Informed Deliberation during Norm-Governed Practical Reasoning

Martin J. Kollingbaum and Timothy J. Norman

Department of Computing Science
University of Aberdeen
Aberdeen, AB24 3UE, UK
{mkolling,tnorman}@csd.abdn.ac.uk

Abstract. A norm-governed agent takes social norms into account in its practical reasoning. Such norms characterise its role within a specific organisational context. By adopting a role, the agent commits to fulfil and adhere to the social norms associated with that role. These commitments require the agent to act in a way that does not violate any of its prohibitions or obligations. In adopting different sets of norms, an agent may experience conflicts between these norms as well as inconsistencies between possible actions for fulfilling its obligations and its currently adopted set of norms. In order to resolve such problems, it must be informed about conflicts and inconsistencies. The NoA architecture for norm-governed agents implements a computationally efficient mechanism for identifying and indicating such problems – possible candidates for action are assigned a specific label that contains cross-referenced information of actions and norms. As actions are indicated as problematic and not simply filtered out, the agent can still choose to either act according to its norms or against them. The labelling mechanism presented in this paper is therefore a critical step towards enabling an agent to reason about norm violations – the agent becomes norm-autonomous.

1 Introduction

Norm-governed agents are able to reason about rules and regulations established in an organisational context. With that, their practical reasoning is not only based on what they believe, desire and intend, but what they are actually obliged, permitted or forbidden to do in a specific social context. Norms are essential for the creation of organisational structures, because they characterise the rights and duties of individuals taking on specific organisational roles. Agents in such roles must be norm-governed - they must be able to take their current normative position into account in their decision-making [14]. To provide an agent with abilities to reason about norms, a set of issues must be investigated:

- How are norms and actions represented?
- How do norms influence the practical reasoning of the agent?
- How do agents resolve conflicts between norms they currently hold and deal with inconsistencies between their actions and their norms?

A specific model and architecture for norm-governed practical reasoning has been developed in the form of the NoA architecture [12]. NoA is a reactive planning architecture in the tradition of concrete implementations of practical reasoning systems [10] with extensions that allow the reasoning about norms. Specific care has been taken to make NoA agents *norm-autonomous* [5] – a NoA agent can decide whether to honour its obligations and prohibitions. This requires that the agent, in its attempt to fulfill obligations, does not simply filter out options for action that are inconsistent with its current set of norms, but that the complete set of options for action are taken into account during deliberation. NoA agents use a labelling mechanism to characterise options for action as either consistent or inconsistent with their current set of norms. In this paper, we use concepts introduced in [14] and [12] and investigate in more detail the concept of "informed" deliberation. For this purpose, an enriched form of a label for candidate actions is introduced that guides or "informs" the deliberation of a norm-governed agent. In its deliberation, the agent can use this label to reason about consistency of a possible option for action – whether an action is norm-compliant or not. In case of inconsistencies, it will be beneficial for the agent to become informed about the reasons of such an inconsistency – which norms are responsible for the inconsistency of an action? Are all options inconsistent, or is there still a possibility to remain norm-compliant? Can the normative authority, which issued such norms, be convinced to revoke existing prohibitions or obligations or at least temporarily grant a permission that overrides a prohibition? Which violation of a norm results in the least damage to the agent's reputation? To support the agent in resolving inconsistency, the labelling mechanism described in this paper holds cross-referenced information between possible candidates for action of the agent and its currently held set of norms.

2 Norm-Governed Agents

Norm-governed agents are able to reason about norms and take them into account in their practical reasoning. Such an agent must be socially aware – it must be able to (a) adopt norms such as obligations, permissions and prohibitions as they are established within a community of agents, (b) process them correctly and (c) anticipate the possible interactions between the effects of its actions and its norms. The NoA system [12, 14] comprises an abstract model of norm-governed agency and a concrete agent architecture for the implementation of norm-governed agents. In the development of this model and architecture, a set of design decisions were made: (a) practical reasoning is based on reactive planning, with a set of pre-specified plan procedures representing the agent's behavioural repertoire, (b) obligations are the principal motivators for the agent to act, (c) plan procedures are declared with explicit effect specifications – this allows the agent to reason about the effects of its actions, whether they are consistent with its currently held norms and (d) a clear distinction is made between the agent achieving a state of affairs or performing an action (see [16]). This distinction is reflected in the NoA norm and plan specifications, with norms refering

to an *activity* that is either the achievement of a state or the performance of an action. Norms are central to the NoA model of norm-governed practical reasoning. In this model, the norms held by the agent are its obligations, permissions and prohibitions:

- Obligations are the principal *social* motivators within NoA — they motivate the agent to either achieve a state of affairs or to perform an action. Based on such a motivation, a norm-governed agent may select an appropriate plan for execution. Obligations can be viewed as analogous (although not identical) to goals (or desires) within traditional Belief-Desire-Intention agent architectures such as Jason [1].[1] The analogy lies in the fact that, as with goals (or desires), it may not be the case that the agent will instantiate and select a plan (i.e. adopt an intention) to satisfy an obligation; this will depend on other social constraints on the agent's activities along with its capabilities (encoded in its available plans) and the current circumstances it finds itself in (that leads to the generation of a set of instantiated plan options).
- Prohibitions require the agent to not achieve a state of affairs or perform an action – the agent is forbidden to pursue a specific activity. Prohibitions are not motivators for the agent, but explicitly restrict the choices of activities the agent can ideally employ.
- Permissions explicitly allow the achievement of a state of affairs or the performance of an action.

