Dynamic Discovery and Maintenance of Role-based Performance Standards

Ramón Hermoso[1] and Henrique Lopes Cardoso[2]

[1] CETINIA, University Rey Juan Carlos
Tulipán s/n, 28933, Madrid, Spain
ramon.hermoso@urjc.es

[2] LIACC / DEI, Faculdade de Engenharia, Universidade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
hlc@fe.up.pt

Abstract. Standards have been deeply studied in economics in order to assure a certain quality of service in bilateral contracts. More specifically, in multi-agent systems performance standards may be used in order to articulate contracts among partners in environments dealing with uncertainty. However, little effort has been made on defining how standards are created and, what is more important, how to ensure standards compliance over time. In this work we put forward a mechanism that, on one hand, creates standards from roles denoting task specialization skills and, on the other hand, tries to maintain role performance standards by applying incentives and/or punishments to agents identified as being able to play those roles.

1 Introduction

A number of research proposals have been made recently concerning the development of infrastructures for supporting interaction in open multi-agent systems. In such systems agents enter and leave the interaction environment, and behave in an autonomous and not necessarily cooperative manner, exhibiting self-interested behaviours. Even when agents establish commitments among them, the dynamic nature of the environment may jeopardize such commitments if agents are not socially concerned enough, valuing more their private goals when evaluating the new circumstances.

Moreover, in open systems one cannot assume that agents will behave consistently along time. This may happen either because of agent's ability or benevolence. In some cases, an agent may not be capable of maintaining a certain behaviour standard throughout its lifetime. In other cases, the agent may intentionally deviate from its previous performance. It is therefore important, when considering open environments, to take into account also the evolution of an agent's internal skills or motivations, besides the dynamics of the interaction environment as a whole.

Taking an organizational approach, and looking at the society from a role-specialization perspective, Hermoso has proposed role evolution [6] as a guideline to develop a coordination mechanism that enables agents to select a partner to delegate a specific task to. The proposed approach is to look at the agent society and identify "run-time roles" that cluster agents with similar skills for a (set of) tasks. From this perspective, the mechanism allows one to identify the role that labels the agents most suitable to perform a specific task.

Looking at this role taxonomy as an artificial organization, in this paper we address the problem of organizational maintenance. Given the evolving nature of agents, as pointed above, a problem faced by the organization in which agents have been (artificially) embedded is that of timeliness: are the agents within a role still performing as well as they did at the time of their assessment? Two different actions can be taken when agents start under-performing. One is to reorganize so that the role taxonomy becomes accurate again. But assuming that this reorganization may be costly, another approach is to influence the agents' reasoning by making use of incentives or punishments, as an attempt to keep them on track.

The rest of the paper is structured as follows. Section 2 summarises the preliminaries this paper is based on. We present the standardisation process from roles in Section 3. Then we put forward a model to establish and adjust incentives in order to maintain standards over time in Section 4. We relate our work to others' in Section 5. Finally, we sum up the paper and sketch the future work in Section 6.

2 Background

The rationale behind creating and maintaining performance standards relies on the concept of role proposed by Hermoso et al. [6]. In this work the authors claim that in a society of agents social relationships may evolve, so roles – defining the positions of agents in terms of skills and importance as seen from others – should also do. They use the role as a piece of reasoning when looking for an interaction partner. The main contribution of the work is that a society of agents may be covered by an overlay role taxonomy formed by extracting capacities and trust relationships among agents over time.

Previous work on the field [6] shows a proposal for a coordination mechanism for OMAS which presents a societal structure where agents try to improve their individual utilities. In order to do that, agents may interact with others in the sense that they will delegate certain tasks other agents may carry out. Such systems are called *Task-oriented Multiagent Systems* (T-MAS). We assume that agents in these systems accumulate their experiences of past interactions and implement some kind of trust model that allows them to establish which agents are more appropriate as possible interaction partners in the future. Based on these trust models, that approach evolves role taxonomies and assigns agents to roles. Agents can then request this information and use it in their decision-making processes. In particular, they can use the assignments of roles to agents in order to improve their trust models, that is, in order to evaluate the expected behaviour or outcome when delegating tasks to or using services from others.

The mechanism has been exhaustively tested in different conditions with open MAS [6], with heterogeneous and dynamic populations, showing a significantly good adaptation in order to provide a useful role taxonomy to agents.

