Methods and models for the design and study of dynamic agent organizations
Ghijsen, M.

Citation for published version (APA):
Ghijsen, M. (2014). Methods and models for the design and study of dynamic agent organizations
SELECTING COORDINATION MECHANISMS

In this chapter we take a closer look at the decision for selecting coordination mechanisms as described by the AgentCoRe framework in Chapter 4. In this chapter we take two approaches to the study of selecting coordination mechanisms. First, we introduce an abstract task environment in which agents have to cooperate to achieve their goals. These agents are capable of using two coordination mechanisms, a centralized and a decentralized mechanism. We show how the decision making framework is operationalized in this abstract task environment. We compare the performance of two static organizations with an organization in which agents have the ability to switch between coordination mechanisms. The second approach is a comparison of the coordination mechanism selection mechanism with three well known approaches to the dynamic selection of coordination mechanisms found in the literature. We analyze each of these approaches and demonstrate that each of these approaches can be explained in terms of our model.

A short version of this chapter has been published in “Proceedings of the 2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, pages 87-91” (Ghijsen, Jansweijer, and Wielinga, 2008a).

6.1 INTRODUCTION

The organizational design of a Multi-Agent System (MAS) together with the task-environment in which the MAS is embedded, are two main aspects which determine its performance (So and Durfee, 1998). In a dynamic environment, a MAS may encounter new situations where its organizational design is no longer the most effective. Possible negative effects caused by dynamics in the task-environment can be mitigated or reduced by continuously adapting the organization of a MAS to these changes in the task-environment (Carley, 1995).

The organization of a MAS consists of many aspects such as relations and interaction patterns between agents (Carley and Gasser, 1999), agent roles (Zambonelli et al., 2000), and coordination mechanism (Jennings, 1996) and thus a MAS organization can be changed in each of these aspects. We view the coordination mechanism as the main organizational aspect of a MAS and therefore the focus in this chapter is on dynamic adaptation of coordination mechanisms in a MAS. Moreover, research by Excelente-Toledo and Jennings (2004), Martin and Barber (2006), Rosenfeld et al. (2008) has shown that enabling agents to dynamically select a coordination mechanism yields better performance compared to agents using a fixed coordination mechanism.
In Excelente-Toledo and Jennings (2004), agents are enabled to dynamically select the coordination mechanism for a collaborative task. However these agents cannot dynamically switch between coordination mechanisms because the agents stick to the selected mechanism for the duration of the task. In Martin and Barber (2006) this issue is addressed and agents are given the capability to switch between coordination mechanisms while executing a single task. However the problem of how to determine the costs of the available coordination mechanisms in complex domains remains unsolved in both approaches. In Rosenfeld et al. (2008) this issue is addressed by using a heuristic approach for determining the expected costs of coordination mechanisms.

Existing research on dynamic selection of coordination mechanisms, as discussed in the previous paragraph, has in common that it does not show the actual decision making process where agents decide whether to switch, which coordination mechanism to use, and how to switch to a different coordination mechanism. Therefore we address this issue in this chapter by presenting such a process and identify the domain independent elements in that process. By separating domain independent and domain dependent knowledge, we increase understanding of the dynamic selection of coordination mechanisms and enable a MAS designer to focus on solving only domain specific issues. Our research is motivated in a similar fashion as research on knowledge engineering (Schreiber et al., 2000) where the first expert systems were analyzed for generic problem solving principles (Clancey, 1985).

In the next section we present our domain independent decision making framework for selecting coordination mechanisms. In order to demonstrate how domain dependent knowledge can be operationalized, Section 6.3 describes an environment in which the agents have the ability to perform two different coordination mechanisms. In Section 6.4 we show the operationalization of the domain dependent aspects of the decision making framework in this environment. Section 6.5 presents an experiment in which we show the effects of applying dynamic selection of coordination mechanisms. In Section 6.6 we argue that our decision making framework is also applicable to previous research by Excelente-Toledo and Jennings (2004), Martin and Barber (2006) and Rosenfeld et al. (2008) on the dynamic selection of coordination mechanisms. We end this chapter with our conclusions and directions for future work in Section 6.7.

