COPYRIGHT NOTICE

Adaptive Coordination in Distributed and Dynamic Agent Organizations

Kathleen Keogh[12] * and Liz Sonenberg[2]

[1] School of Science, Information Technology and Engineering,
The University of Ballarat.
P.O. Box 663 Ballarat. VIC 3353 Australia.
[2] Department of Computing and Information Systems, The University of Melbourne
{k.keogh@ballarat.edu.au}

Abstract. We elaborate the rationale and design of OJAzzIC (Organizations Joining Adaptively with Improvised Coordination), a model for agents in (Jazzy) Organizations that need to engage in dynamic adaptation to respond to a dynamic situation. OJAzzIC provides an adaptive data structure and framework for creation of multiple instances of organizations within a distributed system, with knowledge sharing across organizational boundaries achieved through overlapping instances.

Keywords: Teamwork, Multiagent Systems, Coordination, Adaptation, Organizations

1 Introduction

When working in complex dynamic scenarios such as in emergency management or naval navigation, people adjust their own plans to coordinate and fit in with others in order to achieve goals, rather than following strict scripts, protocols or role descriptions [20, 27]. Indeed, in complex settings it is not possible to consider all alternatives and create a complete plan, rather an incomplete plan is created based on current knowledge and a sequence of incremental problem solving processes is involved in elaborating this plan, whilst actions begin toward fulfilling the plan. Social scientists have studied ways of designing human organizations to support such improvisation, e.g. [25], in settings that involve uncertainty, incomplete knowledge, changing situations and interdependencies across multiple tasks.

Formal predefined organizations can exist based on structured entities, such as an Emergency Rescue Unit, Military Unit or Service Organization. There are also numerous organizations, sometimes termed *adhocracies* [24] that emerge in a dynamic distributed system [32], due to a local problem (shared location) or a coordination need (e.g. resource contention or shared goal). An adhocracy involves multiple groups making decisions in a rapidly changing environment [24,

* Postgraduate student in the Department of Computing and Information Systems, The University of Melbourne

26]. An examination of adaptive organizations found that the complexity and time constraints involved are such that the organization changed via localised adaptive planning and improvisation rather than a broad re-structure at a design level [20].

For similar reasons, it has been recognised that introducing organizational concepts in the design of complex multiagent systems provides capabilities to promote appropriate communication, interaction and coordination, e.g. [18, 19, 38]. In dynamic domains, the interaction and coordination cannot all be prescripted, but must be adaptable at runtime, so that as situations change, the collective of agents can change goals and reallocate tasks or collaborate on tasks in response to availability changes. We are working towards the design of a flexible, coordinated organization-based agent system comprising multiple agents working toward a shared goal. We are not only looking at plan elaboration, the OJAzzIC (Organizations Joining Adaptively with Improvised Coordination) model also supports appropriate knowledge transfer within and across organizations and obligations to ensure relevant knowledge is shared, and is intended to provide a framework to enable coordinated, improvised activity.

The capabilities we expect of our sophisticated agents in order to cope in a complex, uncertain and dynamic environment are based on problem settings with the following characteristics:

- Multiple agents are working with at least one high level shared objective;
- Agents work with individual rationality (self-interest and individual utility function) *as well as* some form of group rationality;
- Interaction, coordination and cooperation between individuals or groups is needed in order to achieve goals with interdependencies;
- Membership of groups is fluid — agents may come and go;
- Roles are not fixed — members are required to improvise in order to achieve goals in a timely way — based on who is available and their capabilities;
- The problem is distributed across multiple locations, central coordination and control is not possible.

These capabilities require a sophistication in terms of agent knowledge, group knowledge and awareness. The OJAzzIC model is based on an organizational approach. Using the structure of an organization, agents have contracts that define how knowledge is shared and held consistent between agents.

We consider an individual agent to have individual mental attitudes such as beliefs, goals, intentions and plans to enact those intentions, and accordingly we build on the traditions of the Beliefs, Desires and Intentions (BDI) architecture for individual agents, extended to group activity, e.g. [38]. We adopt an agent organization as defining a structure for a group of agents with some shared mental attitudes associated with the organization, in addition to individual attitudes. The agent organization is a structured group of agents with definitions of roles defining responsibilitie and relationships between roles, and rules defining obligations on members. For some, an agent organization is a group whose roles and interactions are typically expected to be relatively stable and change slowly over time [29]. Others have introduced terminology of groups, teams, congregations, coalitions etc, and these are variously associated with properties of coordination, correlation, cooperation, and other Co-X words [18, 31], and levels of autonomy between individuals and the group [4, 9, 15, 34]. In this paper we adopt a rather relaxed view, and simply use the term organizations to cover both longer term, stable groupings and those that can be relatively short-lived and changed via localised adaptive planning.

In this paper we target a structure enabling both reallocation of agents to tasks and dynamic goal decomposition and achievement. Previously, we examined SharedPlans [16] in the context of human coordination in the emergency management domain and highlighted that agents need to ensure that they cultivate knowledge about their organizational structure as well as domain knowledge — plans and situation awareness [21]. In future work, we hope to give attention more directly to the management of interdependent resources. These requirements are not unique to the emergency management domain and could be relevant to many emergent situations where agents initially form or enlist in an organization with a common goal, then within that organization, smaller organizational groups form to autonomously work on distributed, but possibly interdependent sub goals. The issue to highlight is that each organization needs to be aware and coordinated within the organization and across any overlapping organization.

We are particularly interested in awareness and coordination of knowledge and behaviour between, across and within organizations. In this paper, we present a high level design for our organizational approach and describe how multiple dynamic instances of organizations are created in order to enable appropriate awareness in the organization. We do not yet address important issues including: policy creation, agent negotiation protocols, resource contention and processes for creation of and disbanding of organizations.

The paper is structured as follows. In the next section, we present key background material: we describe an extension to the traditional agent BDI deliberation cycle to include two levels of agent awareness of others within an organization — i.e. agents not only consider other agents in their own individual deliberations, but agents also deliberate with others in an organization; in subsection 2.1, we describe a scenario involving detection robots to highlight key requirements for our design; and in subsection 2.2 we summarise the main elements of the model OMACS [6–8] on which we build. OJAzzIC builds on the adaptable organizational structure used in OMACS [7] and combines with features from contract based systems [10] and intentional approaches to joint planning [16]. Section 3 contains a discussion of the key ideas of our model, including features, metamodel, and a discussion of goals and roles. In section 4, we briefly cover related work on adaptive agent organizations. We conclude by highlighting future directions of our work.

