Specifying Open Agent Systems: A Survey

Alexander Artikis[1,2] and Jeremy Pitt[2]

[1] Institute of Informatics and Telecommunications,
National Centre for Scientific Research "Demokritos", Athens 15310
[2] Electrical & Electronic Engineering Department,
Imperial College London, SW7 2BT
a.artikis@acm.org, j.pitt@imperial.ac.uk

Abstract. Electronic markets, dispute resolution and negotiation protocols are three types of application domain that can be viewed as open agent systems. Members of such systems are developed by different parties and have conflicting goals. Consequently, they may choose not to, or simply fail to, conform to the norms governing their interactions. It has been argued that many practical applications in the future will be realised in terms of open agent systems of this sort. Not surprisingly, recently there is a growing interest in open systems. In this paper we review and compare four approaches for the specification of open systems, pointing out the extent to which they satisfy a set of requirements identified in the literature.

1 Introduction

Agents, although required to make decisions and act locally, operate in the context of multi-agent systems (MAS). A particular kind of multi-agent system is one where the members are developed by different parties and have conflicting goals. A key characteristic of this kind of MAS, that is due to the globally inconsistent sets of goals of the members, is the high probability of non-conformance to the specifications of the systems. A few examples of this type of MAS are electronic marketplaces, dispute resolution protocols, Virtual Organisations and digital media rights management protocols. Multi-agent systems of this type are often classified as 'open'. It has been argued that many practical applications in the future will be realised in terms of 'open agent systems' (OAS) of this sort.

For the purposes of this paper, we consider a multi-agent system as open if it exhibits the following characteristics:

- The internal architectures of the members are not publicly known.
- Members do not necessarily share a notion of global utility [38].
- The behaviour and the interactions of the members cannot be predicted in advance [24].

The first characteristic implies that an OAS may be composed of agents with different internal architectures. Therefore, we will treat OAS as *heterogeneous* ones. Moreover, there is no direct access to an agent's mental state and so we

A. Artikis, G. Picard, and L. Vercouter (Eds.): ESAW 2008, LNAI 5485, pp. 29–45, 2009.

can only infer things about it. The second characteristic implies that the members of an OAS may fail to, or even choose not to, conform to the specifications (of that system) in order to achieve their individual goals. In addition to these characteristics, OAS are always subject to unanticipated outcomes in their interactions [24].

Often in the literature OAS are those where agents may enter or leave the system at any time (see, for example, [24,38,55,60]). Usually, agents enter a system after having successfully executed a *role-assignment protocol*. The specification of such a protocol is application-specific. Our definition of an agent system as open is irrespective of the protocols that specify the ways by which agents enter or leave the system.

Recently there is a growing interest in the MAS community in OAS. The aim of this paper is to present a critical survey of approaches for the specification of OAS. We review the following lines of work: (a) artificial social systems, (b) enterprise modelling, (c) commitment protocols, and (d) electronic institutions. We chose to review the aforementioned lines of work because they are well-known and illustrate different aspects of OAS. Clearly there are other prominent approaches for specifying OAS, including [3,4,5,6,8,18,22,28,32,40,43,54]. Given the space limitations, however, it is not possible to present here a review of all relevant lines of research.

Following [41] the review of each approach is divided into two parts: (i) the description of the approach, and (ii) our commentary on the approach. Consequently the paper may be read in two ways: its main aim is to describe and compare the various lines of research, but it can also be read as a catalogue of abstracts by ignoring the commentary included with each entry.

The commentary of each approach includes a discussion on whether a set of requirements identified in the literature for the specification of OAS are addressed. It has been argued, for example, that OAS may be viewed as instances of *normative systems* [26]. A feature of this type of system is that actuality, what is the case, and ideality, what ought to be the case, do not necessarily coincide. Therefore, it is essential to specify what is permitted, prohibited, and obligatory, and perhaps other more complex normative relations (such as duty, right, privilege, authority, . . .) that may exist between the agents. Among these relations, considerable emphasis has been placed on the representation of 'institutional' (or 'institutionalised') power [27,30]. This is a feature of norm-governed systems whereby designated agents, when acting in specified roles, are empowered by an institution to create relations or states of affairs of special significance within that institution — *institutional facts* [39]. Consider, for example, the case in which an agent is empowered by an institution to award a contract and thereby create a bundle of normative relations between the contracting parties. According to the account given by Jones and Sergot [27], institutional power can be seen as a special case of a more general phenomenon whereby an action, or a state of affairs, A — because of the rules and conventions of an institution — counts, in that institution, as an action or state of affairs B [39] (such as when sending a letter with a particular form of words counts as making an offer).

It has also been argued that the semantics of the rules of an OAS must be, among other things, *formal*, *declarative* (the semantics should describe *what* rather than *how*), and *verifiable* (it should be possible to determine whether an agent is acting according to the rules of an OAS) [50, 52]. In the commentary of each approach we will point out the extent to which these requirements are satisfied. Finally, the commentaries include a discussion on the types of computational task that may be performed on the system specification.

A short survey of interaction protocols for MAS was recently presented in [31]. The criteria against which each protocol was evaluated are different from the ones we are concerned with. In [31] emphasis was placed on the issue of dynamic protocol composition. We do not exclude from our survey protocols that are not dynamically composable; moreover, unlike the survey of [31], we are concerned with the normative relations that are expressed by each approach for specifying OAS, whether the approach has a verifiable semantics, and the supported computational tasks.

2 Artificial Social Systems

Moses, Tennenholtz and Shoham [33, 34, 47, 48, 53] present in various papers an approach to the design of multi-agent systems, called artificial social systems. An artificial social system is based on a set of restrictions on the agents' behaviour, called *social laws.* These laws allow the agents to co-exist in a shared environment and pursue their goals. Social laws are determined at design-time, that is, before the commencement of the agent activities, and enable the members of a system to create their own plans at run-time.

