Policies for role maintenance through incentives: how to keep agents on track

Henrique Lopes Cardoso[1], Ramón Hermoso[2], and Maria Fasli[2]

[1] LIACC / DEI, Faculdade de Engenharia, Universidade do Porto, Portugal
hlc@fe.up.pt
[2] School of Computer Science and Electronic Engineering, University of Essex, UK
{rhermoso, mfasli}@essex.ac.uk

Abstract. Roles are usually seen as a descriptive concept that agents adopt so showing expectation on its future interacting behaviour. These expectations, so called standards, may be used to articulate contracts among partners in environments dealing with uncertainty. However, few work has been done in the field of multi-agent systems addressing the evolving nature of roles, especially in open systems, in which changes in the population should bring about changes in the way expectations are assessed. In this paper, we put forward a mechanism to search for appropriate incentives aiming to keep agents fulfilling the expectations generated by the roles they play. Furthermore, we also present role evolution policies that allow the system to re-arrange role players when incentives are not effective. We present some empirical results supporting our approach.

1 Introduction

In open multi-agent systems, heterogeneous agents may enter and leave at will and exhibit self-interested behaviours while interacting with others. Commitments, as a means to make interactions more predictable, may be jeopardized by the dynamic nature of the environment: when facing new circumstances, if agents are not sufficiently socially concerned, they may value their private goals more highly than the commitments they previously established.

In general, in open systems we cannot ensure that participants will behave consistently over time either because of their ability or willingness. The primary objective for agents to operate in such environments is to achieve their design objectives and goals. In some cases, an agent may not be capable of maintaining a certain standard of behaviour throughout its lifetime. In other cases, an agent may deviate from its previous and typical behaviour either intentionally, or because its objectives and goals have changed, and hence, this needs to be reflected in the actions taken or in extreme cases because of underlying errors in its program. Therefore in the context of open environments, it is important to take into account the evolution of an agent's internal goals, motivations and preferences and even skills and abilities, in addition to the dynamics of the interaction environment as a whole.

We adopt the notion of *role* given by Hermoso *et al.* in [4]: roles are described as expectations over a set of tasks. In the same work, a role evolution mechanism is presented to enhance partner selection for task delegation purposes. That approach is based on examining the agent society and identifying “run-time roles” – so building a role taxonomy – that cluster agents with similar skill patterns for a certain (set of) task(s). From this perspective, a mechanism allows identifying the role that labels the agents most suitable to perform a specific task.

In previous work [5], we have proposed a model to transform these run-time roles into standards that agents may commit to. Given the evolving nature of open environments and inherent uncertainty in agent behaviour as pointed out above, the problem faced by the organization in which agents have been (artificially) embedded is that of accurateness: this translates to the need to assess whether agents originally designated as capable of enacting certain roles are still up to the challenge. If they are not, one may choose to reorganize and assign new roles to them according to their up-to-date performance. If, however, this reorganization involves a non-trivial cost, another potentially less costly approach is to influence the agents’ decision making through incentives in an attempt to keep them on track. The goal of this paper is to explore these two approaches when addressing the problem of role maintenance.

The rest of the paper is structured as follows. Section 2 provides some preliminary definitions in which our model is based, namely roles and standards. Section 3 explains the decision making of agents that are seen as playing roles identified by the system and introduces the notion of incentive schedule. Section 4 proposes two different policies to maintain roles. An incentive-based policy implementing a local search procedure tries to find which incentive schedules should be applied in order to maintain role quality. Reorganization is imposed when incentives are not cost-effective. We present some empirical results in Section 5. The paper closes with the conclusions in Section 6.

2 Background

The rationale behind creating and maintaining performance standards relies on the concept of role proposed by Hermoso *et al.* [4]. 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 perceived by others – should also do. A society of agents may be covered by an overlay role taxonomy formed by extracting capacities and trust relationships among agents over time. In particular, the authors show an approach for a coordination mechanism for Task-oriented MAS (T-MAS) in which agents may interact with others by delegating certain tasks. The authors assume that agents participating in the system are rational, that is, they behave as utility maximisers. Thus, the main task of the mechanism is twofold: i) to capture similar behaviour among participants that play a role; and ii) to 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 out-performing others and, consequently, being more trusted by the participants. This mechanism has been exhaustively tested in different conditions.