6.2 DECISION MAKING

A brief explanation of the decision making process for selecting a coordination mechanism has already been given in Chapter 4 of this thesis. In this section we will elaborate on the four elementary reasoning processes that constitute the decision making process. Figure 48 shows the inference structure for the selection of coordination mechanisms. This inference structure consists of the following four reasoning processes; (1) whether to initiate the decision
6.2 Decision Making

making process by selecting a possible alternative coordination mechanism from the strategy library, (2) which parameters to use to compare two different coordination mechanisms, (3) which coordination mechanism is the best (i.e. has the lowest costs) by comparing the two coordination mechanisms, and in the case the current coordination mechanism is no longer the most suitable, (4) how to replace the current coordination mechanism in the organization of the MAS by another coordination mechanism.

6.2.1 Initiate decision making

The first decision to be made by an agent is whether to initiate the decision making process by selecting a possible alternative coordination mechanism that is stored in the strategy library. This decision depends on information obtained from the agent’s world model including the role of the agent in the organization of a MAS, the environment in which the agent is embedded and the agents view on that environment.

The strategy for this decision making step depends on the type of organization in which it takes place. In a centralized organization, the decision to switch the organization or a part of the organization to another coordination mechanism can be made by a single agent that has the authority to make such decisions. In other, more decentralized organizations, agents may decide to propose a change of coordination mechanism. In this case, the outcome of the agent’s decision still needs to be accepted by the other agents in the organization. This is subject of the final step of our decision making framework.

A likely factor to be used as input for the strategy to initiate change is the occurrence of certain events in the environment that may require a change in coordination mechanism. Examples of such events are changes in the communication infrastructures (e.g. communication failures or changes in available bandwidth), changes in the tasks that need to be performed. Different coordination mechanisms might be available for different types of tasks or different amounts of tasks.

Another factor that can initiate the change to another coordination mechanism is a change in the organization itself. The size of the organization may
change or agents might obtain or loose certain capabilities which influence the
types of coordination mechanisms that the organization might be capable of.

Ultimately, the quality of information gathered by an agent that is used in
subsequent steps in the decision making process will influence the outcome of
each of these steps. In general, more accurate information will lead to better
decisions and therefore, an agent should only continue the decision procedure
when the quality of information is sufficient.

6.2.2 Selection Criteria

The second decision in the decision making process is the selection of the crite-
ria that are used to compare coordination mechanisms. This step is included to
enable agents to be flexible in the way coordination mechanisms are compared.
This need for flexibility, also recognized by Excelente-Toledo and Jennings
(2004), is motivated by the fact that in some situations we want to select a
coordination mechanism that is fast while in other situations we would want a
coordination mechanism with a high quality result. To provide more insight in
the possible selection criteria, we propose a set of domain independent criteria
that are shown below in set \( C \). Although these selection criteria show on which
basis cost calculations can be made, their operationalization remains a domain
dependent issue.

\[
C = \{ \text{time-to-goal-achievement, solution-quality,}
\text{communication-costs, probability-of-success,}
\text{resource-consumption} \} 
\] (1)

The time-to-goal-achievement criterion captures performance measures
that are related to the amount of time consumed by agents until a goal is
achieved. Examples are the time that is needed for the setup of a coordination
mechanism, or how a coordination mechanism influences the speed at which
agents exchange information or coordinate their actions.

Solution quality criteria focus on the result when a goal is achieved. Such
criteria are especially useful in domains where the amount of time is limited
and the performance of the MAS is measured in terms of the quality of work
that is achieved within that time. A reward received by agents is an often used
operationalization since the size of the reward is usually correlated with the
quality of the work by the agents.

A communication-costs criterion is useful as a selection criteria for mini-
mizing the amount of communication needed by coordination mechanisms
when communication resources are scarce or when costs are involved for using
a communication resource.

Not all coordination mechanisms provide agents with a good chance of
actually reaching a certain goal. This may depend on the type of goal. For
example, a coordination mechanism may be suitable only for a specific set
of tasks. The probability-of-success criterion can be used in a trade-off with other selection criteria such as solution-quality, where the chance of achieving a goal may be low but in the case that goal is achieved, the quality of work is very good.

