A Computational Framework for Institutional Agency

Guido Governatori[*]
School of Information Technology and Electrical Engineering, The University of Queensland, Brisbane, Queensland, QLD 4072, Australia. (guido@itee.uq.edu.au)

Antonino Rotolo[†]
CIRSFID, Law Faculty, University of Bologna, Via Galliera, 3, 40121 Bologna, Italy. (rotolo@cirsfid.unibo.it)

Abstract. This paper provides a computational framework, based on Defeasible Logic, to capture some aspects of institutional agency. Our background is Kanger-Lindahl-Pörn account of organised interaction, which describes this interaction within a multi-modal logical setting. This work focuses in particular on the notions of counts-as link and on those of attempt and of personal and direct action to realise states of affairs. We show how standard Defeasible Logic can be extended to represent these concepts: the resulting system preserves some basic properties commonly attributed to them. In addition, the framework enjoys nice computational properties, as it turns out that the extension of any theory can be computed in time linear to the size of the theory itself.

Keywords: Institutional agency, agent societies, counts-as, defeasible logic

1. Background and Motivation

Recent works on agents and their societies assume that as in human societies, also in artificial societies normative concepts may play a decisive role, allowing for the flexible co-ordination of intelligent autonomous agents [12, 40]. In line with this trend, in [16] the authors of this paper and other colleagues proposed to model organisations of agents in terms of rule-based normative systems; accordingly, an organisation should be characterised by specifying the normative positions relevant to design its structure. These positions include not only duties and permissions, but also powers, as for instance powers of creating further normative positions on the head of other agents. Technically, in this paper we develop a formal machinery to capture some building blocks among those analysed in [16]. In particular, we focus on some basic aspects of agency and institutionalised power. These concepts are embedded in a non-monotonic framework based on Defeasible Logic (DL).

As in [16], the background of this paper is Kanger-Lindahl-Pörn [31, 32, 42] theoretical account of organised interaction (see [14]). The main references here are some recent contributions [45, 29, 30], which have enriched

[*] This work was supported by the Australian Research Council under the Discovery Project No. DP0558854.

[†] This work was supported by the European project for Standardized Transparent Representations in order to Extend Legal Accessibility (ESTRELLA, IST-4-027655).

Artificial Intelligence and Law, forthcoming

this framework with some substantial refinements. The basic idea is to describe agents' interaction within a multi-modal logical setting. The resulting view is abstract but flexible, as social agency is captured by simply combining different modal operators, each of them corresponding to notions such as those of action, power, obligation, and belief.

The paper is confined to two basic aspects of the above line of research: the modal notion of agency and that of institutionalised power.

Despite some limitations (see [47, 43]), modal logic of agency [14] is still very much adopted thanks to its flexibility, as actions are simply taken to be relationships between agents and states of affairs. We will focus on two well-known agency notions. The first is the idea of personal and direct action to realise a state of affairs. In the mentioned logical framework, it is formalised by the modal operator E, such that a formula like E_iA means that the agent i brings it about that A. Different axiomatisations have been provided for it [20]. Here we will consider two basic logical properties of this operator[1]:

$$E_iA \rightarrow A \tag{T}$$

$$E_iE_jA \rightarrow \neg E_iA \tag{EE¬E}$$

Schema (T) expresses the successfulness of actions that is behind the common reading of the "bring about" concept. Schema (EE¬E) is a specific axiom advanced, for example, in [45]. The brings-it-about operator expresses actions performed directly and personally. Hence, (EE¬E) states a principle of rationality for modelling co-ordination in institutional organisations: it is counter-intuitive that the same agent brings it about that A and brings it about that somebody else achieves A.

The second aspect of agency considered here is that of attempt, formalised by the operator H [45, 30]. H_iA says that i attempts to make it the case that A. The operator H_i is not necessarily successful. Here we will simply assume that each successful action is also an attempt[2]:

$$E_iA \rightarrow H_iA \tag{1}$$

[1] Besides these schemata, the logic for E is usually closed under logical equivalence. Other common properties, which are not considered here, correspond to $\neg E_i\top$ (No) and $(E_iA \wedge E_iB) \rightarrow E_i(A \wedge B)$ (C).

[2] Besides that, H usually enjoys (C) and is closed under logical equivalence. In [45, 30] a third operator G has been also defined, corresponding to the idea of indirect successful action. The reading of G_iA is that i ensures that A. G enjoys the same general properties of E. However, instead of (EE¬E), it is adopted $G_iG_jA \rightarrow G_iA$ (GGG). (GGG) differentiates G from E insofar as the former is meant to represent indirect actions. This operator will not be considered explicitly here. Besides its most general reading, it can be argued that G_iA, if strictly analysed in terms of agency, can be thought as any iteration of the form $E_iE_{i_1}\dots E_{i_n}A$, where $n \geq 0$. Notice that this specific reading of G is compatible with that originally assigned to it, since the schemas $E_iA \rightarrow G_iA$, $E_iE_jA \rightarrow E_iG_jA$ and $G_iE_jA \rightarrow G_iG_jA$ are adopted in [45].

Let us focus now on the idea of institutionalised power. This notion is central for describing norm-governed organisations of agents and comes from the distinction between the practical ability to realise a state of affairs –which is not considered in this paper [14, 20]– and the institutional power to do this [34]. For example, if in an auction *i* raises one hand, this implies that the act of making a bid is also obtained. In principle, this kind of ability should be distinguished from the practical capacity to obtain a certain state of affairs. The attempt to make a bid may not be successful: its being successful, within the institutional context (the auction), depends on whether that institution makes it effective. It is up to institutional (constitutive) *rules* to establish whether *i*'s act makes so that a bid is effective or not, namely, that *i*'s act *counts as* bidding.

The logical nature of this kind of rules has been recently investigated following different directions (see, e.g., [23, 6, 29, 16]). Many of these approaches explicitly recognise that constitutive rules are defeasible. In fact, it is intuitive that, e.g., if the agent *i* raises one hand, this may count as making a bid but this does not hold if *i* raises one hand *and* scratches his own head. This paper will adopt the approach provided in [16]. In that work, it is argued that constitutive rules of the form "X counts as Y in the context C" [46] are represented within a conditional logic enjoying at least the basic properties (Reflexivity, Cut, and Cautious Monotonicity) of cumulative reasoning (system **CU** [4]). In [16] the logic was enriched by the modality D_s –originally introduced in [29] but with a different meaning– to represent institutional facts. In that specific perspective, the expression "A counts as B in the institution s", formally $A \Rightarrow_s B$, was stated to be equivalent by definition to $(A \Rrightarrow D_s B) \wedge (D_s A \Rrightarrow D_s B)$, where $\Rrightarrow$ is the conditional obeying the principles of cumulative reasoning. This view is meant to capture the fact that counts-as rules may specify when (1) a brute fact (e.g., destroying the receipt) counts as a type of institutional act (e.g., freeing the debtor from his obligation), and (2) an institutional act (e.g., a contract made by person j in the name of person k) has the same effects of another institutional act (e.g., a contract made by k). D_s represents the domain of institutional facts and it corresponds to a classical non-normal modality. However, in this paper we will not consider the modality D_s, as it is mainly relevant when more institutional contexts are compared and so the modality is used to mark the different institutions where institutional facts hold. Accordingly, leaving aside D_s, the modelling of counts-as rules will simply amount in this paper to dealing with cumulative reasoning.

Notice that the framework we have just recalled is able to capture some composite concepts regarding the normative co-ordination of agents. In particular, [16] shows that the introduction of the notion of proclamation allows to account for the ideas of declarative power and delegation [9, 35]. The logical representation of these ideas has a counts-as structure. Institutional proclamations are formalised by the modal operator *proc*: the expression $proc_i A$ means that agent i proclaims A[3]. The combination of *proc*, agency operators, and the counts-as link enables us to capture two forms of normative delegation, intended as kinds of true representation [16]. The first is $proc_j(proc_i A) \Rightarrow_s E_j(proc_i A)$, that is, when j proclaims that i proclaims that A, this counts as j's making so that i proclaims that A[4]. In addition, we can have $proc_j(E_i A) \Rightarrow_s E_j(E_i A)$. This type of representation is necessary when the representative substitutes a principal which would not be able to perform directly the activity which is delegated to the representative.

Although the above building blocks supply an intermediate level of fine conceptual analysis, it seems difficult to use them directly for implementation. This is due to the inherent computational complexity of multi-modal logics (see, e.g., [25]). In general, the addition of modal operators to the classical propositional base leads to the increase of complexity of the logic. This is mainly due to: (1) the rules to introduce modalities, (2) the axioms governing the behaviour of modalities and their mutual interaction. But something similar applies as well to the logic of counts-as, due to the well-known computational limits of conditional logics (see, e.g., [4]).

The aim of this paper is to show how to introduce modalities in a (computationally oriented) non-monotonic formalism (Defeasible Logic), and then to apply this methodology to deal with the mentioned basic properties of institutional agency. In this perspective, some basic patterns of defeasible reasoning will be extended to account for the institutional dynamics insofar as counts-as links interact with the notions of direct action and attempt. Notice that the use of DL to model the counts-as link is immediate, as its basic form corresponds to cumulative reasoning enjoying the properties we previously mentioned [5]. As we will see, extending DL to treat modal logic of agency requires some adjustments and integrations.

The layout of the paper is as follows. Section 2 makes provision of the basics of standard DL. In Section 3 we show how DL can be extended to deal with the notion of institutional agency we previously recalled; the formal system will be illustrated with the help of some simple examples. In Section 4 we

[3] As is well-known, agent communication concepts play an important role in modelling agent coordination. In [16] the speech act of proclaiming has been defined to capture some minimal properties of all speech acts that are intended to modify the institutional world. However, notice that in this paper we will make a trivial use of the *proc* operator, as we will not model its logical properties. We will simply use it to denote acts of proclamation. At any rate, the logic of *proc* is characterised by very minimal properties: it is closed under logical equivalence and includes at least the axiom $(proc_i A \wedge proc_i B) \equiv proc_i(A \wedge B)$. Of course, *proc* is not necessarily successful: $proc_i A$ is just an attempt to achieve A. Whether it is successful or not, within a certain institution s, depends on whether s makes it effective by means of appropriate counts-as rules.

[4] Of course, the achievement of A will depend on the presence on another rule which states that $proc_i A$ counts as $E_i A$.

provide some formal results of our system. Section 5 presents a discussion of some related work, while Section 6 provides some directions for future work. The interested reader will find an Appendix with proofs or proof sketches of the formal properties mentioned in Section 4.

2. Overview of Defeasible Logic

DL is a simple, efficient but flexible non-monotonic formalism which has been proven able to deal with many different intuitions of non-monotonic reasoning [3]. Here we propose a non-monotonic logic of agency based on the framework for DL developed in [1].

It is not possible to give here a complete formal description of the logic. We hope to give enough information to make the discussion intelligible and we refer the reader to [37, 2] for more thorough treatments. As usual with non-monotonic reasoning, we have to specify 1) how to represent a knowledge base and 2) the inference mechanism.