This paper addresses the problem of how to maintain the agents' performance on the level of quality determined by the roles they are playing. In order to take this step, we need to move from roles as expectations of behaviour [10] to the explicit handling of such expectations as conventions and further as norms [2] that can be committed to. In other words, we need to transform information about role specialisations into performance standards that agents could use to estimate interaction outcomes. In order to do that, we adhere to the definition of standard given in previous work [5]. Standards emerge from the notion of task specialisation as an aggregate level of performance that agents playing a role have shown to achieve in past interactions.

3 Dynamics

In this section, we introduce the decision making apparatus that agents use when executing tasks. Our approach is based on 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, who will therefore be interested in influencing the effort that the agent puts in performing the task. The effort is expressed in terms of available actions which have associated execution costs. In the so-called *hidden action* setting, it is assumed that the actual actions as executed by the provider agent are unobservable to the principal. Instead, only some performance measures of such actions are observed. Actions determine, usually stochastically, the ensuing 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 provider agent to choose the actions that are more likely to lead to an intended performance standard.

3.1 Targeting standards

As described in previous work [5], 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 sub-optimal outcome. Figure 1 illustrates this notion, where ς represents the target standard that the requester would expect, and each concentric circle labelled with a δ_i denotes equidistant performances to the target. Concentric lines highlight the fact that we shall consider deviations

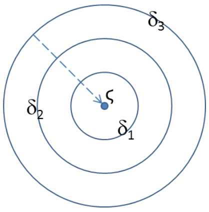


Fig. 1. A standard as a target

in any direction to be equally harmful in terms of expected values. The arrow pointing towards the centre discloses the aim of our incentive-based approach with which we will try to encourage providers to better conform to the standard.

3.2 Actions and outcomes

In our model, we will assume that each provider has a set of actions at its disposal, each with a cost and a probability function for obtaining different performance outcomes. Actions can be thought of as the *effort* the provider puts in when executing a given task: employing more effort, will more likely lead to a higher outcome. Of course, expending more effort also means bearing a higher cost. Following a finite model for actions and outcomes we have that:

Definition 1 *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 $Cost(a_i) < Cost(a_j)$, i.e., a_i is less costly for the provider to execute than a_j.*

Definition 2 *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 outcome than $\bar{x}_j$ and therefore indicates worse performance).*

Definition 3 *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. 4), holds for every provider. This property indicates that greater efforts are more likely to produce better outcomes.

Definition 4 *The MLRP 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.*

3.3 Incentives

Given the fact that only outcomes, and not efforts, are observable by the principal, incentives are specified through an *incentive schedule* mapping possible outcomes to incentive values to be collected or paid by the provider (Def. 5).

Definition 5 *An* incentive schedule $I : \bar{\mathcal{X}} \to \mathcal{I}$ *maps each possible outcome in* $\bar{\mathcal{X}}$ *to a specific incentive value in* $\mathcal{I}$.

We look at incentives as producing some change in the utility that the agent would get if no incentives were in place. In this sense, $\mathcal{D}(\mathcal{I}) = [-1, 1]$, where positive values denote percentage increases in utility and negative values denote percentage decreases in utility (i.e., they are seen as penalties). A null value means that there is no incentive in place.

3.4 Providers' decision rationale

Based on the stochastic model of action outcomes explained above, each provider is assumed to be an expected utility maximizer agent. Therefore, when choosing the action a to perform it will seek to maximize expected utility [9]:

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

where $u(\bar{x}_i, I(\bar{x}_i))$ is the utility the agent gets from obtaining performance outcome $\bar{x}_i$, taking into account the incentive $I(\bar{x}_i)$ it will get from such a performance. Utility is therefore defined as follows:

$$u(\bar{x}_i, I(\bar{x})) = u(\bar{x}_i) \times (1 + sens(I(\bar{x}))) \tag{2}$$

This function encompasses two sub-functions: the prior utility $u(\bar{x}_i)$ collected according to the outcome obtained, and the effect on this utility of the incentive value applied (prior utility remains unchanged when the agent is immune to incentives: $sens(I(\bar{x})) = 0$). In order to accommodate different sensitivities, $sens : I(\bar{x}) \to [0, 1]$ is modelled as follows, where $B \in \mathbb{N}^+$:

$$sens(I(\bar{x})) = \frac{1}{1 + e^{-I(\bar{x}).B}} \tag{3}$$

Higher B values make the agent more sensitive to incentives, while with lower ones the agent will tend to behave the same regardless of any incentives.

