A Model of Dynamic Resource Allocation in Workflow Systems

Guido Governatori[1] **Antonino Rotolo**[2] **Shazia Sadiq**[1]

[1] School of Information Technology and Electrical Engineering
The University of Queensland, Brisbane, QLD 4072, Australia
Email: {guido,shazia}@itee.uq.edu.au
[2]CIRSFID, University of Bologna
Via Galliera 3, 40121, Bologna, Italy
Email: rotolo@cirfid.unibo.it

Abstract

Current collaborative work environments are characterized by dynamically changing organizational structures. Although there have been several efforts to refine work distribution, especially in workflow management, most literature assumes a static database approach which captures organizational roles, groups and hierarchies and implements a dynamic roles based agent assignment protocol. However, in practice only partial information may be available for organizational models, and in turn a large number of exceptions may emerge at the time of work assignment. In this paper we present an organizational model based on a policy based normative system. The model is based on a combination of an intensional logic of agency and a flexible, but computationally feasible, non-monotonic formalism (Defeasible Logic). Although this paper focuses on the model specification, the proposed approach to modelling agent societies provides a means of reasoning with partial and unpredictable information as is typical of organizational agents in workflow systems.

1 Introduction

Typical business processes generally consist of a number of interrelated activities that are undertaken to achieve business goals. Process management technologies such as workflow systems provide a means of automating the coordination and scheduling of constituent activities. The successful completion of these activities depends on the availability of required resources, most important of which are the processing agents (systems and/or people). Thus it is critical that process activities are assigned to the right agent at the right time. In a highly idealised world, each activity of a process could be allocated to a specific agent (or role) specialised in the particular task. Thus, in such case, the assignment of work-items to agents could be regarded as a build-time specification, to be enacted at run-time. This scenario is hardly realistic. Most of the activities in a large organization are executed by a number of agents being placed at different levels of the organizational hierarchy and with a different list of privileges/obligations and expertise within the organization. Of course, due to unexpected events, an agent may not be available for a task at the scheduling time either for a shorter or a longer period of time.

The problem of allocation of the task on hand to the right person at a given moment of time is in practice a difficult one. Its effective solution, to a large extent, depends not only on the mechanism controlling the job allocation, its sophistication and intelligence, but also on the data availability to support such system functionality. To obviate these problems, many workflow systems have referred to an underlying organizational model in both the design phase and in the execution phase.

In general, recent experience with workflows deployment demonstrates a great deal of success in the solutions provided by this technology in terms of productivity enhancements in many process-intensive industry sectors. As with other cases of IT products reaching practical deployment, there is a complementary list of drawbacks found in current workflows systems. One of the limitations is that the workflow management systems do not provide an effective means of dealing with uncertainty and lack of data availability in resource allocation. (Kumar, van der Aalst & Verbeek 2001) provides further shortcomings which are to a large extent a result of the aforementioned limitation. These include selectivity, that is, the work is offered to too many, too few, or simply to those already involved in the work; advanced and context dependent constraints which cannot be expressed without incorporating advanced and contextual knowledge; no distinction between soft and hard constraints, so the systems are not flexible and have limited support for exceptions; no concept of substitutability or delegation (or other normative positions relevant to the underlying organization).

There have been some results reported in the literature to overcome some of the above shortcomings. (van der Aalst & Kumar 2001) proposes the use of UML, in particular Object Constraint Language (OCL) to model teams of agents and their relationships in an organization. However OCL and UML do not support concepts usually adopted in the characterisation of organizations. Similar considerations also apply to (Momotko & Sumieta 2002), where an object-oriented language is proposed as an extension of the allocation mechanism advanced by the WfMC (Hollingsworth 1995). (Kumar et al. 2001) advocates some organizational notions (for example, responsibility, authorization, delegation). (Casati, Castano & Fugin 2001) presents a solution for managing workflow authorization constraints based on ECA rules. What is lacking in many proposals is a clear description of the meaning of various organizational notions. Consequently the interpretation is left to the intuition of users of such techniques, with the obvious impact on the automation of organizations. The lack of models with clear semantics to describe organizations clearly prevents systematic formal analysis of organizations represented in this way

There has been a considerable amount of research dedicated to the development of appropriate models of automation of organizations, and it is recognised that the success of those models depends on the adoption of explicit organizational models. We believe that the allocation of tasks to participants should be guided by the relevant organizational model, in particular by the normative relationships among the agents, and by the attributes (abilities, capabilities, opportunities,...) of single agents or teams of agents. In the build-time phase the organizational model is used to accommodate the evolution of the process schemas, and in the run-time to optimise the execution of the process in a variable scenario, given the available resources.