The concept of role is often considered from a macro perspective describing the objectives, goals and also the constraints applied to players in an organizational context. However, we consider roles from another perspective: as representing expectations of behaviours. In particular, we consider roles from a micro perspective (that is, from the perspective of the agents), where the role other agents are playing in the system provides information about their expected capacities regarding certain interactions (e.g., the provisioning of certain services or tasks). The proposed mechanism evolves role taxonomies over time and adapts itself to changes in the system, which is useful when dealing with the open and dynamic nature of a MAS. We assume that agents participating in the system are rational, that is, they try to maximise their utility in every action they plan to perform. The main task of the mechanism will be twofold: i) capture similar behaviour among participants that play a role; and ii) manage the role taxonomy that structures different positions of agents in the system. The mechanism uses a K-Means clustering algorithm to identify patterns of behaviour, so distinguishing those agents outperforming others and, consequently, being more trusted by the participants. The input for the clustering is the trust network generated from the opinions agents have on the trustworthiness of other participants in different interactions they have had over time.

Next we review some basic notation of previous work in order to correctly understand the remainder of the paper.

Task-oriented Multi-Agent Systems A Task-oriented Multi-Agent System (T-MAS) is a multi-agent system in which participants have to perform a set of tasks. Among this set of tasks, agents will decide based on their capabilities which ones can be performed by themselves and which others are better to be delegated to other agents. A T-MAS is defined as follows:

Definition 1 *A T-MAS is a tuple $TM = \langle \mathcal{A}g, \mathcal{T}, \mathcal{U}, OS \rangle$ where $\mathcal{A}g$ stands for the set of rational agents participating in the system, $\mathcal{T}$ is the set of tasks that can be performed in the system, $\mathcal{U}$ represents a function that measures the system's global utility at time k and $OS \subseteq \overline{OS}$ stands for the T-MAS' organizational structure. $\overline{OS}$ denotes all possible subsets of organisational structures.*

On the organisational structure, we are particularly interested in role taxonomies to articulate coordination in the delegation process. We define a role as follows:

Definition 2 *Let TM be a T-MAS, and let $\mathcal{R}$ be a set of unique role labels. A role in TM is a pair $\langle r, \mathcal{E}_r \rangle$, in which r is the name of the role, and $\mathcal{E}_r = \{t_1, ..., t_n\}$ with $t_i \in \mathcal{T}$ is a finite set of tasks.*

The intended semantics of a role $\langle r, \mathcal{E}_r \rangle$ is that agents playing the role r are qualified "performers" of the tasks contained in $\mathcal{E}$, in the sense that they are "good" in performing any of those tasks. We assume that agents could perform any task in the T-MAS, but they may only be well qualified for some of them. For instance, in the world of conference program committees a role of *GA expert* is more specific than the role *reviewer* for reviewing papers on Genetic Algorithms. Based on the definition of role above, we define a role specialization taxonomy.

Definition 3 *Let $TM = \langle \mathcal{A}g, \mathcal{T}, \mathcal{U}, \{\Delta_{\mathcal{R}}\}\rangle$ be a T-MAS, and $\mathcal{R}$ a set of unique role labels, a specialization role taxonomy in TM is a structure $\Delta_{\mathcal{R}} = \langle \Pi, \rhd_r \rangle$, containing a set of roles Π (over $\mathcal{R}$ and $\mathcal{T}$), and a partial order relationship.*

Role specialisation taxonomies have the following two main properties: **i) Inception**: there exists a root role ($\langle r_{root}, \mathcal{E}_{root}\rangle$) that contains every achievable task ($\mathcal{T}$) in the T-MAS and that is not a specialisation of any other. This property reflects the assumption on the possibility of any agent to perform any type of task. Therefore, any agent in $\mathcal{A}g$ can play, at least, the role $\langle r_{root}, \mathcal{E}_{root}\rangle$ in TM, and **ii) Specialisation**: there exists a partial order relationship that defines the taxonomy ($\rhd_r$), based on the different expectations that agents have on those who play distinct roles, or in other words, the quality with which those other agents play different roles. That is, given two different roles $\langle r_1, \mathcal{E}_1\rangle$ and $\langle r_2, \mathcal{E}_2\rangle \in \Pi$, the relationship $\langle r_1, \mathcal{E}_1\rangle \rhd_r \langle r_2, \mathcal{E}_2\rangle$ holds iff. there exists a subset of tasks in $\mathcal{E}_1$ for which those agents that play the role $\langle r_2, \mathcal{E}_2\rangle$ are expected to perform better on average than the agents playing role r_1.

3 Creating standards from roles

Although in previous work [6] the focus was on providing a role specialization taxonomy in order to better estimate trust on others in a T-MAS, in this paper we try to go beyond by claiming that this notion of specialized role may also be used to allow establishing performance standards and so facilitating the agreement on contracts among agents.

We therefore aim at going from roles as expectations of behavior [12] to the explicit handling of such expectations as conventions and further as norms [2] that are committed to. Our approach is to assign a performance *standard* to every task specialized in a role, so allowing agents agents to formulate normative contracts and reason about them (commit, refuse, fulfill, ...).