4 Policies

As pointed out in Section 2, each provider is assigned to a role according to the performance that it exhibits. Given the evolving nature of the environment in which the agent operates, it may be the case that the agent alters its performance, for better or for worse. We should emphasize at this point that, since the agent is an expected utility maximizer, its decision regarding how much effort to employ is conditioned by a number of factors which we can identify by analysing Equation 1. Any changes in these factors are thus possible causes for a deviation from the standard characterizing the agent's assigned role: i) costs of the efforts that the agent has at its disposal; ii) effectiveness of efforts, that is, their probability distributions over performance outcomes; iii) utility the agent gets from obtaining each possible outcome; and iv) sensitivity of the agent with respect to any incentives it may be offered. In this paper, we assume the agent somehow becomes aware of changes in these factors in order to take them into account when deciding which action to perform.

These deviations in performance render the role clustering unfit to represent the current performances of agents in the system, 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 $I : \bar{\mathcal{X}} \rightarrow \mathcal{I}$ (see Def. 5). Since actions are not observable, this schedule is based exclusively on the measurable outcomes of task execution, which for the sake of defining appropriate incentive schedules are compared with the target outcomes characterizing the roles. We call the entity responsible for maintaining role stability an *incentive policy maker* (IPM). The IPM does not have access to the factors influencing the agents' decision making as this is considered to be private information.

The goal of the IPM is to keep on target the agents playing a specific role, i.e., agents should obtain outcomes as close as possible to the target outcome of the role. We assume the IPM prefers to achieve this aim with the least incentives needed. In case of failure to accomplish this aim, or if by doing so the IPM has to apply a too costly incentive schedule, then it is time to rearrange the role – agents are seen as no longer being able to perform the role at a bearable cost for the system and should thus be reassigned to a different role.

4.1 Incentive schedules

Given an incentive schedule offered to the agents playing a specific role, we may determine its effectiveness by looking at the outcomes that have been obtained. We should also take into account the cost of applying such an incentive schedule. Given the stochastic nature of agent efforts in terms of obtained outcomes, an incentive schedule's quality oscillates around some value, regardless of there being any changes in the environment that lead agents to deviate from previously obtained outcomes. In order to compute an incentive schedule's quality $Q(I)$, we aggregate a sequence $\bar{X} = \langle \bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n \rangle$ of n obtained outcomes ($\bar{x}_i \in \bar{\mathcal{X}}$), and compare them with the target outcome $\bar{x}$. We define $Q(I)$ as:

$$Q(I) = \left(w_o \times \mathit{targetHit}(\bar{X}) - w_c \times \mathit{totalCost}(I, \bar{X})\right) / (w_o \times n) \tag{4}$$

$$\mathit{targetHit}(\bar{X}) = n - \sum_{i=1}^{n} |\bar{x}_i - \bar{x}| \tag{5}$$

$$\mathit{totalCost}(I, \bar{X}) = \sum_{i=1}^{n} |I(\bar{x}_i)| \tag{6}$$

Target hit measures the incentive schedule's effectiveness in inducing agents to meet the target. Any values outside the target are seen as deviations that need to be minimized in terms of role maintenance – for simplicity we assume $\bar{\mathcal{X}} \subset [0,1]$. A perfect performance consists of n outcome values at the target $\bar{x}$. The *total cost* of the incentive schedule takes into account actually paid incentives, which depend on the outcomes obtained. In order to normalize $Q(I)$ to the range $[0,1]$, the weighted sum of these two components is divided over the maximum quality value, where all n outcomes are on target and no incentive values are applied. Weights w_o and w_c allow us to define the relative importance of these two conflicting goals, e.g., by giving priority to obtained performance over how much it costs to achieve it in terms of incentives paid.

