\neg Viol$ (for all paths, it holds globally that $\neg Viol$). Please note that a general and automated manner for extracting landmarks from a large set of norms is still future work. In this section we give an example to show the intuitions of the idea.

[3] Anderson's reduction consists of interpreting a deontic operator in terms of an alethic one in combination with a violation constant: $O\phi := \Box(\neg\phi \rightarrow Viol)$. Such reduction strategy has the advantage of enabling deontic notions in a simple and intuitive way. However, it suffers the typical shortcomings lying in the use of classical material implication. For a discussion of these issues see [13].

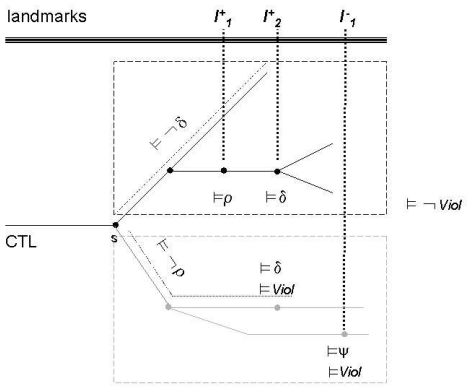

Fig. 2. From norms to landmarks

Let us consider the simple case in which the only norms are $O(\rho \leq \delta)$ and $F\psi$. It is easy to see that the following semantic constraints is obtained:

$$\forall\sigma \text{ with } \sigma_0 = s, \forall j: \text{ either } M, \sigma_j \models \neg\delta \text{ and not } \mathcal{M}, \sigma_j \models \psi$$
$$\text{or } \exists i, 0 \leq i \leq j : \mathcal{M}, \sigma_i \models \rho \text{ and not } \mathcal{M}, \sigma_j \models \psi.$$

As we would intuitively expect, ψ never occurs and either the condition δ also never occurs, or, if it occurs at a certain state, then ρ is the case in some preceding state. In other words, among the paths that abide by $F\psi$, there are two types of paths which abide by $O(\rho \leq \delta)$: the ones in which the condition δ never occurs, and the ones in which the condition does occur after the required state ρ has been reached. Given that we want our protocols to be not just norm-abiding (*safety*), but also goal directed (*liveness*)[4], a trivial landmark pattern for $O(\rho \leq \delta)$ and $F\psi$ would then be the structure $\mathfrak{L} = \langle L^+, L^-, \leq \rangle$ where $L^+ = \{l^+_1, l^+_2\}$, $L^- = \{l^-_1\}$ and $\leq = \{(l^+_1, l^+_2)\}$ and $l^+_1 = \rho$, $l^+_2 = \delta$, $l^-_1 = \psi$; this is expressed in figure 2.

This way of understanding the relation between norms and landmark patterns presupposes the idea that, from one set of norms, many landmark patterns can actually be extracted which are equivalent as far as that set of norms is concerned. Trivially, another landmark pattern for the simple example above can be obtained strengthening the positive landmarks or weakening the negative one.

2.5 From landmark patterns to protocols

Given the landmark structure, we design a protocol which abides by the norms of the domain. In this process the landmarks are considered to be sub-goals that protocols need to fulfil. The idea is then that certain protocol states can

[4] The point is that a "do nothing" protocol is usually norm-compliant. The liveness issue has been discussed in [1].

be linked to the landmark states that were obtained from the norms. For the protocol to be norm-compliant, the linked states of the protocol should satisfy the relational constraints that are included in the landmark structure.

Technically, we have to define a formal relation between definitions 1 and 2.

Definition 5. ($\mathfrak{P}$ compliance with $\mathfrak{L}$)
Given a landmark pattern $\mathfrak{L} = \langle L^+, L^-, \leq \rangle$ and a protocol $\mathfrak{P} = \langle S, \{R_\alpha\}_{\alpha \in \mathcal{A}} \rangle$, we say that $\mathfrak{P}$ complies with *$\mathfrak{L}$ if it is possible to define a relation $\mathcal{R} \subseteq L^+ \cup L^- \times S$ such that:*