Consider for example the workflow in Figure 1 in a typical service support business process[1]. Such processes are found in many domains such as telecommunication services, road vehicle services, city council inquiries etc. This example represents a hypothetical and simplified process, wherein, upon receipt of a customer call, a call centre agent will create a request with relevant details in the system, and allocate an engineer based on skills and availability. The request appears on the work list of the allocated engineer, who will now have to make decisions on how to resolve the customer request based on his/her expertise. In typical environments, the problem is addressed and solved in accordance with pre-determined procedures. Temporal/control constraints and quality assurance parameters are associated with these procedures. However, the engineer assigned at this level may attempt to resolve the problem, but may not always be successful. Thus, in some circumstances, a greater level of support may be required (so called level 2 support). The need for level 2 support is identified through a proclamation of the original engineer. This proclamation will generally include essential instance specific data such as a detailed problem specification, tests performed, classification of the problem (Problem A is of Type X).

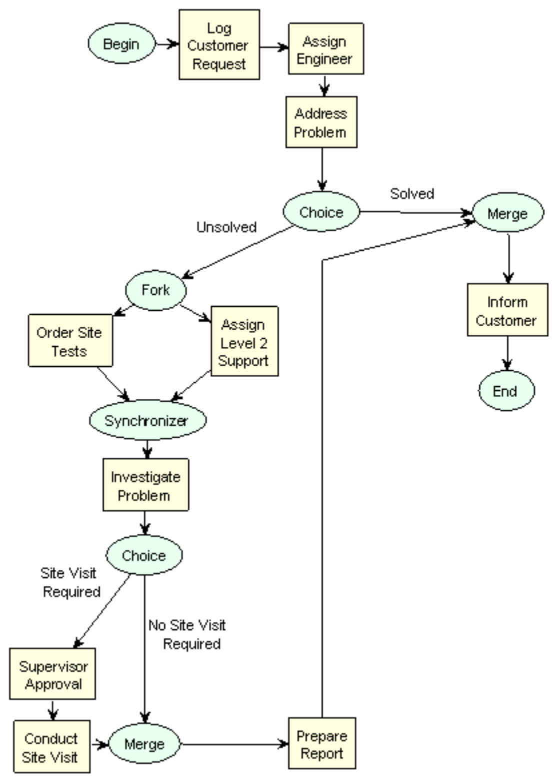

Figure 1: Service Support Workflow

Provision of further levels of support is typically more complex. It is difficult, if not impossible to determine the exact response warranted for every possible customer request. Thus some flexibility has to be afforded to the engineers-in-charge. The underlying objective is to effectively meet the customer request within the given constraints, while making the best use of individual expertise and experience, and taking into consideration a number of factors such as availability, work load, problem classification, proximity of location etc. In other words, one can anticipate that resource allocation decisions need to be based on an integrated view of the instance, organizational and resource usage data.

For example, assigning an expert engineer for level 2 support would need to take into consideration the factors identified above. Furthermore the investigation task may be decomposed and carried out through multiple agents requiring the notion of *delegation*. In fact, various rules of delegation can be applied in such scenarios, where tasks are assigned to peers and/or subordinates, these tasks are attempted, and re-assigned as more results are obtained.

In recent work on agents and agent societies a specific normative line of research has been emerging. This research assumes 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 (see, e.g., (Conte & Dellarocas 2001)). In this perspective, a precise logical analysis of normative notions such as obligations, institutions, responsibilities, delegation, powers, etc., has been recognised as one precondition for the development of norm-governed societies.

In general, we believe that an appropriate formal model of organization should conceive an organization as a *policy-based normative system*; accordingly an organization should be characterised by specifying

1. 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. The idea behind the process of allocating powers is thus necessary to account for other organizational notions such as those of *responsibility* and *delegation* (Gelati, Governatori, Rotolo & Sartor 2002). This model provides conceptual tools for define both the normative patterns of behaviour of each agent within an organization, and how the agents can relate to each other to attain the normative co-ordination needed to achieve the organizational goals;

2. a notion of agency capable of expressing agents' behaviour also without referring to their concrete actions. Specifically, the concepts of *direct* and *indirect action*, *attempt*, and their relationships with the relevant normative positions, seem to be central for describing the organization of a set of agents;

3. a non-monotonic system that formalises the business rules governing and describing the activities of an organization. The use of an appropriate non-monotonic system facilitates the formalisation of the dynamism needed to implement the flexibility required by exceptional behaviours and incomplete user's specifications and profiles.