In the quest to find out the best incentive schedule, measured both in terms of effectiveness and cost, the number of different incentive schedules available to the IPM is quite high. In order to reduce this search space, we can limit ourselves to incentive schedules composed of values within the set $\lfloor \mathcal{I} \cdot 10 \rfloor / 10$, which gives us discrete incentive values with 0.1 steps. Depending on the number of outcomes to consider, this may still give us a huge number of schedules to experiment with. We can slightly alleviate this issue by taking into account the intuitive heuristic that we should promote outcomes closer to the target no less than outcomes farther away.

4.2 A local search approach

Given the high number of schedules to experiment with, we follow a *local search* approach to find the optimal incentive schedule. More specifically, we employ a hill-climbing procedure, by successively trying to find out neighbouring incentive schedules that are better than the currently employed one. In order to find them, however, we need to try out incentive schedules before we know how worthy they are, which makes the search more stochastic. Furthermore, given the dynamics of the environment, these quality values are not constant over time, and thus exploration must always be an option.

To generate the neighbours of an incentive schedule, we introduce a step change (upwards or downwards) on the incentive value being applied to the target outcome. If the change is downwards, we correct the schedule obtained so that outcomes closer to the target have at least the same incentive as outcomes farther away. This gives us a cardinality of at most 2 in the set of neighbours of each possible schedule.

The local search procedure is illustrated in Algorithm 1. At each step, we compare the current schedule with the best known (line 4), and update it (line 5). Whenever the current schedule is not better than the best known (which means we have tried a neighbour that was not found to be better), we go back to the best as the current schedule (line 7). Then we randomly explore, with probability $1-Q(best)$, the neighbours of the best schedule (line 9) – the better the schedule is, the less likely we will explore, exploiting instead the best schedule we know of. In order to compute the quality of each incentive schedule we need to apply it enough time to aggregate new evidence to fill in sequence $\bar{X}$ (see Eq. 6).

Algorithm 1 Local search for incentive schedules

```
best {the best known incentive schedule}
current {the currently applied incentive schedule}
loop
    if $Q(current) > Q(best)$ then
        $best \leftarrow current$
    else
        $current \leftarrow best$
    end if
    with probability $1 - Q(best)$: $current \leftarrow RandomNeighbour(best)$
end loop
```

4.3 Re-organization

Given the organizational flavour of role taxonomies, we have preferred an incentive-based policy to a reorganization of the system in terms of up-to-date skills of agents. This preference is due to both practical and fundamental reasons. The computational complexity of creating or updating a role-taxonomy may be considerable, especially when little is known about the agents that are to take part in this organization, and thus about the roles that are to be created (see e.g. the clustering approach in [4]). Furthermore, the very notion of roles as a descriptive facet of an organization presumes some notion of stability as they provide a means to identify specific agents within the organization. If we reorganize too often, this property of long-term existence is lost.

Nevertheless, there will be situations in which reorganization is a better option as compared to applying incentive schedules with the aim of influencing the providers' behaviour. On the one hand, although an effective incentive schedule may be found, applying it may bear a significant cost. On the other, there will be situations in which an effective incentive schedule is not found. This may happen either because the search procedure employed is not able to find it (the local search approach described in the previous section is vulnerable to local optima), or because environmental changes have reduced the influencing ability of the IPM, therefore lowering the effectiveness of the most effective incentive schedule available. In either case, the quality $Q(I)$ of the best incentive schedule found will be lower than a certain value. We may use this information to decide when to reorganize the role taxonomy. By reorganizing we mean to obtain roles that exhibit a higher cohesion in terms of the skills of the agents to which those roles have been assigned.

Given the permanent search for better incentive schedules (see Algorithm 1), the stochastic nature of outcomes, and possible environmental changes, $Q(I)$ may oscillate. Therefore, in order to infer that the incentive-based policy is not able to obtain incentive schedules that are cost-effective we must allocate enough time for the search procedure to run. The approach that we take to make such a call is to collect n_q quality values $\langle q_1, q_2, \ldots q_{n_q} \rangle$ for the last n_q employed incentive schedules, and to compare the average of those values with a predefined threshold $q_{min} \in [0, 1]$. Reorganization is the choice if $\left(\sum_{i=1}^{n_q} q_i \right) / n_q < q_{min}$.

By increasing the value of parameter q_{min}, we demand more cost-effective incentive schedules before a reorganization decision is taken. Lower q_{min} values will allow for less cost-effective schedules, meaning that we consider reorganization as being a more costly operation to undertake.