2 Background material

2.1 Motivating Scenarios

A robotic search for weapons of mass destruction scenario previously used in adaptive agent system design will be outlined and then we will modify this scenario to highlight some of the requirements we are addressing. This scenario was used previously to describe a simple adaptable organization based on OMACS (Organization Model for Adaptive Computational Systems) [7].

The scenario is based on a number of robot sensors that need to search an area and identify suspicious objects. Each suspicious object needs to be checked to see if it is a biological, chemical or radioactive weapon. Not all sensor robots have the capabilities to perform each object identification. When a suspect object is found, all 3 checks need to be performed until one matches. So, each weapon may only be one type of weapon, and the checks may be performed in any order. There are six agent types: Base Robot, Sophisticated Robot, Chemical Robot, Biological Robot, Nuclear Robot and Remover Robot. Both the Chemical Robot and the Sophisticated Robot can identify chemical weapons, but the Sophisticated Robot's chemical detector is not as good as the Chemical Robot's chemical detector. All robots can search and find suspect objects. When a weapon has been successfully identified, the Remover Robot removes it.

We now propose three modified scenarios to highlight more complex requirements for coordination:

Scenario 1. Goals involving multiple agents:
   Example 1a. Suppose the chemical weapon can only be detected successfully by two robots working together simultaneously — Chemical Robot and Sophisticated Robot. The two robots need to coordinate their behaviour to both move to the same object simultaneously in order to perform the detection task.
   Example 1b. Suppose two agent types Base Robot and Chemical Robot, can combine their individual capabilities in order to achieve the removal role capabilities of one Removal Robot agent. In this case, if an agent Removal Robot fails or leaves the scene and another Removal Robot is not available, these two agents could together combine to achieve the tasks that were previously allocated to one agent: Removal Robot.

Scenario 2. Resource contention and interdependencies: Suppose the removal robot agents require an additional resource: a trolley, to help remove the detected weapons. Each removal robot has access to at least one trolley, shared by other removal robots within a close proximity. Sharing of this resource requires that the agents coordinate their use and movement of the trolley. As agents move about, they may need to become aware of 'new' trolleys closer to them.

These modifications highlight issues that we are interested in addressing in the context of agent organizations. We focus on situations where tasks require multiple agents acting in a coordinated way to complete them, and therefore we require organizations of agents who share a goal. Within the organization, agents can work autonomously on some tasks, but must coordinate where necessary. In some cases, there may be a commander or leader role, however sometimes the coordination may be established directly between members of the organization. Agents should be able to dynamically reorganize and reallocate tasks if agents leave or become unable to fulfill responsibilities.

Based on an analysis of human coordination in Emergency Management [21], we seek to address the following requirements for coordination in a dynamic organization:

- *Awareness* All players in the organization are obliged to work with awareness of others, including to be aware of what is relevant to others.
- *Appropriate knowledge sharing* Information, relating to a goal or resource, may flow within and out of one organization into other organization(s) as members identify relevance to other organizations that they belong to, or are aware of.
- *Flexible adjustment of behaviour* In a dynamic situation involving uncertainty, agents adjust their behaviour to fit in with others. The action sequence emerges over time based on the situation and adaptation to address changes as they are realised. Goals may also change.

These requirements apply to complex domains that demand flexibility due to uncertainty, distributed knowledge and interdependencies that must be managed. Agents need to be able to improvise — adjust plans and modify goals to fit in with others. It is not possible to prescribe behaviour exactly for all circumstances at design time so agents need to have access to organizational knowledge to enable reasoning to change individual and organizational goals and plans.

2.2 OMACS

OMACS (Organization Model for Adaptive Computational Systems) has been developed as an organization design based framework model that is capable of adaptation so that a system organization can organize and reorganize itself dynamically at run time. The organization is defined to include the following entities: Goals, Roles, Agents, Domain Model and Policies [6–8]. OMACS addresses some of our requirements for flexible and adaptive organizations and has influenced our design.

In OMACS, Deloach introduces the *Capabilities* abstraction to enable flexible and dynamic reallocation of agents to roles [7]. A Role Model is used to define a list of tasks or responsibilities to be fulfilled. Each Role definition within the Role Model includes a required Capabilities list as well as a function that enables capabilities to be prioritised as to their relative importance. This function enables the measurement of an agent's utility to play each role. It is possible to define multiple alternative roles that are capable of achieving a particular goal, however only one role can be allocated to one goal at any one time. DeLoach and colleagues have proposed that adaptability in planning can be addressed by having alternative paths available in a goal decomposition.

When agents are no longer available and goals cannot be met according to the original goal-role-agent assignment, the system automatically reorganizes and newly revised roles or goals are selected based on the currently available agents' capabilities [7]. However, if there is no agent available with an exact match to the capabilities for a required role, the OMACS system does not address this situation. It does not provide for dynamic flexibility at the level of coordination of multiple agents together performing one role to achieve a goal.

In OMACS, most of the information required for collaboration between roles is embedded in the goals that are instantiated [7]. OMACS is associated with the creation of a dynamic goal design and run time representation using Goal Model for Dynamic Systems (GMoDS) [6]. Using careful goal design according to a goal decomposition tree, goals can be ordered so that goal dependency is represented in the goal model, as well as alternative options, so that if one set of goals cannot be satisfied, an alternative set may be chosen. This enables dynamic and flexible re-allocation of goals. The capabilities abstraction enables flexibility in dynamic allocation and re-allocation of agents to roles based on a changing context. However, coordination between agents is implicit in the plans that individual roles have to enact in order to achieve autonomous goals. OMACS does not provide explicit coordination mechanisms toward agent cooperation on goals or cooperation on synchronizing loosely coupled activities so as not to interfere with other agents. If two agents need to act together concurrently, they would need to each have separate goals for the actions and coordination would be achieved implicitly according to the predefined script or plan for each goal that each agent follows autonomously.