In the following, we present a detailed specification of the NoA model of norm-governed agency.

2.1 The Abstract Model

The NoA model of norm-governed agency maintains a set of $BELIEFS$ as a representation of the current state of the world, the set $PLANS$ containing the plan specifications, the set $NORMS$ representing the adopted set of norms, and the set $ROLES$ comprising all those roles the agent has adopted. Each role is characterised by a set of norms – when the agent adopts a role it adopts all the norms annotated to this role as well. All norm specifications over all the adopted roles comprise the set $NORMS$. An agent joins an organisation and adopts (one or more) roles within this organisation by signing a contract with members (representatives) of the organisation. Each role $r \in ROLES$ is specified in a contract $c \in CONTRACTS$. To allow a unique identification of elements within these sets, the concept of an identifier is introduced. These concepts are plans, norms, roles, agents and contracts:

[1] In the research reported here, we do not discuss the distinctions between desires (internal motivators) and obligations (social motivators), but focus exclusively on the way that norms are interpreted; this is clearly a topic for future investigation, but see, for example, Castelfranchi [3] for some insights into this issue.

Definition 1. *The set* $I^{NORMS} = \{n_1, ..., n_n\}$ *describes a finite set of norm identifiers. The set* $I^{Plans} = \{p_1, ..., p_n\}$ *describes a finite set of plan identifiers. The set* $I^{Roles} = \{r_1, ..., r_n\}$ *describes a finite set of role identifiers. The set* $I^{Agents} = \{a_1, ..., a_n\}$ *describes a set of agent identifiers. The set* $I^{Contracts} = \{c_1, ..., c_n\}$ *describes a finite set of contract identifiers.* $IDENTIFIERS = I^{Roles} \cup I^{Agents} \cup I^{Plans} \cup I^{Contracts}$ *is the set of all identifiers, where* I^{Roles}, I^{Agents}, I^{Plans} *and* $I^{Contracts}$ *are mutually disjunct.*

In the context of NoA, the norm-governed agent is described as pursuing either a state-oriented or action-oriented *activity* [16]. Norm declarations, therefore, contain a so-called *activity specification*:

Definition 2. *An activity A determines either the achievement of a state of affairs, called state-oriented activity, or the performance of an action, called action-oriented activity. The expression* $achieve(p)$ *expresses the achievement of a state of affairs p. The expression* $perform(\sigma)$ *expresses the performance of action* σ*, where* σ *describes the signature of a pre-specified plan procedure formulated in the NoA language. An agent can be allowed, forbidden or required to achieve or* ***not*** *achieve a state of affairs (or its negation):*

- *"achieve a state of affairs p":* $achieve(p)$
- *"achieve a state of affairs* $\neg p$*":* $achieve(\neg p)$
- *"not achieve a state of affairs p":* $\neg achieve(p)$
- *"not achieve a state of affairs* $\neg p$*":* $\neg achieve(\neg p)$

An agent may also be obliged, forbidden or allowed to perform or to ***not*** *perform an action:*

- *"perform action* σ*":* $perform(\sigma)$
- *"not perform action* σ*":* $\neg perform(\sigma)$

Norm specifications, comprising the set $NORMS$ and expressing either an obligation, permission or prohibitions, contain such activity specifications expressing that a state or action is either *obliged*, *permitted* or *prohibited*. A label is introduced to identify a norm specification as either an obligation, permission or prohibition.

Definition 3. *The set* $L^{Norms} = \{obligation, permission, prohibition\}$ *is the set of labels used to identify obligations, permissions and prohibitions*[2]*.*

A norm specification can then be defined in the following way:

Definition 4. *A norm specification, expressing an obligation, permission, prohibition is a tuple* $\langle n, i^{Roles}, A, a, e \rangle$*, where*

- $n \in L^{NORMS}$

[2] A label "sanction" exists as syntactic sugar, as it is an obligation for an agent in the role of a so-called "authority" to pursue certain activities that represent such sanctions (see [13] for more details).