5 Evaluation

The software we have used to empirically evaluate our approach is a simulation framework built using RePast Simphony. We have designed experiments to present how our proposal learns appropriate incentives for different types of providers along time and, in case the search for incentive schedules fails, how the role evolution mechanism can be launched to re-allocate providers in roles more representative of their current skills.

5.1 Provider profiling

There are three functions that together determine the behaviour of each provider. Firstly, we define a function for effort costs, as mentioned in Def. 1; these costs are used in the provider's decision making (see Equation 1). For this purpose, we use Equation 7 to define different profiles of providers. This means that different providers may have different costs for the same efforts.

$$Cost(a) = \alpha \cdot (k + (1 - k) \cdot a^{1/\beta}) \tag{7}$$

We model five provider profiles, whose effort cost functions are depicted in Figure 2. Due to length constraints we only use some of them in the experiments.

- **Flat.** The provider's efforts cost is the same for any effort ($\alpha = 0.5$, $k = 1$ and $\beta = 1$).
- **Linear.** Effort costs increase linearly. We set $\alpha = 1$, $k = 0$ and $\beta = 1$.
- **Concave.** Effort costs are modelled with a concave-shaped function. We use $\alpha = 1$, $\beta = 0.3$ and $k = 0$. We do not use it.
- **Convex.** Effort costs are modelled with a convex-shaped function. We use $\alpha = 1$, $\beta = 3$ and $k = 0$.
- **Random.** Values for α, k and β are randomly selected with $0.5 \leq \alpha \leq 1$, $0.3 \leq \beta \leq 3$ and $0 \leq k \leq 0.5$.

We use the same approach to define the second function: the outcome utility of the providers, that is, the utility the provider obtains from each possible outcome (this is the prior utility $u(\bar{x}_i)$ mentioned in Equation 2). The same profiles defined above are applied to this function.

The third function we need to put forward is the one relating provider efforts and obtained outcomes, as mentioned in Def. 3 and used in Equation 1. We have modelled this relation by using beta distributions, whose shape is controlled by two parameters, α and β. For each beta distribution, we set $\alpha = 1 + (c * p - c)$

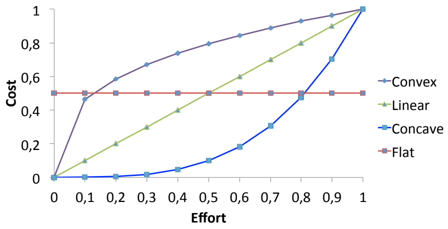


Fig. 2. Effort-Cost and Outcome-Utility provider profiles

and $\beta = p - (c * p - c)$, where c is a centre value and p is a peak factor that we have set picking a random value between 50 and 70.

To be more specific, we consider as possible outcomes the set $\{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_7\}$. For simplicity, the cardinality of efforts available to providers is the same (although it need not be): $a_1, a_2, \ldots, a_7$. We generate the outcome probability obtained by each effort a_i as a beta distribution with centres selected in a non-decreasing fashion from $\{\bar{x}_1, \ldots, \bar{x}_7\}$, so ensuring that the MLRP property holds.

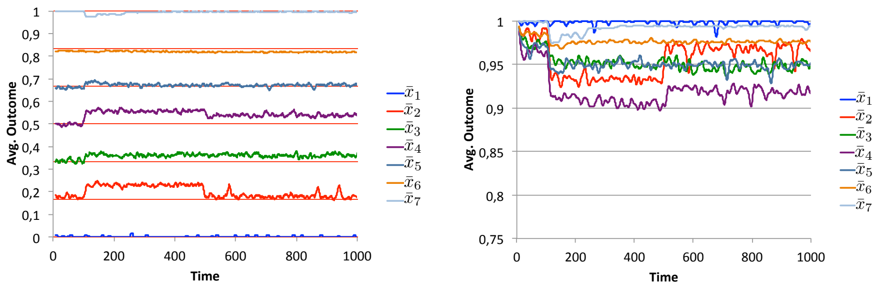
5.2 Empirical evaluation

We assume at the beginning of each experiment that the role evolution mechanism was run and providers have been allocated in one out of seven possible roles, each targeting a different outcome. Thus, we will have a population with providers allocated in the role they perform better (that is, corresponding to the outcome they most often obtain).