2.3 Extending the BDI agent deliberation cycle

Our approach is based on the popular BDI architecture that enables goal-directed behaviour in the presence of explicit deliberation about changes in the environment, but extended to accommodate reasoning about other agents e.g. [38]. Traditional BDI agents deliberate based on a self-interested cycle, i.e.

```
repeat
        perceived-events := event-selector(event-queue);
        update-attitudes();
        plan-options := option-generator(perceived-events,current-goals);
        selected-plan-options := deliberate(plan-options);
        update-intentions(selected-plan-options);
        execute();
end repeat
```

Table 1. Individual Agent BDI Deliberation Cycle

As Corkill [5] and others argue, agents within an organization, need also to consider organizational objectives in addition to their own goals. Agents must hence ensure that individual mental attitudes are managed so that they are not inconsistent with organizational attitudes. Corkill describes organizationally adept agents that use adjustable preferences to value self, others and the organization to different degrees whilst evaluating utility functions in selecting actions to perform. We propose that this be further extended, so that agents within an organization might not only deliberate individually, but as an organization. Agents therefore would be capable of individual utility based (self-interested) reasoning, other-centred reasoning — with *awareness of* others, as well as organizational reasoning *with* others. In our approach, the obligations to update individual mental attitudes are made explicit in a social contract that defines congruence between organizational attitudes and individual attitudes.

```
repeat
        perceived-events := event-selector(event-queue);
        process-individual-and-organizational-attitudes(current-social-contracts):
        plan-options := option-generator(perceived-events, current-goals);
        selected-plan-options :=
                    deliberate(plan-options, other-agents-and-organizations);
        update-intentions(selected-plan-options);
        execute();
end repeat
```

Table 2. Organizational Agent BDI Deliberation Cycle

In the extended deliberation cycle, an agent perceives environmental input, updates individual mental attitudes but, before selecting an intention the organizational agent will deliberate *with others in the organization.* This organizational deliberation is defined by obligations and policies in the social contract within the organization and is discussed further in Section 3.4. Following the first stage of organizational deliberation and any needed adjustment to individual attitudes, the agent continues the deliberation process *considering others* — when the agent may revise individual intentions and plans to ensure that they are not intending anything that will hamper others and possibly to add new intentions to help others.

3 OJAzzIC: Agents in Organizations Joining Adaptively with Improvised Coordination

In this section, we outline our model, OJAzzIC and how this is used to instantiate a network of multiple coordinated organizations. This model is for Organizations Joining Adaptively with Improvised Coordination (OJAzzIC), the adaptive requirements result in a model that can capture the necessary static and dynamic knowledge in such as way as members can behave as a Jazz musician might — to improvise and adapt their script on the fly, but not in such a way that it would interfere with the script or plan adopted by others. The plans need to be clear, but flexible. Behaviour needs to be coordinated, but not prescribed. OJAzzIC builds on the adaptable organizational structure used in OMACS [7] and combines with features from contract based systems [10] and intentional approaches to joint planning [16].

3.1 Adaptability in design — features of OJAzzIC

The novelty in our approach is to combine the adaptive nature of a dynamic organizational structure—enabling both reallocation of agents to tasks and dynamic goal decomposition— with a dynamic social contract that defines explicit obligations and coordination policies on the fly. The social contract may be based on a predefined script that can be well defined or loosely governed by predefined landmarks [40]. Additionally, our proposal for use of multiple *instances* of overlapping organizations in a dynamic way enables adaptable coordination within meta organizations.

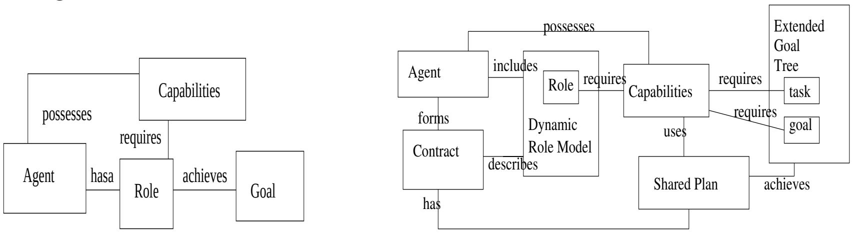
An organizational instance is created dynamically whenever a complex problem arises that requires some coordination over time. This coordination may involve an emergent plan that needs revision when further information becomes available. It may also require coordination of members in terms of a shared resource or goals that require multiple members to coordinate activity dynamically in order to achieve the goal. It may also be that some initiative will be required in terms of using members outside of their usual role descriptions where they have capabilities toward helping achieve a goal.

Key to the approach of OJAzzIC is the creation, as required, of dynamic instances of organizations. Within each organizational instance, context specific dynamic contracts define agent allocations, obligations and roles. These ensure that coordination, knowledge sharing and behavioural obligations can be dynamically defined within a particular network (organization) of agents. Multiple organization instances may be created within the original organization. Each organization has at least one goal that requires coordination. Each organizational instance (including the high level organization) has a set of dynamic organization attributes with values available to all members. These attributes include an organizational structure (role model including dynamic role definitions and coordination roles), set of agents, goal tree, domain beliefs, resource list, fixed domain policies and role definitions and a dynamic contract. The contract defines the allocation of agents to tasks/coordination roles and dynamic coordination and knowledge-sharing policies to ensure consistent beliefs are maintained within the organization.

In order to address our requirements and establish an organizational design with the flexibility to adapt, the following major decisions were made: The model would include agentified organizations as first class entities [38]; Agent-Role-Task mapping using Capabilities would be used to enable flexible automated reallocation [7, 36]; Goals could be shared by multiple roles using Capabilities and Tasks; and Organization instances created would include social contracts to define coordination obligations dynamically [40].

Agentifying the organization means we can treat the organization as one agent, with mental attitudes that can then be semantically related to individuals in the organization as desired [38]. This also means that no one individual needs to stay in a particular role (e.g. Leader) for appropriate communication with the organization. The organization is addressable as an agent in its own right [28].

The organization is a static predefined structure that may be instantiated by an actual organization instance that is created at run time. This organization instance may be quite a stable and permanent structure based on formal roles, but it may also be a short term organization created so that a group of agents can work together in a coordinated way. In the latter case, the coordination may be negotiated dynamically rather than be based on predefined scripts. In the former case, the coordination may be based on default scripts, though these can still be adapted. All organization instances are considered to be dynamic, first class, agentified entities and from here on, we shall refer to these as organizations. The organizational contract is part of the organization's knowledge, so accessible to all agent members.