– $i^{Roles} \in I^{Roles}$ *is a role identifier for a norm addressee*
– A *is the activity specification*
– a *is the activation condition*
– e *is the expiration condition*

With such a definition in place, norms can be specified in NoA. Norms are declared according to the possibilities of expressing a specific activity. For example, according to this definition of a norm specification, an obligation can express that a norm addressee is obliged to *see to it* that a specific state of affairs is achieved (or not achieved) or that it is obliged to **not** *see to it* that a specific state of affairs is achieved (or not achieved).

Norms in NoA are conditional entities — they are *relevant* to an agent under specific circumstances only. Our model of norm-governed agents includes a concept of explicit norm activation and deactivation: norms carry two conditions, an activation condition and an expiration condition. These two conditions allow an exact specification of circumstances under which a norm becomes active and, therefore, relevant to the agent, and when it expires. A separate expiration condition allows a more precise specification of the circumstances when a norm is actually active:

– As soon as the activation condition holds, a norm is activated and becomes relevant to the agent.
– It continues to be activated, even if the activation condition ceases to hold.
– A norm is transferred from an activated into a deactivated state only if the expiration condition holds.

With that, the two conditions test two events — the occurrence of a state of affairs that activates the norm and the occurrence of a state of affairs that deactivates the norm.

NoA is a reactive planning system. Characteristic for a reactive planning system is the provision of pre-specified plan procedures at design time as the behavioural repertoire of an agent. A NoA agent adopts a set of such plans as its set $PLANS$. Obligations can motivate either the achievement of a state of affairs or the performance of an action. Plan procedures in NoA service both cases. If a *state-oriented* activity is required, plans are selected according to their *effects* – NoA introduces explicitly specified effects into plan declarations. If an *action-oriented* activity is required, plans are selected directly according to their signature. An abstract definition of a plan is given in the following:

Definition 5. *A plan is defined as a tuple* $P = \langle \sigma, precondition, effects, body \rangle$*, where:*

– σ *is the signature of the plan specification, with* $\sigma = \langle I^{Plans}, \{par_1, .., par_n\} \rangle$ *comprising a plan identifier and a set of parameters,*
– precondition *comprises an expression over predicates and operators* $\wedge, \vee, \neg$*; if the set* $BELIEFS$ *reflects a state of affairs that evaluates the precondition to true, the plan becomes activated,*

– effects *comprises a list of terms expressing possible effects occurring during the execution of the plan body,*
– body *comprises an executable specification of the plan.*

2.2 Activation, Selection and Execution

The concept of *activation* is essential in NoA. As described before, norm and plan declarations contain conditions that determine under what circumstances norms are activated (and instantiated in the course of this activation) and, therefore, *relevant* to the agent and when plans are activated and, therefore, instantiated and available as potential options for execution. The currently activated norms determine the agents current *normative position*. The currently activated plans determine its current potential behavioural repertoire. Two sets express the current activation state of an agent: (a) the set $INSTNORMS$, representing the set of activated and instantiated norms and (b) the set $INSTPLANS$, representing the set of activated and instantiated plans.

Definition 6. *The set* $INSTNORM = INSTOBL \cup INSTFOR \cup INSTPER$ *is the set of currently activated and, therefore, instantiated norms. Subsets of the set* $INSTNORMS$ *are* $INSTOBL$, $INSTFOR$ *and* $INSTPER$, *which are the sets of currently instantiated obligations, prohibitions and permissions.*

The sets $INSTNORMS$ and $INSTPLANS$ are permanently changing according to changes in the set of beliefs of the agent. Therefore, at any time, a specific set of norms is activated and and a set of plans instantiated. A subset of these activated norms are the currently activated obligations of the agent, $INSTOBL$. Each obligation $o \in INSTOBL$ motivates the agent to act – either to achieve a state of affairs or to perform an action. The agent has to select options or *candidates* for action from the set of currently instantiated plans, $INSTPLANS$. The set $CANDIDATES$ is formed, containing all those plan instantiations that are candidates for obligations in the set $INSTOBL$.

Traditionally, agents based on reactive planning architectures have to select one specific candidate for execution from this set (which is described here as the set $CANDIDATES$) – in a process of deliberation, the agent has to apply specific strategies for this selection. Norm-governed agent have to take norms into account in their practical reasoning. With the introduction of norm-awareness into an agent architecture, the agent is enabled to reason about the *consistency* of its actions in terms of norms – certain actions, which are possible candidates for fulfilling an obligation are maybe forbidden. One way of dealing with such *inconsistent* candidates would be to simply filter them out – but with such a strategy the agent becomes completely benevolent and is not norm-autonomous. Norm-autonomy is essential to NoA agents – the agent can decide whether to honour its obligations and prohibition. Therefore, before the agent decides which candidate from the set $CANDIDATES$ will be executed, it has to investigate the consistency of these options. For this, NoA introduces a labelling mechanism

that identifies each candidate as either consistent or inconsistent with the set $INSTNORMS$.