1. *the restriction $L^+\rceil\mathcal{R}$ of the domain of $\mathcal{R}$ to L^+ is non-empty and such that: if $(l, s) \in L^+\rceil\mathcal{R}$, then there is an $\alpha \in \mathcal{A}$ such that $(s_0, s) \in R_\alpha$; and there is at least a pair $(l_i, s_i) \in L^+\rceil\mathcal{R}$ where landmark $l_i \in L^+$ and $s_i \in S_f$.*
2. *the restriction $L^-\rceil\mathcal{R}$ of the domain of $\mathcal{R}$ to L^- is either empty, or such that if $(l, s) \in L^-\rceil\mathcal{R}$, then there is no $\alpha \in \mathcal{A}$ such that $(s_0, s) \in \mathcal{R}_\alpha$.*
3. *there is no state $s \in S$ such that $(l_i, s), (l_j, s) \in \mathcal{R}$ with $l_i \in L^+$ and $l_j \in L^-$.*

Condition 1 can be strengthened in order to force an embedding of the landmark pattern on the protocol, we say that $\mathfrak{P}$ is linear compliant with *$\mathfrak{L}$:*

- *the restriction $L^+\rceil\mathcal{R}$ of the domain of $\mathcal{R}$ to L^+ defines an embedding $f : \mathfrak{L} \longrightarrow \mathfrak{P}$. That is to say, that f is a mapping from L^+ to S such that, for all $l_1, l_2 \in L^+$: $l_1 \leq l_2$ iff there exists an $\alpha \in \mathcal{A}$, s.t. $f(l_1) R_\alpha f(l_2)$; and there is at least a pair $(l_i, s_i) \in L^+\rceil\mathcal{R}$ where landmark $l_i \in L^+$ and $s_i \in S_f$.*

Condition 1 says that positive landmarks are related to states in the protocol such that those states are always reachable in the protocol from the starting state and that at least one landmark is related to one of the protocol's final states[5]; condition 2 states that $\mathfrak{P}$ does not contain states which count as negative landmarks and if it contains them they are innocuous since they are not reachable from the starting point; condition 3 states that a state cannot be at the same time linked to a positive and a negative landmark. In case $\mathfrak{P}$ is linearly compliant with $\mathfrak{L}$, the set of positive landmarks is actually mapped on (and not just related to) the protocols. Intuitively, in order for a protocol to embed a landmark pattern, the protocol should behave linearly with respect to the pattern, avoiding branches which require a multiplication of the landmark corresponding states. The example analysed in the following section displays such a protocol.

3 Landmarks in practice

In this section we show how the theory, explained in previous sections, can be used to guide the behaviour of normative multiagent systems. To do so let us return to the example. Let it be the case that the police in region A has an investigation towards a suspect X that operates in region A. A, however, suspects that X is operating in region B as well, and therefore A assumes that B might have an investigation towards X as well. Moreover, as A suspects that X has connections to corrupt police officers it is imperative that A does not simply

[5] This is a way of capturing the liveness condition we touched upon in Section 2.4.

asks B "Do you know anything about X?", since that would expose that X is a suspect in an investigation of A, and thereby jeopardising his investigation.

To ensure the safety of A's investigation, A has to abide to the norms holding for this domain. That would mean that A should be aware of whom he is talking to (not making sure that he asks his questions to B would jeopardise his investigations even more) and that he has to make certain that B knows about X before asking for information about X. Also, by regulation, police regions are not allowed to ask or exchange personal details about persons not being suspected of a criminal offence. The norms that are applicable to this domain are:

1. The identity of police officers should be know to both parties before they begin interacting.
2. Police regions are obliged to confirm the knowledge of other police regions about suspects (without leaking that information) before exchanging information on this suspect.
3. Sharing information about persons who are not under suspicion (of a crime) is forbidden.