(a) Agent-Goal Abstraction in OMACS

(b) Agent-Goal Abstraction in OJAzzIC

Fig. 1. Comparison of Agent to Goal Relationships in OMACS and OJAzzIC

Figure 1 shows an abstract high level view of the relationships in an OJAzzIC organization alongside a partial view of similar components in an OMACS organization [7]. Figure 1(b) is expanded upon in Figure 2. In Figure 1(a) Agents are related to Goals using the Role abstraction. As discussed in section 3.3, this assumes that one role will achieve one goal. We introduce an extra level of separation between roles and goals. In OJAzzIC, the Goal Tree is extended to include Tasks as a possible decomposition of Goals. Agents can be allocated based on responsibilities for Roles or based on Capabilities to fulfil Tasks. Extending Goal Trees using Tasks will be discussed further in section 3.3.

The Role Model in OMACS is a fixed relationship of predefined roles (represented as 'Role' in Figure 1(a)) whilst in OJAzzIC (Figure 1(b)), the Role Model is dynamic, it is created based on context and represents the roles instantiated by agents in the organization. The Contract is an explicit mental attitude adopted by the organization and defines obligations regarding knowledge sharing and will be discussed further in section 3.4. Figure 1(b) is a simplified conceptual model, more detail is presented in Figure 2. Our design is motivated by reality. In real situations, a role description may change based on context and roles might need to be shared. In OJAzzIC, the dynamic Role Model enables goals to be shared between multiple roles and where necessary coordination roles are created. The structure of the organization — role relationships and role definitions — are dynamically defined.

We are not alone in proposing the need for shared mental models. Commitments toward maintaining and proactively sharing information in teamwork has been addressed in similar work with agent/human teams [41]. In order to establish information relevancy, explicit information-needs graphs have been used along with explicit mental models of team structure, team processes, and domain knowledge [12]. Information flow within groups has been described in terms of the relationships that form in a coordination loop [32]. In our case, we propose that encouraging information sharing in each such network requiring coordination is possible if each is considered a dynamic instance of an organization. Within each organizational instance, obligations exist ensuring appropriate knowledge sharing. Each organizational instance has a shared goal — such as a knowlege seeking goal or a goal to manage a dependency, or a goal to achieve a particular set of tasks. Agents may belong to multiple organizational instances simultaneously. In this way, an agent's knowledge can be shared across organizational boundaries where it is relevant to more than one organizational group.

3.2 OJAzzIC Organization Model

Figure 2 shows the OJAzzIC organizational model. Each organizational instance is created following this model. Our organization entity is loosely based on OMACS [7], with extensions to provide for more flexible and dynamic goal/role sharing. Where we adopt OMACS concepts without extension, we do not provide details here, but direct the reader to details elsewhere [8].

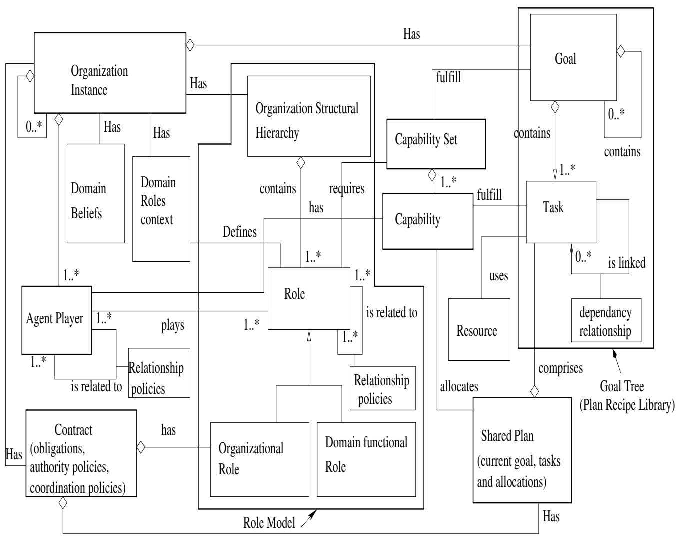


Fig. 2. The OJAzzIC organization model

To achieve goals the organization may require more than one agent to coordinate behaviour for related tasks. Goals are described in terms of tasks and tasks require capabilities. A player has capabilities, enacts a role and is also allocated tasks based on that role. Plans are based on instantiating general default plans or creating dynamic plans based on the decomposed Goal Tree, according to the particular context. Organizational plans to achieve goals are established dynamically using SharedPlans [16].

Agents start by belonging to a large organization responsible for the main high level goal (e.g. the entire system). When two or more players need to coordinate, a new organization is formed and within that organization an explicit contract is formed that dictates policies, obligations, agreed goals and agreement to coordinate with others in that organization. Coordination within the organization relies on appropriate communication to share relevant information and share plans. The new organizations that form overlap with existing organizations. We propose that in a dynamic organization, multiple smaller organizations are created. These organizations each need to be explicit so that appropriate coordination can be established within each. Each organizational instance created would be based on the OJAzzIC organizational entity structure.

In OJAzzIC, an organization O is a tuple:
$<G*, R, Re, Contract, A, C, P, \sum, \beta,$ *oaf, achieves, requires, possesses* $>$

This extends OMACS with G*, R, Re and Contract. These are explained in more detail in the subsequent sections.

G*: extended Goal Tree, including ordered tasks where possible. This defines the goals and how they could be decomposed into sub goals and ordered tasks;

R: the Role Model including a set of Roles, relationships between Roles and context based Role Definitions. This also includes coordination roles created dynamically as necessary;

Re: a dynamic Resources list defining objects in the environment that can be used to help perform tasks;

Contract: The dynamic Contract contains a social contract and an information contract. The social contract comprises a SharedPlan [16] and a set of coordination Roles agreed for the organization. The Shared Plan outlines the current selection of tasks to achieve the goal and the allocations thus far assigning responsibilities for tasks. The contract in OJAzzIC replaces ϕ in OMACS (ϕ is a relation over G x R x A providing goal/role/agent assignments). The information contract is a set of agreed policy obligations and commitments to intentions to ensure consistency of beliefs within the agent organization. The information contract contains β the current Beliefs set that includes beliefs about the environment, including resources.