2.3 Investigating Norm Consistency

In essence, two problems have to be investigated: (a) *Possible Conflicts* between permissions and prohibitions and (b) *Possible Inconsistencies* between candidate plans and norms. Permissions and prohibitions *configure* the normative position of an agent, either restricting or expanding the set of possible actions (plans) the agent can employ without causing norm violation. In terms of inconsistency, obligations may motivate the creation of a set $CANDIDATES$, where *none*, *some* or *all* plan instantiations contained in this set are prohibited because either the execution of the plan itself is prohibited or because the plan produces at least one (side-)effect that is prohibited. Conflicts between permissions and prohibitions have to be resolved so that the consistency of candidates in the set $CANDIDATES$ can be investigated. For this purpose, NoA puts forward conflict resolution strategies that are discussed in detail in [12, 14].

For a definition of consistent execution of plans in NoA, it is necessary to observe the relationship between candidates – plan instantiations – and norms. The set $INSTNORMS$ expresses that either the achievement of certain states of affairs or the performance of certain actions (plan instantiations) is either allowed, forbidden or obliged:

Definition 7. *The set S_O describes those states of affairs obliged by currently active obligations contained in the set $INSTOBL$, whereas the set T_O describes actions obliged by currently active obligations contained in the set $INSTOBL$. Similarly, the sets S_F and S_P and the sets T_F and T_P describe states of affairs prohibited / permitted and actions prohibited / permitted by currently active norms.*

According to definition 10, a plan instantiation in the set $INSTPLANS$ is a consistent candidate for a specific obligation $o \in INSTOBL$, if this plan instantiation is (a) not a currently forbidden action, (b) none of its effects are forbidden states of affairs and (c) none of its effects *counteracts* any obligation in the set $INSTOBL$. To allow the investigation of possible effects of an instantiated plan $p \in INSTPLANS$, a function *effects*(p) is introduced:

Definition 8. *For a plan instantiation $p \in INSTPLANS$, the function effects(p) provides the set of fully instantiated effect specifications:*

$$effects(p) = \{\ e \mid e \text{ is an effect of plan instantiation } p \in INSTPLANS\}$$

A second function is needed that allows us to refer to states of affairs that are the negation of states expressed by plan effects. The function producing this set is called *neg_effects*(p).

Definition 9. *For a plan instantiation $p \in INSTPLANS$, the function neg_effects(p) describes a set that contains a negated version for each element e of the set described by effects(p):*

$$neg\_effects(p) = \{\ n \mid e \in effects(p) \wedge n = \neg e\}$$

With these definitions in place, a norm-consistent execution of a plan can be expressed in the following way:

Definition 10. *The execution of a plan instantiation $p \in INSTPLANS$, with $p \notin T_F$ (p is not a currently forbidden action), is consistent with the current set of active norms, $INSTNORMS$, of an agent, if none of the effects of p is currently forbidden and none of the effects of p counteracts any currently active obligation:*

$$\begin{aligned} consistent(p, T_F, S_F, S_O)\ \textit{iff}\quad & p \notin T_F \\ & \textit{and}\ S_F \cap effects(p) = \emptyset \\ & \textit{and}\ S_O \cap neg\_effects(p) = \emptyset \end{aligned}$$

An investigation into the consistency in NoA takes place according to this definition of consistent execution of a plan instantiation. The result of such an investigation will be the set of prohibitions that either forbid the candidate to be executed directly or that forbid the candidate's effects to occur as states of affairs, and the set of obligations that are counteracted by the effects of the candidate. In NoA, this information is accumulated in the consistency label for candidates:

Definition 11. *A label, expressing consistency / inconsistency of a plan instantiation $c \in CANDIDATES$, is a tuple*

$$L = \langle c, MOTIVATORS, PROHIBITORS \rangle,$$

where

- $c \in CANDIDATES$ *is the labelled candidate for a set of motivating obligations*
- $MOTIVATORS = \{\ o^c \mid o^c \in INSTOBL \wedge c \in CANDIDATES \wedge effects(c) \cap S_O \notin \emptyset\ \} \cup \{\ o^c \mid o^c \in INSTOBL \wedge c \in CANDIDATES \wedge c \in T_O\ \}$ *is the set of obligations that motivated the addition of this candidate to the the set $CANDIDATES$, because (a) one of its effects achieves the state of affairs demanded by this obligation or (b) it is the action demanded by these obligations*
- $PROHIBITORS = \{\ f^c \mid f^c \in INSTFOR \wedge c \in CANDIDATES \wedge c \in T_F\} \cup \{\ f^c \mid f^c \in INSTFOR \wedge c \in CANDIDATES \wedge effects(c) \cap S_F \notin \emptyset\} \cup \{\ o^c \mid o^c \in INSTOBL \wedge c \in CANDIDATES \wedge neg\_effects(c) \cap S_O \notin \emptyset\}$ *is the set of conflicting prohibitions or obligations*

From this labelling, the agent can derive the consistency of its current normative position. For a candidate $c \in CANDIDATES$, a label expresses consistency, if the set of $PROHIBITORS$ is empty:

- Label expressing *consistency*: $\langle\ c, MOTIVATORS, \{\}\ \rangle$

A partitioning of the set $CANDIDATES$ emerges into *consistent* and *inconsistent* candidates. By translating the set $CANDIDATES$ into a *labelled* set $CANDIDATES^L$, this partitioning occurs, where each element is annotated with a label L expressing consistency or inconsistency.