3.1 Standard creation

Let $\mathcal{X}$ be the set of attributes that characterizes any task in $\mathcal{T}$. For instance, in an e-commerce domain $x \in \mathcal{X} = \{delivery\_time, quality\}$ would be a set of attributes that might characterize the task *supply a good*. Let $\bar{x}$ be a set of possible values for x, where $\bar{x}_i$ corresponds to a value for the attribute x_i. For example, $\bar{x} = \{5, B\}$ means that the value for the attribute *delivery_time* is 5 and the *quality* of the good is of type B.

Firstly we define the concept of standard:

Definition 4 *A* ***standard*** ς *is a tuple* $\langle r, t, x, \bar{x}\rangle$ *that establishes, for a role* r *and a task* t*, a certain expected level of quality for a set of attributes* x*.*

The level of quality of different attributes is meant to be a target ($\bar{x}_i$) that separates acceptable behavior of an agent performing a task t from less preferred behavior for the same task.

In the literature, the concept of standard has mainly been used in economics, especially in commerce finances and industry, where it defines what is expected (in terms of outcomes) from a commercial interaction or an industrial process, respectively. This notion of standard fosters interactions among different stakeholders in a system by allowing to enact contracts in terms of standard's attributes (and its corresponding values) they adhere to. Analogously as it occurs with roles, the inherent nature of open systems makes it complicated to define standards a priori for any kind of process. In the case of T-MAS, the potentially changing environment (regarding the system's population and agents' behaviors) makes necessary the use of evolution mechanisms in order to update organizational structures (in our case role taxonomies). Since standards are directly related with the notion of task and role, the evolving nature of the latter entails also evolution of standards over time.

In Section 2 we summarized a previous work on role evolution to cover this issue. In that approach roles are created with the aim of allowing a more accurate partner selection process in T-MAS, thus roles are considered as specializations of agents skilled to perform (with a sufficient quality) a certain set of tasks. The authors propose there to use subjective information based on agents' trust models to aggregate societal opinions about the goodness of every individual in the system on their performance carrying out different tasks.

In this paper we claim that once this role evolution mechanism is in place, roles may be used to create standards which, in turn, can be included in contracts that regulate interactions in the system. The underlying idea relies on the aggregation of values of different attributes that characterize (the performance of) a task within the role, to establish a standard for that role/task pair, as defined in Def. 4.

Let TM be a T-MAS with a set of roles Π and a role specialization taxonomy $\Delta_{\mathcal{R}} = \langle \Pi, \rhd_r\rangle$. We use a function $f : \Pi \times \mathcal{T} \rightarrow \mathcal{S}$ (*role to standard*) where $\mathcal{S}$ is the set of possible standards for the task in $\mathcal{T}$ specialized by the role in Π. Algorithm 1 shows the description of function f. Input parameters for this function are the role from which the standard is created and one of the tasks the role is specialized for. The first loop takes the attributes the task is characterized by (line 1). Then there is a double loop in which the algorithm aggregates the values of the attributes x_i ($\bar{x_i}$) gathered from the agents in the system regarding those agents assigned to role r when performing task t (lines 2, 3 and 4). Function $ag(r)$ returns the set of agents playing the role r. The *aggr* function might be implemented in several ways, e.g. by weighting differently the information coming from different agents. Function e (line 4) is defined as $e_{a_k} : \mathcal{A}g \times \mathcal{T} \times \mathcal{X} \rightarrow \bar{\mathcal{X}}$. Thus $e_{a_k}(a_1, buy\_cotton, delivery\_time) = 5.3$ means that agent a_k has experienced an average *delivery_time* of 5.3 days when buying cotton to agent a_1. We assume that there is a monitoring process periodically launched in charge of annotating actual outcomes for every attribute in the system. Thus function e retrieves information already gathered. For every different attribute the algorithm stores a standard value (line 7). This value might again be calculated in different ways, e.g. by using an arithmetic mean. Finally, a vector of standard values for that pair $\langle r, t \rangle$ is returned. For instance, in open electronic markets (e.g. eBay) roles might be created in order to place providers in different categories of provision, while standards would emerge in order to ensure better interaction processes, and so keep agents from undesirable behaviors, such as longer delivery times, price changes, decrease of quality, etc.