Accordingly a defeasible theory D is a structure $(F,R,>)$ where F is a finite set of facts, R a finite set of rules (either strict, defeasible, or defeater), and $>$ a binary relation (superiority relation) over R.

Facts are indisputable statements. *Strict rules* are rules in the classical sense: whenever the premises are indisputable so is the conclusion; *defeasible rules* are rules that can be defeated by contrary evidence; and *defeaters* are rules that cannot be used to draw any conclusions. Their only use is to prevent some conclusions. In other words, they are used to defeat some defeasible rules by producing evidence to the contrary. The *superiority relation* among rules is used to define priorities among rules, that is, where one rule may override the conclusion of another rule.

A *rule* r consists of its *antecedent* (or *body*) $A(r)$ ($A(r)$ may be omitted if it is the empty set) which is a finite set of literals, an arrow, and its *consequent* (or *head*) $C(r)$ which is a literal. Given a set R of rules, we denote the set of all strict rules in R by R_s, the set of strict and defeasible rules in R by R_{sd}, the set of defeasible rules in R by R_d, and the set of defeaters in R by R_{dft}. $R[q]$ denotes the set of rules in R with consequent q. If q is a literal, $\sim q$ denotes the complementary literal (if q is a positive literal p then $\sim q$ is $\neg p$; and if q is $\neg p$, then $\sim q$ is p).

A *conclusion* of D is a tagged literal and can be either:

$+\Delta q$: q is definitely provable in D (i.e., using only facts and strict rules).

$-\Delta q$ meaning that we have proved that q is not definitely provable in D.

$+\partial q$ meaning that q is defeasibly provable in D.

$-\partial q$ meaning that we have proved that q is not defeasibly provable in D.

Provability is based on the concept of a *derivation* (or proof) in D. A derivation is a finite sequence $P = (P(1),\ldots,P(n))$ of tagged literals satisfying four conditions (which correspond to inference rules for each of the four kinds of conclusion). $P(1..n)$ denotes the initial part of the sequence P of length i.

$+\Delta$: If $P(n+1) = +\Delta q$ then
(1) $q \in F$ or
(2) $\exists r \in R_s[q] : \forall a \in A(r) + \Delta a \in P(1..n)$

$-\Delta$: If $P(n+1) = -\Delta q$ then
(1) $q \notin F$ and
(2) $\forall r \in R_s[q] \exists a \in A(r) : -\Delta a \in P(1..n)$

The intuition behind the proof conditions is to give conditions under which we can append a (tagged) literal at the end of a derivation. The definition of Δ describes forward chaining of strict rules or, in other terms, it corresponds to Modus Ponens for strict rules. Accordingly, for a literal q to be definitely provable we need to find a strict rule with head q, of which all antecedents have been definitely proved previously. To establish that q cannot be proven definitely we must establish that for every strict rule with head q there is at least one antecedent which has been shown to be non-provable.

The inference conditions for negative proof tags are derived from the inference conditions for the corresponding positive proof tag by applying the Principle of Strong Negation introduced in [1]. The strong negation of a formula is closely related to the function that simplifies a formula by moving all negations to an innermost position in the resulting formula and replace the positive tags with the respective negative tags and viceversa. For example, if in a proof condition for $+\#$ we have $\forall s(+\#_1 A(s) \wedge -\#_2 B(s))$, the strong negation of the condition is $\exists s(-\#_1 \mathit{StrongNegation}(A(s)) or + \#_2 \mathit{StrongNegation}(B(s)))$. Accordingly, in what follows, we will often list only the positive version of the inference rules.

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

Let us work through this condition. To show that q is provable defeasibly we have two choices: (1) We show that q is already definitely provable; or (2) we need to argue using the defeasible part of D as well. In particular, we require that there must be a strict or defeasible rule with head q which can be applied (2.2). But now we need to consider possible “attacks”, i.e., reasoning chains in support of $\sim q$. To be more specific: to prove q defeasibly we must show that $\sim q$ is not definitely provable (2.1). Also (2.3) we must consider the set of all rules which are not known to be inapplicable and which have head $\sim q$ (note that here we consider defeaters, too, whereas they could not be used to support the conclusion q; this is in line with the motivation of defeaters given earlier). Essentially each such rule s attacks the conclusion q. For q to be provable, each such rule s must be counterattacked by a rule t with head q with the following properties: (i) t must be applicable at this point, and (ii) t must be stronger than s. Thus each attack on the conclusion q must be counterattacked by a stronger rule. In other words, r and the rules t form a team (for q) that defeats the rules s. In an analogous manner we can define $-\partial q$ (see, for example [2]). The purpose of the $-\partial$ inference rules is to establish that it is not possible to prove $+\partial$. This rule is defined in such a way that all the possibilities for proving $+\partial q$ (for example) are explored and shown to fail before $-\partial q$ can be concluded. Thus a conclusion tagged with $-\partial$ is the outcome of a constructive proof that the corresponding positive conclusion cannot be obtained.

We illustrate how the proof conditions work with the help of the following theory:

$$\begin{aligned}
F &= \{A, C\}\\
R &= \{r_1 : A \Rightarrow B,\\
&\quad r_2 : C \Rightarrow E,\\
&\quad r_3 : A, D \Rightarrow \neg B,\\
&\quad r_4 : E \Rightarrow \neg B\}\\
&\quad \{r_3 > r_1, r_1 > r_4\}
\end{aligned}$$

Since $A, C \in F$, we have $+\Delta A$ and $+\Delta C$; by clause (1) we also have $+\partial A$ and $+\partial C$. To prove $+\partial B$ we have to ensure that its negation cannot be definitely proved (i.e., proved using only facts and strict rules). This follows immediately since $\neg B$ is not a fact and there are no strict rules for $\neg B$. r_1 is a defeasible rule for B whose antecedent $A(r_1)$ is $\{A\}$, and we have already proved $+\partial A$, thus clause (2.2) is satisfied. We have two rules for $\neg B$, namely r_3 and r_4. Using the same reasoning we can show that $+\partial E$ (clause 2.3 for the derivation of $+\partial E$ is vacuously satisfied since there are no rules for $\neg E$). For r_3 we have that $-\partial D$ ($D \notin F$ and there are no rules for it), so clause 2.3 is satisfied for r_3 based on clause 2.3.1. For r_4 we can use the superiority relation $r_1 > r_4$, to exhibit a rule (i.e., r_1) for B which is stronger than r_4. Thus clause 2.3 is true also for r_4, and then we are justified to append $+\partial B$ at the end of the derivation.

Sometimes all we want to know is whether a literal is *supported*, that is if there is a chain of reasoning that would lead to a conclusion in absence of conflicts. This notion is captured by the following proof conditions:

$+\Sigma$: if $P(n+1) = +\Sigma p$ then
(1) $+\Delta p \in P(1..n)$ or
(2) $\exists r \in R_{sd}[p] : \forall a \in A(r) + \Sigma a \in P(1..n)$

The notion of support corresponds to monotonic proofs using both the monotonic and non-monotonic parts of defeasible theories.

3. A Computational Framework for Institutional Agency

As we have seen in Section 1 multi-modal logics have been put forward to capture the intensional nature of (institutional) agency. Usually multi-modal logics are extensions of classical propositional logic with some intensional operators. Thus any multi-modal logic should account for three components: (1) the underlying logical structure of the propositional base; (2) the logic behaviour of the modal operators; and (3) the relationships among the modal operators. Alas, as is well-known, classical propositional logic is not well suited to deal with real life scenarios. The main reason is that the descriptions of real-life cases are, very often, partial and somewhat unreliable. In such circumstances classical propositional logic might produce counter-intuitive results insofar as it requires complete and consistent information. Hence modal logics based on classical propositional logic are doomed to suffer from the same problems.

On the other hand the logic should specify how modalities can be introduced and manipulated. Some common rules for modalities are, e.g., Necessitation (i.e., $\vdash A / \vdash \Box A$) and RM (i.e., $\vdash A \rightarrow B / \vdash \Box A \rightarrow \Box B$) [10]. Both dictate conditions for introducing modalities in contrast with the analysis of institutional agency as outlined in Section 1. To comply with the properties of this notion, in the setting provided by DL we have to set 1) the rules describing the logical inferences and 2) the rules to introduce the modal operators of agency E_i (*the agent i brings about*), and H_i (*the agent i attempts*). Accordingly we will consider two types of rules (strict, defeasible, and defeaters): a set of rules for the notion of *counts-as*, and a set of rules for the notion of *results-in*.

Since we want to reason about actions we extend the language of DL with a set of action symbols; we will use $\alpha_i, \beta_i, \gamma_i$ to denote atomic actions. The meaning of an action symbol, for example α_i, is that the action corresponding to it has been performed by agent i, while we use $\neg\alpha_i$ to denote that the action described by α_i has not been performed. Given the modal operators E_i, H_i, and $proc_i$ we form new literals as follows: i) if l is a literal then $proc_i l$ is a literal; ii) if l is a literal then $E_i l$, $\neg E_i l$, $H_i l$ and $\neg H_i l$ are literals if l is different from $E_i m$, $\neg E_i m$, $H_i m$ and $\neg H_i m$, for some literal m. We will use Lit to denote the set of literals.

In this perspective a defeasible institutional action theory is a structure $I = (A, F, R^c, \{R^i\}_{i\in A}, >)$ where, A is a finite set of agents, F is a set of facts, R^c is a set of counts-as rules (i.e., $\rightarrow_c$, $\Rightarrow_c$, $\leadsto_c$), $\{R^i\}_{i\in A}$ is a family of sets of results-in rules (i.e., $\rightarrow^i$, $\Rightarrow^i$, $\leadsto^i$, $\forall i \in A$), and $>$, the superiority relation, is a binary relation over the set of rules (i.e., $> \subseteq (R^c \cup R^A)^2$), where $R^A = \bigcup_{i\in A} R^i$.

The intuition is that, given an institution, F consists of the description of the raw institutional facts, either in form of states of affairs (literal and modal literal) and actions that have been performed. R^c describes the basic inference mechanism internal to an institution, while R^A encodes the transitions from state to state occurring as the results of actions performed by the agents within the organisation. The rules in R^A are used to introduce modal operators. To capture these notions we impose some restrictions on the form of rules: literals of the form $E_i l$, $\neg E_i l$, $H_i l$ and $\neg H_i l$ are not permitted in the consequent of results-in rules for i, while actions symbols are not permitted in the consequent of results-in rules. The first restriction is motivated from the fact that 1) results-in rules are the rules to introduce the modalities and in the present context sequences of modalities for the *same* agent are useless[5] 2) counts-as rules make possible the derivation of institutional actions (modalised literals) only when they follow from specific actions (intentionally) performed by the agent. The second restriction is due to the idea that results-in rules describe, as their name suggests, the results of actions, not actions themselves.