Moses and Tennenholtz [34] define artificial social systems first by defining multi-agent systems in general, and then by defining *normative systems* and *social systems.*

A multi-agent system $\mathcal{M}$ is represented as a tuple of the form $\mathcal{M} = (N, W, K_1, \ldots, K_n, A, Able_1, \ldots, Able_n, I, T)$, where:

- $N = \{1, \ldots, n\}$ is a set of agents.
- W is a set of possible worlds (states).
- $K_i \subseteq W \times W$ are accessibility (equivalence) relations that capture the knowledge of each agent i ($i \in N$).
- A is a set of primitive individual actions.
- $Able_i : W \rightarrow 2^A$ are functions that determine the set of actions that each agent i is physically capable of performing ($i \in N$).
- I is a set of possible external inputs for the agents. The elements of this set are intended to capture any messages that an agent may receive from outside the system.
- $T : W \times (A \times I)^n \rightarrow W \cup \{\perp\}$ is a state transition function that determines the state immediately after the current one as a function of the actions that each agent performs, and the input each agent receives in the current state.

A normative system is a multi-agent system associated with a set of social laws. Tennenholtz [53] structures the states of a system in order to define social laws.

Given disjoint sets of states $S_1, \ldots, S_n$, the set of system states W is defined as the set of all tuples $(s_1, \ldots, s_n)$ where $s_i \in S_i$ for each agent i. Given a system state $w \in W$, w_i is the i-th element of w.

Given a set of system states W, a first-order language $\mathcal{L}$ (with an entailment relation $\models$), and a set of actions A, a social law is a set of constraints of the form (α_j, φ_j) where $\alpha_j \in A$ and $\varphi_j \in \mathcal{L}$, at most one for each $\alpha_j \in A$.

The language $\mathcal{L}$ is used to describe what is true and false in different system states. Intuitively, an action a is *prohibited* for agent i at state w by the constraint (a, φ) if and only if $w_i \models \varphi$. $\varphi \in \mathcal{L}$ is the most general condition that prohibits the performance of action a.

Every physically possible action that is not prohibited by a social law at a particular state is considered *allowed* (according to that law) in that state. The $Legal_i : W \rightarrow 2^A$ functions represent the set of actions that agent i is allowed to perform in state $w \in W$, according to the system's laws. The $Legal_i$ functions satisfy the three following properties:

1. Each agent knows what actions it is allowed to perform (*epistemological adequacy*).
2. The set of actions that an agent is allowed to perform are physically possible for that agent (*physical adequacy*): $\forall i \in N,\ \forall w \in W,\ Legal_i(w) \subseteq Able_i(w)$.
3. In every state, there is at least one action an agent is allowed to perform (*non-triviality*): $\forall i \in N,\ \forall w \in W,\ Legal_i(w) \neq \emptyset$.

A normative system $\mathcal{N}$, extending a multi-agent system $\mathcal{M}$, is defined as a pair $\mathcal{N} = (\mathcal{M}, \{Legal_i\}_{i \in N})$.

In a normative system the state transition function takes as an argument not only the current state and the actions performed by the agents but also the social laws that are in force (which are associated with the performed actions). Given such input, the state transition function produces a prediction about the set of possible next states, provided that all agents comply with the laws.

A social system is regarded as an instance of a normative system. All possible states in a social system must be 'socially acceptable'. This will happen if the initial states are 'socially acceptable' and as long as every agent obeys the laws of the system. Moreover, each agent should always be able to attain a set of 'socially acceptable' goals — these are the goals that the social system allows the agents to achieve.

The set of 'socially acceptable' states is represented by the W_{soc} set ($W_{soc} \subseteq W$) whereas the set of 'socially acceptable' goals is represented by the G_{soc} set. An agent's goal g is associated with a set of states $W_g \subseteq W$; these are the states in which the goal has been achieved. The W_{soc} set is intended to capture the 'safety' and 'fairness' of the system while the G_{soc} set is intended to capture the 'liveness' of the system. The definition of both of these sets is application-specific.

Social systems are defined in terms of the agents' *legal plans*. A plan for an agent i is defined as a function $Ch_i : W \rightarrow A$ that satisfies the following property:

- Each chosen action must be physically possible:
  $\forall i \in N,\ \forall w \in W,\ Ch_i(w) \in Able_i(w)$.

The above property is strengthened in order to define a *legal* plan:

- Each chosen action must be allowed according to the system's laws: $\forall i \in N,\ \forall w \in W, Ch_i(w) \in Legal_i(w)$.

A social system $\mathcal{S}$ for a multi-agent system $\mathcal{M}$, consistent with W_{soc} and G_{soc}, is a normative system extending $\mathcal{M}$ that satisfies the following:

1. A state $w \in W$ is *legally reachable* only if it is 'socially acceptable', that is, $w \in W_{soc}$.
2. For every legally reachable state w, if the goal of agent i in w is 'socially acceptable', that is $g \in G_{soc}$, then there is a legal plan for i that, starting in w, will attain g as long as all other agents act according to the laws of the social system.

Fitoussi and Tennenholtz [16,17] define two properties of social laws. These properties are called *minimality* and *simplicity.* The rationale for defining minimal social laws is that a minimal law provides agents more freedom in choosing their behaviour (that is, it prohibits fewer actions) while ensuring that agents conform to the system specification. The rationale for defining simple social laws is that a simple law relies less on the agents' capabilities rather than a non-simple one. Learning a simple law is likely to be faster and its representation is likely to be more succinct.