Algorithm 1 RoleToStandard process

Require: $\langle r, \mathcal{E}_r \rangle \in \Pi$ {the role to use}
Require: $t \in \mathcal{E}_r$
1: **for** $x_i^t \in \mathcal{X}^t$ **do**
2:     **for** $a_k \in \mathcal{A}g$ **do**
3:         **for** $a_j \in ag(r)$ **do**
4:             $values[i] \leftarrow aggr(values[i], e_{a_k}(a_j, t, x_i^t))$
5:         **end for**
6:     **end for**
7:     $stdValues[i] \leftarrow stdEval(values[i])$
8: **end for**
9: **return** $stdValues$

In the perspective of the role taxonomy, where roles specialize tasks, the semantics of a role entails a list of tasks the role is specialized for, but not addressing objectively measurable quantitative information. In the case of standards, these are specific values that can be used to specify the terms of a contract. We assume that any task in a T-MAS is characterized as a multi-attribute entity, which means that the quality or measurement of its performance will be assessed in terms of those attributes. For instance, a task *Supply cotton* might be characterized by different attributes such as *delivery time*, *origin* or *color*.

Although we define standards as a set of attribute values related with a certain task performance, standards can be used as a means to articulate contracts. Thus a contract could be seen as an agreement, between at least two parties, in which the counterparts know about the standards to be fulfilled. Formally:

Definition 5 *A contract is a tuple $\langle a_i, a_j, \varsigma_k \rangle$, in which $a_i, a_j \in \mathcal{A}g$ and $\varsigma_k = \langle r, t, x, \bar{x} \rangle \in \mathcal{S}$.*

Following the definition above, the meaning of a contract is as follows: agent a_i agrees with agent a_j that the latter shall fulfill a standard ς_k when executing a task t the standard ς_k refers to.

Algorithm 2 Standard matching process for requester agent a_k

Require: a_k {Requester agent}
Require: $t \in \mathcal{T}$ {The task agent a_k is interested in}
Require: $S \in \mathcal{S}$ {Denotes the set of standards on t in the repository}
1: **for** $s \in S$ **do**
2: $eval[s] \leftarrow m_{a_k}(s, t)$
3: **end for**
4: $eval \leftarrow sort(eval)$
5: **return** $eval$

3.2 Standards dynamics

Once the standards have been created for every role-task pair in the T-MAS it is necessary to explain how those standards become part of contracts between different stakeholders in the system. Let a_k be an agent willing to request a task (or a service) t. There exists a repository that collects standards (created from the current taxonomy) in the T-MAS. Agent a_k will look up which standard on t it is more interested in. Let us call this process *standard matching process.* This process is detailed in Algorithm 2. This algorithm attempts to rate the potential standards the agents might be adhered to with a numerical value. Function $m_{a_k} : \mathcal{S} \times \mathcal{T} \rightarrow \mathbb{R}$ is used to encapsulate the matching rate of the agent a_k's preferences to the current standards in the system (lines 1–3). That is, agent a_k has freedom of choice among the standards that the system provides. Typically rates will be in the range $[0, 1]$. The algorithm returns a sorted list of standards ordered by their ratings (lines 4–5).

However, a contract involves profit and risk sharing between the two parties: the requester agent and the provider [9]. Thus, once the agent has selected a standard that considers good enough for her expectations, then it needs to look for providers in the system that want to adhere to the proposed standard and so perform the task. This is designed by using a typical *contract-net protocol* we name *Call For Standard Acceptance* protocol (CFSA protocol). In this protocol, the initiator sends a call for standard (s) acceptance to every agent $a_i \in ag(r)$ such that s was created from r. The potential providers evaluate the proposal by using an instance of the function m. Nevertheless, providers use m to match their preferences with the standard proposed by the initiator trying to rate the convenience of an eventual acceptance. If the proposal evaluation is reasonably promising for the provider then the standard proposal is accepted, otherwise is refused. Among the agents that replied with an acceptance the initiator must choose one to finally formalize a contract and eventually perform the interaction. This selection process is out of the scope of this paper.

4 Maintaining performance standards through incentives

After we have explained the path from roles as expectations to committed contracts based on standards, we are now in a position to elaborate on enforcement schemes that enable us to maintain the stability of the role taxonomy, which is obtained as explained in Section 2. We will base our approach in the well known *principal-agent* model [7, 1] from economics, in which a principal (a service requester) requests an agent (the provider) to perform a specific task.

The outcome of the task execution affects the principal's utility, and therefore the latter will be interested in influencing the efforts that the agent puts when performing the task. Those efforts are expressed in terms of available actions, which have associated execution costs. In the so-called *hidden action* setting there is an assumption that the actual actions executed by the agent are unobservable to the principal. Instead, only some performance measures of such actions are observed. The actions determine, usually stochastically, the obtained performance. Performance is therefore a random variable whose probability distribution depends on the actions taken by the agent. This stochastic nature captures the fact that there are externalities in the environment that the agent does not control. The principal will therefore want to establish an incentive schedule in order to encourage the agent to choose the actions better leading to an intended performance standard.