In this paper we provide a formal machinery to capture some of the building blocks mentioned above. In particular, we focus on some basic aspects of agency and institutionalised power (see points 1 and 2). These concepts are embedded in a non-monotonic framework to account for the fundamental activities performed within an organization (see point 3). The paper is organized as follows: Section 2 outlines a

[1]The syntax of the workflow modelling language presented in this example can be found in (Sadiq & Orlowska 2000)

model of organization based on a multi-modal logic of agency; Section 3 briefly introduces Defeasible Logic, and we argue that it offers a suitable and flexible non-monotonic formalism. Section 4 shows how to adapt Defeasible Logic to cope with agency and multi-agent systems. Finally Section 5 discusses some directions for future work.

2 Institutional Agency

The background of this paper comes from the well-known Kanger-Lindahl-Pörn (Kanger 1972, Lindahl 1977, Pörn 1977) logical theory to account for agency and organized interaction (see (Elgesem 1997)). Our starting point is to take advantage of some recent contributions (Santos, Jones & Carmo 1997, Jones & Sergot 1996, Jones 2003), which have enriched this framework with some substantial refinements. As we have alluded to, the notion of agency is described in a multi-modal logical setting. Despite some well-known limitations (see (Elgesem 1997, Segerberg 1992, Royakkers 2000)), such an approach is very general since actions are simply taken to be relationships between agents and states of affairs, and very flexible since it allows the easy combination of actions and concepts like powers, obligations, beliefs, etc. As recently pointed out regarding the design of computerised multi-agent systems, such a multi-modal logic "[is] a means of supplying an intermediate level of description, falling somewhere between [...] ordinary-language account of what a system [...] is supposed to be able to do and [...] the level of implementation" (Jones 2003).

The paper is confined to two basic notions of agency and the concept of institutionalised power. The first of these notions is the idea of personal and direct action to realise a state of affairs, formalised by the modal operator E: E_iA means that the agent i brings it about that A. Different axiomatisations have been provided for it but almost all include

$$E_iA \to A$$
$$\neg E_i\top$$
$$(E_iA \wedge E_iB) \to E_i(A \wedge B)$$

and are closed under logical equivalence

$$\frac{A \equiv B}{E_iA \equiv E_iB}\ .$$

If these are some general properties for E, a specific axiom advanced in (Santos et al. 1997) –and adopted here– to characterise specifically this operator is

$$E_iE_jA \to \neg E_iA. \qquad \text{(EE}\neg\text{E)}$$

It corresponds to the idea that the brings-it-about operator expresses actions performed directly and personally. In other words this axiom states a principle of rationality for modelling co-ordination in institutional organizations: it is counterintuitive 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_i (Santos et al. 1997, Jones 2003). H_iA says that i attempts to make it the case that A. The operator H_i is not necessarily successful. Besides that, it enjoys Agglomeration and is also closed under logical equivalence. Notice that we have

$$E_iA \to H_iA.^2$$

The notion of institutionalised power is central for describing norm-governed organizations of agents and comes from the distinction between the practical ability to realise a state of affairs –which is not considered in this paper (Elgesem 1997)– and the institutional power to do this. For example, if i signs a document on behalf of her boss j, such a document is as it were signed by j only if i has been empowered to do this. 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 an institutional context (an auction), depends on whether that institution makes it effective. It is up to the institutional rules to establish whether i's act, in the conditions in which it is made, makes so that a bid is effective or not. According to Searle (Searle 1995), the rules through which institutions make effective these attempts are constitutive in character and have the form "X counts as Y in the context C". Their function is to create a special kind of facts, whose nature is institutional and conceptually distinct from that of empirical facts.

In their seminal (Jones & Sergot 1996), Jones and Sergot developed a formal approach to 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

$$E_iA \Rightarrow_s E_iB$$
$$E_iA \Rightarrow_s E_jB$$

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 (Jones & Sergot 1996, Jones 2003)).

Similarly, but more closely to Searle's intuition, (Governatori, Gelati, Rotolo & Sartor 2002, Gelati et al. 2002) argue that the counts-as link is composed by a normative conditional $\Rrightarrow$ corresponding at least to cumulative logic (system **CU** (Artosi, Governatori & Rotolo 2002)) plus the modality D_s –introduced in (Jones & Sergot 1996) but with a different meaning– to represent institutional facts. In this perspective,

$$A \Rightarrow_s B =_{def} (A \Rrightarrow D_sB) \wedge (D_sA \Rrightarrow D_sB).$$

The idea behind the above definition of the counts-as link is that we want 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 so it cannot be a normal modality. In fact, the weakening of counts-as consequents is not acceptable in the setting of (Governatori et al. 2002, Gelati et al. 2002) since, from

$$D_s(making\_a\_bid)$$

[2] In (Santos et al. 1997, Jones 2003) 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 $G_iG_jA \to G_iA$ (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 $E_iA \to G_iA$, $E_iE_jA \to E_iG_jA$ and $G_iE_jA \to G_iG_jA$ are adopted in (Santos et al. 1997).

should not follow

$$D_s(making\_a\_bid \lor drinking\_some\_water).$$

In this sense, D_s is a non-normal modality closed under logical equivalence and satisfying Agglomeration

$$D_sA \land D_sB \rightarrow D_s(A \land B)$$

and Consistency

$$\neg D_s\bot.$$

Of course, necessitation does not hold: it sounds strange that $\top$ is an institutional fact for any institution s. Finally, notice that the axiom

$$D_sE_iA \rightarrow D_sA$$

guarantees successfulness also within the domain of every institution s.