A: set of Agents;

C: set of Capabilities;

P: fixed policy constraints to apply to all members and to the allocation of tasks;

$\sum$: domain model used to specify environment objects and relationships

In OMACS, policies are abstractly used to define the processes for allocation of agents to roles (Assignment Policies), define behavioural obligations and relations between roles (Behavioural Policies) and define structural reorganizational processes such as how to reallocate tasks (Reorganization Policies). OMACS defines policies that must be held as *Law* policies and policies that can be prioritised and hold when possible as *Guidance* policies. OMACS also defines additional supporting functions *oaf, achieves, requires, possesses.* The definition of function achieves has been extended to include tasks and SharedPlans *oaf*: organization assignment function measures the utility of a particular SharedPlan assignment of Agents to Roles to Tasks;

achieves: function defining role assignment — how effective the behaviour of roles can be to achieve task T or goal G in a SharedPlan. This is used if a default plan needs revision or if a default plan cannot be found for a context. The Goal-Tree can be used to derive a plan.

requires: defines the capabilities required to play a role R or task T; and

possesses: function defining the quality of an agent's capability for a particular Task. To decide how well an agent can play a role, the requires and possesses functions are combined into a function: capable. To decide how well an agent can play a role to achieve a goal, the capable function and achieves function are combined as a function: potential.

Based on our requirements, in OJAzzIC, we incorporate additional *Authority* Policies defining a process for explicit acceptance of allocations by an agent as well as *Coordination* Policies to help resolve multi-agent plan coordination dynamically.

3.3 Goal Trees and Dynamic Role Model

In order to achieve our aim of flexibility, we choose to keep separate the goals of an organization and the available roles that may be used to define (or allocated) responsibility to achieve these goals.

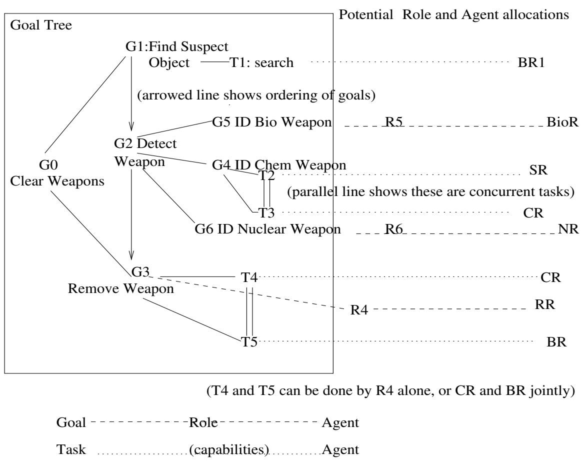


Fig. 3. OJAzzIC Goal Tree showing potential allocations with Players

Synchronized Tasks If a goal cannot be achieved by one role, then multiple roles or agents can combine to achieve a goal by working together. We describe goals as composing synchronised tasks and tasks can be performed by agents with the appropriate capabilities. This abstraction is introduced to enable flexible and dynamic planning by agents to establish a coordinated SharedPlan to work together to achieve a goal. We choose to split goals into separate synchronised tasks rather than split roles, as intuitively, this abstraction fits with our observations from real life examples in the Emergency Management domain.

In figure 3 an example OJAzzIC Goal Tree for the sensor case from section 2.1 is shown, as defined at design time. The Goal Tree encapsulates knowledge about goal decompositions and in this case, we have indicated on the right side how the goal tree could be expanded at run time to link to capabilities of particular agents and generate potential plans. The Goal Tree can be thought of as a plan recipe library. Default or preferred plans can be defined by indicating preferred paths at design time, however the flexibility to use the Goal Tree dynamically allows for dynamic planning and revision of plans.

As figure 3 shows, a goal may comprise multiple sub goals. Goals may also be decomposed into tasks. Tasks and goals can be ordered. This abstraction is to enable the splitting of goals to share between multiple players. When a goal is split, performing the tasks requires coordination between the players. In figure 3, goal G0 is decomposed into sub goals G1, G2 and G3. G1 must be performed before G2. G1 is described in terms of search task T1. In some cases, one goal can be achieved by one role directly, as with G3 achieved directly by R4. In the absence of an agent allocated to Role R4 for whatever reason, an alternative is to split goal G3 into two separate tasks T4 and T5. These tasks then can be allocated to individual agents based on individual capabilities. When goals are split and shared then the agents need to coordinate their behaviour using a SharedPlan. In figure 3, different dashed lines indicate whether an agent is allocated to perform a role that directly satisfies a goal (big dashes) or whether agents are allocated individual tasks (small dotted line). In the latter case, the SharedPlan will ensure coordination between the agents. This figure shows potential allocations of players (sensor robots) to these goals and tasks at run time. BR1 has the capabilities to perform task T1 and in this example has been allocated that task. Player RR has the capabilities to fulfil role R4 and thus could be allocated in that role to perform both tasks T4 and T5. Alternatively, CR could perform T4 and BR could perform T5. Tasks T4 and T5 must be done simultaneously — indicated by the parallel lines connecting them.

Using the sensor agent case as a very simple example, we can identify that if allocations were made based on Figure 3 with CR and SR working together to achieve the goal G4 Detect Chemical Weapon, then CR and SR would need to coordinate and create a SharedPlan. This is a simple instance of an organization. Creating an organization would then obligate each agent (as defined in the social contract) to appropriately share information such as a SharedPlan to facilitate coordination between these agents. There could be initially an organization involving BR1 and perhaps some other agents conducting a search of the area. Then as suspect objects are found, new organizations could spring up including agents able to detect weapons. For example, an organization with BioR, SR, CR, NR could form with BR1 as a leader to achieve the goal, G2 Detect Weapon. Then when a weapon has been identified, a new organization involving these or other agents would form to remove the weapon. We could imagine a more complex scenario where there were hundreds of agents involved in the weapons search over a large area. Multiple organizations could be created dynamically as agents elect to work together to achieve goals. When one agent is part of multiple organizations, this overlap allows for relevant information (e.g. location, detection status of object) to be propagated across the network across organizational boundaries.

Resources are explicit entities in the OJAzzIC organization's knowledge base. This is because of the potential for resource contention amongst agents in the organization and thus the need to coordinate interdependencies. Having this knowledge explicit will enable direct reasoning within the Agent Organization, based on priorities, negotiated protocols or the use of coordination artifacts [30, 1]. We do not address such reasoning further in this paper.