By means of the logical formalism described in 2.2 and 2.3 we can translate these norms into the following formulas (we use P_1 and P_2 as variables for police regions, and Y as variable for a person):

1. $O(\mathit{authenticated}(P_1, P_2) \leq \mathit{interacted}(P_1, P_2))$
2. $O(\mathit{confirmed\_know}(P_1, P_2, \mathit{suspect}(Y)) \leq \mathit{exchanged\_info}(P_1, P_2, Y))$
3. $F(\mathit{exchanged\_info}(P_1, P_2, \mathit{non\_suspect}(Y)))$

From these formal norms we can derive, by use of the process described in section 2.4, the positive and negative landmarks and the landmark pattern. From the first norm we obtain the positive landmarks $l_1^+ = \mathit{authenticated}(P_1, P_2)$ and $l_2^+ = \mathit{interacted}(P_1, P_2)$, and the sub-pattern $(l_1^+, l_2^+) \in \leq$. The landmarks we derive from the second norm are $l_3^+ = \mathit{confirmed\_know}(P_1, P_2, \mathit{suspect}(Y))$ and $l_4^+ = \mathit{exchanged\_info}(P_1, P_2, Y)$, and the sub-pattern $(l_3^+, l_4^+) \in \leq$. Finally we obtain a negative landmark $l_1^- = \mathit{exchanged\_info}(P_1, P_2, \mathit{non\_suspect}(Y))$ from the third norm. When combined this forms the following landmark structure:

$$\mathfrak{L} = \langle \{l_1^+, l_2^+, l_3^+, l_4^+\}, \{l_1^-\}, \{(l_1^+, l_2^+), (l_3^+, l_4^+)\} \rangle$$

As described in section 2.5 we use this landmark structure to guide the behaviour of the multiagent system used to assist the officers in regions A and B. The protocol that we obtain from the landmark structure given above is basically made of three separate parts. The first part is a protocol for determining the identity of the different parties involved. This can be anything from the exchange of identity-papers (or, in the case of agents, digital certificates hashed/encoded by some cryptographic key), to something as complex as a cryptography-based authentication protocol for determining identities.

1. A sends B its certificate signed by A's private key ($s_0 \rightarrowtail s_1$).
2. B sends A its certificate signed by B's private key ($s_1 \rightarrowtail s_2$).

After obtaining the certificate from the other party, A needs to decide whether he wants to continue (in case he is conviced of the identity of B), or that he wants to halt the protocol (steps 3.1 ($s_2 \rightarrowtail s_{3.1}$) and 3.2 ($s_2 \rightarrowtail s_{3.2}$)); we are now in landmark l_1^+.

The part of the protocol that A and B execute when A decides to go forth is, in itself, a complex protocol, taken from [14], that needs to be executed so that A knows that B already knows about X and vice versa, i.e., the protocol is used such that both parties prove their knowledge about X to the other party. Note that starting this part of the protocol is considered *interacting*, and we therefore reached landmark l_2^+.

4. Region A chooses an Information Block (IB) $I_A \in KB_A$ of which they want to prove their knowledge to region B, and of which they want to test B's possession ($s_{3.1} \rightarrowtail s_4$).
5. A computes $I_{A^*} \subseteq KB_A$ and generates a random challenge C_A such that it discriminates within I_{A^*} ($s_4 \rightarrowtail s_5$).
6. A sends B the message $\{H_1 = hash(pad(I_A, \{N\})), C_A\}$ ($s_5 \rightarrowtail s_6$).
7. B computes $I_{B^*} \subseteq KB_B$ ($s_6 \rightarrowtail s_7$).
8. B does one of the following:
   (1) B generates a random challenge C_B such that it discriminates within I_{B^*}, and sends A the message $\{C_B\}$ ($s_7 \rightarrowtail s_{8.1}$).
   (2) B sends A the message $\{halt\}$ and the protocol is halted ($s_7 \rightarrowtail s_{8.2}$).
9. A sends B the message $\{H_{2_A} = hash(pad(I_A, \{N, A, C_B\}))\}$ ($s_{8.1} \rightarrowtail s_9$).
10. B verifies whether the received H_{2_A} equals any $hash(pad(I_{B_i}, \{N, A, C_B\}))$, where $I_{B_i} \in I_{B^*}$ (locally computed). If they are equal, B concludes that I_A equals the matching I_{B_i}, and thereby verifies that A knows the matching I_{B_i} (which is called I_B from here on) ($s_9 \rightarrowtail s_{10}$).
11. If B is willing to prove his knowledge of I_B to A, B sends A the message $\{H_{2_B} = hash(pad(I_B, \{N, B, C_A\}))\}$ ($s_{10} \rightarrowtail s_{11}$).
12. A verifies whether the received H_{2_B} is equal to $hash(pad(I_A, \{N, B, C_A\}))$ (locally computed). If they are equal, A concludes that I_A equals I_B, and thereby verifies that B knows the matching I_A ($s_{11} \rightarrowtail s_{12}$).