As already mentioned, work on artificial social systems focuses on the design-time specification of laws that allow agents to achieve their goals at run-time. Sometimes it is not feasible to design social laws at design-time. For example, it might be the case that all the characteristics of a system are not known at design time. In such a case the members of a system will *converge* to social laws. In a complementary (to the artificial social systems approach) work, Shoham and Tennenholtz [46, 49] examine the case in which laws emerge at run-time. The issue of law emergence is beyond the scope of this paper; therefore, we do not present here this line of work.

Commentary

The work on artificial social systems has been very influential in the field of multi-agent system specification. We make the following comments with respect to the suitability of this approach for OAS specification. First, social laws only refer to the permissions of the agents, regulating their behaviour. No other normative relations are considered. This is probably due to the fact that the applications that have been studied in the context of the artificial social systems approach (for example, mobile robots moving along a two-dimensional grid [48, Section 2] or a circular automated assembly line [17, Section 3.1]) can be mainly described in terms of physical actions rather than communicative actions, and *brute facts* rather than institutional facts [39]; being in physical possession of an object, for example, is a brute fact (it can be observed), whereas being the owner of that object is an institutional fact.

Second, in the artificial social systems approach it is assumed that all members comply with the social laws. However, as Shoham and Tennenholtz [47, p.281] mention, the members of agent systems (and especially OAS) do not always act according to the social laws. There is a need, therefore, to consider and develop mechanisms that deal with non-conformance to the laws.

The semantics of social laws are formal and declarative. Regarding the verification of compliance to social laws, it is necessary that the laws make no assumptions about the internal architectures of the agents, that is, ϕ in law (a, ϕ) does not refer to the mental states the agents — recall that there is no access to the agents' internal states in an OAS. Finally, to the best of our knowledge, there is no discussion about a software implementation allowing for reasoning about the social laws of a system (thus, informing the decision-making of the agents by computing legal plans, allowed actions, and so on).

3 Enterprise Modelling

In this section we review the work of Fox, Grüninger and colleagues [19, 23] on modelling computational enterprises/organisations. In this approach an organisation is viewed as a set of agents playing roles in which they are acting to achieve specific goals according to various constraints defining the 'rules of the game'. More precisely, an organisation consists of: *divisions* and *subdivisions* (recursively defined), a set of *organisation agents* (members of divisions), a set of *roles* that members play in the organisation, and an *organisation goal tree* that specifies the goal (and the sub-goals) that members are trying to achieve.

A role R is associated with, among other things, a set of: (i) *goals* that each agent occupying R is obliged to achieve, (ii) *processes*, that is, activity networks defined to facilitate the achievement of the goals, (iii) *policies* constraining the performance of R's processes, and (iv) *skills* that are required to achieve the goals of R. An element of a role on which considerable emphasis is placed is that of *authority*. Depending on the type of authority agent Ag_1 has over agent Ag_2, due to the roles Ag_1 and Ag_2 occupy:

- Ag_1 may assign a set of goals to Ag_2, thus making Ag_2 obliged to achieve these goals.
- Ag_1 may assign a set of new roles to Ag_2.
- Ag_1 may *empower* Ag_2 to perform a set of actions. Empowerment — a key feature of this approach on enterprise modelling — may be necessary in order to achieve the set of goals associated to the roles one occupies; for instance, an agent may manipulate/consume a set of resources only if it is empowered to access these resources.

The constraints on the members of an organisation ('rules of the game'), such as the policies of each role, are expressed with the use of a dialect of the Situation Calculus [36]. Moreover, a logic programming implementation of the Situation Calculus formalisation has been developed supporting the following computational tasks:

- Planning; what sequence of actions must be performed to achieve a particular goal? In order to minimise the cycle time for a product, for example, we may compute a plan maximising the concurrency of actions.
- Narrative assimilation (temporal projection); what are the effects of the occurrence of a set of actions? For instance, how are the obligations and powers of a set of agents affected by the actions performed so far? Or what will happen if we move one task ahead of schedule and another behind schedule?

Commentary

Fox and colleagues, in their model of computational enterprises/organisations, identify and represent several normative relations of the member agents, such as obligation, authority and empowerment. This is in contrast to the artificial social systems approach in which only permissions are considered. This difference is not surprising as the modelling of computational organisations, unlike the applications studied in the context of the artificial social systems approach, requires the representation of institutional facts and communicative actions. The representation of 'empowerment', however, is not very clear. In some cases 'empowerment' seems to coincide with the concept of institutional power presented in the introduction, in the sense that an empowered agent may create a set of institutional facts, such as the establishment of a contract with a third party, while in other cases 'empowerment' seems to coincide with permission, in the sense that performing a permitted action will not be penalised.

Like the artificial social systems approach, Fox and colleagues do not consider mechanisms for dealing with non-compliance to the specifications (for instance, violating obligations).

The semantics of the organisational constraints, expressed by means of a version of the Situation Calculus, are formal, declarative and verifiable. Finally, there is no discussion on the complexity of the computational tasks supported by the logic programming implementation of the Situation Calculus formalisation — for instance, can verification of compliance be performed in computational systems composed of hundreds of agents?

4 Commitment Protocols

The concept of *commitment* has been frequently used in the literature for specifying protocols for OAS. The term 'commitment' (or 'social commitment') has been used in the MAS field to refer to a wide variety of different concepts [42]. A common usage of this term is in the sense of an obligation directed from one agent to another. In this paper we review the work of Singh and colleagues [7,9,11,12,51,52,56,59] on the so-called *commitment protocols*.