4.1 Targeting standards

As described in Section 3, standards are generated through the use of an averaging function applied to task execution outcomes of a group of provider agents that have been clustered within a specific role. Since, according to our model, standards allow requesters to identify expected values for the outcomes of tasks when executed by a specific provider, we consider a standard as a target that agents should meet. Any deviation from the standard is considered as a suboptimal outcome. Figure 1 illustrates this idea, where ς represents the target standard that the requester expects, and each concentric circle labelled with a δ_i denotes equidistant performances to the target. These concentric lines highlight the fact that we shall consider deviations in any direction (left or right, upwards or downwards) to be equally harmful in terms of expected values. The arrow pointing towards the centre discloses the aims of our incentive-based approach, with which we will try to encourage providers to better target the standard.

In our model, we will assume that each provider has a set of actions at his disposal, each with a cost and a probability function for obtaining different performance outcomes. As follows from Figure 1, an outcome is seen as a *distance* to the intended standard values. This allows us to think of actions as *efforts* the provider puts in when executing a given task: the more effort is invested, the higher the likelihood that the obtained outcome will be closer to the standard. Of course, expending more effort also means bearing a higher cost.

4.2 Actions, outcomes and incentives

More formally, and following a finite model for actions and outcomes, we have that:

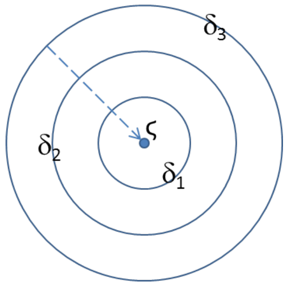

Fig. 1. A standard as a target

- The provider has an ordered set of possible actions $\mathcal{A} = \{a_1, ..., a_n\}$, where $a_i \prec a_j$ if $i < j$. This means that $Cost(a_i) < Cost(a_j)$ (a_i is less costly to the provider than a_j).
- The possible observable outcomes that the provider may obtain is an ordered set $\bar{\mathcal{X}} = \{\bar{x}_1, ..., \bar{x}_m\}$, where $\bar{x}_i \prec \bar{x}_j$ if $i < j$ ($\bar{x}_i$ is a worse performance than $\bar{x}_j$). For simplification, we will assume that $\bar{x}_i \in [0, 1]$, for all $i \in [1, m]$: each $\bar{x}_i$ will denote the percentage of the target standard that has been achieved.
- There is a probability distribution function for $\bar{\mathcal{X}}$ given an action in $\mathcal{A}$, where $p(\bar{x}_k|a_i)$ is the probability of obtaining outcome $\bar{x}_k \in \bar{\mathcal{X}}$ when performing action $a_i \in \mathcal{A}$. We have that $\sum_{k=1}^{m} p(\bar{x}_k|a_i) = 1$, for all $i \in [1, n]$.

We assume that the monotone likelihood ratio property (MLRP) [1], relating actions with outcomes (as defined in Def. 6), holds for every provider. This property indicates that greater efforts are more likely to produce better outcomes.

Definition 6 *The* monotone likelihood ratio property *holds iff for any* $a_i, a_j \in \mathcal{A}$ *with* $a_i \prec a_j$ *we have that the likelihood ratio* $p(\bar{x}_k|a_i)/p(\bar{x}_k|a_j)$ *is non-increasing in* k*.*

Incentives are specified through an incentive schedule mapping possible outcomes to payments to be collected by the provider:

- Function $I : \bar{\mathcal{X}} \to \mathcal{I}$ maps each possible outcome in $\bar{\mathcal{X}}$ to a specific incentive value in $\mathcal{I}$. We assume that I is non-decreasing, that is, $I(\bar{x}_1) \leq ... \leq I(\bar{x}_m)$, meaning that higher outcomes must have at least the same incentive as lower ones. Furthermore, we look at incentives as producing some change in the utility the agent would get if no incentives were in place; in this sense, $\mathcal{I} = \{\iota : \iota \in [-1, 1]\}$, where positive values denote percentage increases in utility and negative values denote percentage decreases in utility (i.e., they are seen as penalties). When $\iota = 0$ there is no incentive in place.

Based on the stochastic model of action outcomes explained above, each provider is taken to be expected utility maximizer. Therefore, when choosing

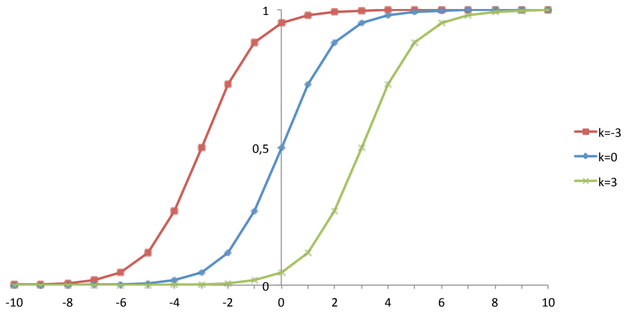

Fig. 2. Different $u(I(\bar{x}))$ functions varying κ

the action to perform it will maximize expected utility [11]:

$$\arg\max_{a \in \mathcal{A}} \mathbb{E}_a = \sum_{i=1}^{m} p(\bar{x}_i|a)u(I(\bar{x}_i)) - Cost(a) \tag{1}$$

where $u(I(\bar{x}_i))$ is the utility the agent gets from obtaining performance outcome $\bar{x}_i$. More precisely, this utility is not directly dependent on performance, but on the payment $I(\bar{x}_i)$ it will get from such a performance.