Let us see by means of some examples the intuition behind this formalism. We focus here on defeasible rules but similar remarks can be applied to the other kinds of rules. Suppose the agent i is acting in the context of an auction. Then we may have cases like the following[6]:

$$\mathbf{bids_i},\ \mathit{auction\_begun} \Rightarrow^i \mathit{offer} \tag{2}$$

This rule is an example corresponding to the introduction of the modality E_i. In fact, agent i's fulfilment of the conditions in the antecedent produces the occurrence of *offer*: agent i's action of bidding has the result that i has made an offer. As we will see, if *offer* can be derived, this permits the introduction of $E_i(\mathit{offer})$.

$$\mathit{auction\_begun} \Rightarrow^i \neg \mathit{offer} \tag{3}$$

The example above does not specify any action in the antecedent (empty action). This means that, when the auction is begun, agent i's refraining from doing any action has the result to have no offer. In logical terms, also this case can lead to the introduction of E[7].

[5] An expression like E_iE_iA is useless since it is equivalent to E_iA.

[6] Bold type expressions correspond to action symbols, the italicised ones to state of affairs.

[7] The ideas of empty action and refraining from doing a specific action should not be confused with what it is expressed by $\neg E_iA$. As we will see, this last corresponds to the non-derivability of A within I, which can depend also on reasons that have nothing to do with agent i's refraining from acting to realise A.

Now suppose that agent i is acting on behalf of agent j.

$$\mathbf{bids_i}, \mathit{proc}_i(E_j \mathit{offer}) \Rightarrow^j \mathit{offer} \tag{4}$$

This formula means that the fact that agent i makes a bid and proclaims that agent j makes the offer permits to introduce E_j, namely that $E_j \mathit{offer}$.

Let us consider examples of counts-as rules.

$$\mathbf{raises\_hand_i}, \mathit{auction\_begun} \Rightarrow_c \mathbf{bids_i} \tag{5}$$

This rule says that that agent i's action of raising one hand counts as agent i's action of bidding, when the auction is begun.

$$\mathit{auction\_begun}, E_i(\mathit{offer}) \Rightarrow_c \neg\mathbf{raises\_offer_i} \tag{6}$$

Also here we have agent i's generic refraining from doing any action in the antecedent. This example represents the institutional connection linking such refraining, and *the fact* that agent i made an offer when the auction is begun, to agent i's specific refraining from raising a new offer. Notice that the same meaning is assigned to counts-as rules where the antecedent contains only non-modal literals.

$$\mathit{auction\_begun}, \mathbf{raises\_hand_i} \Rightarrow_c \mathit{offer} \tag{7}$$

This rule is an example of the institutional analogous of results-in rules, where an action and a state of affairs occur respectively in their antecedent and consequent. However, in this case the result is an institutional fact and follows by convention only within the institution. In fact, that an offer is a consequence of agent i's raising one hand is not a simple matter of agent i's action results. The attempt of agent i to make an offer by raising the hand is effective only if the institution recognises this.

Let us see a couple of examples with more than one agent. As above, agent i is acting on behalf of agent j.

$$\mathit{proc}_i(E_j \mathit{offer}) \Rightarrow_c E_i(E_j \mathit{offer}) \tag{8}$$

This rule says that if agent i proclaims that agent j makes an offer, then this counts as agent i brings it about that agent j makes such an offer.

$$\mathit{proc}_i(E_j \mathit{offer}), \mathbf{raises\_hand}_i \Rightarrow_c \mathbf{bids}_j \tag{9}$$

Rule (9) expresses that agent i's proclamation that agent j makes an offer counts as agent j's action of bidding.

It is worth noting that no explicit reference is made here to the modality D_s [16], as discussed in Section 1. In fact, the present setting accounts for the idea of institution in terms a special kind of defeasible theory. Each institutional action theory I encodes in itself all possible inferences that can be drawn within the domain of institutional facts relative to a given s. This means that s may be identified with I since all action results are obtained within such a domain of facts. In other words, the introduction of the modality D_s corresponds here to the general definition of derivability using counts-as and results-in rules. Technically, counts-as rules are meant to capture the case $D_sA \Rrightarrow D_sB$ mentioned in Section 1. Roughly speaking, on the other hand, the case $A \Rrightarrow D_sB$ will be treated as a special kind of results-in rule, where the manipulation of the consequent is made under the constraints designed to account for the idea of institutional consequence. This is just a technical device to differentiate the two cases: the logical behaviour of the counts-as link as described in [16] is here encoded in the whole formal machinery corresponding to the definitions of the proof conditions.

Before moving to the proof conditions we have to introduce the notion of complementary literals. In standard DL two literals are complementary to each other if one is the negation of the other. This means that the two literals cannot hold at the same time. The extension with modal operators has to consider when modal literals are in conflict with each other. Since the agency operator E is successful (i.e., $E_iA \rightarrow A$), it is not possible to have together E_iA for some agent i and A. In a similar way we have to capture the strong notion of agency we intend to model within our framework, i.e., where $E_iE_jA \rightarrow \neg E_iA$.

Given an atomic literal p we use Ep to denote any string $E_{i_1} \dots E_{i_n}p$ where $E_{i_1} \dots E_{i_n}$ is a (possibly empty) string of positive modal operators such that $\forall 1 \leq j < n, i_j \neq i_{j+1}$. Let l be a literal, $\mathscr{C}(l)$ denotes the complement of l, i.e., the set of literal that cannot be true when l is.

- if $l = p$, then $\mathscr{C}(l) = \{E{\sim}p\}$;
- if $l = E_ip$, then $\mathscr{C}(l) = \{E{\sim}p, E\neg E_ip\}$;
- if $l = \neg E_ip$, then $\mathscr{C}(l) = \{EE_ip\}$.

The meaning of the first condition is that if p is true then no agent prevented p; for the second condition we have that if an agent i has realised p, then no other agent prevented p and no agent prevented i from realising p. Finally if an agent i has refrained from doing p, then it is not possible that some other agents achieved that i did p.

We are now ready to give the proof conditions for institutional agency. We begin with the conditions for counts-as derivations.

$+\Delta_c$: if $P(n) = +\Delta_c p$, then either:
(1) $Ep \in F$ or
(2) $\exists r \in R_s[Ep]: \ \forall a, \alpha, E_jb \in A(r), +\Delta_c a, +\Delta_c \alpha, +\Delta_j b \in P(1..n)$.

The conditions are in essence the same as those for definite conclusions for DL given in Section 2. The first difference is in clause (1) where to prove a literal p we can use any fact of the form Ep, let us say, for example E_iE_jp. This is due to the successfulness of the E_i operator (see Section 1); in the limit case E is the empty sequence, and we recover the basic condition of DL. Similarly, in clause (2) we look for applicable counts-as rules for Ep instead of simply p. The last difference is that a rule is now applicable if the literals in the antecedent are proved with the right mode: $+\Delta_c$ for unmodalised literals and action literals and $+\Delta_i$ for modal literals whose main operator is E_i. This follows the intuition that modal rules for agent i behave as introduction rules for the modal operator E_i.

$-\Delta_c p$: if $P(n) = -\Delta_c p$, then both:
(1) $Ep \notin F$, and
(2) $\forall r \in R_s[Ep]$: $\exists a \in A(r), -\Delta_c a \in P(1..n)$ or
$\exists \alpha \in A(r), -\Delta_c \alpha \in P(1,,n)$ or
$\exists E_i b \in A(r), -\Delta_i b \in P(1..n)$.

The intuiton for the condition for $-\Delta_c$ is similar to that of $-\Delta$ with the remarks about the condition for $+\Delta_c$. The only issue we want to point out is that to reject a rule (to show that a rule cannot be applied) we have to show that there is at least one literal in the antecedent which is not provable with the appropriate mode. Finally, it is easy to verify that the condition for $-\Delta_c$ is the strong negation of the condition for $+\Delta_c$.

We can introduce the conditions for defeasible derivations. Again, the basic intuition is the same as DL with the additional considerations as the conditions for strict derivations.

$+\partial_c$: if $P(n) = +\partial_c p$, then:
(1) $+\Delta_c p \in P(1..n)$, or
(2.1) $-\Delta\mathscr{C}(p) \in P(1..n)$ and
(2.2) $\exists r \in R_{sd}[Ep]$ $\forall a, \alpha, E_i b \in A(r)$:
$+\partial_c a, +\partial_c \alpha, +\partial_i b \in P(1..n)$ and
(2.3) $\forall s \in R[\mathscr{C}(p)]$: either
(2.3.1) $\exists a \in A(s), -\partial_c a \in P(1..n)$ or
(2.3.2) $\exists \alpha \in A(s), -\partial_c \alpha \in P(1..n)$ or
(2.3.3) $\exists E_i b \in A(s), -\partial_i b \in P(1..n)$ or
(2.3.4) $\exists t \in R[Ep]$ $\forall a, \alpha, E_i b \in A(t)$:
$+\partial_c a, +\partial_c \alpha, +\partial_i b \in P(1..n)$ and $t > s$.

The conditions for $-\partial_i$ are obtained from that for $+\partial_i$ using the mentioned principle of strong negation.

$-\partial_c$: if $P(n) = -\partial_c p$, then:
(1) $-\Delta_c p \in P(1..n)$ and
(2.1) $+\Delta_c \mathscr{C}(p) \in P(1..n)$ or
(2.2) $\forall r \in R_{sd}[Ep]$: either
(2.2.1) $\exists a \in A(r) : -\partial_c a \in P(1..n)$ or
(2.2.2) $\exists \alpha \in A(r) : -\partial_c \alpha \in P(1..n)$ or
(2.2.3) $\exists E_i b \in A(r) : -\partial_i b \in P(1..n)$ and
(2.3) $\exists s \in R[\mathscr{C}(p)]\ \forall a, \alpha, E_i b \in A(r)$:
$+\partial_c a, +\partial_c \alpha, +\partial_i b \in P(1..n)$ and
(2.3.1) $\forall t \in R[Ep]$: either $t \not> s$ or
$\exists a \in A(t), -\partial_c a \in P(1..n)$ or
$\exists \alpha \in A(t), -\partial_c \alpha \in P(1..n)$ or
$\exists E_i b \in A(t), -\partial_i b \in P(1..n).$.

To conclude the presentation of the proof conditions for counts-as conclusions we give the conditions for support.

$+\Sigma_c$: if $P(n) = +\Sigma_c p$, then
(1) $Ep \in F$ or
(2) $\exists r \in R_{sd}[p]\ \forall a, \alpha, E_i b \in A(r)$:
$+\Sigma_c a, +\Sigma_c \alpha, +\Sigma_i b \in P(1..n)$.

$-\Sigma_c$: if $P(n) = -\Sigma_c p$, then
(1) $Ep \notin F$ and
(2) $\forall r \in R_{sd}[p]$: either
$\exists a \in A(r), -\Sigma_c a \in P(1..n)$ or
$\exists \alpha \in A(r), -\Sigma_c \alpha \in P(1..n)$ or
$\exists E_i b \in A(r), -\Sigma_i b \in P(1..n)$.