Basically, we will follow here the intuitions presented in (Governatori et al. 2002, Gelati et al. 2002). Though in different perspectives, however, an important point shared by (Jones & Sergot 1996, Jones 2003) and (Governatori et al. 2002, Gelati et al. 2002) is that the counts-as link is defeasible. This is a crucial feature of this notion. 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.

In a more general perspective, notice that the above framework is able to capture some composite concepts regarding the normative co-ordination of agents. For example, (Governatori et al. 2002, Gelati et al. 2002) show that the introduction of the notion of proclamation allows to account for the ideas of declarative power and delegation.

The idea of delegation plays an important role in this setting. To delegate means basically to entrust a representative to act on behalf of someone else. As pointed out by (Castelfranchi & Falcone 1998), the very notion of agent makes itself explicit reference to delegation. The allocation of tasks via delegation may be articulated according to different strategies and levels. A task is in fact a goal that can be decomposed through a plan into a set of the sub-goals to be realised by different agents. Second, different forms of delegation may be specified on the basis of the cognitive characterisation of the agents involved in the process, i.e., on the kind of interaction between the delegating agent and the delegated one (i and j respectively). In (Castelfranchi & Falcone 1998), for example, it is argued that delegation is

(a) weak, when there is no agreement, request or influence and i just exploits a fully autonomous action of j,

(b) mild, when it is based on induction, namely on the active indirect achievement by i of the task,

(c) strict, where there is an explicit agreement between i and j.

In addition, delegation may be close or open, depending on the degree of specification of the tasks assigned, namely on the level of granularity of them in the plan. We will not explore here these issues, which would require a much more complex logical framework than that developed in the subsequent sections.

More important, the idea of delegation, insofar as it is applied to workflows, will be analysed within a non-intention-based approach. No reference is made here to the notion of cognitive agent (with its beliefs, intentions, desires, etc.), since delegation is viewed as a purely normative concept in a policy-based system. Normative delegation is usually associated with normative co-ordination, i.e., with a specific way of achieving co-ordination by the allocation of normative positions (Norman & Reed 2001, Gelati et al. 2002, Governatori et al. 2002). This happens when, for example, a manager, entrusted with a certain task, can design a multi-agent plan for achieving this task by assigning duties to subordinated agents. Notice that the allocation of duties –which is not considered in this paper– is not the fundamental notion in this process. The key concept is that of power. This holds not only for the obvious reason that the allocation of duties requires the power to do this. Rather, it is intuitive to say that the idea of normative delegation takes place just when an agent entrusts its representative to act on its behalf and the allocation of this task or goal corresponds to conferring the power to achieve it.

The logical representation of this institutional power has a counts-as structure. However, as emphasised in (Gelati et al. 2002, Governatori et al. 2002) (cf. (Norman & Reed 2001, Norman & Reed 2002)) an important role is also played by agent communication concepts. Recently, the link between speech acts theory and normative positions has been widely investigated (cf. (Jones 1990, Castelfranchi, Dignum, Catholijn & Treur 2000, Singh 1999, Colombetti 2000). In (Gelati et al. 2002, Governatori et al. 2002) it is defined a unique speech act (proclaiming) to model all speech acts characterised by a world to word direction of fit, that is all speech acts which are intended to modify the institutional world. Such a notion is formalised by the modal operator *proc*. Its logic is characterised by some very minimal properties: it is closed under logical equivalence and includes at least the axiom

$$(proc_iA \land proc_iB) \equiv proc_i(A \land B).$$

Of course, *proc* is not necessarily successful. $proc_iA$ is just an attempt to achieve A:

$$proc_iA \rightarrow H_iA.$$

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.

This minimal characterisation permits to provide *proc* with a new reading that is suitable to model workflows settings. In particular *proc* can be used to model "declarative data flow" intended to supplement or override previous (system) decisions and procedures. For example, in the scenario depicted in Figure 1, a call-centre operator classifies a customer request according to a well-defined procedure. Let us say that she classifies a particular request as a problem of type A. As a result a task will be assigned to an engineer specialized in problems of type A. After a preliminary analysis the engineer realizes that the request has been misclassified, and that it does not fall in the realm of her main expertise; therefore the task cannot be completed successfully. However, the engineer has a better understanding of the field than the call-centre operator and she is in a position to offer a better classification. Thus she can override the previous classification by proclaiming that the problem is of a different type, let us say of type B ($proc_E\,Type\ B$).