Via characterising the consistency of candidate plans, we can define the consistency of an obligation. To be able to address the subset of candidates that are options for a specific obligation, the function $options(o)$, with $o \in INSTOBL$, is defined:

Definition 12. *For a specific instantiated obligation $o \in INSTOBL$, the function $options(o)$ describes a subset of elements from the set $CANDIDATES$, where each element of this subset is a candidate for obligation o:*

$$options(o) = \{\ c^o \mid c^o \in CANDIDATES\ \wedge\ o \in INSTOBL \wedge is\_candidate(c^o, o)\}$$

For a specific obligation $o \in INSTNORMS$, a specific subset of the set $CANDIDATES^L$ represents the set $options(o)$ of possible candidates. There are three possible configurations for this set: (a) all elements in $options(o)$ are labelled consistent, (b) at least one element in $options(o)$ is labelled consistent or (c) all elements are labelled inconsistent. According to these three possibilities, we introduce three so-called *consistency levels* for a specific obligation:

- *Strong Consistency.* An obligation is strongly consistent if all $options(o) \subseteq CANDIDATES^L$ are labelled as consistent:

  $$strong\_consistent(o, S_F, S_O, T_F)\ \textit{iff}\ \forall p \in options(o).\ consistent(p, T_F, S_F, S_O)$$

- *Weak Consistency.* An obligation is weakly consistent if at least one candidate in the set $options(o)$ is labelled as consistent:

  $$weak\_consistent(o, S_F, S_P, S_O, T_F)\ \textit{iff}\ \exists p \in options(o)\ s.t. consistent(p, T_F, S_F, S_P, S_O)$$

- *Inconsistency.* An obligation is inconsistent if no candidate in the set $options(o)$ is labelled as consistent:

  $$inconsistent(o, S_F, S_O, T_F)\ \textit{iff}\ \forall p \in options(o).\ \neg consistent(p, T_F, S_F, S_O)$$

For a NoA agent, this norm-annotated set of candidates, $CANDIDATES^L$, is the input into the subsequent deliberation process to find a single plan for execution for each obligation in the set $INSTOBL$. According to the concept of *norm-autonomy* [5], norm-inconsistent options for action are not simply filtered out but remain – albeit inconsistent – options for the agent's deliberation. During deliberation, the agent can then decide whether to honour its obligations and prohibitions by only selecting norm-consistent options or to act against its currently held norms. NoA agents are, therefore, norm-autonomous.

3 Informed Deliberation

Informed Deliberation is the mechanism within NoA for dealing with consistency between the agent's actions and its currently held set of norms. For the agent to be able to deliberate about its actions, it needs information about a partitioning of the set $CANDIDATES$ of applicable actions into allowed and forbiddent actions. Such a partitioning must be "complete" – if the normative situation for specific candidates is not decided because of conflicts in the set of norms, then these conflicts have to be resolved. In the context of NoA, specific conflict resolution strategies are proposed (see [12]). The following strategies are under investigation in the context of the NoA model: (a) Arbitrary decision, (b) Recency, (c) Seniority, (d) Cautiousness, (e) Boldness, (f) Social power and (g) negotiation with the norm issuer. These are conflict resolution strategies that can be employed during the agent's deliberation. It helps the agent to achieve a complete partitioning of its candidate set into allowed and forbidden plans. The strategy "*arbitrary decision*" can be utilised as the simplest form of conflict resolution as it does not take into account any information about the conflict situation itself. If the agent chooses "*recency*" or "*seniority*", then a form of time stamp is required that records the activation time of a norm. With that, a ranking according to activation time can be established and selections according to "*recency*" or "*seniority*" can take place. The agent is pursuing a "*cautious*" strategy, if prohibitions always overrule permissions and it is pursuing a "*bold*" strategy, if permissions always overrule prohibitions.

An agent can also "*renegotiate*" specific norms and reach agreements to either revoke prohibitions or receive additional permissions that override prohibitions.