A commitment is viewed as a four-place relation involving a proposition (p) and three agents (x, y and G). More precisely, $c = \mathbf{C}(x, y, G, p)$ denotes a commitment from x toward y in the context of G and for the proposition p. In this case x is the debtor, y is the creditor, G is the context group and p is the discharge condition of commitment c. The context group is viewed as the adjudicating authority that resolves disputes between the debtor and the creditor. *Explicit commitments*, unlike *implicit* ones, are explicitly represented by one or more agents. Implicit commitments are assumed to be common knowledge, or mutually believed in a multi-agent system. Commitments can also be divided into *base-level commitments* and *meta-commitments.* A commitment c is a meta-commitment if the proposition p refers to another commitment; c is a base-level commitment if p does not refer to other commitments. Regarding agent societies, Singh states that "[m]etacommitments create a society. The metacommitments are the norms of the society" [51, p.106].

Singh [51] has discussed the relationship between his interpretation of commitment and related concepts from deontic logic. For instance:

- A *pledge* is an explicit commitment.
- Singh claims that the classical approaches in deontic logic that express *ought* are unsuitable because 'ought' is inherently contextual. Therefore, he defines 'ought' to be relativised to the context group:
  $\mathbf{Ought}(x, G, p) \equiv (\exists G : x \in G \;\&\; c = \mathbf{C}(x, G, G, p))$. Notice that 'ought' is directed to the context group, not to a particular agent.
- *Obligation* has two main readings: one close to pledge and one close to 'ought'.
- A *convention* or a *custom* is considered to be an implicit meta-commitment relativised to a debtor and a context group.
- A *power* refers to the ability of an agent to force (if it desires) the alteration of a legal relation in which the another agent participates. Power is expressed as follows: $\mathbf{Power}(x, y, r) \equiv \mathbf{C}(G, x, G, \mathit{request}(x, G, r) \Rightarrow \mathit{perform}(G, r))$ where r is an operation on commitments whose creditor or debtor is an agent y and the context group is G. '$\Rightarrow$' represents strict implication.

Commitment protocols have been formalised in various ways, giving different types of semantics to the concept of commitment. Here we will briefly present three such formalisations.

Yolum and Singh [58,59] have specified commitment protocols with the use of a subset of Shanahan's 'full version' of Event Calculus (EC) [44]. All the 'operations' on commitments — *creating*, *discharging*, *cancelling*, *releasing*, *delegating*, *assigning* commitments — are formalised with the use of EC axioms. Consider, for example, the 'release' operation: $\mathit{Release}(E(y), c)$ releases the debtor x from the commitment $c = \mathbf{C}(x, y, p)$ (here, c is a 'base-level' commitment — in the interest of clarity we omit the context group G from the representation of c). The creditor y performs an action denoted by $E(y)$ that 'terminates' the debtor's commitment to bring about p [58]:

$$\begin{aligned}&\mathit{Release}(E(y), \mathbf{C}(x, y, p)) \equiv \\ &\qquad \mathit{Happens}(E(y), t) \wedge \mathit{Terminates}(E(y), \mathbf{C}(x, y, p), t)\end{aligned} \tag{1}$$

Yolum and Singh have employed an abductive EC planner [45] in order to compute planning queries regarding the EC specifications of commitment protocols.

Given an initial state of a commitment protocol, a final state and the specification of the protocol, the planner will compute the protocol runs (if any), that is, the sequences of actions, that will lead from the initial state to the final one. Consequently, agents may use the EC planner to compute protocol runs that lead to a 'desirable' outcome. In the case where agents are not capable of processing a commitment protocol specification, however, a subset of a commitment protocol specification may be compiled to a finite state machine where no commitments are explicitly mentioned [57].

Singh and colleagues have also used the $C+$ action language [21] — a formalism with *explicit* transition systems semantics — to express commitment protocols (see [7, 11, 12] for a few recent papers). Like EC, $C+$ has been employed to express all operations on commitments as well as the effects of the agents' actions. The Causal Calculator [21], a software implementation using satisfiability solvers to compute planning and narrative assimilation queries regarding $C+$ formalisations, allows for the execution of the commitment protocol specifications.

In a line of work primarily aimed at protocol composition [9,10], commitment protocols have been formalised with the use of the Web Ontology Language (OWL) [1] and the Semantic Web Rule Language (SWRL) [25]. SWRL has been used for formalising protocol rules that include instances of OWL-P, an OWL ontology for protocols expressing, among other things, the concept of commitment. Protocol composition is beyond the scope of this paper; therefore, we do not expand our presentation of this line of research.

Commentary

The notion of commitment is very important when specifying OAS, that is, systems in which the members may have competitive individual goals and there is no guarantee about their behaviour. It is difficult to see, however, how an interaction protocol for an OAS (such as, for example, a protocol for e-commerce, negotiation, argumentation or voting) can be specified simply in terms of commitments in this sense. Other normative relations are at least as important when specifying a protocol and the meaning of protocol actions. For example, when a contract is established between two parties by means of accepting an offer, a party may be interested in finding out the circumstances in which it has the 'legal capacity' or institutional power to initiate proceedings against the other party. In an auction house it is important to know the circumstances in which an agent has the institutional power to place a bid and, consequently, change the current bidding price. Similarly there are other examples in which more normative relations than simply commitments need to be considered.

Like the other two reviewed approaches (artificial social systems and enterprise modelling), the formalisations of commitment protocols do not cater for mechanisms dealing with non-compliance with the system specifications (commitments, in this case).

The semantics of the aforementioned formalisations of commitment protocols are formal and declarative. Moreover, no assumptions are made about the internal architectures of the agents, allowing for the verification of compliance in OAS.