Another important issue is that the combination of *proc* and the counts-as link enables us to capture two forms of normative delegation, intended as kinds of true representation (Gelati et al. 2002). The first is:

$$proc_j(proc_iA) \Rightarrow_s E_j(proc_iA) \qquad (1)$$

that is, when j proclaims that i proclaims that A, this counts as j making so that i proclaims that A[3]. In addition, we can have:

$$proc_j(E_i A) \Rightarrow_s E_j(E_i A) \quad (2)$$

This type of representation is necessary when the representative substitutes a principal which would not be able to perform directly the activity delegated to the representative.

Back to the example in Figure 1, let us suppose that the supervisor is responsible for writing the report. However, for routine reports he might delegate this task to the engineering who has worked on the problem. This particular type of delegation is represented by the statement:

$$proc_s(E_e report\_ready) \Rightarrow_s E_s(E_e report\_ready) \quad (3)$$

Now all the supervisor has to do to delegate this task is just to proclaim that the engineer has to prepare the report, i.e., $proc_s(E_e report\_ready)$.

In the next sections we develop a computational framework, based on Defeasible Logic, able to treat the basic mechanisms of institutional agency. Although (Jones & Sergot 1996, Jones 2003) and (Governatori et al. 2002, Gelati et al. 2002) provide an interesting analysis, they can hardly be used directly for implementation. This is clearly due at least to the well known computational limits of conditional logics (see, e.g., (Artosi et al. 2002)). In this perspective, some basic patterns of defeasible reasoning will be extended to account for the institutional dynamics insofar as they are interplayed with the notions of direct action and attempt.

3 Overview of Defeasible Logic

Defeasible Logic is a simple, efficient but flexible non-monotonic formalism which has been proven able to deal with many different intuitions of non-monotonic reasoning (Antoniou, Billington, Governatori, Maher & Rock 2000), has been applied in many fields in the last few years. Here we propose a non-monotonic logic of agency based on the framework for Defeasible Logic developed in (Antoniou, Billington, Governatori & Maher 2000).

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 (Nute 1987, Antoniou, Billington, Governatori & Maher 2001) 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 have one of the following four forms:

$+\Delta q$ meaning that 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 n

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

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

The definition of Δ describes just forward chaining of strict rules. For a literal q to be definitely provable we need to find a strict rule with head q, of which all antecedents have already been definitely proved. And 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.

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

Let us work through this condition. To show that q is defeasibly provable 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 either a strict or a defeasible rule with head q which can be applied (2.1). 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 (2.2). 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

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

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 $-\partial q$ as

$-\partial$: If $P(n+1) = -\partial q$ then
(1) $-\Delta q \in P(1..n)$ and
(2.1) $\forall r \in R_{sd}[q]\, \exists a \in A(r)\colon -\partial a \in P(1..n)$ or
(2.2) $+\Delta{\sim}q \in P(1..n)$ or
(2.3) $\exists s \in R[{\sim}q]$ such that
(2.3.1) $\forall a \in A(s)\colon +\partial a \in P(1..n)$ and
(2.3.2) $\forall t \in R_{sd}[q]$ either $t \not> s$ or
$\exists a \in A(t)\colon -\partial a \in P(1..n)$.

The purpose of the $-\partial$ inference conditions 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 conclusions tagged with $-\partial$ are the outcome of a constructive proof that the corresponding positive conclusion cannot be obtained.

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) $\exists r \in R_{sd}[p]\ \forall a \in A(r)\colon +\Sigma a \in P(1..n)$.

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

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

The inference conditions for a negative proof tag are derived from the inference conditions for the corresponding positive proof tag by applying the Principle of Strong Negation introduced in (Antoniou, Billington, Governatori & Maher 2000). 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 replaces the positive tags with the respective negative tags and vice-versa. Accordingly, in what follows, we will list only the positive version of the inference rules.

4 A Defeasible Logic of Institutional Agency

As we have seen in Section 2 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 counterintuitive 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,

$$\frac{\vdash \varphi}{\vdash \Box\varphi}\text{Necessitation} \qquad \frac{\vdash \varphi \supset \psi}{\vdash \Box\varphi \supset \Box\psi}\text{RM}$$

Both dictate conditions for introducing modalities in contrast with the analysis of institutional agency as outlined in Section 2. To comply with the properties of this notion, in the setting provided by Defeasible Logic 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 it about that*), 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 be able to reason about actions we extend the language of Defeasible Logic with a set of action symbols; we will use $\alpha_i, \beta_i, \gamma_i$ to denote atomic actions. The intended 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.