The sharing of relevant knowledge within and across organizations is important as knowledge may be distributed. Obligations to ensure appropriate knowledge-sharing about SharedPlans [16] has been well defined and we adopt this intention based approach to planning. We have adopted the use of contracts to manage obligations to ensure agents will share domain-knowledge as well as structural and coordination knowledge within each organization. Collective obligations can be implemented as policies to govern joint activity and teamwork [39]. We leave implementation details aside and in the next section, describe at a conceptual level the contents of the contracts that need to be created.

3.4 Contracts

As agents join (or apply to join) an organization, then agents must agree (commit) to a social contract that defines interaction within the organization. Beliefs, values, objectives, protocols and policies may be defined in the context of the social relationships that exist within the society. At the time a new organization is instantiated in OJAzzIC, players in each organization, explicitly form an organizational contract. Each organization exists for the duration of time in which there is a need for that group of agents to be coordinated. The organizational contract comprises a social contract that defines the social structure of the organization and an information contract that defines how information is shared within the organization.

The *social contract* defines role descriptions and agreed role allocations. Role descriptions may be abstractly defined at organizational design time and adapted dynamically. Roles are defined with associated capabilities, authority levels and obligations. These are made explicit in the social contract to enable dynamic and adaptive revisions. Having an explicit social contract provides the abilitiy to predict others' behaviour and flexibly adapt individual goals in anticipation of others' needs and behaviour. The social contract also defines an agreed model for command, control and coordination. Coordination Roles such as Leader, Resource Manager, Knowledge Manager and Contract Manager are identified and allocated if needed.

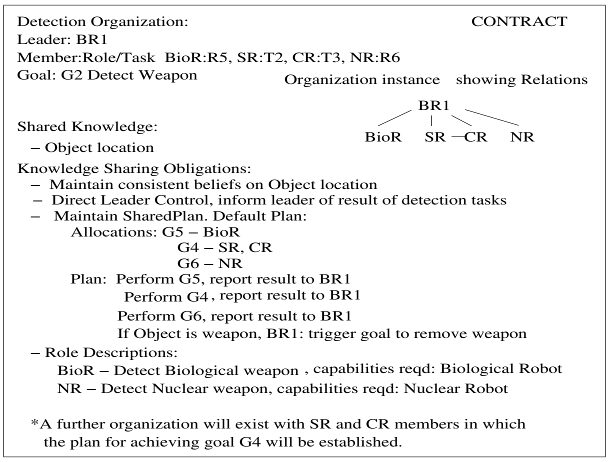


Fig. 4. Partial contract for the detection team in the sensor robot case scenario

Based on the need to cultivate knowledge sharing [21], *information contracts* include policies that obligate members in an organization to all adopt joint intentions to cultivate mutual knowledge within the organization. Obligations to share information are limited to the agents within each individual organization that forms. As an agent can belong to multiple organizations, the overlap enables relevant information to be dispersed across a wider network as necessary. Figure 4 shows parts of an example contract for the organization with goal G2: Detect Weapon, in the sensor robot case study.

4 Related work

We have looked particularly at the following adaptive organizational models for agents: OMACS [7], KB-ORG [36], and OperA [11]. Each addresses part of our requirements. We also consider adaptivity in relationships achieved by associating Agents to Roles and Goals as discussed, for example, in previous work by Ferber [13] and Odell [28].

In these proposed metamodels for Agents, Roles and Groups [13, 28], within a group context, agents are associated with an agent role to determine the sorts of activities in which the agent may participate. The interactions between agents are governed by the roles played by the agents. An agentified group can then communicate, take on a role and act as an agent. Roles enable a layer of abstraction between agents and their allocated tasks facilitating re-allocation and reorganization of agent groups. Tidhar has similarly defined an organization as a set of related teams as a first class (agentified) entity [38].

We have looked for flexibility in our goal design and have extended the OMACS definition, so that a goal might be broken into synchronized/ordered tasks that could be assigned to agents. This approach has also been adopted in AGR [13] and in the MOISE system [14] where a distinction is made between a separate structural specification and a functional specification. In MOISE-Inst [19], goals are decomposed into missions, then allocated to a set of responsible agents. The goal tree specifies potential tasks that can be associated with individual plan recipes that achieve each leaf goal [18, 36].

A related approach has been promoted in the KB-ORG system, designed for automatic allocation of tasks to agents in a dynamic organization [36]. In KB-ORG, roles contain an assignable list of responsibilities and if necessary, roles can be split between a set of agents and explicit coordination roles are created. KB-ORG differs from our approach in that we are attempting to create a design for organizationally aware agents, rather than using an external management system. Importantly, in order to establish dynamic coordination by agents in organizations, explicit coordination roles are needed. In KB-ORG, coordination roles are created when an application level role is split between a set of agents. Similar coordination roles are adopted in the role model and agreed in a social contract in OJAzzIC.

We are not unique in articulating agent interactions as requirements and modelling these separately in the design process for an agent system [33]. Others have described interactions as part of an organizational design [2, 14]. Relationships and awareness of relationships between agents in a dynamic organization are important to enable the appropriate coordination and communication.

Functional specification and decomposition of tasks in MAS using a goal-tree to specify tasks with synchronization or coordination relations is not unique. It is found in models including for example STEAM [37] and TAEMS [23]. High level guidelines have been used to describe constraints on how organizational objectives should be decomposed in a hierarchy. Separately, operational objectives represented as leaf goals in their goal decomposition can be operationally coordinated as required by the individuals involved (not at an organizational level) [5, 36]. This abstraction to 'leave the details' to the smaller groups is similar to ours, although we make the distinction that these smaller groups may be considered as temporary organizations rather than teams. The value in creating short to medium term organizations, is that for the duration of the organization, obligations and some infrastructure including shared mental attitudes, can be used to help ensure that our complex, dynamic, coordination requirements may be addressed. A separate approach is to use Petri Net models to monitor and coordinate hierarchical team plans [3].

Coordination by proxies or intermediate layers within an agent architecture has been suggested to enable open systems with heterogenous agents to work together. For example, Scerri *et al* assign a proxy agent responsible for coordination to each team player [35]. This enables domain specialised agents to work as part of a larger team, without the need for knowledge about the team itself. However, with this approach, the agent players are not able to directly reason about team issues or coordination and this limits its applicability in our context.