As the goal of this section is to give some evidence on the potential of the approach, we have divided the experiments into three different scenarios. The first two are meant to test the well functioning of the mechanism of incentive search detailed in Section 4.2, with different provider profiles in the population. The third scenario has to do with the need for reorganization in some cases. Besides we use two different search approaches: *individual search*, in which we learn incentive schedules for each provider separately; and *role search*, in which we learn incentive schedules for all the providers in the same role.

After a few empirical tests, we set the value of B to 10, in order to tune the sensitivity of providers to incentives (see Eq. 3). Moreover, also after empirical consideration, in order to evaluate the quality of the incentive schedules found we set $w_o = 2$ and $w_c = 1$, so giving priority to obtained performance over the cost of its application, in terms of incentives paid. The population size is 100 providers and we simulate task executions for every different provider in every time step. We show average results from different runs.

In the first scenario, we initiate the population with providers all with a Random profile, for both effort costs and outcome utility functions. We introduce changes into the system at time step 100. These changes consist of modifying 25% of the providers by randomising the parameters for effort outcome and outcome utility functions. In Figure 3, we show the results of this experiment running the mechanism with *individual search*, so seeking incentive schedules personalised for each provider in order to make them perform as close as possible to the standard created from the role they play.



(a) Avg. Outcome per role

(b) Avg. $Q(I)$ per role

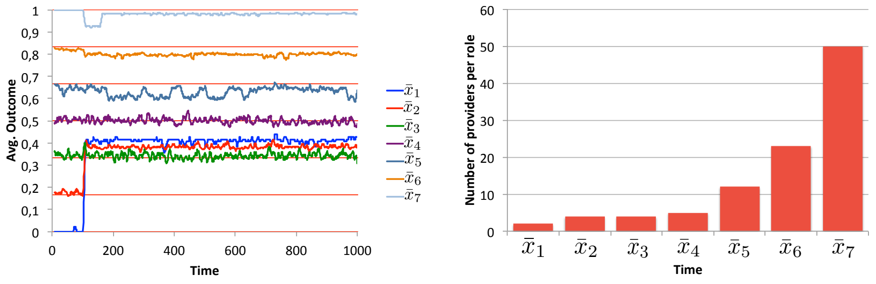
Fig. 3. Results for the random population changing 25% at time $t = 100$ to random profile (individual search)

In Figure 3(a), we observe the average outcome of the providers in the seven different roles. As expected, each provider starts fully attached to its role standard (thinner flat lines). Then, at $t = 100$ changes in the population make some providers deviate from the standard. The local search (applied individually) is able to incentivize the providers deviating from their standards with adequate incentives. Note that for some roles, to incentivize providers with a proper incentive schedule takes longer due to the stochastic nature of the search.

Figure 3(b) shows how $Q(I)$ evolves through time for the same experiment. We can observe how in the first 100 ticks, incentives are mostly not used, since there is no need for them (notice that until $t = 100$ providers behave as they did when allocated in the role they play). We also observe a slight oscillation around the optimum value. This is a consequence of the stochastic nature of effort outcome distributions and the exploration of the local search method. After the changes introduced at $t = 100$, the quality of incentive schedules temporarily drops, while the system is striving to find better schedules eventually recovering around $t = 500$. Although an increase on $Q(I)$ is clearly observable for some roles, the cost of applying the incentives prevents it from rising back to initial values (before changes in the population were introduced).

The second scenario starts with a population of providers all with a Random profile, for both effort costs and outcome utility functions. We then introduce changes into the system at time step 100. These changes consist of modifying all providers by randomising the parameters for effort outcome and outcome utility functions. In Figure 4, we show the results of this experiment running the mechanism using *role search*, so seeking distinct incentive schedules for different roles to apply collectively to all providers playing each role, in order to make them perform as close as possible to the standard the role defines.