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., $\rightarrow_c$, $\Rightarrow_c$, $\rightsquigarrow_c$), $\{R^i\}_{i\in A}$ is a family of sets of results-in rules (i.e., $\rightarrow^i$, $\Rightarrow^i$, $\rightsquigarrow^i$ for each agent $i \in A$), 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 organization[4]. Technically the rules in R^A are used to introduce modal operators. To correctly 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 meaningless 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 the results of actions, not actions themselves.

[4]Accordingly, counts-as rules correspond to the case $D_s A \Rightarrow D_s B$ mentioned in Section 2. Roughly speaking, the case $A \Rightarrow D_s B$ 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. In this sense, no reference to the modality D_s is required in this setting. At any rate, this fact will be clear when we will give the definitions of the proof conditions.

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 agents e (engineer) and s (supervisor) are acting in the context of the workflow of Figure 1. As we have seen in Section 2, the notion of delegation corresponds to a counts-as rule where the antecedent is a particular type of proclamation. Accordingly the rule[5]

$$proc_s(E_e report\_ready) \Rightarrow_c E_s(E_e(report\_ready))$$

represents the eventuality that a supervisor delegates the writing of the report to the engineer who worked on the problem.

$$\mathbf{prepare\_report}_e,\ proc_s(E_e report\_ready) \Rightarrow^s report\_ready$$

This rule is an example corresponding to the introduction of the modality E_s. In fact, the fulfilment of the conditions in the antecedent produces the occurrence of *report_ready*: e's action of writing the report when delegated by his supervisor has the result that s has prepared the report. Formally the derivation of *report_ready* permits the introduction of $E_s(report\_ready)$.

$$proc_s(E_e report\_ready) \Rightarrow^s \neg report\_ready$$

The example above does not specify any action in the antecedent (empty action). This means that e's refraining from doing any action, when delegated to do it, has the result to keep the problem unsolved. In logical terms, also this case can lead to the introduction of E_s[6].

Let us consider two further examples of counts-as rules.

$$proc_s(E_e report\_ready), \mathbf{prepare\_report}_e \Rightarrow_c \mathbf{prepare\_report}_s$$

This rule says that e's action of writing the report counts as the action of s of writing the report.

$$proc_s(E_e report\_ready), \mathbf{prepare\_report}_e \Rightarrow_c report\_ready$$

This rule is an example of the institutional analogous of a results-in rule, where an action and a state of affairs occur respectively in their antecedent and consequent. In this case the result is an institutional fact and follows by convention only within the institution. That the report is ready is a consequence of e's writing it is not a simple matter of e's action and s delegation. The attempt of e to prepare the report is effective only if the institution recognises this.

We are now ready to give the definitions of the proof conditions. For counts-as derivability (R^c) we assume the basic conditions of Defeasible Logic given in Section 3. Thus $\pm\Delta_c$, $\pm\partial_c$ correspond, respectively, to $\pm\Delta$ and $\pm\partial$.

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 $I \vdash +\partial_i p$, 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 of 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 2 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 the same way we have that $-\partial_i p$ corresponds to $\neg E_i p$ and $-\Sigma_i p$ to $\neg H_i p$. 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) $E_i p \in F$; or
(2) $\exists r \in R^c_s[p]\ \forall a, \alpha \in A(r)$:
$+\Delta_i a, +\Delta_c \alpha \in P(1..n)$; or
(3) $\exists r \in R^i_s[p]\ \forall a, E_j b, \alpha \in A(r)$:
$+\Delta_c a, +\Delta_j b, +\Delta_c \alpha \in P(1..n)$.

To prove an indefeasible brings-it-about, we need either that it is given as a fact (1), or that we have a strict rule for results-in (an irrevocable policy) whose antecedent is indisputable (3). However we have another case (2): if an agent knows that B is an indisputable consequence of A in the institution (it always is the case that A counts as B), and it produces A, then it must realise B. This is in contrast with the NML interpretation whereby the agent has to bring about all consequences of his/her actions.

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

The inference conditions for H_i are very similar to those for strong brings-it-about; 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.