Given a task execution with an actual outcome and an incentive schedule, the provider's utility is given by

$$U(S, \bar{x}, a) = u(I(\bar{x})) - Cost(a) \tag{2}$$

Function $u : \mathcal{I} \rightarrow [0..1]$ is the utility function for the incentive received, which is taken to be strictly increasing. We assume that risk aversion is an internal characteristic of provider agents. In order to implement this function we use a sigmoid approach as follows:

$$u(I(\bar{x})) = \frac{1}{1 + e^{-I(\bar{x})*B+\kappa}} \tag{3}$$

where $\kappa \in \mathbb{R}$ represents a parameter to tune the center of the sigmoid function and B is a constant in $\mathbb{N}^+$. Figure 2 shows several examples of Equation 3 to model $u(I(\bar{x}))$ with different κ values and $B = 10$. Observing the curves we notice that when a provider agent is not offered any incentive, function $u(I(\bar{x}))$ returns a default value representing the personal gain the agent gets from the contract (driven by the value of κ) regardless of how it performs. Back to Equation 2, $u(I(\bar{x}))$ somehow determines the utility of the actions according to the incentives provided. Thus negative incentives provided to the agent (punishment) results in lower utility for the provider whilst higher incentives results in a higher utility. The κ value encapsulates different characteristics of the agent, such as capability, willingness, cooperativeness, dependency, attitude, etc. Varying κ we obtain different profiles of provider agents with different default gain values.

4.3 Deviations and responses

Given the previous performance of each provider, on which standards (via roles) have been defined (as described in Section 3), one might ask why those agents may start to under-perform, in the sense that they are not able to meet the standards they agree with in a contract. We identify two possible sources for such deviations, which naturally come to surface from analyzing Equation 1:

- The cost of actions has increased, leading the agent to choose actions that do not obtain the same level of performance (because those actions have lower probabilities for each possible performance outcome);
- The probabilities for the performance outcomes of an action have changed, e.g. due to environmental factors not under the control of the agent, meaning that a same action is not as effective as before.

In this paper we do not address the issue of how the agent becomes aware of these changes in order to take them into account when deciding which action to perform (according to Equation 1). In both cases we can easily think of external factors causing these changes. For instance, in a supply chain, fluctuations on prices for different inputs (e.g. parts or raw materials obtained from suppliers) will certainly influence the cost of executing the task. As for outcome probabilities, the agent may be able to update these estimations on-line, according to run-time experience.

These deviations in performance make the role clustering (obtained as described in Section 2) unfit to represent the current performances of agents in the T-MAS, in terms of the standards extracted from the roles. Therefore, in order to maintain role stability when agents deviate from agreed standards, the system may determine and employ an appropriate incentive schedule $I : \bar{\mathcal{X}} \rightarrow \mathcal{I}$. Since actions are not observable, this schedule is based exclusively on the measurable outcomes of task execution. As mentioned in Section 4.2, an outcome denotes the percentage of the target standard that has been met.

Unlike typical approaches in game theory, we do *not* assume that action costs, their probability distributions on outcomes, or utility functions on incentive values are known to the incentive policy maker. Thus, we see the problem of searching for an optimal incentive schedule as a *reinforcement learning* (RL) [10] problem. We assume that the principal prefers higher outcomes with the lowest incentives needed.

In any reinforcement learning problem, an agent (the incentive policy maker in our case) perceives the environment and determines the *state* it is facing. Based on this state, it will try to determine, by exploring its action set, the best possible *action* in terms of a *reward* obtained from the environment. The goal of the learning agent is to maximize the reward it receives in the long run. Rewards are used in RL to tell the learning agent what we want it to achieve, not how to achieve it. Therefore, rewards are strongly connected with arrival states as a consequence of the action (an incentive schedule) that the agent chooses to employ. It may be the case that a particular incentive schedule only obtains the desired effect a number of iterations after it has been applied.

We see our problem as a continuing task that the learner needs to solve. Furthermore, because of the sources for deviating behaviors identified above, providers may decide differently when facing a specific incentive schedule at different times. Reinforcement learning naturally encompasses this kind of situations by allowing for an adjustable trade-off between exploitation (taking advantage of the actions that have been found as good) and exploration (trying out other actions whose effect is not totally known).