In some domains such as robotics, the amount of resources consumed while coordinating determines the suitability of a coordination mechanism. Fuel consumption, CPU cycles and amount of memory needed are all operationalizations of the resource-consumption criterion.

This decision corresponds to the select parameters decisions in the Agent-CoRe inference structure for strategy selection. The available parameters and the rules for selecting certain parameters are stored as strategy rules in the inference structure.

6.2.3 Cost calculation

Once an agent has decided on the selection criteria to be used, the agent can compare the coordination mechanisms based on their costs. Calculation of the costs of a coordination mechanism can be a difficult problem and it depends on the domain in which this takes place. As mentioned by Lesser (1998) “purely symbolic reasoning about costs and benefits” [of coordination mechanisms] “can be extremely complex, particularly in large systems and open environments, or where agents can simultaneously pursue multiple goals.” A solution for this problem is to adopt a heuristic approach as done by Rosenfeld et al. (2008).

Multiple selection criteria can be combined by using a weighted sum of the criteria or by using a rule based approach, e.g. if the value of a criterion α rises above a threshold, use the value of that criterion, else use the value of criterion β.

6.2.4 Change procedure

When an agent decides to switch to a different coordination mechanism, the agent starts a change procedure to replace one coordination mechanism by the other. This procedure is stored in the change rules library and describes how the transition from one coordination mechanism to the other should take place.

If the agent that initiates the change procedure has authority over the agents that should adopt this coordination mechanism, the change to another coordination mechanism can be achieved by sending a command to switch to the other agents. Because the authority relation exists, the other agents will accept this decision and switch to the other coordination mechanism.

In cases where the decision to switch cannot be enforced by authority, the change procedure should contain a mechanism to establish agreement.
on the new coordination mechanism (Excelente-Toledo and Jennings, 2004),
(Barber, Martin, and McKay, 2001). In this case, the decision to switch is sent
to other agents as a proposal on which other agents can respond. When all
agents agree on what the new coordination mechanism will be, some further
organizational adaptation may be required. A new coordination mechanism
may impose changes in agent roles, interaction patterns, or agent relations
and the change procedure should include a mechanism to transform the
multi-agent organization into the desired state.

In other cases, when agents have a high degree of autonomy and are loosely
coupled. The decision of an agent to switch to another coordination mechanism
might not even affect other agents. In such cases, the decision to switch does
not require any cooperation from other agents and no change procedure is
needed.

6.3 ABSTRACT TASK ENVIRONMENT

Typical coordination problems occur when agents cannot achieve their goals
alone because a task requires effort from multiple agents or when resources
that are needed for performing tasks are scarce (Malone and Crowston, 1994).

The abstract task environment (ATE) is a simulation environment with
discrete time in which a set of agents $A$ of size $n$, have to cooperate to perform
a set of tasks $B$ with size $m$. In the initial stage of the simulation, each agent $a_i$
is randomly assigned a to a subset $B_i \subset B$ such that the set of all subsets $B_i$
is a partition of $B$. Agents do not know the tasks of the other agents nor the
size of $B$. The goal of the agents is to complete all tasks in $B$ as fast as possible.
A task $b_{j,w}$ (the $j$th task in $B$) can only be completed if $w$ agents perform an
action $p(i,j)$ (action performed by agent $a_i$ on task $b_j$) at the same time step.
Each agent can only perform one action per time step. It is assumed that
a task always succeeds when the right number of agents execute that task
simultaneously.

In the ATE, agents communicate via a single communication channel. Each
time step only one agent can use this communication channel to broadcast a
message to all other agents on the channel. We assume agents can “see” all
other agents on the communication channel so they know the recipients of
their message. Sending and receiving a message occurs in a single time step,
so there is no delay in communication. The bandwidth of the communication
channel is limited.

Dynamics in the simulator are introduced by breaking the communication
channel in one or more places at a random time during a simulation. The
occurrence of such a split event results in two or more fully separate commu-
nication channels and effectively breaks the MAS into two or more smaller
groups of agents. The basic characteristics of a communication channel remain
unchanged by a split event. The bandwidth will remain the same and the
assumption that agents can “see” the other agents on the communication
channels still holds. However, after a split event agents can no longer “see” the agents that are on other channels. Furthermore, for each communication channel, only one agent is allowed to send a message per time step.