A comparison of the Event Calculus (EC) and the C+ language, that were used for expressing commitment protocols, and a comparison of EC and the Situation Calculus, employed by Fox and colleagues for enterprise modelling, may be found in [35], for example. (See [2] for a comparison of EC and C+ with respect to specifying OAS.) To the best of our knowledge, the relationship between the aforementioned action languages, and SWRL and OWL-P, that have also been employed for specifying commitment protocols, has not been discussed.

The abductive EC planner and the Causal Calculator can be used, in principle, for the provision of both design-time services — for instance, proving protocol properties — and run-time services — for example, calculating the commitments current at each time for the benefit of the agents or their designers. These implementations, however, can become inefficient when considering large OAS. Moreover, it is assumed that the domain of each variable is finite and known from the outset (for instance, we need to know all possible propositions that an argumentation protocol participant may claim). Consequently, due to these issues, the use of these implementations for the provision of run-time services may be limited.

5 Electronic Institutions

Esteva and colleagues [13, 14, 15, 20] have developed a systematic approach to the design and development of multi-agent systems which incorporate aspects of conventional behaviour and organisational structures in a formal specification of structured interactions. Such a specification of a multi-agent system is called an Electronic Institution (EI).

The core notions defining an EI are [14]:

- *Roles.* A named role defines dialogic capabilities, that is, an agent must occupy a role in order to make a communicative action. Roles also determine the organizational structure, through the definition of a role hierarchy to indicate, for example, subsumption and exclusivity between roles.
- *Dialogic framework.* The dialogic framework of an EI defines the acceptable message format of communicative actions, by specifying a named list of illocutions (a name for each type of communicative action), the content language (the language of propositions embedded in the illocution), and the ontology (a vocabulary of terms). In the specification of an EI, the dialogic framework also includes the roles and the organizational structure. The dialogic framework determines the space of illocutions that can be exchanged, the pertinent exchanges are defined by scenes.
- *Scenes.* A scene is directed graph specifying a structured exchange of messages between roles. Essentially a scene is a finite state diagram defining a communication protocol, with each state transition labelled by an illocution scheme, itself composed from elements of the dialogic framework (an illocution, two agent/role pairs (the sender/receiver), an expression of the content language, and a time stamp).

- *Performative structure.* A directed graph of scenes defines the performative structure(s) of the EI. The specification of a performative structure describes how agents can move between scenes by satisfying certain pre-conditions, specifically related to roles they occupy and communicative actions they make.
- *Normative rules.* Normative rules in an EI are expressions which impose obligations or prohibitions on communicative actions within scenes, which constrain or regulate the behaviour of agents which occupy the roles that are the subject of the rule.

In [20] two ‘sorts’ of normative rule are given. One sort concerns the so-called ‘strict’ obligations to perform communicative actions according to the scene specification. A second sort are (what is called) ‘less strict’ normative rules which express other permissions, obligations and prohibitions with a conditional or temporal constraint.

A ‘strict’ normative rule is of the form:

$$\left(\bigwedge_{i=1}^{m}\{\mathit{illocutions}\}_i \wedge \bigwedge_{j=1}^{n}\{\mathit{conditions}\}_j\right) \rightarrow \bigwedge_{k=1}^{m'}\{\mathit{illocutions}\}_k \wedge \bigwedge_{l=1}^{n'}\{\mathit{conditions}\}_l$$

with the intuitive meaning that if the illocutions $i, 1 \leq i \leq m$ have been communicated (‘uttered’) in the appropriate scene states, and the expressions $j, 1 \leq j \leq n$ are satisfied, then the illocutions $k, 1 \leq k \leq m'$ satisfying the expressions $l, 1 \leq l \leq n'$ ‘must be’ communicated in the appropriate scene states. If the right-hand side is $\perp$ instead of a conjunction of illocutions and conditions, then uttering the illocutions i is said to be a violation.

Note that no deontic operators are used in a ‘strict’ normative rule. Such operators are employed to express the ‘less strict’ normative rules [20]:

OBLIGED *illocution*
BEFORE *illocution*
IF *condition*

PERMITTED *illocution*
BETWEEN (*illocution, illocution*)

Here, the first expression contains a conditional obligation to perform a certain action (utter an illocution) prior to a certain other action, subject to certain other conditions (which might include performance of past actions). For example, in an auction scene, a buyer might be obliged to pay for an item before the auctioneer closes the auction, if the buyer performed the action (uttered the illocution) that was the winning bid. The second expression denotes a permission in an interval of time: for instance, in a voting scene, a voter might be permitted to vote after the chair has called for votes, and before the chair closes the ballot.

Two types of mechanism have been devised in order to deal with the possibility of non-compliance with the normative rules. In [37] devices called ‘interagents’ were employed to *enforce* the rules of an auction house to the buyer and seller agents. In this setting, forbidden actions were physically blocked by interagents. In [20] sanctions were applied in the case of rule violation; failure to comply with an obligation, for example, could result in a monetary penalty.

Software tools for computational support of the EI specification languages have been developed. Islander [13], for instance, is an integrated development environment for specifying EIs. It offers a graphical user interface which allows the user to edit performative structures, scenes and illocutions. A verification module is implemented which, it is claimed, checks for 'integrity' (whether all elements of an EI are defined), protocol correctness (standard properties, such as liveness, reachability and termination), and 'norm consistency'. In [13] verification of 'norm consistency' is performed only over the so-called 'strict' normative rules. Another tool for computational support of the EI specification languages is presented in [20]; this is a rule-based system for executing a set of normative rules, including the 'less strict' normative rules, with the aim of providing run-time services, such as the computation of the permissions and obligations of the agents current at each state.