The conditions are the same as $+\Delta$ and $-\Delta$; the only difference is that for support we consider both strict and defeasible rules instead of only strict rules, and the two conditions are the strong negations of each other.

The conditions for derivations involving results-in rules are more complicated since we have to cater for more possibilities. First of all we have that $I \vdash E_i p$ if either $I \vdash +\Delta_i p$ or $I \vdash +\partial_i p$,[8] and $I \vdash H_i p$ if $I \vdash +\Sigma_i p$. In other words it is possible to derive $E_i p$ if we have either a strict or defeasible derivation of p using both results-in and counts-as rules, and that agent i (in an institution I) attempts p ($H_i p$) if I supports p using counts-as ad results-in rules. The output of a results-in rule produces E_i modal literals, and we have seen in Section 1 that the E_i operator is a success operator; therefore we add the conditions that it is possible to derive $+\Delta_c p$ from $+\Delta_i p$ and $+\partial_c p$ from $+\partial_i p$. In particular, it is worth noting that a counts-as rule can be used as it were a results-in rule if

[8] It is possible to prove $E_i p$ from a theory I also in the case that $I \vdash +\Delta_c E_i p$ or $I \vdash +\partial_c E_i p$ and similarly for H_i.

all the literals occurring in its antecedent are proved as appropriate results-in conclusions. In this case, we say that we have a *conversion* from a counts-as rule into a results-in rule. For example, suppose we have that

$$\mathit{auction\_begun},\ \mathbf{raises\_hand_i} \Rightarrow_c \mathit{offer}$$

If we have $\mathbf{raises\_hand_i}$ and prove *auction_begun* as a results-in conclusion, in particular as $E_i\mathit{auction\_begun}$, then we can say that agent i brings *offer* about, namely that $E_i\mathit{offer}$. More on conversions can be found in [22].

In the same way we have that $-\partial_i p$ corresponds to $\neg E_i p$ and $-\Sigma_i p$ to $\neg H_i p$ in addition to the cases where the modal literal is provable with a counts-as derivation (e.g., $I \vdash +\partial_c E_i p$). This is in agreement with the principle of strong negation used to define the inference conditions.

$+\Delta_i$: if $P(n+1) = +\Delta_i p$ then
(1) $EE_i p \in F$; or
(2) $+\Delta_c E_i p \in P(1..n)$; or
(3) $\exists r \in R_s^i[p]\ \forall a, \alpha, E_j b \in A(r)$:
$+\Delta_i a, +\Delta_i \alpha, +\Delta_j b \in P(1..n)$ or
(4) $\exists r \in R_s^c[p]$: $\exists a \in A(r) \cap \mathrm{Lit}$, and
$\forall a, \alpha \in A(r)$: $+\Delta_i a, +\Delta_c \alpha \in P(1..n)$.

To prove non-defeasible brings-it-about, we need either that it is given as a fact (or the set of facts contains a chain of brings-it-about operators where the last one is E_i) (1), or that $E_i p$ has been proved using counts-as rules, or that we have a strict rule for results-in (an irrevocable policy) whose antecedent is indisputable (3). However we have another case (4): if an agent knows that B is an indisputable consequence of A in the institution (it is always the case that A counts as B), and it produces A, then it must realise B.

$-\Delta_i$: if $P(n) = -\Delta_i p$ then
(1) $EE_i p \notin F$ and
(2) $-\Delta_c E_i p \in P(1..n)$ and
(3) $\forall r \in R_s^i[p]$: either
$\exists a \in A(r), -\Delta_c a \in P(1..n)$ or
$\exists \alpha \in A(r), -\Delta_c \alpha \in P(1..n)$ or
$\exists E_j b \in A(r), -\Delta_j b \in P(1..n)$, and
(4) $\forall r \in R^c[p]$, either
$A(r) \cap \mathrm{Lit} = \emptyset$ or
$\exists a \in A(r) : -\Delta_i a \in P(1..n)$ or
$\exists \alpha \in A(r) : -\Delta_c \alpha \in P(1..n)$.

As usual the condition for $-\Delta_i$ is the strong negation of that for $+\Delta_i$. The only points to notice are clause (2) where we have to consider that $E_i p$ is not provable using counts-as rules, and the first condition of clause (4) that imposes that conversions from counts-as rules to results-in rules is not possible if the antecedent of the counts-as rule does not contain any literal (even if it may contain actions). According to clause (4) of the two conditions above, given the facts E_iA and β_j we can use the rule $A, \beta_j \rightarrow_c B$ to derive $+\Delta_i B$ and consequently E_iB, but not the rule $\beta_j \rightarrow_c B$.

We give now the proof condition for support for i ($\pm\Sigma_i$).

$+\Sigma_i$: if $P(n+1) = +\Sigma_i p$ then
(1) $E_i p \in F$; or
(2) $\exists r \in R^i_{sd}[p]\ \forall a, E_j b, \alpha \in A(r)$:
$+\Sigma_c a, +\Sigma_j b, +\Sigma_c \alpha \in P(1..n)$; or
(3) $\exists r \in R^c_{sd}[p]\ \exists a \in A(r) \cap \mathrm{Lit}$ and,
$\forall a, \alpha \in A(r) : +\Sigma_i a, +\Sigma_c \alpha \in P(1..n)$.

The inference conditions for H_i are very similar to those for strong results-in rules; essentially they are monotonic proofs using both the monotonic part (strict rules) and the supportive non-monotonic part (defeasible rules) of a defeasible institutional action theory. Given the close similarity between the conditions for $+\Delta_i$ and $+\Sigma_i$ and the fact that all pairs of proof conditions for the proof tags given in this paper are in agreement with the principle of strong negation the conditions for $-\Sigma_i$ are omitted.

To capture the results of defeasible actions we have to use the superiority relations to resolve conflicts. Thus the inference conditions for $+\partial_i$ are:

$+\partial_i$; $P(n) = +\partial_i p$ then
(1) $+\Delta_i p \in P(1..n)$ or
(2.1) $-\Delta\mathscr{C}(E_i p), -\Delta_i E_k p \in P(1..n)$ and
(2.2) $\exists r \in R^i_{sd}[p] \cup R^c_{sd}[EE_i p]:\ \forall a, \alpha, E_j b \in A(r)$,
$+\partial_c a, +\partial_c \alpha, +\partial_j b \in P(1..n)$ or
$\exists r \in R^c_{sd}[p]$: $A(r) \cap \mathrm{Lit} \neq \emptyset$, and
$\forall a, \alpha \in A(r), +\partial_i a, +\partial_c \alpha \in P(1..n)$; and
(2.3) $\forall s \in R[\mathscr{C}(E_i p)] \cup R^i[E_k p]$: either
$\exists a \in A(s) : -\partial_c a \in P(1..n)$ or
$\exists \alpha \in A(s) : -\partial_c \alpha \in P(1..n)$ or
$\exists E_j b \in A(s) : -\partial_j b \in P(1..n)$, and
$\forall s \in R^c[E_k p]$: either
$A(s) \cap \mathrm{Lit} = \emptyset$ or
$\exists \alpha \in A(s) : -\partial_c \alpha \in P(1..n)$ or
$\exists a \in A(s) : -\partial_i a \in P(1..n)$; or
(2.3.3) $\exists t \in R^i[p] \cup R^c[EE_i p] : t > s$ and
$\forall a, \alpha, E_j b \in A(t), +\partial_c a, +\partial_c \alpha, +\partial_j b \in P(1..n)$ or
$\exists r \in R^c[p]$: $A(t) \cap \mathrm{Lit} \neq \emptyset$, and
$\forall a, \alpha \in A(t), +\partial_i a, +\partial_c \alpha \in P(1..n)$

The conditions for proving the results of defeasible actions are essentially the same as those given for defeasible derivations in Section 2. Also here, at each stage, we have to check for two cases, namely: (1) the rule used is a results-in rule; (2) the rule is a counts-as rule. In the first case we have to verify that factual antecedents are defeasibly proved/disproved using counts-as ($\pm\partial_c$), and brings-it-about antecedents are defeasibly proved/disproved using results-in rules ($\pm\partial_i$). In the second case we have to remember that a conclusion of a institutional counts-as rule can be transformed (converted) into a results-in if all the literals in the antecedent are defeasibly executed. For the attack phase (clause 2.3) we have to consider all rules in $\mathscr{C}(E_i p)$ as well as all results-in rules for agent i for $E_k p$, i.e., rules meaning that agent i does something so that agent k personally does p (again, see Section 1 for the motivation and intuition behind this condition). Finally, for the same reason we have to ensure that all counts-as rules for E_k ($k \neq i$) do not behave as results-in rule for agent i. This means we have to verify that either the rule cannot be converted into a results-in rule for i (i.e., $A(r) \cap \mathrm{Lit} = \emptyset$) or that the conversion is blocked, i.e., that there is a literal which is not provable for ∂_i. This means that the event corresponding to the literal is not under the control of agent i, and so the whole conclusion, which would correspond to the delegation to agent k, is not under the influence of agent i.

For $-\partial_i$ we have:

$-\partial_i$: if $P(n) = -\partial_i p$ then
(1) $-\Delta_i p \in P(1..n)$ and
(2.1) $+\Delta\mathscr{C}(E_i p) \in P(1..n)$ or $+\Delta_i E_k \in P(1..n)$
(2.2.1) $\forall r \in R^i_{sd}[p] \cup R^c[EE_i p]$ either
    $\exists a \in A(r) : -\partial_c a \in P(1..n)$ or
    $\exists \alpha \in A(r) : -\partial_c \alpha \in P(1..n)$ or
    $\exists E_j b \in A(r) : -\partial_j b \in P(1..n)$, and
(2.2.2) $\forall r \in R^c[p]$ either
    $A(r) \cap \mathrm{Lit} = \emptyset$ or
    $\exists a \in A(r) : -\partial_i a \in P(1..n)$ or
    $\exists \alpha \in A(r) : -\partial_c \alpha \in P(1..n)$, or
(2.3) $\exists s \in R[\mathscr{C}(E_i p)] \cup R^i[E_k p] : \forall a, \alpha, E_j b \in A(s)$,
    $+\partial_c a, +\partial_c \alpha, +\partial_j b \in P(1..n)$ or
  $\exists s \in R^c[E_k p]$: $\exists a \in A(s) \cap \mathrm{Lit}$, and
    $\forall a, \alpha \in A(r), +\partial_i a, +\partial_c \alpha \in P(1..n)$, and
(2.3.1) $\forall t \in R^i[p] \cup R^c[EE_i p]$ either $s \not> t$ or
    $\exists a \in A(t) : -\partial_c a \in P(1..n)$ or
    $\exists \alpha \in A(t) : -\partial_c \alpha \in P(1..n)$ or
    $\exists E_j b \in A(t) : -\partial_j b \in P(1..n)$, and
(2.3.2) $\forall t \in R^c[p]$ either
    $A(s) \cap \mathrm{Lit} = \emptyset$ or
    $\exists a \in A(t), -\partial_i a \in P(1..n)$ or
    $\exists \alpha \in A(t), -\partial_c \alpha \in P(1..n)$ or $s \not> t$.