6.3.1 Coordination

To enable agents to share information about the world and coordinate their actions we introduce four types of messages that can be exchanged between agents:

1. The allocate time slots message is used by an agent to coordinate the use of the communication channel by indicating when agents are allowed to communicate. By using this message, agents are able to avoid the situation that multiple agents want to use the communication channel at the same time and information getting lost. The message consists of a list of tuples \( <t,i> \) to indicate that agent \( a_i \) is allowed to send a message at time \( t \). For example, \( [<t2,agent14>, <t3,agent25>] \) means that at \( t2 \) agent14 is allowed to use the communication channel and at \( t3 \) agent25 is allowed to communicate.

2. The gather task info message contains a list of uncompleted tasks that can be used by an agent to indicate which tasks are still uncompleted. Each uncompleted task is represented by a tuple \( <j,w> \) where \( j \) is the task id and \( w \) is the workload of the task expressed by the number of agents required to perform the task. For example, \( [<job2,2>, <job5,3>, <job6,4>] \) means that \( job2 \) requires 2 agents to perform, \( job3 \) requires 3 agents and \( job6 \) requires 4 agents.

3. The allocate tasks message is a plan message and it can be used by an agent to tell other agents when they should perform a certain task. A plan message contains of a number of plan elements represented by a tuple \( <t,A,j> \) which indicates that at time \( t \), the agents in set \( A \) should perform task \( j \). Furthermore a plan message also contains the overall workload of an agent represented by the number of uncompleted tasks the agent is responsible for. By providing this information, other agents are able to use this information to determine the overall workload. For example the plan message \( <8, [<t1,[agent1,agent2],job2>, <t1,[agent4,agent5,agent6],job5>] \) means that an agent still has 8 uncompleted tasks and at time \( t1 \), agent1 and agent2 should perform task \( job2 \). Also at time \( t1 \), agent4, agent5 and agent6 should perform task \( job5 \).

4. The switch message is used to tell the recipients of this message to switch to another coordination mechanism. This message consists of a tuple \( <\text{switch},cm> \) that contains the switch command and the name
of the coordination mechanism (cm) that will be switched to. For example, \texttt{<switch,decentral>} tells the agents to switch to a coordination mechanism called decentral.

The ATE poses two coordination challenges; communication coordination and task coordination. Communication resources are scarce (one communication channel with limited access and limited bandwidth) which makes coordination of this resource necessary. This problem is similar to communication coordination problems in other domains such as the RoboCupRescue competition (Kitano et al., 1999) where bandwidth is limited and the number of messages that can be received by agents per time step is also limited.

Furthermore in the ATE, each agent can only execute one task each time step and executing a task only succeeds if it is performed by \( w \) agents at the same time step. This requires coordination of which task should be performed when and by which agents. Coordinating actions is a problem which occurs in many domains, for example the predator-prey domain (Alcazar and Garcia, 2006) where predator agents have to coordinate their movements to capture a prey.

Next, we present two coordination mechanisms for the ATE. The first can be characterized as a centralized approach based on authority and the second as a decentralized approach based on social conventions.

### 6.3.1.1 Centralized Coordination

The centralized coordination mechanism is a command driven approach in which one agent acts as a central coordinator and the others act as subordinate agents. The basic task of the coordinator agent is to gather information about the state of the world and issue plans to the subordinate agents. The agent with the lowest id on the communication channel will always be the coordinator agent and all other agents also know who their manager is.

![Interaction pattern for the centralized coordination mechanism.](image)

The interaction pattern for the centralized coordination mechanism is shown in Figure 49. It starts with the coordinator agent who determines who can...
use the communication channel by allocating time slots to the agents. When a simulation starts, the coordinator agent communicates the time slots that tell the other agents when to send their local task information. If the available bandwidth is high, this information can be communicated in a single broadcast message. If the bandwidth is small, multiple smaller messages may be required to communicate this information. When all information is received, the coordinator agent starts coordinating tasks. After this, each agent will take one time step to communicate their local task information. Once the coordinator agent has received this information, the coordinator agent will start sending plan messages. In each time step, the coordinator sends a plan message that prescribes which tasks have to be performed in that time step. The coordinator makes sure that the plan elements do not interfere (i.e. no agents have to execute more than one task in a single time step) and that as much tasks are executed as possible.