Commentary

Electronic Institutions (EI) are, in some sense, a result of an almost ethnographic study of the fishmarket auction extended to other (human) institutions. The five core notions of EI can be used to give order and structure to MAS intended to support similar types of activity (e-commerce, negotiation, and so on). As such, it is a specification formalism whose analytic and formal basis lends itself well to computational support.

We note though that the normative rules concern only the permissions and obligations of the agents — this is similar to the work on commitment protocols where the focus was on directed obligations. It has been argued that other normative relations are at least as important as those when specifying the meaning of protocol actions.

The general strategy of designing mechanisms to force compliance and eliminate non-permitted behaviour — *regimentation* [26] — is rarely desirable (it results in a rigid system that may discourage agents from entering it), and not always practical (violations may still occur due to, say, a faulty regimentation device). For all of these reasons, allowing for sanctioning mechanisms may be a better option than relying exclusively on regimentation devices (interagents, in this case).

The semantics of the EI specification languages are formal and declarative. Moreover, verification of compliance has been performed in settings where there is no access to the internal architectures of the agents. However, the representation of the agents' actions and their effects is not as expressive as those of enterprise modelling, where the Situation Calculus was used, and commitment protocols, where the Event Calculus and the $C+$ language were used.

To the best of our knowledge, a precise equivalence between (fragments of) the formalism used for verification of 'norm consistency' with that used for supporting run-time activities has not been established. Consequently, it is not possible to, say, verify a system specification for 'norm consistency' and then translate that specification to an equivalent for the provision of run-time services.

The verification module for 'norm consistency' operates under the assumption that the domain of each variable is finite and known from the outset, while the complexity is exponential to the number of variables and their possible values. Consequently, like the Event Calculus planner and the Causal Calculator that were employed in the context of commitment protocols, it may not be possible to use this module in large systems.

6 Discussion

We reviewed four approaches for the specification of multi-agent systems exhibiting the features open systems; no access to the code of the agents, no guarantee of benevolent behaviour, and no possibility of predicting the agent interactions. Most reviewed approaches focused on the representation of the permissions and obligations (or commitments) of the agents. These normative relations may be adequate for expressing the system rules when a system can be mainly described in terms of brute facts. When this is not the case, however, it is necessary to explicitly represent the concept of institutional (or institutionalised) power in order to express the circumstances in which an agent may create a set of institutional facts. Jones and Sergot state the following with respect to this issue:

> “[W]e need to make it explicit at the outset that ‘empowering’ is not an *exclusively* legal phenomenon, but is a standard feature of any norm-governed organisation where selected agents are assigned specific roles (in which they are empowered to conduct the business of that organisation). Thus although it is perhaps legal examples, [..], which come most immediately to mind, it is clear that illustrations of essentially the same sort of phenomenon also occur frequently in other than legal contexts: the Chief Librarian is empowered to waive lending restrictions; the Head of Department alone is empowered to assign duties to members of the department; [..]. An adequate account of the mechanisms by means of which an organisation conducts its affairs will have to incorporate, one way or another, the phenomenon of *institutionalised power*, as we shall here choose to call it.” [27, p.430]

The work on enterprise modelling included a representation of institutional power, although the distinction between institutional power and permission was not always clear. The other reviewed approaches did not explicitly represent institutional power.

Although all reviewed approaches specified rules regulating the behaviour of agents (in terms of permissions, obligations, commitments), only the work on Electronic Institutions developed mechanisms for dealing with rule violation.

All reviewed approaches specified rules with a formal, declarative and verifiable semantics. They all employed different formalisms to express the rules, though, thus the expressiveness of each formalisation varied. Details about the relationship between some of the employed formalisms for expressing the rules of a system can be found in the papers cited in earlier sections.

The support for computational tasks regarding the system specification, offering design-time and run-time services, is crucial for the realisation of open systems. All reviewed approaches apart from the work on artificial social systems included software implementations executing the system specifications with the aims of proving properties of the specifications, checking whether agents comply with the rules, or informing the decision-making of the agents by computing plans that lead to a desired state, or by computing their current permissions, obligations, and so on. There was no demonstration of an implementation framework, however, providing all of the above services at run-time in a large multi-agent system.

References

1. OWL Web Ontology Language: Overview. W3C Recommendation (2004), `http://www.w3.org/TR/owl-features/`
2. Artikis, A., Sergot, M., Pitt, J.: Specifying norm-governed computational societies. ACM Transactions on Computational Logic 10(1) (2009) (retrieved July 6, 2008), `http://www.acm.org/pubs/tocl/accepted/304artikis.pdf`
3. Bandara, A.: A Formal Approach to Analysis and Refinement of Policies. Ph.D thesis. Imperial College London (2005)
4. Boella, G., van der Torre, L.: Security policies for sharing knowledge in virtual communities. IEEE Transactions on Systems, Man and Cybernetics 36(3), 439–450 (2006)
5. Boella, G., van der Torre, L.W.N.: The ontological properties of social roles in multi-agent systems: Definitional dependence, powers and roles playing roles. Artificial Intelligence and Law 15(3), 201–221 (2007)
6. Bradshaw, J., Uszok, A., Jeffers, R., Suri, N., Hayes, P., Burstein, M., Acquisti, A., Benyo, B., Breedy, M., Carvalho, M., Diller, D., Johnson, M., Kulkarni, S., Lott, J., Sierhuis, M., Van Hoof, R.: Representation and reasoning about DAML-based policy and domain services in KAoS. In: Rosenschein, J., Sandholm, T., Wooldridge, M., Yoko, M. (eds.) Proceedings of Conference on Autonomous Agents and Multi Agent Systems (AAMAS), pp. 835–842. ACM Press, New York (2003)
7. Chopra, A., Singh, M.: Contextualizing commitment protocols. In: Proceedings of Conference on Autonous Agents and Multi-Agent Systems (AAMAS), pp. 1345–1352. ACM, New York (2006)
8. Dastani, M., Dignum, V., Dignum, F.: Role-assignment in open agent societies. In: Proceedings of Conference on Autonomous Agents and Multi-Agent Systems, pp. 489–496. ACM, New York (2003)
9. Desai, N., Mallya, A., Chopra, A., Singh, M.: Interaction protocols as design abstractions for business processes. IEEE Transactions on Software Engineering (TSE) 31(12), 1015–1027 (2005)
10. Desai, N., Mallya, A., Chopra, K., Singh, M.: OWL-P: A methodology for business process modeling and enactment. In: Kolp, M., Bresciani, P., Henderson-Sellers, B., Winikoff, M. (eds.) AOIS 2005. LNCS, vol. 3529, pp. 79–94. Springer, Heidelberg (2006)
11. Desai, N., Singh, M.: A modular action description language for protocol composition. In: Proceedings of Conference on Artificial Intelligence, AAAI (2007)
12. Desai, N., Singh, M.: Checking correctness of business contracts via commitments. In: Proccedings of International Conference on Autonomous Agents and Multi-Agent Systems (2008)
13. Esteva, M., de la Cruz, D., Sierra, C.: Islander: an electronic institutions editor. In: Castelfranchi, C., Johnson, L. (eds.) Proceedings of the First International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), pp. 1045–1052. ACM Press, New York (2002)