Let us examine the above conditions at work with the help of some examples. We assume the following theory:

$$
\begin{aligned}
&F = \{\alpha_i, p, E_j q\},\\
&R = \{r_1 : \alpha_i, p, E_j q \Rightarrow^i s; \quad r_2 : s \Rightarrow^i r; \quad r_3 : r \Rightarrow_c t\}.
\end{aligned}
$$

In this theory we are able to prove $E_i t$. The facts fire r_1, thus we can prove $+\partial_i s$ ($E_i s$). Now, since s has been brought about, s is the case. We can use this to fire the rule r_2. Hence we obtain $+\partial_i r$, which is $E_i r$. This implies that all the requisites of r_3 have been brought about; but r_3 states that r counts as t; this means that t has been brought about, hence $+\partial_i t$ and Et.

Let us replace r_3 with $r_3' : p, r \Rightarrow_c t$. This time we can prove $+\partial_c t$, but not $E_i t$ ($+\partial_i t$). The reason is that p is the case without a specific "intention" of the agent to bring it about. Similarly, if we replace r_3 by $r_3'' : E_i r \Rightarrow_c t$ we can no longer derive $E_i t$. Here $E_i r$ is understood as a mere institutional fact, and not as the successful intention of the agent to realise r in order to realise t.

In the previous example we have seen how we can argue in favour of $E_i p$ (for same literal p). Let us examine the conditions to attack it. Let I be the following institutional defeasible theory

$$
\begin{aligned}
&F = \{\alpha_i, p, q\},\\
&R = \{r_1 : \alpha_i, p \Rightarrow^i s; \quad r_2 : q \Rightarrow_c r; \quad r_3 : p, r \Rightarrow_c \neg s\}
\end{aligned}
$$

Clearly $E_i s$ ($+\partial_i s$) is not derivable from the given theory since there is an applicable rule for $\neg s$. r_3 is applicable since we can derive $+\partial_c r$. Similarly, if we replace r_2 with $q \Rightarrow_i r$, r_3 is still applicable. We can prove $+\partial_i r$: this means that there is a successful action resulting in r. In general to discard a rule we have to show that some of the premises cannot be derived. With a factual literal we have to show that the literal is not the case (or, in other terms, that there are no literals that count as it), and that the literal is not the result of a successful action: results of successful actions are indeed the case. Finally we replace $r3$ with $r_3'' : p, r \Rightarrow^i E_j s$. Again we cannot conclude $E_i s$; see the motivation for the principle (EE¬E) in Section 1.

Let us now consider how to represent the following business scenario. For normal orders a company has pre-defined invoices and the finance department can delegate the preparation of the invoices to the shipping department. The preparation of an invoice requires to check that the details in it are correct and to sign it. However special orders require more care and processing, and the finance department is in charge for their invoices. Finally goods can be delivered only after the finance department has prepared the invoice. This scenario is depicted by the following institutional theory,

$$
\begin{aligned}
&r_1 : \mathit{proc}_F(E_S(\mathit{invoice\_ready})), E_S(\mathit{invoice\_ready}) \Rightarrow^F \mathit{invoice\_ready}\\
&r_2 : \mathit{special\_order}, E_S(\mathit{invoice\_ready}) \Rightarrow_c \neg \mathit{invoice\_ready}\\
&r_3 : \mathbf{sign\_invoice}_X \Rightarrow^X \mathit{invoice\_checked}\\
&r_4 : \mathit{invoice\_checked} \Rightarrow_c \mathit{invoice\_ready}\\
&r_5 : E_F(\mathit{invoice\_ready}) \Rightarrow_c \mathit{ship\_order}
\end{aligned}
$$

where $r_2 > r_1$ and $r_2 > r_4$. Here rule r_1 is the rule governing the delegation of the preparation of the invoice, where r_2 is an exception to it. r_3 is a schema that establishes that the act of signing an invoice by an agent (a role) X results in the invoice being checked by X. The meaning of r_4 is that according to the business rule of the company is that once an invoice has been checked then the invoice is ready to be sent. Finally r_5 states that items can be shipped only after their invoice has been approved by the finance department.

Let us consider the following scenario. The company receives an order. The finance department considers the order to be a standard order and it delegates the whole process to the shipping department, which processes it and a clerk in this department signs the invoice. In this case the facts are $\mathit{proc}_F(E_S(\mathit{invoice\_ready}))$, and $\mathbf{sign\_invoice}_S$. We can apply r_3 to derive $E_S(\mathit{invoice\_checked})$. According to rule r_4 we have that the invoice is ready. However the invoice has been signed by a clerk in the shipping office, the result of this action is qualified as an act performed by the shipping department. This means that we carry over the qualification from the antecedent to the consequent of rule r_4. Hence we obtain $E_S(\mathit{invoice\_ready})$. Since the shipping department was delegated by the finance department to process the invoice, we can apply rule r_1 to derive that the invoice had been prepared by the finance department via delegation ($E_F(\mathit{invoice\_ready})$) and the order can be delivered. On the other hand, if an order is classified as a special order, then the only alternative is that the finance department process the invoice by itself, that is somebody in the finance department has to sign the invoice.

4. Properties of the Logic

First of all, as it was mentioned in Section 1, it is worth noting that the consequence relation induced by the defeasible relation for the counts-as –which is characterised by proof conditions for standard DL– is a cumulative consequence relation and thus it obeys the basic properties of Reflexivity, Cautious Monotonicity and Cut we previously required for the counts-as conditional. The proof for this result can be found in [5].

Let us see some properties of the logic we have just described.

The purpose of the $-\Delta$ and $-\partial$ inference rules is to establish that it is not possible to prove a corresponding tagged literal. These rules are defined in such a way that all the possibilities for proving $+\partial p$ (for example) are explored and shown to fail before $-\partial p$ can be concluded. Thus we have a constructive proof that the corresponding positive conclusion cannot be obtained.

As a result, there is a close relationship between the inference rules for $+\partial$ and $-\partial$, (and also between those for $+\Delta$ and $-\Delta$, and $+\Sigma$ and $-\Sigma$). The structure of the inference rules is the same, but the conditions are negated in some sense. This feature allows us to prove some properties showing the well behaviour of defeasible logic.

THEOREM 1. *Let $\# = \Delta_c, \partial_c, \Sigma_c, \Delta_i, \partial_i, \Sigma_i$, and I be an institutional action theory. There is no literal p such that $I \vdash +\#p$ and $I \vdash -\#p$.*

The above theorem states that no literal is simultaneously provable and demonstrably unprovable, thus it establishes the coherence of the defeasible logic presented in this paper.

THEOREM 2. *Let I be an acyclic institutional action theory, and $M \in \{c,i\}$, $i \in A$. $I \vdash +\partial_M p$ and $I \vdash +\partial_M {\sim} p$ iff $I \vdash +\Delta_M p$ and $I \vdash +\Delta_M {\sim} p$.*

This theorem gives the consistency of defeasible logic. In particular it affirms that it is not possible to bring conflicting states about ($+\partial_i p$ and $+\partial_i {\sim} p$) unless the information given about the environment is itself inconsistent. Notice, however, that the theorem does not cover attempts (Σ_i). Indeed it is possible to attempt something and its negation. We will say that an institutional theory is consistent if Theorem 2 holds for the theory.

Let I be an institutional action theory I. With Δ_c^+ we denote the set of literals strictly provable using the counts-as part of I, i.e., $\Delta_c^+ = \{p : I \vdash +\Delta_c p\}$. Similarly for the other proof tags.

THEOREM 3. *Let I be an institutional action theory, and $M \in \{c,i\}$, $i \in A$.*

1. *$\Delta_M^+ \subseteq \partial_M^+ \subseteq \Sigma_M^+$;*
2. *$\Sigma_M^- \subseteq \partial_M^- \subseteq \Delta_M^-$;*
3. *Let I be a consistent institutional action theory such that $I \vdash -\Delta_i p$. If $I \vdash +\partial_i E_j p$ then $I \vdash -\partial_i p$.*
4. *For any i, $\Delta_i^+ \subseteq \Delta_c^+$, and $\partial_i^+ \subseteq \partial_c^+$.*

Since $+\partial_i$ and $+\Sigma_i$ correspond to E_i and H_i, we have that that 1. and 2. correspond to the axiom $E_i A \rightarrow H_i A$. 3. is an immediate consequence of clause 2.3.2 of the inference condition for $+\partial_i$. This property corresponds to the axiom (EE¬E) of Section 1. Finally 4. corresponds to the successfulness of the E_i operator (i.e., axiom T).

To conclude this section we give a result justifying the choice of DL as our computational framework. Given an institutional action theory I, the universe of I, U^I is the set of atomic propositions and action symbols occurring in it. The extension of I is the set of all proof tags derivable from I, restricted to the (modal) literals that can be built from U^I.

THEOREM 4. *Let I be an institutional action theory. The extension of I can be computed in time linear to the size of the theory, i.e.,* $O(|R| * |U^I| * |A|)$.

The proof is based on a variation of the data structure used by Maher [33] to prove that the basic DL has linear complexity, see [18, 22].

5. Related Work

An impressive amount of literature has been devoted to agent interaction and coordination. Our work presents a rule-based system and so it fits into a long and extensive AI tradition. As regards agent interaction, we can identify in particular two recent strands: (a) a cognitive account of agents that specifies their mental attitudes; (b) modelling agents' behaviour by means of normative concepts. In this section we simply comment some contributions which are strictly related to the specific perspective adopted here, a perspective originating from [16] and which belongs to the research line mentioned under point (b) above. The current work is a technical extension of [16], as it takes some of the building blocks used there and re-defines them within a computational framework, where by "computational framework" we mean a logical system which enjoys nice computational properties and which is directly designed for implementation.

Different formal theories of action have been used to deal with institutional agents. Logics such as Event and Situation Calculi, the STIT approach, Dynamic Logics—just to mention a few examples—were all proven useful in combination with normative concepts, and especially with deontic notions (for recent applications in the field, see [13, 15, 26, 7]). However, the aim of this paper is not to develop an alternative methodology to these theories, as our approach focuses on very minimal and abstract properties of agency in the spirit of the modal logic of agency described in [14]. In this specific perspective, our contribution is meant to show how such minimal properties can be embedded in DL, and so how they can be re-interpreted within a non-monotonic system specifically oriented to implementation.

A further goal of this paper was to see how agency can interplay with counts-as rules. As far as the logical nature of these rules is concerned, the literature provides different views.

In a first perspective, the institutional status of constitutive rules is directly related to some epistemic notions [6]: using the metaphor of normative systems as agents, this approach attributes to them some peculiar mental attitudes. Hence, norms are considered as mental objects and constitutive rules, in particular, are modelled as beliefs.