Again, at the end A needs to decide whether he wants to go through or not, depending on whether B succeeded in proving to A that he knows about A (step 13.1 ($s_{12} \rightarrowtail s_{13.1}$) and 13.2 ($s_{12} \rightarrowtail s_{13.2}$)). Note that B has a similar decision point at step 8. By now we have arrived landmark l_3^+.

The final part (to get from l_3^+ to l_4^+) can then be as simple as:

14. A tells B everything he knows about X ($s_{13.2} \rightarrowtail s_14$).
15. B tells A everything he knows about X ($s_14 \rightarrowtail s_{15}$).

More complex interaction and information exchange protocols can be used instead if desired, though.

Given the protocol specification above we obtain the following formal protocol structure (as specified in definition 2):

$$\mathfrak{P} = \langle \{s_0, s_1, s_2, s_{3.1}, s_{3.2}, \ldots, s_{15}\}, \{R_i\}_{i \in \mathcal{A}} \rangle$$

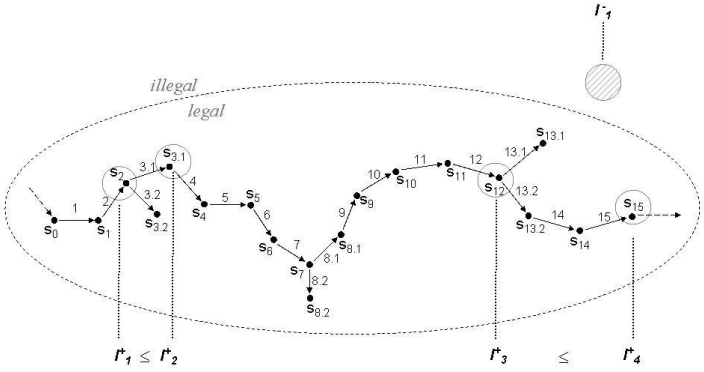

Fig. 3. From landmarks to protocol

where $\mathcal{A}$ is the set $\{1, 2, 3.1, 3.2, \ldots, 14, 15\}$ closed under ; and $\cup$ operations. Figure 3 depicts this protocol and its compliance with the landmark pattern. Compliance of $\mathfrak{P}$ is guaranteed, on the basis of definition 5, by the following relation between landmarks and states in the protocols:

$$\mathcal{R} = \{(l_1^+, s_2), (l_2^+, s_{3.1}), (l_3^+, s_{12}), (l_4^+, s_{15})\}.$$

Please note that a) $(l_1^+, l_2^+) \in \leq$ iff $(s_2, s_{3.1}) \in R_{3.1}$, and $(l_3^+, l_4^+) \in \leq$ iff $(s_{12}, s_{15}) \in R_{13.2;14;15}$; b) there is no $s \in \{s_0, \ldots, s_{15}\}$ such that $(l_1^-, s) \in \mathcal{R}$; and c) that landmark l_4^+ is associated to one of the final states of the protocol.

4 Conclusions

In this paper we proposed a formal framework to design agent protocols from a normative specification. As norms are declarative in nature, they cannot be directly connected to a protocol (operational in nature). In order to tackle the problem, we introduced landmarks as an intermediate level. Landmarks reduce the complexity of normative reasoning by capturing a) the important states of affairs, as defined in the norms, and b) the operational constraints between those states. This information can then be used to design a norm-compliant protocol.

Norm compliance has also been studied in [1], where the main focus was on checking the norm compliance of a given protocol against the norms by means of a formal framework. Here instead, we introduce the idea of extracting landmarks from the norms to guide the protocol design. We also foresee landmarks as a way for agents to evaluate norm compliance of protocols on-line, i.e. at runtime.