In the following subsections we briefly discuss how states, actions and rewards can be addressed in the problem faced by the incentive policy maker.

States. A state is characterised by the performance outcomes the agents belonging to a given role are currently obtaining. States exhibiting performances farther away from the target standard need to be addressed with stronger incentive policies. On the other hand, states that denote abidance to agreed standards need no intervention from the policy maker.

The first issue to take into account is related with the meaning of "timeliness": the state encompasses information regarding the most recent task executions of agents within a role. The aggregation of such data may take different forms, depending on how role performance quality is to be interpreted. One possibility is to average over the last executions:

$$role\_perf = \left(\sum_{i=t-\Delta}^{t} \bar{x}^i \right) / \Delta$$

where t is the current time step, $\bar{x}^i$ denotes outcome obtained at time step i and Δ is the size of the time window, i.e. the number of task executions to consider. Another possibility is to consider the worse task execution as a reference, with the aim of obtaining a more sensitive and pro-active incentive mechanism:

$$role\_perf = min(\bar{x}^{t-\Delta}, ..., \bar{x}^t)$$

This would indicate that the policy maker wants every agent within the role to perform equally well.

In order to reduce the size of the state space, states are discretized according to the number of levels of deviation that are to be addressed differently, as illustrated in Figure 1. In order to discretize the obtained value, we define a δ parameter that tells us in how many intervals we should split the distance to target standards (which is always a value between 0 and 1, interpreted as a percentage of the target that has been achieved, as explained before):

$$state = \begin{cases} 1 & \text{if } role\_perf = 1 \\ \lfloor role\_perf * \delta \rfloor / (\delta - 1) & \text{if } role\_perf < 1 \end{cases}$$

This function ensures that we will have δ possible states, represented by values within $[0, 1]$. If role performance is maximum (i.e. 1), then the state denotes an optimal situation; as performances get farther from 1, higher δ values will bring us more quickly to different (less desirable) states.

Actions[3]**.** Available learner actions concern incentive schedules I that specify, for any $\bar{x} \in \bar{\mathcal{X}}$, an incentive value $\iota \in \mathcal{I}$. As mentioned in Section 4.2, this function I is non-decreasing (better outcomes should not receive lower incentives than worse outcomes).

Based on the approach described in [1], each action can be represented as an incentive vector $\iota = (\iota_1, ..., \iota_m)$, where m is the number of possible outcomes, each $\iota_i \in \mathcal{I}$ and, as already mentioned, $\iota_i \leq \iota_j$ for all $i < j$. In order to reduce the action space, we can consider only actions within the set $\lfloor \mathcal{I} * 10 \rfloor / 10$, which gives us discrete action values with 0.1 steps.

Depending on the number of outcomes to consider, this may still give us a big number of actions to experiment with. In any case, given the MLRP property described in Def. 6 and the fact that providers are expected utility maximizers, we may be able to guide the way exploration of different actions is carried out. Techniques such as action refinement [5] can also be a very useful approach here.

Rewards. Just as states can be defined differently depending on the level of granularity we need the system to respond to, different reward configurations can help us tune the goal we want the learner to pursuit. An incremental approach to assigning rewards to different states will help the agent to find which states are more likely to be in the path to the best possible outcomes. If, on the other hand, we need the learner to quickly influence providers to escape from undesired outcomes, only states in which the outcomes are at the target standard should be awarded.

Assuming that we want to minimize the incentives needed to obtain a certain level of performance, the cost of implementing a specific incentive schedule must be taken into account when choosing among actions. In RL, quality values of the form $Q(s, a)$ are computed that determine the expected return for executing an action a in a state s. It is therefore important to consider, when computing Q values, the estimated costs of such actions. It should be noted that these costs do not result automatically from deciding to implement a specific incentive schedule; instead, they depend on the actual performances that such a schedule has lead to, since incentives are paid (if positive) or collected (if negative) according to actual outcomes.

5 Related work

In this paper we have attempted to put forward a theoretical approach in order to build standards from roles in dynamic task-oriented MAS. Although the creation of standards has been more deeply studied in the fields of economics and finance, in the MAS community there have been some attempts to dynamically build social structures to foster interactions. For instance, there are many approaches

[3] We emphasize that these are the learner's actions (i.e., those available to the incentive policy maker), and not the actions of the provider agent as discussed in Section 4.2. We here use the same term *action* because it is well established in RL literature.

on how norms are formed and how they emerge from expectations. In [12] the authors present a work that gathers users expectations for social interactions to transform then into logic formulae that can be used in order to check an eventual outcome. The main difference with our approach is that they use explicit requests to the users to gather their expectations, while we automatise that process by using the role creation mechanism. Other approaches related to this issue are [2] and [4], in which the authors put forward how prescriptions might emerge from individual expectations eventually forming norms.