In a second perspective, the attention is rather focused on the role of institutional rules intended as external factors constraining agents' behaviour. Clearly, our paper draws inspiration from this second perspective. Within this view, we can mention at least two alternative options (see [24] for a fine discussion of the different meanings of the counts-as link).

A first approach is in line with Goldman's theory of actions generating actions [17]. It may be argued that the generation of institutional facts via counts-as rules is close to the idea of causality. If so, counts-as relations cannot be reflexive since "it is precisely the property of non-reflexiveness that distinguishes a generation relation as such" [28]. In [29], Jones and Sergot basically follow this approach and develop an analysis of the notion of institutionalised power by introducing a new (classical but not normal) conditional connective "$\Rightarrow_s$". This connective expresses the "counts as" connection holding in the context of an institution *s*. In particular, when applied to action descriptions, formulas like $E_iA \Rightarrow_s E_iB$ and $E_iA \Rightarrow_s E_jB$ represent respectively *i*'s institutional power to produce *B* when *A* is realised and *i*'s power to perform an action as if something else were made by *j* (see [29, 30])[9]. In addition, the logic for $\Rightarrow_s$ is integrated by the **KD** modality D_s, such that D_sA means that *A* is a "constraint on the institution *s*". The connection between $\Rightarrow_s$ and D_s is characterised by the schema $(A \Rightarrow_s B) \rightarrow D_s(A \rightarrow B)$. This approach differs from our view, as Jones and Sergot state that the counts-as be non-reflexive and transitive, while we see it as at least enjoying Reflexivity, Cut, and Cautious Monotonicity. Reflexivity affects the meaning ascribed to the count-as link. If the defeasibility of counts-as must be accepted, we have to decide whether reflexivity must prevail over transitivity or the other way around, since transitivity and reflexivity imply monotonicity (see [4]). As in [16], we assume that the counts-as link has a classificatory nature, and defeasible classificatory relations, such as typicality, normally enjoy reflexivity.

A second approach, by Grossi, Meyer, and Dignum [23], views counts-as statements as yielding contextual classifications. Hence, as we do here, it is emphasised the classificatory role of the notion of counts-as, a notion which is investigated by Grossi and colleagues by means of modal logic techniques from a semantics-driven perspective. In particular, the authors model

[9] A computational framework for modelling the counts-as link, insofar as it is viewed as a kind of causal relation, has been later devised by Sergot [48]. He developed the language $(C/C+)^{++}$ to represent counts-as relations between actions in terms of conventional generations of actions.

the counts-as connection within the multi-modal logic $\mathbf{KD45_n^{i-j}}$. Despite its conceptual clarity, this analysis has two drawbacks. First, the defeasible nature of constitutive rules is disregarded. Second, contextual modalities suffer of the so-called "omniscience problem", a problem which also affects Jones and Sergot's D_s modality: if *making_a_bid* is an institutional act, this would imply in that approach that $\mathit{making\_a\_bid} \vee \mathit{drinking\_some\_water}$ holds as well within an institution. Our approach tries to avoid this difficulty, as institutional consequences are derived only if stronger reasons do not block these derivations.

6. Discussion and Future Work

Our aim was to develop a computational treatment of the notion of institutional agency as described in [16]. In this perspective, our contribution does not include any explicit refinement (e.g., in terms of articulating new axioms) of what has been already proposed in [16]. This does not mean, however, that the model presented here cannot be a potential starting point to achieve new proof-theoretical results. Let us recall that the propositional base of the modal logic of agency is classical propositional logic [44, 14]. On the other hand, any refinement to introduce non-monotonic reasoning as a crucial aspect of institutional agency has been confined both in [16] and in [29, 30] to account only for the counts-as link. Although this paper provides a machinery to reason about actions only with regard to institutional domains, it proposes some inferential mechanisms that may be generalised to define a non-monotonic theory of agency. How to do this and which is the axiomatisation resulting from such a generalisation is a matter of future research.

The logic presented here is just one of the many logics that can be defined using the main idea of the paper. Non-monotonic reasoning is a complex phenomenon with many facets. Several variants of defeasible logic have been put forward to deal with different intuitions behind non-monotonic reasoning. Accordingly a designer of a defeasible logic of agency has to chose the most appropriate defeasible inference mechanism and the degree of provability corresponding to the modalities at hand for the intended application. Similarly, the designer can chose more or less liberal conditions to use counts-as rules to derive brings-it-about literals. For example in this paper we have assumed that we can use a counts-as rule to derive a brings-it-about literal if all the literal in the antecedent of the rule can be derived as results-in conclusions. A more liberal condition could just require that only one of them is derived in such a way.

The aim of the paper was to provide a computationally oriented framework for the notion of counts-as and institutional agency. The model was given by a multi-modal extension of Defeasible Logic, and we have shown that the complexity of the resulting logic is linear. At the same time it is possible to use the logic as both a conceptual and executable specification of an institution. Accordingly it is natural to ask whether the logic has been implemented. While specific implementations for the logic do not exist, [18] describes an implementation of a similar modal (deontic) variant of Defeasible Logic. The implementation follows very closely the data structures and algorithm used to prove Theorem 4. Therefore the logic presented here can be easily implemented (indeed a Python prototype for the inference engine can be implemented with a few hundred lines of code).

Finally, we suggest some conceptual refinements for our future research.

First, the model should cope with a wider range of properties and with other concepts of agency, such as those mentioned in Section 1, i.e. the notions of ability [14] and indirect successful action. Both of them are crucial in modelling the co-ordination of agents: (a) the inference of institutional facts may be conditioned by the practical capability of an agent to do things that generate by convention these facts; (b) the characterisation of the institutionalised power may require that an agent is empowered to realise indirectly a state of affairs without specifying the chain of agents that will bring about such a state of affairs.

Another issue concerns the relations between different institutions. These relations are relevant when an action takes place in different institutional contexts and produces diverse, and possibly contradictory, results. Following [16], multi-institutional contexts are captured by stipulating that $A \Rightarrow_s B =_{def} (A \Rrightarrow D_s B) \wedge (D_s A \Rrightarrow D_s B) \wedge (D_s A \Rrightarrow D_{s'} B)$. They may be represented here introducing counts-as rules indexed by different institutions: the superiority relations would play an important role in settling possible contradictions between different institutional contexts. But that is not all since the matter regards the complex problem of the relation between normative systems [41].

We also have to develop a more accurate mechanism to deal with conflicting institutional results arising from the exercise of different powers and which lead to dropping institutional facts which were previously derived. This question requires to develop a dynamic account of the institutional mechanisms. Of course, the idea, according to which the generation of institutional facts is close to the concept of causality, is a feasible option in this regard. However, as we said, this diverges from our view of the counts-as link. An alternative possibility is thus to introduce explicit temporal dimensions, as done in [21], in order to make explicit when an institutional fact p is cancelled by a conflicting one which results from the subsequent exercise of a different power, or even of the same power that produced p.

Finally, we have to introduce in the current framework deontic modalities, as they are, too, crucial in modelling the normative coordination of agents. Some promising results in this perspective are already provided, for example, in [21, 22], but a more extensive work is needed in studying the properties of the system when deontic concepts are added.

References

1. G. Antoniou, D. Billington, G. Governatori, and M. J. Maher. A flexible framework for defeasible logics. In *Proc. AAAI 2000*, pages 401–405, AAAI Press, 2000.
2. G. Antoniou, D. Billington, G. Governatori, and M. J. Maher. Representation results for defeasible logic. *ACM Transactions on Computational Logic*, 2(2):255–287, 2001.
3. G. Antoniou, D. Billington, G. Governatori, M. J. Maher, and A. Rock. A family of defeasible reasoning logics and its implementation. In *Proc. ECAI 2000*, pages 459–463. IOS Press, 2000.
4. A. Artosi, G. Governatori, and A. Rotolo. Labelled tableaux for non-monotonic reasoning: Cumulative consequence relations. *Journal of Logic and Computation*, 12(6):1027–1060, 2002.
5. D. Billington. Defeasible Logic is Stable. *Journal of Logic and Computation*, 3:370–400, 1993.
6. G. Boella and L. van der Torre. Regulative and constitutive norms in normative multiagent systems. In *Proc. KR 2004*, pages 255-266. Morgan Kaufmann, 2004.
7. J. Broersen. Action negation and alternative reductions for dynamic deontic logics. *Journal of Applied Logic*, 2(1): 153–168, 2004.
8. C. Castelfranchi, F. Dignum, M.J. Catholijn, and J. Treur. Deliberative normative agents: Principles and architecture. In *Proc. ATAL 1999*, LNCS 1757, pages 364–378. Springer, 2000.
9. C. Castelfranchi and R. Falcone. Towards a theory of delegation for agent-based systems. *Robotics and Autonomous Agents*, 24:141–157, 1998.
10. B. Chellas. *Modal Logic. An Introduction*. Cambridge University Press, 1980.
11. M. Colombetti. A commitment-based approach to agent speech acts and conversations. In M. Greaves et al., editors, *Proc. 4th International Conference on Autonomous Agents, Workshop on Agent Languages and Conversation Policies*, pages 21–29, Barcelona, 2000.
12. R. Conte and C. Dellarocas. *Social Order in Multiagent Systems*. Kluwer, 2001.
13. R. Demolombe and A. Herzig. Obligation Change in Dependence Logic and Situation Calculus. In A. Lomuscio and D. Nute, editors, *Proc. Deon 2004*, LNAI 3065, pages 57–73. Springer, 2004.
14. D. Elgesem. The modal logic of agency. *Nordic Journal of Philosophical Logic*, 2:1–48, 1997.
15. A.D. Farrell, M. Sergot, M. Sallé, and C. Bartolini. Using the event calculus for tracking the normative state of contracts. *International Journal of Cooperative Information System*, 4(2-3):99–129, 2005.
16. J. Gelati, G. Governatori, A. Rotolo, and G. Sartor. Normative Autonomy and Normative Co-ordination: Declarative Power, Representation, and Mandate. *Artificial Intelligence and Law*, 12(1-2): 53–81, 2004.
17. A. Goldman. *A Theory of Human Action*. Prentice Hall, Princeton, 1970.
18. G. Governatori and D.H. Pham. A Semantic Web Based Architecture for e-Contracts in Defeasible Logic. In *Proc. RuleML 2005*, LNCS 3791, pages 145–159. Springer, 2005.
19. G. Governatori, V. Padmanabhan. A defeasible logic of policy-based intention. In *Proc. Australian AI 2003*, LNAI 2903, pages 414–426. Springer, 2003.