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 as follows:

$+\partial_i$: if $P(n+1) = +\partial_i p$ then
(1) $+\Delta_i p \in P(1..n)$ or
(2.1) $-\Delta_c{\sim}p, -\Delta_i{\sim}p \in P(1..n)$ and
(2.2) $\exists r \in R^c_{sd}[p]\ \forall a \in A(r) : +\partial_i a \in P(1..n)$, or
$\exists r \in R^i_{sd}[p]\ \forall E_j b, a, \alpha \in A(s)$:
$+\partial_j b, +\partial_c a, +\partial_c \alpha \in P(1..n)$; and
(2.3) $\forall s \in R[{\sim}p] \cup R^i[E_k p]$ either
(2.3.1) $\exists \alpha \in A(s) : -\partial_c \alpha \in P(1..n)$, or
(2.3.2) if $s \in R^c[{\sim}p]$ then $\exists a \in A(s)$:
$-\partial_c a \in P(1..n)$; and
if $s \in R^A[{\sim}p] \cup R^i[E_k p]$ then either
$\exists E_j a \in A(s) : -\partial_j a \in P(1..n)$ or
$\exists a \in A(s) : -\partial_c a \in P(1..n)$; or
(2.3.3) $\exists t \in R[p]$ such that $t > s$ and
$\forall \alpha \in A(t) : +\partial_c \alpha \in P(1..n)$ and
if $t \in R^c[p]$ then $\forall a \in A(t)$:
$+\partial_c a \in P(1..n)$; and
if $t \in R^i[p]$ then $\forall a, E_j b \in A(t)$:
$+\partial_c a, +\partial_j a \in P(1..n)$.

The conditions for proving the results of defeasible actions are essentially the same as those given for defeasible derivations in Section 3. The only difference is that 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

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

[6] The ideas of empty action and refraining from a doing a specific action should not be confused with $\neg E_i A$; as we will see, this corresponds to the non-derivability of A within I, which can depend also on reasons that have nothing to do with i's refraining from acting to realise A.

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 an institutional counts-as rule can be transformed into a results-in if all the literals in the antecedent are defeasibly executed.

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 about. Similarly, if we replace r_3 by

$$r_3'' : E_i r \Rightarrow_c t$$

we can no longer derive $E_i t$. In this case $E_i r$ 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

$$r_2' : q \Rightarrow^i r,$$

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 r_3 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 2.

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 conclusions with these tags are the outcome of 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 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.

Let I be an institutional action theory. 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 *For every institutional action theory I, 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. *if $I \vdash +\partial_i E_j p$ then $I \vdash -\partial_i p$.*

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 2.

5 Conclusion and Future Work

We have presented a modelling framework for agent societies through the combination of an intentional notion of institutional agency, and a (computationally oriented) non-monotonic system (Defeasible Logic). We have shown how to provide a sound theoretical and practical non-monotonic framework to reason about the organizational notions (proclamation, declarative power, delegation, ...) that constitute the proposed model. We believe that this approach shows potential for an intelligent and dynamic resource allocation protocol when one has to reason under uncertainty and limited information which is a typical characteristic of current organizational setups.

The logic presented here is just one of the many logics that can be defined using the main idea of the paper (see Section 4). 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. In a similar way the designer can chose more or less liberal conditions to use counts-as rules to derive brings-it-about literals. 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 derive as results-in. A more liberal condition could just require that only one of them is derived in such a way.

Finally, one important question, which we could not address here, is to embed deontic notions into our system. Defeasible deontic logic is widely discussed in literature (Nute 1997). How to adapt or re-frame existing models (see (Nute 1998)) in the present setting is a matter of future research.

References

Antoniou, G., Billington, D., Governatori, G. & Maher, M. J. (2000), A flexible framework for defeasible logics, *in* 'Proc. American National Conference on Artificial Intelligence (AAAI-2000)', AAAI/MIT Press, Menlo Park, CA, pp. 401–405.

Antoniou, G., Billington, D., Governatori, G. & Maher, M. J. (2001), 'Representation results for defeasible logic', *ACM Transactions on Computational Logic* **2**(2), 255–287.

Antoniou, G., Billington, D., Governatori, G., Maher, M. J. & Rock, A. (2000), A family of defeasible reasoning logics and its implementation, *in* W. Horn, ed., 'ECAI 2000. Proceedings of the 14th European Conference on Artificial Intelligence', IOS Press, Amsterdam, pp. 459–463.

Artosi, A., Governatori, G. & Rotolo, A. (2002), 'Labelled tableaux for non-monotonic reasoning: Cumulative consequence relations', *Journal of Logic and Computation* **12**(6), 1027–1060.

Casati, F., Castano, S. & Fugin, M. (2001), 'Managing workflow authorization constraints through active database technology', *Information Systems Frontiers* **3**(3), 319–338.

Castelfranchi, C., Dignum, F., Catholijn, M. & Treur, J. (2000), Deliberative normative agents: Principles and architecture, *in* 'Proceedings of ATAL 1999', Springer, Berlin, pp. 364–378.

Castelfranchi, C. & Falcone, R. (1998), 'Towards a theory of delegation for agent-based systems', *Robotics and Autonomous Agents* **24**, 141–157.

Colombetti, M. (2000), A commitment-based approach to agent speech acts and conversations, *in* M. Greaves, F. Dignum, J. Bradshaw & B. Chaibdraa, eds, 'Proceedings of the Fourth International Conference on Autonomous Agents, Workshop on Agent Languages and Conversation Policies', Barcelona, pp. 21–29.