In Figure 4(a), we can observe the results of this experiment. As in the previous experiment, providers start fully attached to their role's standard (thinner flat lines) at the beginning of the simulation. After the changes take effect we note a deviation in the average outcomes for each role. From that moment on, the local search starts to be effective in some of the roles. For instance, it barely takes 80 time steps to get back role defined by outcome $\bar{x}_7$ close to the standard again. It takes longer to do the same with the role defined by outcome $\bar{x}_5$ (about 500 time steps). These differences are due to the stochastic nature of the mechanism, which can get trapped sometimes in local optima. Another feature to remark is the inability of the mechanism to fix some of the providers that clearly deviated from their required behaviour. This is the case for roles defined by outcomes $\bar{x}_1$ and $\bar{x}_2$. In this case, the change in the parameters of the effort costs and outcome utility functions has been significant, since providers playing those roles are too heterogeneous with regard to the expected outcomes, resulting in the absence of an appropriate incentive schedule suitable for them. Actually, this is more likely to happen in those roles with a smaller number of providers. In them, a change in the providers' profile makes the role "behave" in a totally different manner from what was expected when it was created. We can notice this issue in Figure 4(b) in which we display the size, in terms of providers, of every role in the system. Observing it we notice that low populated roles for lower target outcomes $(\bar{x}_1, \bar{x}_2)$ are more sensitive to changes, that is, providers will very easily change outcomes with any changes in the effort-costs and outcome-utility functions. In other words, it is more difficult to bring the providers back to those expected outcomes $(\bar{x}_1, \bar{x}_2)$.



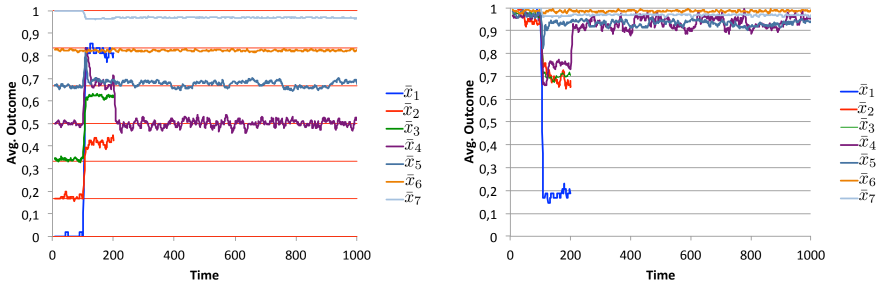
(a) Avg. Outcome per role

(b) Number of providers in each role

Fig. 4. Random population changing 100% at $t = 100$ to random profile (role search)

In order to address the inability to meet the standards in some roles through an incentive-based approach, we described in Section 4.3 a policy to consider if a re-allocation of providers is needed as a last resort of the system attempting to keep providers on track. We have implemented this policy with a threshold $q_{min} = 0.8$, the minimum incentive schedule quality that prevents re-organisation. If q_{min} is not met, a re-organisation of providers is carried out by re-allocating providers from one role to another. Re-organisation is carried out only for those roles that need it, not for the entire role taxonomy. The aim of the re-organisation is to get rid of the problem of providers for whom incentives are useless.

With the aforementioned in mind we put forward a third experiment that tries to verify our assumption on this. We start again with a population of providers all with a Random profile. After 100 time steps we change all providers to have a Convex profile (see Eq. 7). After that, in time step 700 another change occurs, this time converting all agents into Linear providers. Moreover, we let our mechanism search incentives for each role (role search), and we let the re-organisation policy consider re-allocation of providers every 200 time steps. Results are shown in Figure 5.



(a) Avg. Outcome per role

(b) $Q(I)$ per role

Fig. 5. Results for random population changing 100% at $t = 100$ and $t = 700$ to convex and linear profiles, respectively (role search). Re-organisation running every 200 units

We can observe in Figure 5(a) how at the beginning providers keep their behaviour according to their role. After the first change ($t = 100$), providers start to deviate from the target outcome marked by the standard. We can see how the local search method achieves to make providers behave according to their roles in some cases (role for target outcome $\bar{x}_5$ and more slightly in roles for target outcomes $\bar{x}_4$ and $\bar{x}_7$). When the first re-organisation takes place ($t = 200$), looking at Figure 5(b) we notice that there exist four roles (with target outcomes for standards $\bar{x}_1, \bar{x}_2, \bar{x}_3$ and $\bar{x}_4$) for which incentives are being useless (compared to q_{min} threshold). For that reason, the re-organisation policy decides to re-allocate agents playing those roles. In the case of the first three of them ($\bar{x}_1, \bar{x}_2, \bar{x}_3$), the policy removed the roles, re-allocating all the providers to other roles according to their changed behaviour. In the case of role for $\bar{x}_4$, the policy re-allocated some of the providers in other roles but some others remain (since they still behave according to the expected behaviour for this role). After the second change ($t = 700$), no changes are noticed in the providers, since the effectiveness of the incentives for the convex population remains over time.