20. G. Governatori and A. Rotolo. On the Axiomatization of Elgesem's Logic of Agency and Ability. *Journal of Philosophical Logic*, 34 (4):403–431, 2005.
21. G. Governatori and A. Rotolo and G. Sartor. Temporalised Normative Positions in Defeasible Logic. In *Proc. ICAIL'05*, pages 25–43. ACM, 2005.
22. G. Governatori and A. Rotolo and V. Padmanabhan. The Cost of Social Agents. In *Proc. AAMAS'06*, pages 513–520. ACM, 2006.
23. D. Grossi, J-J. Meyer, and F. Dignum. Modal Logic Investigations in the Semantics of Counts-as. In *Proc. ICAIL'05*, pages 1–9. ACM, 2005.
24. D. Grossi, J-J. Meyer, and F. Dignum. Counts-as: Classification or constitution? An answer using modal logic. In *Proc. Deon'06*, LNCS 4048, pages 115–130. Springer, 2006.
25. J.Y. Halpern and Y.O. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1990.
26. J.F. Horty. *Agency and Deontic Logic*. Oxford University Press, Oxford, 2001.
27. A. Jones. Towards a formal theory of communication and speech acts. In P. Cohen and M. Pollack, editors, *Intentions in Communication*. MIT Press, 1990.
28. A. Jones, X. Parent, and A. Stolpe. Private communication, 2003.
29. A. Jones and M. Sergot. A formal characterisation of institutionalised power. *Journal of IGPL*, 3:427–443, 1996.
30. A. J. Jones. A logical framework. In J. Pitt, editor, *Open Agent Societies: Normative Specifications in Multi-Agent Systems*. Wiley, forthcoming.
31. S. Kanger. Law and logic. *Theoria*, 38:105–32, 1972.
32. L. Lindahl. *Position of change: A Study in law and logic*. Reidel, 1977.
33. M.J. Maher. Propositional defeasible logic has linear complexity. *Theory and Practice of Logic Programming*, 1(6):691–711, 2001.
34. D. Makinson. On the formal representation of rights relations. *Journal of Philosophical Logic*, 15:403–25, 1986.
35. T. Norman and C. Reed. Delegation and responsibility. In C. Castelfranchi and Y. Lesperance, editors, *Proc. Intelligent Agents VII*, LNAI 1986, pages 136–149. Springer, 2001.
36. T. Norman and C. Reed. A model of delegation for multi-agent systems. In M. d'Inverno et al., editors, *Proc. Foundations and Applications of Multi-Agent Systems*, LNAI 2403, pages 185–204. Springer, 2002.
37. D. Nute. Defeasible logic. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 353–395. Oxford University Press, 1987.
38. D. Nute, ed. *Defeasible Deontic Logic*. Kluwer, 1997.
39. D. Nute. Norms, priorities and defeasibility. In P. McNamara and H. Prakken, editors, *Norms, Logics and Information Systems*, pages 83–100. IOS Press, 1998.
40. J. Pitt, editor. *Open Agent Societies: Normative Specifications in Multi-Agent Systems*. Wiley, forthcoming.
41. H. Prakken. *Logical Tools for Modelling Legal Argument*. Kluwer, 1997.
42. I. Pörn. *Action Theory and Social Science: Some Formal Models*. Reidel, 1977.
43. L. Royakkers. Combining deontic and action logics for collective agency. In *Legal Knowledge and Information Systems (Jurix)*, pages 135–146. IOS Press, 2000.
44. F. Santos and J. Carmo. Indirect Action. Influence and Responsibility. In M. Brown and J. Carmo, editors, *Deontic Logic, Agency and Normative Systems*, pages 194–215. Springer, 1996.
45. F. Santos, A. Jones, and J. Carmo. Action concepts for describing organised interaction. In *Proc. 13th HICSS*, pages 373–382. IEEE Computer Society Press, 1997.
46. J. Searle. *The Construction of Social Reality*. Penguin Press, 1995.

47. K. Segerberg. Getting started: Beginnings in the logic of action. *Studia Logica*, 51:347–58, 1992.

48. M. Sergot. The Language $(C/C+)^{++}$. In J. Pitt, editor, *Open Agent Societies: Normative Specifications in Multi-Agent Systems*. Wiley, forthcoming.

49. M.P. Singh. An ontology for commitments in multiagent systems: toward a unification of normative concepts. *Artificial Intelligence and Law*, 7:93–113, 1999.

Appendix

A. Proofs of the Theorems in Section 4

THEOREM 1. *Let* $\# = \Delta_c, \partial_c, \Sigma_c, \Delta_i, \partial_i, \Sigma_i$, *and* I *be an institutional action theory. There is no literal* p *such that* $I \vdash +\#p$ *and* $I \vdash -\#p$.

Proof. The result is a straightforward consequence of the principle of strong negation [1, 3] used to define the proof conditions for the logic at hand. According to the principle of strong negation the condition for $+\#$ is the constructive negation of that of $-\#$ and the other way around. Thus if the condition for $+\#$ is satisfied the condition for $-\#$ fails and the other way around.

THEOREM 2. *Let* I *be an institutional action theory, and* $M \in \{c, i\}$, $i \in A$. $I \vdash +\partial_M p$ *and* $I \vdash +\partial_M {\sim} p$ *iff* $I \vdash +\Delta_M p$ *and* $I \vdash +\Delta_M {\sim} p$.

Proof. We have to show that if we have both $+\partial_M p$ and $+\partial_M \neg p$ then the only possible derivation is one where the two are both justified by clause (1) of the proof conditions for $+\partial$, and combinations of justifications where one of them is justified in terms of clause (2) lead to a contradiction.

It is clear that a combination of clauses (1) does not lead to any contradiction. Thus we have that both $+\Delta_M p$ and $+\Delta_M \neg p$.

Let us consider the cases where one is justified by clause (2). This means that clause (2.1) is satisfied thus for $M = c$ we have $-\Delta_c \mathscr{C}(p)$, and for $M = i$ both $-\Delta_i \mathscr{C}(E_i p)$ and $-\Delta_i E_k p$. But both $\neg p \in \mathscr{C}(p)$ and ${\sim} p \in \mathscr{C}(E_i p)$, thus we have $-\Delta_M p$. By Theorem 1 it is not possible that both $I \vdash +\Delta_M p$ and $I \vdash -\Delta_M p$. Thus in this case we get a contradiction.

We examine now the situation where $M = c$ and both conclusions are justified by clause (2). This means that $\exists r^+ \in R_{sd}[Ep]$ such that the rule is applicable (i.e., the condition of clause (2.2) is satisfied), and at the same time we have that $\exists r^- \in R_{sd}[E{\sim}p]$ such that r^- is applicable. $R_{sd}[E{\sim}p] \subseteq R[\mathscr{C}(p)]$, thus there must be a rule $t_0 \in R[Ep]$ such that t_0 applicable and $t_0 > r^-$ (according to clause (2.3.4)). We have two cases (i) t_0 is maximal (i.e., $\neg\forall s > t_0$) (ii) t_0 is not maximal. For (i) we have that $t_0 \in R[\mathscr{C}({\sim}p)]$ and is applicable thus t_0 satisfies clauses (2.3) of $-\partial_c$. Therefore, $-\partial_c {\sim} p$, and we have a contradiction according to Theorem 1. For (ii) let us consider the set $T_0 = \{s : s > t_0 \wedge s \in R[E{\sim}p]\}$. Notice that $r^- \notin T_0$, since the theory I is acyclic. From T_0 we eliminate the rules that are not applicable for condition (2.2) of $+\partial_c$, and we call the resulting set S_0. Since we have $+\partial_c p$ this means that for every rule $s \in S_0$ there is a rule $t' \in R[Ep]$ such that the rule is applicable and stronger than s. Let s' and t_1 be such rules, If t_1 is maximal we are done as in the previous reasoning. We build the set $T_1 = \{s : s > t_1 \wedge s \in R[E{\sim}p]\}$ and then S_1 in the same way as S_0. Since the theory is acyclic we have that $S_0 \subset S_1$. We can repeat this construction n times for each rule in S_0 until we reach a point where the rules we have for Ep are maximal, and we get that $-\partial_c{\sim}p$, from which we get again a contradiction.

For $M = i$ we have to consider rules in $R^c[\mathscr{C}(E_i p)] \cup R^i[E_k p] \cup R^c[E_k p]$ and the conditions of applicability of clause (2.2) for $+\partial_i$, when we build the sets T_n and S_n, but the reasoning for ∂_i carries over this case as well.

THEOREM 3. *Let I be an institutional action theory, and $M \in \{c, i\}$, $i \in A$.*

1. $\Delta^+_M \subseteq \partial^+_M \subseteq \Sigma^+_M$;

2. $\Sigma^-_M \subseteq \partial^-_M \subseteq \Delta^-_M$;

3. *Let I be a consistent institutional action theory such that $I \vdash -\Delta_i p$. If $I \vdash +\partial_i E_j p$ then $I \vdash -\partial_i p$.*

4. *For any i, $\Delta^+_i \subseteq \Delta^+_c$, and $\partial^+_i \subseteq \partial^+_c$.*

Proof. For 1. The inclusion $\Delta^+ \subseteq \partial^+$ is immediate given clause (1) of the the proof condition for $+\partial_M$, which allows us to extend a derivation with $+\partial_M p$ if $+\Delta_M p$ is already in the derivation.

For the inclusion $\partial^+_M \subseteq \Sigma^+_M$, the proof is by induction on the length of the derivation of $+\partial_M p$. Notice that it is not possible to have a defeasible derivation consisting of a single step: a minimal defeasible derivation has at least two lines. We will use this case as inductive base. We have two possibilities. We have (i) $P(1) = +\Delta_M p$ for $p = \alpha$, $\alpha \in F$ or $E_i p \in F$, and then $P(2) = +\partial_M p$ justified by $P(1)$; or (ii) $P(1) = -\Delta{\sim}p$ (there are no strict rules for ${\sim}p$), and $P(2) = +\partial_M p$, justified by the fact that there is a strict or defeasible rule r in R^M, $A(r) = \emptyset$, and $R[\mathscr{C}(E_i p)] \cup R[E_k p] = \emptyset$ and either $R^c[E_k p] = \emptyset$ or $\forall s \in R^c[E_k p], A(s) \cap \mathrm{Lit} = \emptyset$, for $k \neq M$.

For (i) we have that the justification for $P(1)$ corresponds to clause (1) of the proof condition for $+\Sigma_M$, thus we can create a proof for $+\Sigma_M p$. For (ii) $r \in R_{sd}[p]$ and the conditions (2) and (3) of the proof condition for Σ_M are vacuously satisfied. We can now assume that the property holds for the derivation of $+\partial_M p$ of length n. For the inductive step we have to consider whether $+\partial_M p$ is justified by clause (1) or clause (2) of the proof condition for $+\partial_M$. For (1) we have two sub-cases: the conclusion is a fact and we can repeat the argument of the inductive base or either clauses (2) or (3) of $+\Delta_M$ apply. This means that by inductive hypothesis there is a strict rule that satisfies either the condition of clauses (2) or (3) of Σ_M. In case $+\partial_M p$ is justified by clause (2.2) of $+\partial_M$, then all we have to notice is that we consider the same sets of rules as in clause (2) and (3) of $+\Sigma_M$, plus the inductive hypothesis.

For 2. The property follows immediately from 1 and the principle of strong negation.