5 Conclusion

We have provided an elaborated description of a model for agent organizations requiring adaptability and improvisation. By giving agents access to organizational information that they can change, we allow agents to adjust their own attitudes to fit in others in a changing situation. We have taken features from existing systems [7, 11, 16, 36] and extended these to meet our requirements.

We have organizations as first class entities so membership does not need to be fixed and members have access to the mental state of the organization. Organizations are created based on a need to coordinate actions or resources. Within organizations, agents are obliged to share information and maintain consistent plans. We have contract based dynamic organizations so that organizational structure is defined/agreed and modifiable at run time. We have a flexible goal-task decomposition that enables definition of concurrent tasks and goals that require multiple agents. We allow for goals to be shared, where multiple agents can combine their capabilities to collaborate to achieve a goal. We propose that this model be implemented so that multiple organizations are created, as needed. As the OJAzzIC system is an organization of organizations, reasoning at different levels of abstraction is possible.Each organization manages obligations to ensure relevant knowledge is shared within and between organizations.

Future work is needed to formalise this design and validate it with an implementation. We hope to test it with a multi-agent organization implementation based on the modifications to the search for weapons discussed early in this paper [7]. Ultimately, our intent is to use our model in organizations involving agents and humans, c.f. [41].

Agents may leave or join an organization at any time. Upon creation or joining of an organization, agents agree to obligations within the organizational contract. In particular these obligations guide communication and shared intentions to keep mutual knowledge consistent between members. Agents may have individual utility functions for private goals as well as global utility functions at an organizational level. Agents may also use initiative when they have capabilities outside of their designated role to locally adapt the organizational structure to assist by performing tasks in the interest of the organization. These policies and obligations need to be formally specified in future work.

OJAzzIC organizations would be suited to work in highly dynamic, complex domains that require flexible adaptive interactive behaviour. These could include Emergency Management systems, Naval management coordination and to some extent military command and control (when local decision making occurs separate from the formal vertical command hierarchy). Where short term coordinated tasks are to be performed by a group of agents and are well specified, not likely to change during execution of a plan, an organizational structure is not necessary. In these cases, agents might form a different less structured collective such as a group with a SharedPlan [17].

Multiagent systems enabled with characteristics to work with humans have potential benefits in simulation and training. Sophisticated agents could potentially be used as surrogate humans in virtual organizations for training exercises.

To be useful, such agents need to behave in a predictable and believable way similar to humans [22] and need to be designed to coordinate behaviour with other players. Our work contributes by proposing a conceptual design for dynamic and adaptable agent organizations working together in a coordinated way.

Acknowledgements The authors thank Dr Gil Tidhar for his conversations and suggestions regarding OJAzzIC. Also, thanks to the participants in the COIN workshop at AAMAS 2011 for their feedback and to the anonymous reviewers for their constructive suggestions.

References

1. D. Acay, G. Tidhar, and L. Sonenberg. Extending agent capabilities: Tools vs. agents. In *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 2, pages 259–265, 2008.
2. E. Argente, V. Julian, and V. Botti. MAS modeling based on organizations. In *Proceedings of Agent-Oriented Software Engineering workshop IX, 2008, LNCS*, volume 5386, pages 16–30. Springer-Verlag, 2009.
3. O. Bonnet-Torrès and C. Tessier. A Formal Petri Net Based Model for Team Monitoring. In V. Dignum, editor, *Handbook of Research on Multi-Agent Systems: Semantics and Dynamics of Organizational Models*, pages 568–589. IGI Global. Information Science Reference, 2009.
4. C. Brooks and E. Durfee. Congregation formation in multiagent systems. *Journal of Autonomous Agents and Multiagent Systems*, 7:145–170, 2002.
5. D. Corkill, E. Durfee, V. Lesser, H. Zafar, and C. Zhang. Organizationally adept agents. In *Proceedings of COIN Coordination Organization Institutions and Norms workshop AAMAS 2011*, volume in press, 2011.
6. S. DeLoach and M. Miller. A goal model for adaptive complex systems. *International Journal of Computational Intelligence: theory and Practise*, 5(2), 2010.
7. S. A. DeLoach. OMACS: a framework for adaptive, complex systems. In V. Dignum, editor, *Multi-Agent Systems: Semantics and Dynamics of Organizational Models*. IGI Global: Hershey, PA. ISBN: 1-60566-256-9, 2009.
8. S. A. DeLoach and G.-O. J. Carlos. O-MaSE: A customizable approach to designing and building complex, adaptive multiagent systems. *International Journal Agent Oriented Software Engineering*, 4(3), 2010.
9. V. Dignum. *A Model for Organizational Interaction: based on Agents, founded in Logic*. PhD Thesis, Universiteit Utrecht, 2004.
10. V. Dignum, J. Meyer, and H. Weigand. Towards an organizational model for agent societies using contracts. In *Proceedings of AAMAS02, First International Joint Conference on Autonomous Agents and Multi-agent Systems, Bologna, Italy*, 2002.
11. V. Dignum, J. Vázquez-Salceda, and F. Dignum. OMNI: Introducing social structure, norms and ontologies into agent organizations. In *Programming Multi-Agent Systems: Second International Workshop ProMAS, 2004*, volume 3346, pages 181–198. Springer, 2005.
12. X. Fan, S. Sun, B. Sun, G. Airy, M. McNeese, and J. Yen. Collaborative RPD-enabled agents assisting the three-block challenge in command and control in complex and urban terrain. In *Proceedings of 2005 BRIMS Conference Behavior Representation in Modeling and Simulation*, pages 113 – 123, Universal City. CA, 2005.