A conflict resolution strategy according to "*social power*" would utilise relationships of dependency and influence between roles. Such relationships can be used to determine, if a norm is "more powerful" to override a conflicting norm. If the issuer of norms, acting in a position of power, issues multiple conflicting norms, the agent, despite being able to detect such conflicts, will not be able to resolve the conflict according to "*social power*" as all conflicting norms are issued by the same source. The agent may claim that this source is inconsistent itself and require it to resolve these conflicts and to reissue a set of norms without conflicts. Such a situation can be regarded as a *distributed conflict resolution strategy.*

Finally, the agent may not be able to remove prohibitions on its actions. If these actions are necessary for the fulfilment of its obligations, it may decide to act against existing prohibitions. In such a case, it may investigate the *consequences* or *sanctions* for such a violation – according to a rational reasoning, the agent may decide to choose an action that incurs a minimum of costs in terms of sanctions. This would require the enhancement of the NoA labelling mechanism to capture such costs.

The consistency label of candidates is used in NoA to indicate the consistency of specific candidate plans – candidate plans for execution are simply identified as either consistent or inconsistent. In the following discussion, the information conveyed by the label in the form of the set $PROHIBITORS$ is taken into

account in a more detailed fashion. The goal is to give an agent means to *remove* inconsistencies so that it can pursue its intended activities. The agent has to change its consistency level.

The normative situation within a society can be quite complex. An agent can take on different roles and, with that, adopt different sets of – possibly conflicting – norms. NoA employs a model of norm specifications with conditions that determine under what circumstances norms are "active" and, therefore, "relevant" to the agent. Inconsistencies between norms and actions are, therefore, apparent only if specific circumstances activate inconsistent norms and actions.

3.1 Example

For example, let us assume that an agent holds a set $INSTNORMS = \{ p_1, p_2 \}$ with two plans p_1 and p_2 as its current (instantiated) capabilities. We assume that these plans will produce the following states of affairs as their effects during execution:

- plan p_1 : $\mathit{effects}(p_1) = \{ s, t \}$
- plan p_2 : $\mathit{effects}(p_2) = \{ s \}$

We also assume that the agent adopts two roles, $ROLES = \{ r_1, r_2 \}$ and, consequently, two sets of norms annotated to these roles. If we use specific syntactic forms to express norm specifications according to Definition 4 (see [14, 12] for details), then we can describe the two role-related norm sets in the following way:

- role r_1 : $\{ \mathit{obligation}(r_1, \mathit{achieve}(s), \phi, \psi), \mathit{prohibition}(r_1, \mathit{perform}(p_2), \phi, \psi) \}$
- role r_2 : $\{ \mathit{prohibition}(r_2, \mathit{achieve}(t), \phi, \psi) \}$

According to Definition 4, norm specifications are characterised by a reference to a role, an activity specification and two conditions, the activation and expiration condition (denoted here as ϕ and ψ). For the following discussion, we assume that these two norm sets are issued by two different normative authorities, authority A_x and A_y. With that, the agent's set $INSTNORMS$, comprising these two role-related norm sets, contains norms issued by different normative authorities.

This agent is motivated by its obligation $\mathit{obligation}(r_1, \mathit{achieve}(s), \phi, \psi)$ to achieve this state of affairs. Consequently, it forms the set $CANDIDATES$. Plan p_1 as well as p_2 produce s as one of their effects and, therefore, comprise the set $CANDIDATES$:

- $CANDIDATES = \{ p_1, p_2 \}$

The investigation of consistency yields following problems: candidate p_1 is inconsistent with the prohibition to achieve state t, as $t \in \mathit{effects}(p_1)$ and candidate p_2 is inconsistent with the prohibition to perform action p_2. A set of labels emerges, characterising these inconsistencies (see Definition 11):

$$L_{p_1} = \langle p_1, \{obligation(r_1, achieve(s), \phi, \psi)\}, \{prohibition(r_2, achieve(t), \phi, \psi)\}\rangle$$

$$L_{p_2} = \langle p_2, \{obligation(r_1, achieve(s), \phi, \psi)\}, \{prohibition(r_2, perform(p_2), \phi, \psi)\}\rangle$$

In both labels, the set $MOTIVATORS$ (see Definition 11) contains the one motivating obligation. In both cases, the set $PROHIBITORS$ is not empty but contains the corresponding conflicting prohibitions. The motivating obligations responsible for forming this set $CANDIDATES$ is at a level of *inconsistency*.

In this situation, the agent has two options:

- although the agent is in a state of inconsistency, it acts by selecting one of the candidates for execution.
- the agent tries to *improve* the level of consistency for its obligation, so that at least one of the candidates becomes a consistent option

3.2 Improving the Level of Consistency

As outlined before, the consistency of candidate plans defines the consistency of an obligation. For a specific obligation $o \in INSTNORMS$, a specific subset of the set $CANDIDATES^L$ represents the set $options(o)$ of possible candidates for this obligation. This set can have one of the three following states: (a) all candidates are consistent, (b) at least one of them is consistent or (c) none of them is consistent. According to the consistency situation of $options(o)$, the obligation o is then either *strongly consistent*, *weakly consistent* or *inconsistent*. An obligation can be fulfilled without violating other norms, if it is at least weakly consistent. A change of such a consistency level may take place because of the activation of new permissions and prohibitions. Permissions allow actions to occur whereas prohibitions declare certain actions as forbidden.