14. Esteva, M., Padget, J., Sierra, C.: Formalizing a language for institutions and norms. In: Meyer, J.-J.C., Tambe, M. (eds.) ATAL 2001. LNCS, vol. 2333, pp. 348–366. Springer, Heidelberg (2002)
15. Esteva, M., Rodríguez-Aguilar, J., Sierra, C., Vasconcelos, W.: Verifying norm consistency in electronic institutions. In: Proceedings of the AAAI 2004 Workshop on Agent Organizations: Theory and Practice, pp. 8–14 (2004)
16. Fitoussi, D., Tennenholtz, M.: Minimal social laws. In: Proceedings of Conference on Artificial Intelligence (AAAI) and Innovative Applications of Artificial Intelligence (IAAI), pp. 26–31. AAAI Press/The MIT Press, Menlo Park (1998)
17. Fitoussi, D., Tennenholtz, M.: Choosing social laws for multi-agent systems: minimality and simplicity. Artificial Intelligence 119(1-2), 61–101 (2000)
18. Fornara, N., Colombetti, M.: Formal specification of artificial institutions using the event calculus. In: Multi-Agent Systems: Semantics and Dynamics of Organizational Models. IGI Global (2008) (to appear)
19. Fox, M., Barbuceanu, M., Grüninger, M., Lin, J.: An organizational ontology for enterprise modeling. In: Prietula, M., Carley, K., Gasser, L. (eds.) Simulating Organizations: Computational Models for Institutions and Groups, pp. 131–152. AAAI Press/The MIT Press (1998)
20. García-Camino, A., Noriega, P., Rodríguez-Aguilar, J.: Implementing norms in electronic institutions. In: Proceedings of the Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), pp. 667–673. ACM Press, New York (2005)
21. Giunchiglia, E., Lee, J., Lifschitz, V., McCain, N., Turner, H.: Nonmonotonic causal theories. Artificial Intelligence 153(1–2), 49–104 (2004)
22. Grossi, D., Dignum, F., Meyer, J.-J.C.: A formal road from institutional norms to organizational structures. In: Durfee, E., Yokoo, M., Huhns, M., Shehory, O. (eds.) Proceedings of Conference on Autonomous Agents and Multi-Agent Systems, pp. 616–623 (2007)
23. Grüninger, M., Fox, M.: The role of competency questions in enterprise engineering. In: Proceedings of the IFIP WG5.7 Workshop on Benchmarking-Theory and Practice (1994)
24. Hewitt, C.: Open information systems semantics for distributed artificial intelligence. Artificial Intelligence 47, 79–106 (1991)
25. Horrrocks, I., Patel-Schneider, P., Boley, H., Tabet, S., Grosof, B., Dean, M.: SWRL: A semantic web rule language combining OWL and ruleML. W3C Submission (2004), `http://www.w3.org/Submission/2004/SUBM-SWRL-20040521/`
26. Jones, A., Sergot, M.: On the characterisation of law and computer systems: the normative systems perspective. In: Deontic Logic in Computer Science: Normative System Specification, pp. 275–307. J. Wiley and Sons, Chichester (1993)
27. Jones, A., Sergot, M.: A formal characterisation of institutionalised power. Journal of the IGPL 4(3), 429–445 (1996)
28. Kagal, L., Finin, T.: Modeling communicative behavior using permissions and obligations. Journal of Autonomous Agents and Multi-Agent Systems 14(2), 187–206 (2006)
29. Lomuscio, A., Sergot, M.: Deontic interpreted systems. Studia Logica 75(1), 63–92 (2003)
30. Makinson, D.: On the formal representation of rights relations. Journal of Philosophical Logic 15, 403–425 (1986)
31. McGinnis, J., Miller, T.: Amongst first-class protocols. In: Artikis, A., O'Hare, G.M.P., Stathis, K., Vouros, G. (eds.) ESAW 2007. LNCS, vol. 4995. Springer, Heidelberg (2008)