There are economic approaches also founded on the emergence of standards, such as [9], in which Sherstyuk proposes a method to set an appropriate performance standard to develop optimal contracts, i.e., contracts in which the provider agent's best choice is to keep the standard through its action. In this paper, however, we are not interested in obtaining optimal performance standards, but we are instead concerned about how to maintain the level of those standards once they have been created.

In the same line Centeno *et al.* [3] present an approach on adaptive sanction learning by exploring and identifying individuals' inherent preferences without explicit disclose of information, i.e. the mechanism learns over which attributes of the system should modifications be applied in order to induce agents to avoid undesired actions. In our case, we adhere to a more formal scenario, in which interactions are regulated by means of contracts and, besides, we assume that attributes that may be modified by means of incentives are already known by the mechanism.

The approach taken in [8] also assumes that the mechanism knows which attributes it should tweak in order to influence agents' behaviors, namely by adjusting deterrence sanctions applicable to contractual obligations agents have committed to. The notion of social control employed there is similar to our notion of role standard maintenance, although instead of a run-time discovered standard a fixed threshold is used to guide the decisions of the policy maker. Moreover, only sanctions (seen as fines) are used to discourage agents from misbehaving, while here we are more interested on incentivating agents to do their best (by using appropriate actions) while executing the tasks they are assigned to.

6 Conclusions and future work

Standards are used as a means to articulate contracts in social interactions. When dealing with organisational multi-agent systems, roles might be used as a reference in order to create standards, since the latter can be seen as a measure of the quality of performance of agents playing them. In this paper we have proposed a mechanism that, on the one hand, creates performance standards from roles discovered at run-time in a multi-agent system and, on the other hand, provides incentives to make agents maintain a level of performance as close as possible to the standards.

An issue that we have intentionally left outside this exercise is the decision regarding when it is less costly to rearrange the role taxonomy than to employ incentive mechanisms to keep the current configuration's quality. This is something left for future work.

We intend to pursue the mechanism presented in this paper, first by refining the learning model of the incentive policy maker, and second by building a simulation that enables us to confirm and improve on the virtues of the proposed approach. It is our belief that the incentive mechanism to develop includes some interesting modeling choices that may find applications in some application domains.

References

1. B. Caillaud and B. Hermalin. Hidden action and incentives. Teaching Notes, U.C. Berkeley, accessed at http://faculty.haas.berkeley.edu/hermalin/agencyread.pdf, 2000.
2. C. Castelfranchi, F. Giardini, E. Lorini, and L. Tummolini. The prescriptive destiny of predictive attitudes: From expectations to norms via conventions. In R. Alterman and D. Kirsh, editors, *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*, Boston, MA, 2003.
3. H. Centeno, Roberto Billhardt and R. Hermoso. An adaptive sanctioning mechanism for open multi-agent systems regulated by norms. In *Proceedings of the 2011 23rd IEEE International Conference on Tools with Artificial Intelligence*, ICTAI '11, page to appear. IEEE Computer Society, 2011.
4. R. Conte and C. Castelfranchi. From conventions to prescriptions. towards an integrated view of norms. *Artificial Intelligence and Law*, 7(4):323–340, 1999.
5. T. G. Dietterich, D. Busquets, R. L. d. Mántaras, and C. Sierra. Action refinement in reinforcement learning by probability smoothing. In *Proceedings of the Nineteenth International Conference on Machine Learning*, ICML '02, pages 107–114, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.
6. R. Hermoso, H. Billhardt, and S. Ossowski. Role evolution in open multi-agent systems as an information source for trust. In *9th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2010)*, pages 217–224. IFAAMAS, 2010.
7. J. Laffont and D. Martimort. *The Theory of Incentives: The Principal-Agent Model.* Princeton paperbacks. Princeton University Press, 2002.
8. H. Lopes Cardoso and E. Oliveira. Social control in a normative framework: An adaptive deterrence approach. *Web Intelligence and Agent Systems*, 9:363–375, December 2011.
9. K. Sherstyuk. Performance standards and incentive pay in agency contracts. *Scandinavian Journal of Economics*, 102(4):725–736, 2000.
10. R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction.* The MIT Press, 1998.
11. J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior.* Princeton University Press, 3 edition, May 1980.
12. M. Winikoff and S. Cranefield. Eliciting expectations for monitoring social interactions. In *Proceedings of the First international conference on Computer-Mediated Social Networking*, ICCMSN'08, pages 171–185, Berlin, Heidelberg, 2009. Springer-Verlag.