In the previous example, the set $MOTIVATORS$ for the two candidates p_1, p_2 contains one obligation to achieve a state of affairs s:

$$obligation(r_1, achieve(s), \phi, \psi)$$

According to the labelling outlined in the example, none of the candidate plans for this obligation are consistent – a prohibition exists for both candidates in the set $CANDIDATES$. The agent is regarded as operating at a "level of inconsistency" in terms of this obligation.

If the agent decides to fulfill this obligation in a norm-consistent way, then it must try to *upgrade* the level of consistency of this obligation. This would mean to free – maybe temporarily – at least one of the candidate plans from its *prohibitors*. This can take place by engaging with the authority that issued the prohibitors in a dialogue and reach an agreement that can be the following:

- the authority revokes the prohibiting norms

– the authority issues a permission that temporarily overrides the existing prohibition (see [14, 12] for details about precedence and overriding between norms and appropriate conflict resolution strategies)

If the authority issues a (temporary) permission, then a situation of conflict occurs with the existing norms contained in the set $PROHIBITORS$ of at least one of the candidates for this obligation. In this case, such a conflict is intentional – during the dialogue with the authority, the agent negotiates the release of such a permission, using knowledge about the possible classes of conflict (as outlined above) between norms. After receiving such a permission, the agent relies on its set of conflict resolution strategies to achieve the correct overriding between norms.

Let us assume that the agent could convince the authority to issue following permission:

$$permission(r_1, achieve(t), \phi, \psi)$$

In our example, the agent has adopted two roles, r_1 and r_2. Let us assume that the agent receives this permission for its role r_1. As this permission allows the achievement of state t that is forbidden by the existing prohibition, a *conflict* occurs. The agent can employ a conflict resolution strategy – for example, one of those outlined in [12] – to make the permission the dominant norm. With that, the agent allows the prohibitor for candidate plan p_1 to be overridden – this candidate becomes a consistent choice. Candidate p_2 is still inconsistent, therefore the agent can fulfill this obligation at a level of *weak consistency*.

In the example above, two different normative authorities are introduced, A_x and A_y. The agent has to decide which authority to contact for relaxing its normative situation. It also has to decide, for which action the prohibition should be either revoked or relaxed. The information contained in the label assigned to each candidate in the set $CANDIDATES$ can be used in this decision process. It gives a clear indication about all the norms that create the current state of inconsistency. If additional information about relationships of power and influence between authorities is made available, these power relationships within organisations can be used to find an authority at a superior level in this hierarchy that has the power to override decisions of subordinates and upgrade the agent's level of consistency. A conflict resolution strategy according to *social power* would require a substantial extension of the role model within NoA to express relationships of dependency and influence between roles. Such relationships can be used to determine, if a norm is "more powerful" to override a conflicting norm. The indication of such role-relationships within the NoA labelling mechanism will be investigated in future work.

As the NoA architecture uses mechanisms to perform plan and norm activations efficiently (using a Rete network implementation [9]), information contained in labels is maintained whenever plan and norms are activated or deactivated. It represents, therefore, an efficient form of informing the deliberation of the agent.

4 Related Work

Research into norm-governed reasoning and the concept of norm-autonomy, as described in this paper, is influenced by related work, especially [4, 6, 7] and [5]. The model of norm-governed agency also takes influences from the work by Jones and Sergot [11] and by Pacheco and Carmo [17]. They describe the modelling of complex organisations and organisational behaviour based on normative concepts. The design of the NoA architecture takes influences from various sources, most prominently the BDI model of agency [18], but also from classical planners regarding the declaration of plans and from production systems regarding plan activation, selection and execution. NoA is a reactive planning architecture [8, 10], where the behaviour of an agent is determined by pre-specified plans. NoA differs from these classic models and systems: (a) a clear distinction is made between agents achieving a state of affairs or performing an action, reflected in norm and plan specifications, (b) plan procedures contain explicit effect specifications to allow a norm-governed practical reasoning, and (c) NoA employs a detailed model of conflict resolution and inconsistencies between actions and norms and inform the deliberation of the agent about possible inconsistencies to make the agent norm-autonomous. In terms of designing a normative architecture, Broersen et al. [2] describe the BOID architecture. Conflict resolution strategies are presented as overruling orders between the concepts "belief", "obligation", "intention" and "desire". NoA takes, in contrast to BOID, a practical approach towards modelling norm-governed agency and provides a design for a specific architecture for norm-governed agents. But similar problems, as conflicts between norms and precedence relationships between them, are also discussed in the context of NoA. NoA, as a practical reasoning system based on reactive planning mechanisms, puts forward a set of conflict resolution strategies. Similarly, Lopez et al. [15] discuss how agents decide whether or not to adopt norms, taking into account issues of consistency.