Conte, R. & Dellarocas, C. (2001), *Social Order in Multiagent Systems*, Kluwer Academic Publishers, Boston.

Elgesem, D. (1997), 'The modal logic of agency', *Nordic Journal of Philosophical Logic* **2**, 1–48.

Gelati, J., Governatori, G., Rotolo, A. & Sartor, G. (2002), Declarative power, representation, and mandate: A formal anaysis, *in* T. Bench-Capon, A. Daskalopulu & R. Winkels, eds, 'Legal Knowledge and Information Systems', IOS Press, Amsterdam, pp. 41–52.

Governatori, G., Gelati, J., Rotolo, A. & Sartor, G. (2002), Actions, institutions, powers. preliminary notes, *in* G. Lindemann, D. Moldt, M. Paolucci & B. Yu, eds, 'International Workshop on Regulated Agent-Based Social Systems: Theories and Applications(RASTA'02)', Fachbereich Informatik, Universität Hamburg, pp. 131–147.

Hollingsworth, D. (1995), *The Workflow Reference Model*, issue 1.1 edn, Workflow Management Coalition, Winchester. WfMC-TC-1003.

Jones, A. (1990), Towards a formal theory of communication and speech acts, *in* P. Cohen & M. Pollack, eds, 'Intentions in Communication', MIT Press, Cambridge, Mass.

Jones, A. J. (2003), A logical framework, *in* J. Pitt, ed., 'Open Agent Societies: Normative Specifications in Multi-Agent Systems', John Wiley and Sons, Chichester, chapter 3.

Jones, A. & Sergot, M. (1996), 'A formal characterisation of institutionalised power', *Journal of IGPL* **3**, 427–443.

Kanger, S. (1972), 'Law and logic', *Theoria* **38**, 105–32.

Kumar, A., van der Aalst, W. M. & Verbeek, E. M. (2001), 'Dynamic work distribution in workflow management systems: How to balance quality and performance?', *Journal of Management Information Systems* **18**(3), 157–129.

Lindahl, L. (1977), *Position of change: A Study in law and logic*, Reidel, Dordrecht.

Momotko, M. & Sumieta, K. (2002), Dynamic changes in workflow partecipant assignment, *in* 'East-European Conference on Advances in Databases and Information Systems', Vol. 2, Bratislava.

Norman, T. J. & Reed, C. A. (2001), Delegation and responsibility, *in* C. Castelfranchi & Y. Lesperance, eds, 'Intelligent Agents VII: Proceedings of the Seventh International Workshop on Agent Theories, Architectures and Languages', Vol. 1986 of *LNAI*, Springer-Verlag, Berlin, pp. 136–149.

Norman, T. J. & Reed, C. A. (2002), A model of delegation for multi-agent systems, *in* M. d'Inverno, M. Luck, M. Fisher & C. Preist, eds, 'Foundations and Applications of Multi-Agent Systems', Vol. 2403 of *LNAI*, Springer-Verlag, Berlin, pp. 185–204.

Nute, D. (1987), Defeasible logic, *in* 'Handbook of Logic in Artificial Intelligence and Logic Programming', Vol. 3, Oxford University Press, pp. 353–395.

Nute, D. (1998), Norms, priorities, and defeasible logic, *in* P. McNamara & H. Prakken, eds, 'Norms, Logics and Information Systems', IOS Press, Amsterdam, pp. 201–218.

Nute, D., ed. (1997), *Defeasible Deontic Logic*, Kluwer, Dordrecht.

Pörn, I. (1977), *Action Theory and Social Science: Some Formal Models*, Reidel, Dordrecht.

Royakkers, L. (2000), Combining deontic and action logics for collective agency, *in* J. B. et al., ed., 'Legal Knowledge and Information Systems (Jurix)', IOS Press, Amsterdam.

Sadiq, W. & Orlowska, M. E. (2000), 'Analysing process models using graph reduction techniques', *Information Systems* **25**(2), 117–134.

Santos, F., Jones, A. & Carmo, J. (1997), Action concepts for describing organised interaction, *in* 'Thirtieth Annual Hawaii International Conference on System Sciences', IEEE Computer Society Press, Los Alamitos.

Searle, J. (1995), *The Construction of Social Reality*, Penguin Press, Harmondsworth.

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

Singh, M. (1999), 'An ontology for commitments in mutliagent systems: toward a unification of normative concepts', *Artificial Intelligence and Law* **7**, 93–113.

van der Aalst, W. M. & Kumar, A. (2001), 'A reference model for team-enabled workflow management systems', *Data & Knowledge Engineering* **38**, 335–363.