For 3. To prove this case we have to show how to build a derivation for $-\partial_i p$ given a derivation of $+\partial_i E_k p$, and give the appropriate conditions. If we can derive $+\partial_i E_k p$ since we have $+\Delta_i E_k p$, then by clause (2.1) we can derive $-\partial_i p$. Otherwise we consider the rule r used to derive the conclusion. We have two cases (a) $r \in R^i[E_k p] \cup R^c[EE_i E_k p]$ or (b) $r \in R^c[E_k p]$. The two cases are analogous, the only difference is in the condition of applicability of the rule. We will say that r is applicable if the appropriate conditions in clause (2.2) of $+\partial_i$ are satisfied. We consider two exhaustive cases: (i) r is maximal, i.e., $\neg\exists s, s > r$, (ii) r is not maximal. For (i) the maximality of r ensures that clauses (2.3.1) and (2.3.2) of $-\partial_i$ are satisfied and then the applicability of r makes clause (2.3) true. Thus in this case we can derive $-\partial_i p$. For (ii) if r is not maximal we consider the set of rules $S_0 = \{s : s > r\}$. Let $R^* = R^i[p] \cup R^c[EE_i p] \cup R^c[p]$ If $S_0 \cap R^* = \emptyset$, then clauses (2.3.1) and (2.3.2) are vacuously satisfied and again we are done. Otherwise, consider a rule $s \in S_0$. If s is discarded, it meets either conditions of clause (2.3) of $+\partial_i$, then we have that s satisfies also either clause (2.2.1) or (2.2.2), and we can remove s from S_0. Otherwise if s is applicable, then there is a rule t that satisfies (2.3.3), and in particular $t > s$. At this stage we consider if t is maximal or not. If t is maximal we have a rule that satisfies clause (2.3) of $-\partial_i$, and again we are done. If t is not maximal, we consider the set $S_1 = \{s : s > t\}$. Since $>$ is acyclic we have that $S_1 \cap R^* \subset S_0 \cap R^*$. We can now repeat the above reasoning for the rules in $S_1 \cap R^*$, and we can repeat it n times for each applicable rule in the set. In this way we remove rules until we arrive at an applicable rule $t' \in R^i[p] \cup R^c[E_i p] \cup R^c[p]$ such that it is either maximal or that all stronger rules than it in R^* are discarded. In this way for every applicable rule in R^* we have a rule that satisfies clause (2.3) of $-\partial_i$, and thus we can conclude $-\partial_i p$.

For 4. The proof is by induction on the length of the proof. We start with definite conclusions. For the inductive base, i.e., $P(1) = +\Delta_i p$, either $EE_i p \in F$ or there is a rule $r \in R^i_s[p]$ such that $A(r) = \emptyset$. If $EE_i p \in F$, then $E' p \in F$ ($E' = EE_i$) and $R^i_s[p] \subseteq R_s[Ep]$, thus in both cases we can build a derivation for $+\Delta_c p$.

For the inductive step, i.e., $P(n+1) = +\Delta_i p$, we assume as usual that the property holds up to derivation of length n. Since $R^i_s[p] \subseteq R_s[Ep]$, and the conditions of applicability of strict rules in clause (3) of $+\Delta_i$ are the same as those of clause (2) of $+\Delta_c$ we have the same situation as in the inductive base.

If $P(n+1) = +\Delta_i p$ is justified by clause (4) of $+\Delta_i$ (conversion), then there is a rule $r \in R^i_s[p]$ such that $A(r) \cap \mathrm{Lit} \neq \emptyset$ and $\forall \alpha \in A(r)$, $+\Delta_c \alpha \in P(1..n)$, and $\forall a \in A(r), +\Delta_i a \in P(1..n)$ are under the inductive hypothesis, thus we have that we can build a proof P' where $\forall a \in A(r), +\Delta_c a \in P'(1..m)$, and thus we can conclude $+\Delta_c p$, using clause (2) of $+\Delta_c$. If $P(n+1) = +\Delta_i p$ is justified according to clause (2) of $+\Delta_i$, then we have that either $EE_i p \in F$ or there is a rule $r \in R_s[E_i p]$ such that $\forall l \in A(l), +\Delta l \in P(1..n)$. In both cases we can repeat the reasoning for the inductive base to prove that there is a proof for $+\Delta_c p$.

For ∂^+ the proof is essentially the same as that for Δ^+. The only differences are that we have to consider the two clauses (2.3). It is immediate to verify that (2.3.1)–(2.3.3) of $+\partial_c$ and (2.3) of $+\partial_i$ are identical; in addition we have that clause (2.3.1) of $+\partial_i$ (reinstatement by conversion) can be transformed into clause (2.3.4) of $+\partial_c$ by inductive hypothesis as we did in the case for Δ^+. Finally, for clause (2.1) all we have to do is to notice that $\mathscr{C}(p) = \{Ep\}$ and $\mathscr{C}(E_i p) = \{E{\sim}p, E\neg E_i p\}$, and thus $\mathscr{C}(p) \subseteq \mathscr{C}(E_i p)$.

THEOREM 4. *Let I be an institutional action theory. The extension of I can be computer in time linear to the size of the theory, i.e.,* $O(|R| * |U^I| * |A|)$.

Proof. The proof is based on a modification of the algorithm given by Maher [33] to show that propositional defeasible logic has linear complexity.

The main idea of the proof is to build appropriate data structure to implement a series of transformations reducing the complexity of the rules, and where each literal and modal literal is examined only once. The focal point of the transformations is based on the following properties:

- Let $D \vdash +\partial p$ then
$$D \cup \{r : p_1, \ldots, p_n, p \Rightarrow q\} \equiv D \cup \{r : p_1, \ldots, p_n \Rightarrow q\}.$$
- Let $D \vdash -\partial p$ then $D \cup \{r : p_1, \ldots, p_n, p \Rightarrow q\} \equiv D$.

The properties allow us (1) to remove already proved literals from the body of rules and (2) to remove rules which have been discarded.

The algorithm has three phases. (1) A pre-processing phase where we use the transformations given in [2] to transform a theory into an equivalent theory without superiority relation and defeaters; the transformation is linear. (2) A *rule loader* that parses the theory obtained in the first phase and generates the data structure that encodes the theory. (3) The *inference engine* applies transformations to the data structure, where at every step it reduces the complexity of the data structure.

We set $V^I = \emptyset$, then the rule loader first scans the set of rules and extracts the set of conclusions $Cn(I)$, and the set of atomic literals in it $Lit(I)$. For each element $l \in Cn(I) \cup Lit(I)$ we add $l, E_i l, \neg E_i$ for every $i \in A$ to V^I if the expressions are well formed according to the formation conditions given in Section 3.[10] At this stage the rule loader builds a data structure where every element of V^I is associated with four hash tables: $+h$ the rules that can prove the elements, $+h$ the rules that can disprove the element, $+b$ the rules that need the element to be applicable, and $-b$ the rules that can be discarded by the element. Each hash table depends on the type of literal it is associated to according to the following conditions.

For α, we have:

- $+h$ is the list of (pointers to) rules in $R^c[\alpha]$;
- $-h$ is the list of rules in $R^c[{\sim}\alpha]$;
- $+b$ is the list of rules in $\{r \in R : \alpha \in A(r)\}$;
- $-b$ is the list of rules in $\{r \in R : {\sim}\alpha \in A(r)\}$.

For p (a plain literal), we have:

- $+h$ is the list of (pointers to) rules in $R[Ep]$;
- $-h$ is the list of rules in $R[E{\sim}p]$;
- $+b$ is the list of rules in $\{r \in R : p \in A(r)\}$;
- $-b$ is the list of rules in $\{r \in R : E{\sim}p \in A(r)\}$.

For $E_i p$ (a modalised literal), we have:

- $+h$ is the list of (pointers to) rules in $R^i[p] \cup R^c[EE_i p] \cup R^c[p]$;
- $-h$ is the list of rules in $R[E{\sim}p] \cup R[E{\sim}E_i p] \cup R^c[E_k p] \cup R^i[E_k p]$ for any $k \neq i$;
- $+b$ is the list of rules in $\{r \in R : E_i p \in A(r)\} \cup \{r \in R^c : p \in A(r)\}$;
- $-b$ is the list of rules in $\{r \in R : E{\sim}p \in A(r)\} \cup \{r \in R : \neg E_i p \in A(r)\} \cup \{r \in R : E_i E_k p \in A(r)\}$, for any $k \neq i$.

Each results-in rule r is represented by the rule loader as a pair (h,b) where h is pointer to the head of the rule and b has pointers to the literals in $A(r)$. On the other hand a counts-as rules is implemented as an $n+3$-tuple ($n = |A|$, the number of agents in I) $(h,a,b,a_1,\ldots,a_n)$. h is as per results-in rules, a is the set of pointers for action literals in $A(r)$, b is the set of pointers for non action literals in $A(r)$, and each a_i is either a set of pointers to non action literals if

[10] Notice that V^I is in general smaller than U^I, but it is easy to see that for every element $e \in U^I - V^I$, we have $I \vdash -\partial_{c,i} e$.

either $A(r) \cap \mathrm{Lit} \neq \emptyset$ or there is no literal of the form $E_i p \in A(r)$; otherwise b is the special symbol *nil*.

The Inference Engine is based on an extension of the *Delores* algorithm/implementation proposed in [33] as a computational model of Basic Defeasible Logic. In turn

1. It asserts each literal $l \in F$ as a conclusion and removes l from all rules in $+b(l)$, and remove all rules (pointers to rules) in the hash tables for $-h$. For counts-as rules, if $l = \alpha$ we remove l from the a part of the rules; if $l = p$, we remove it from the p part of the b part, and if $l = E_i m$, then (1) we remove both m and $E_i m$ from the rules in $+b(E_i p)$, and (2) for counts-as rules we remove $E_i m$ and m from the b part and p from the a_i part as appropriate.

2. Then it scans the set of rules for rules where b is empty. For counts-as rules it looks for rules where both a and either b or a_i are empty for some $i \in A$. For each of such rules it takes $a(r)$ and $E_i a(r)$ (only $E_i a(r)$ for counts-as rule where a_i is empty), and it checks that $-h(a(r))$, $-h(E_i a(r))$ are empty. If so, it adds $a(r)$, $E_i a(r)$ to the set of conclusions as appropriate.

3. It repeats the first step, using the conclusions obtained from the previous step.

4. The algorithm terminates when one of the two steps fails. On termination the algorithm outputs the set of conclusions.[11]

Notice that all the operations described in the above steps correspond to hash functions, thus they have constant complexity $O(1)$. It is immediate to see that the algorithm runs in linear time. Each (modal) atom/literal in a theory is processed exactly once and every time we have to scan the set of rules, thus the complexity of the above algorithm is $O(|V^I| * |R| * |A|)$.

[11] This algorithm outputs ∂^+; ∂^- can be computed by an algorithm similar to this with the “dual actions”. For Δ^+ we have just to consider similar constructions where we examine only the first parts of step 1 and 2. Δ^- follows from Δ^+ by taking the dual actions.