13. J. Ferber, O. Gutknecht, and F. Michel. From agents to organizations: An organizational view of multi-agent systems. In *Agent-Oriented Software Engineering (AOSE) IV, 2003*, volume 2935, pages 214–230, July 2004.
14. B. Gâteau, D. Khadraoui, E. Dubois, and O. Boissier. MOISEInst: An organizational model for specifying rights and duties of autonomous agents. In *Proceedings of Third European Workshop on Multi-Agent Systems (EUMAS 2005)*, pages 484–485. Elsevier Science B.V., 2005.
15. N. Griffiths. Supporting cooperation through clans, cybernetic intelligence - challenges and advances. In *Proceedings IEEE Systems, Man and Cybernetics, 2nd UK&RI Chapter Conference*, 2003.
16. B. Grosz and L. Hunsberger. The dynamics of intention in collaborative activity. *Cognitive Systems Research*, 7(2-3):259–272, 2006.
17. B. Grosz and S. Kraus. The evolution of shared plans. In *Foundations and Theories of Rational Agency*, pages 227–262, 1999.
18. B. Horling and V. Lesser. A survey of multi-agent organizational paradigms. *The Knowledge Engineering Review*, 19(4):281–316, 2005.
19. J. Hübner, R. Kitio, and A. Ricci. Instrumenting multi-agent organisations with organisational artifacts and agents 'giving the organisational power back to the agents'. *Autonomous Agents and Multi-Agent Systems*, 20:369–400, 2010.
20. E. Hutchins. Organizing work by adaption. *Organization Science*, 2(1):14–39, 1991.
21. K. Keogh, E. Sonenberg, and W. Smith. Coordination in adaptive organisations: Extending shared plans with knowledge cultivation. In *Post conference proceedings-Organised Adapatation in Multi-Agent Systems (OAMAS08), LNAI*, volume 5368, pages 90–107. Springer, 2009.
22. G. Klein, P. Feltovich, J. Bradshaw, and D. D. Woods. Common ground and coordination in joint activity. In W. Rouse and K. Boff, editors, *Organizational Simulation*, pages 139–184, New York, 2004. Wiley.
23. V. Lesser, K. Decker, T. Wagner, N. Carver, A. Garvey, B. Horling, D. Neiman, R. Podorozhny, M. N. Prasad, A. Raja, R. Vincent, P. Zuan, and X. O. Zhang. Evolution of the GPGP/TAEMS domain-independent coordination framework. *Autonomous Agents and Multi-Agent Systems*, 9:87–143, 2004.
24. D. Mendonca, T. Jefferson, and J. Harrald. Emergent interoperability: Collaborative adhocracies and mix and match technologies in emergency management. *Communications of the ACM*, 50(3):45–49, 2007.
25. D. Mendonca and W. A. Wallace. A cognitive model of improvisation in emergency management. *IEEE Systems, Man and Cybernetics: Part A*, 37(4):547 – 561, 2007.
26. H. Mintzberg. Structure in 5's: A synthesis of the research on organization design. *Management Science*, 26(3):322–341, 1980.
27. H. Mintzberg. *Structure in Fives: Designing Effective Organizations*. Prentice-Hall, New Jersey, USA, 1983.
28. J. Odell, M. Nodine, and R. Levy. A metamodel for agents, roles and groups. In *Proceedings of Agent-Oriented Software Engineering (AOSE) V*, volume 3382, pages 78–92. Springer, 2005.
29. J. Odell, H. Van Dyke Parunak, and M. Fleischer. The role of roles in designing effective agent organizations. In *Software Engineering for Large-Scale Multi-Agent Systems*, volume 2603, pages 27–38. Springer, Berlin,, 2003.
30. A. Omicini, A. Ricci, and M. Viroli. Coordination artifacts as first-class abstractions for MAS engineering: State of the research. In A. F. Garcia, R. Choren, C. J. P. de Lucena, P. Giorgini, T. Holvoet, and A. B. Romanovsky, editors,

Software Engineering for Multi-Agent Systems IV, Research Issues and Practical Applications [the book is a result of SELMAS 2005], volume 3914, pages 71–90. Springer-Verlag, 2006.

31. H. Parunak, S. Brueckner, M. Fleischer, and J. Odell. A preliminary taxonomy of multi-agent interactions. In *Proceedings of The Second International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS) 2003*, pages 1090–1091, Melbourne, Australia, 2003.
32. B. Prue, M. Voshell, D. Woods, J. Peffer, J. Tittle, and W. Elm. Synchronised coordination loops: A model for assessing distributed teamwork. In *Proceedings of NATO Research and Technology Organisation (RTO) Human Factors and Medicine Panel (HFM) symposium. Adaptability in Coalition Teamwork*, pages 17.1–17.12, April 2008.
33. I. Rahwan, T. Juan, and L. Sterling. Integrating social modelling and agent interaction through goal-oriented analysis. *Computer Systems Science and Engineering*, 3:87–98, 2006.
34. T. Sandholm and V. R. Lesser. Coalitions among computationally bounded agents. *Artificial Intelligence*, 94:99–137, 1997.
35. P. Scerri, D. Pynadath, N. Schurr, A. Farinelli, S. Gandhe, and M. Tambe. Team oriented programming and proxy agents: The next generation. In M. Dastani, J. Dix, and E. Falah-Seghrouchni, editors, *Programming Multi-Agent Systems: First International Workshop ProMAS, 2003*, volume 3067, pages 131–148. Springer-Verlag, 2004.
36. M. Sims, D. Corkill, and V. Lesser. Automated organization design for multi-agent systems. *Autonomous Agents and Mult-Agent Systems*, 16:151–185, 2008.
37. M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7:83–124, 1997.
38. G. Tidhar. *Organization-Oriented Systems: Theory and Practice*. PhD Thesis, Department of Computer Science, University of Melbourne, 1999.
39. J. van Diggelen, J. Bradshaw, M. Johnson, A. Uszok, and P. Feltovich. Implementing collective obligations in human-agent teams using KAoS policies. In *Proceedings of Workshop on Coordination, Organization, Institutions and Norms (COIN), Eighth International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, Budapest, Hungary, 2009.
40. H. Weigand, V. Dignum, J.-J. Meyer, and F. Dignum. Specification by refinement and agreement: Designing agent interaction using landmarks and contracts. In P. Petta, R. Tolksdorf, and F. Zambonelli, editors, *Third International Workshop on Engineering Societies in the Agents World (ESAW2002), LNAI*, volume 2577, pages 257–269. Springer-Verlag, 2003.
41. J. Yen, X. Fan, S. Sun, T. Hanratty, and J. Dumer. Agents with shared mental models for enhancing team decision-makings. *Journal of Decision Support Systems: Special issue on Intelligence and Security Informatics*, 41(2), 2005.