One of the lines we want to explore is how agents may use landmarks to dynamically create or adapt protocols at run-time: given a protocol and the landmarks, agents may reason about acceptable variations of the protocol that are *legal* and that allow them to fulfil their interests or to cope with an unexpected situation not foreseen in the protocol. Given some landmarks, agents

may even negotiate the protocol to use. Another line to explore is the impact of landmarks in norm enforcement: on-line checking the execution of protocols by making sure that the systems does not pass through any negative landmarks.

References

1. H. Aldewereld, J. Vázquez-Salceda, F. Dignum, and J.-J.Ch. Meyer. Proving norm compliancy of protocols in electronic institutions. Technical Report UU-CS-2005-010, Institute of Information and Computing Sciences, Utrecht University, 2005.
2. A.R. Anderson. A reduction of deontic logic to alethic modal logic. *Mind*, 22:100–103, 1958.
3. J. Broersen, F. Dignum, V. Dignum, and J.-J. Ch. Meyer. Designing a Deontic Logic of Deadlines. In *7th Int. Workshop on Deontic Logic in Computer Science (DEON'04)*, Portugal, May 2004.
4. F. Dignum. Abstract norms and electronic institutions. In *Proceedings of the International Workshop on Regulated Agent-Based Social Systems: Theories and Applications (RASTA '02), Bologna*, pages 93–104, 2002.
5. F. Dignum, D. Kinny, and L. Sonenberg. From Desires, Obligations and Norms to Goals. *Cognitive Science Quarterly*, 2(3-4):407–430, 2002.
6. F. Dignum and R. Kuiper. Combining dynamic deontic logic and temporal logic for the specification of deadlines. In R. Sprague Jr., editor, *Proc. of 13th HICSS*, 1997.
7. V. Dignum. *A Model for Organizational Interaction*. SIKS Dissertation Series, 2003.
8. E.A. Emerson. Temporal and modal logic. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, volume B: Formal Models and Semantics*, chapter 14, pages 996–1072. Elsevier Science, 1990.
9. D. Grossi, H. Aldewereld, J. Vázquez-Salceda, and F. Dignum. Ontological aspects of the implementation of norms in agent-based electronic institutions. Accepted for the 1st International Symposium on Normative Multiagent Systems (NorMAS2005), 2005.
10. D. Grossi and F. Dignum. From abstract to concrete norms in agent institutions. In M. G. Hinchey, J. L. Rash, W. F. Truszkowski, and et al., editors, *Formal Approaches to Agent-Based Systems: Third International Workshop, FAABS 2004*, Lecture Notes in Computer Science, pages 12–29. Springer-Verlag, April 2004.
11. D. Harel. Dynamic logic. In D. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic: Volume II: Extensions of Classical Logic*, pages 497–604. Reidel, Dordrecht, 1984.
12. Sanjeev Kumar, Marcus J. Huber, Philip R. Cohen, and David McGee. Toward a formalism for conversation protocols using joint intention theory. *Computational Intelligence*, 18(2):174–228, 2002.
13. J.-J. Ch. Meyer and R.J. Wieringa. *Deontic Logic in Computer Science: Normative Systems Specification*. John Wiley and sons, 1991.
14. W. Teepe. New protocols for proving knowledge of arbitrary secrets while not giving them away. In Sieuwert van Otterloo, Peter McBurney, Wiebe van der Hoek, and Michael Wooldridge, editors, *Proceedings of the 1st Knowledge and Games Workshop*, pages 99–116, Liverpool, July 2004.
15. J. Vázquez-Salceda, H. Aldewereld, and F. Dignum. Implementing norms in multiagent systems. In G. Lindemann, J. Denzinger, I.J. Timm, and R. Unland, editors, *Multiagent System Technologies*, LNAI 3187, pages 313–327. Springer-Verlag, 2004.
16. J. Vázquez-Salceda, V. Dignum, and F. Dignum. Organizing multiagent systems. Technical report, Institute of Information and Computing Sciences, Utrecht University, 2004.
17. M. Wooldridge and P. Ciancarini. Agent-oriented software engineering. In S. K. Chang, editor, *Handbook of Software Engineering and Knowledge Engineering*, volume 1, pages 507–522. World Scientific Publishing Co., 2002.