32. Minsky, N., Ungureanu, V.: Law-governed interaction: a coordination and control mechanism for heterogeneous distributed systems. ACM Transactions on Software Engineering and Methodology (TOSEM) 9(3), 273–305 (2000)
33. Moses, Y., Tennenholtz, M.: On computational aspects of artificial social systems. In: Proceedings of Workshop on Distributed Artificial Intelligence (DAI), pp. 267–284 (1992)
34. Moses, Y., Tennenholtz, M.: Artificial social systems. Computers and Artificial Intelligence 14(6), 533–562 (1995)
35. Mueller, E.: Commonsense Reasoning. Morgan Kaufmann, San Francisco (2006)
36. Pinto, J., Reiter, R.: Temporal reasoning in logic programming: a case for the situation calculus. In: Warren, D. (ed.) Proceedings of Conference on Logic Programming, pp. 203–221. MIT Press, Cambridge (1993)
37. Rodriguez-Aguilar, J., Martin, F., Noriega, P., Garcia, P., Sierra, C.: Towards a test-bed for trading agents in electronic auction markets. AI Communications 11(1), 5–19 (1998)
38. Rosenschein, J., Zlotkin, G.: Rules of Encounter: Designing Conventions for Automated Negotiation among Computers. MIT Press, Cambridge (1994)
39. Searle, J.: What is a speech act? In A. Martinich, editor. In: Martinich, A. (ed.) Philosophy of Language, 3rd edn., pp. 130–140. Oxford University Press, Oxford (1996)
40. Serban, C., Chen, Y., Zhang, W., Minsky, M.N.: The concept of decentralised and secure electronic marketplace. Journal of Electronic Commerce Research (2008)
41. Sergot, M.: The representation of law in computer programs: a survey and comparison. In: Bench-Capon, T. (ed.) Knowledge Based Systems and Legal Applications. Academic Press, London (1990)
42. Sergot, M.: A computational theory of normative positions. ACM Transactions on Computational Logic 2(4), 522–581 (2001)
43. Sergot, M., Craven, R.: The deontic component of action language $nC+$. In: Goble, L., Meyer, J.-J.C. (eds.) DEON 2006. LNCS (LNAI), vol. 4048, pp. 222–237. Springer, Heidelberg (2006)
44. Shanahan, M.: The event calculus explained. In: Veloso, M.M., Wooldridge, M.J. (eds.) Artificial Intelligence Today. LNCS (LNAI), vol. 1600, pp. 409–430. Springer, Heidelberg (1999)
45. Shanahan, M.: An abductive event calculus planner. Journal of Logic Programming 44, 207–239 (2000)
46. Shoham, Y., Tennenholtz, M.: Emergent conventions in multi-agent systems. In: Proceedings of Conference on Principles of Knowledge Representation and Reasoning (KR), pp. 225–231 (1992)
47. Shoham, Y., Tennenholtz, M.: On the synthesis of useful social laws for artificial agent societies. In: Swartout, W. (ed.) Proceedings of Conference on Artificial Intelligence (AAAI), pp. 276–281. The AAAI Press/ The MIT Press (1992)
48. Shoham, Y., Tennenholtz, M.: On social laws for artificial agent societies: off-line design. Artificial Intelligence 73(1-2), 231–252 (1995)
49. Shoham, Y., Tennenholtz, M.: On the emergence of social conventions: modeling, analysis and simulations. Artificial Intelligence 94(1-2), 139–166 (1997)
50. Singh, M.: Agent communication languages: rethinking the principles. IEEE Computer 31(12), 40–47 (1998)
51. Singh, M.: An ontology for commitments in multiagent systems: towards a unification of normative concepts. Artificial Intelligence and Law 7(1), 97–113 (1999)

52. Singh, M.: A social semantics for agent communication languages. In: Dignum, F., Greaves, M. (eds.) Issues in Agent Communication. LNCS, vol. 1916, pp. 31–45. Springer, Heidelberg (2000)
53. Tennenholtz, M.: On computational social laws for dynamic non-homogeneous social structures. Journal of Experimental and Theoretical Artificial Intelligence 7, 379–390 (1995)
54. Uszok, A., Bradshaw, J., Lott, J., Breedy, M., Bunch, L., Feltovich, P., Johnson, M., Jung, H.: New developments in ontology-based policy management: Increasing the practicality and comprehensiveness of KAoS. In: Proceedings of Workshop on Policies for Distributed Systems and Networks, pp. 145–152. IEEE Computer Society, Los Alamitos (2008)
55. van Eijk, R., de Boer, F., van der Hoek, W., Meyer, J.-J.: Open multi-agent systems: agent communication and integration. In: Jennings, N., Lesperance, Y. (eds.) ATAL 1999. LNCS, vol. 1757, pp. 218–232. Springer, Heidelberg (2000)
56. Venkatraman, M., Singh, M.: Verifying compliance with commitment protocols. Journal of Autonomous Agents and Multi-Agent Systems 2(3), 217–236 (1999)
57. Yolum, P., Singh, M.: Commitment machines. In: Meyer, J.-J.C., Tambe, M. (eds.) ATAL 2001. LNCS (LNAI), vol. 2333, pp. 235–247. Springer, Heidelberg (2002)
58. Yolum, P., Singh, M.: Flexible protocol specification and execution: applying event calculus planning using commitments. In: Castelfranchi, C., Johnson, L. (eds.) Proceedings of Conference on Autonomous Agents and Multiagent Systems (AAMAS), pp. 527–535. ACM Press, New York (2002)
59. Yolum, P., Singh, M.: Reasoning about commitments in the event calculus: An approach for specifying and executing protocols. Annals of Mathematics and Artificial Intelligence 42(1-3), 227–253 (2004)
60. Zambonelli, F., Jennings, N., Wooldridge, M.: Organisational rules as an abstraction for the analysis and design of multi-agent systems. International Journal of Software Engineering and Knowledge Engineering 11(3), 303–328 (2001)