5 Conclusion

A norm-governed agent must be able to anticipate whether its actions are violating any norms that are associated with its role in a specific organisational context. The NoA model of norm-governed practical reasoning introduces a labelling mechanism to focus the deliberation of the agent on such violations. The deliberation of the agent is informed about inconsistencies between potential candidate actions it could deploy to fulfill its obligations and its currently held set of norms. Instead of simply filtering out inconsistent candidates for action, a label is attached to each candidate action containing a rich set of information, cross-referencing options for action (plans) with motivating obligations and possible norms that are inconsistent with such an action. With that, an agent may attempt to comply with a specific norm, but still violate others. By informing and focussing the deliberation of the agent on such cases of inconsistencies, the agent can use certain resolution strategies such as, for example, engaging in a

dialogue with a normative authority to reach an agreement about "relaxing" its social constraints. Or it can decide to simply violate a norm. The mechanisms within NoA to identify such violations is an important step in enabling an agent to reason about norm violations.

References

1. R. Bordini and J. Huebner. *Jason: A Java-based AgentSpeak Interpreter used with Saci for Multi-Agent Distribution over the Net, Manual*, 2004.
2. J. Broersen, M. Dastani, J. Hulstijn, Z. Huang, and L. van der Torre. The BOID architecture: Conflicts between Beliefs, Obligations, Intentions and Desires. In *Proceedings of Autonomous Agents 2001*, pages 9–16, 2001.
3. C. Castelfranchi. Modelling Social Action for AI Agents. *Artificial Intelligence*, 103:157–182, 1998.
4. C Castelfranchi, F. Dignum, C. Jonker, and J. Treur. *Deliberate normative Agents: Principles and Architecture*, volume Intelligent Agents VI LNAI 1757 of *Lecture Notes in Artificial Intelligence*, pages 364–378. Springer-Verlag, 2000.
5. R. Conte, R. Falcone, and G. Sartor. Agents and Norms: How to fill the Gap? *Artificial Intelligence and Law*, 7(1), March 1999.
6. F. Dignum. Autonomous Agents with Norms. *Artificial Intelligence and Law*, 7:69–79, 1999.
7. F. Dignum, D. Kinny, and L. Sonenberg. From Desires, Obligations and Norms to Goals. *Cognitive Science Quarterly*, 2(3–4):407–430, 2002.
8. R.J Firby. An Investigation into Reactive Planning in Complex Domains. In *Proceedings of the National Conference on Artificial Intellgence (AAAI)*, pages 809–815, 1987.
9. C.L. Forgy. Rete: A Fast Algorithm for the Many Pattern / Many Object Pattern Match Problem. *Artificial Intelligence*, 19:17–37, 1982.
10. M.P. Georgeff and A. Lansky. Reactive Reasoning and Planning. In *Proceedings AAAI-87*, pages 677–682, Seattle, WA, 1987.
11. A.J.I. Jones and M. Sergot. A Formal Characterisation of Institutionalised Power. *Journal of the IGPL*, 4(3):429–445, 1996.
12. M.J. Kollingbaum. *Norm-governed Practical Reasoning Agents*. PhD thesis, University of Aberdeen, 2005.
13. M.J. Kollingbaum and T.J. Norman. Supervised Interaction - creating a Web of Trust for Contracting Agents in Electronic Environments. In C. Castelfranchi and W. Johnson, editors, *First International Joint Conference on Autonomous Agents and Multi-Agent Systems AAMAS 2002*, pages 272–279. ACM Press, 2002.
14. M.J. Kollingbaum and T.J. Norman. Strategies for Resolving Norm Conflict in Practical Reasoning. In *ECAI Workshop CEAS 2004*, 2004.
15. F. Lopez y Lopez, M. Luck, and M. dInverno. Constraining autonomy through norms. In *In Proceedings of the First International Joint Conference on Autonomous Agents and Multi-agent Systems*, pages 647–681, 2002.
16. T.J. Norman and C.A. Reed. Delegation and Responsibility. In C. Castelfranchi and Y. Lesperance, editors, *Intelligent Agents VII, LNAI 1986*, volume 1986 of *Lecture Notes in Artificial Intelligence*, pages 136–149. Springer-Verlag, 2001.
17. O. Pacheco and J. Carmo. A Role Based Model for the Normative Specification of Organized Collective Agency and Agents Interaction. *Autonomous Agents and Multi-Agent Systems*, 6(2):145–184, 2001.
18. M. Wooldridge. *Reasoning about Rational Agents*. MIT Press, 2000.