In these experiments we have tested two aspects. First, we have shown that our mechanism is able to learn suitable incentive schedules to keep providers to the standards emerged from roles. However, in some cases the method is not able to achieve this because of the providers' dynamics or due to the heterogeneity of the roles. Then we have demonstrated that a re-organisation process is able to re-allocate providers to roles according to their up-to-date skills.

6 Conclusions

In the last few years, a considerable number of research proposals have been made concerning the development of infrastructures for supporting interaction in open multi-agent systems. In this paper, we have built on one such proposal [4] by extending it with the notion of standards, which characterize the roles that agents are able to play. Such standards comprise the focus of two different policies that intend to maximize role stability. Our proposal assumes that a discovered role taxonomy is used to provide a service enabling other providers to find the most suitable performers of specific tasks.

Incentive schedules can be thought of as patches that the system tries to apply in the meantime, as an attempt to keep the current role structure. This approach favours role stability, looking at roles as being part of an overall organization of the population. When such patches are not effective according to some evaluation function, reorganizing the role taxonomy comprises the last resort that the system has at its disposal, with the aim of providing a good agent identification service. One could argue that the proposed incentive mechanism is exploitable by providers once they notice that by intentionally under-performing they are able to get access to incentives that would otherwise not be needed. However, such providers are likely to lose in the long run, as soon as the system decides on reorganizing the role taxonomy.

Other approaches to social control through incentives include the work by Centeno *et al.* [3], who present an approach on adaptive sanction learning by exploring and identifying individuals' inherent preferences without explicit disclosure of information. The mechanism learns which attributes of the system should be modified 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 specific standards; we also assume that incentives are directly inflicted on the agents. An approach based on sanctions is proposed by Lopes Cardoso and Oliveira in [8], which also assumes that sanctions influencing agents' behaviour are directly applied to them, 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 run-time discovered standards, a fixed threshold is used to guide the decisions of the policy maker. Moreover, only sanctions (seen as fines) are employed to have a deterrence effect and discouraging agents from misbehaving, while here we are more interested on incentivizing agents to meet specific targets when executing specific tasks. In a preliminary approach [6], we have put forward an incentive-based mechanism that uses reinforcement learning to find out incentive schedules that influence agents to meet specific targets (much as we do in this paper). However, further assumptions were made in that model, namely that the highest efforts would bring outcomes closer to target values. In this paper, we change the focus of the incentive policy maker: that of targeting the standard, regardless of the efforts that agents must employ. This enables us to experiment with more heterogeneous populations, imposing thus fewer assumptions on the agents' decision models.

Possible applications that we identify for our approach include manufacturing systems, in which agents fulfilling different roles when building a craft are supposed to meet and maintain a standard during their work. Social systems such as ruled electronic markets are another promising domain, in which, as standards may not be known a priori, they can be discovered at runtime and artificially maintained for the sake of the overall market community.

References

1. B. Caillaud and B. Hermalin. Hidden action and incentives. Teaching Notes, U.C. Berkeley, 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. R. Centeno, H. 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, pages 523–530. IEEE Computer Society, 2011.
4. 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*, pages 217–224. IFAAMAS, 2010.
5. R. Hermoso and H. Lopes Cardoso. Dynamic discovery and maintenance of role-based performance standards. In Ossowski, Toni, and Vouros, editors, *Agreement Technologies*, volume 918 of *CEUR Workshop Proceedings*, pages 27–41, 2012.
6. R. Hermoso and H. Lopes Cardoso. Dynamically maintaining standards using incentives. In *Trends in Practical Applications of Agents and Multiagent Systems*, volume 221 of *Advances in Intelligent Systems and Computing*, pages 87–94. Springer International Publishing, 2013.
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. J. Von Neumann and O. Morgenstern. *Theory of Games and Economic Behavior.* Princeton University Press, 3 edition, May 1980.
10. 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. Springer, 2009.