6.3.1.2 Decentralized Coordination

![Figure 50: Interaction pattern for the decentralized coordination mechanism.](image)

In the decentralized coordination mechanism each agent is fully autonomous in its decision making. Coordination of communication and tasks is achieved by social convention. This convention prescribes that each agent in turn is allowed to communicate its plans for the current time step and to make the decision to switch. The time at which agent $a_i$ is allowed to do so is when $i = t \mod n$ where $i$ is the agent ID, $t$ is the current time and $n$ is the number of agents. This approach ensures that agents will not simultaneously try to use the communication channel and that their plans will not interfere. The interaction pattern for this coordination mechanism is shown in Figure 50.

6.3.1.3 Coordinating Dynamics

In the case of a split event, each group of agents continues to use the same coordination mechanism as they used before the split event.
6.4 DOMAIN DEPENDENT DECISIONS

In Section 6.2 we presented a decision making process for the dynamic selection of coordination mechanisms. In this section we demonstrate how we operationalize these decisions in the ATE.

6.4.1 Initiate decision making

The first step is for the agent to decide whether to initiate the decision making process. In the ATE, this is different for the two coordination mechanisms. In the centralized case, an agent will only initiate further decision making when it has the central coordinator role. In the decentralized case, an agent will initiate decision making when the agent has an allocated time slot.

In order to prevent taking wrong decisions based on insufficient information, the agents will not initiate the process until they know how many tasks have to be performed. In the centralized approach, this is the case when the central coordinator has received the complete task lists of the other agents. In the case of decentralized coordination, this is the case when each agent has communicated a plan message containing its workload $l_i$.

6.4.2 Selection criteria

Next, the agent selects the criteria it will use to calculate the costs of a coordination mechanism. In the ATE we use one selection criterion; the time-to-goal-achievement criterion which is operationalized as the time to execute all remaining tasks.

6.4.3 Cost calculation

As mentioned before, cost calculation of coordination mechanisms in general is a difficult problem to solve. The ATE is fully deterministic (except for the occurrence of split events and the initial distribution of tasks) and the coordination mechanisms are also deterministic when the number of agents needed to perform a task is known a priori (for the experiments we have set $w = 3$ for each $b_{j,w}$). Now, agents are able to compute the costs of the coordination mechanisms for each state of the environment. This also assumes that agents have an accurate view on the world which is ensured by the first step in the decision making process.

6.4.4 Change procedure

In the final step of the decision making process, agents perform the change procedure to switch to the new coordination mechanism. To enable the agents
to perform the actual switch between coordination mechanisms the switch message is used. We assume this message requires only a bandwidth of one, so this message can always be sent in a single time step. Sending a switch message adds extra costs of one time step to the remaining execution times calculated in the previous section.

When agents receive such a switch message they know that from the next time step on, the coordination mechanism (cm) specified in that message will be used. Although in the generic case, some mechanism is needed for the agents to agree on whether to switch or not, we assume that in this case all agents are fully cooperative. Thus, when one agent decides to switch, the other agents will accept this decision and also switch to a different coordination mechanism. Furthermore, when agents receive a switch message, each agent will update its own model of the organization of the MAS. If the new coordination mechanism is the centralized mechanism, each agent decides whether it will take the central coordinator role or an operator role (the agent with the lowest id will be the central coordinator). Then, authority relations are created between the central coordinator and the operator agents. The coordinator role prescribes the behavior as described in 6.3.1.1 and the subordinate role prescribes that the agent will only act when ordered by the central coordinator. If the new coordination mechanism is the decentralized mechanism, the agent will remove all authority relations with other agents and it will assume an autonomous role. This role prescribes the behavior as described in Section 6.3.1.2.

6.5 experiment

In this section we present an experiment to demonstrate the effect of the decision making process described in the previous sections. In this experiment we compare the performance of three MAS organizations in the ATE. In the first organization the agents only use the centralized coordination mechanism. In the second organization the agents only use the decentralized coordination mechanism. In the third organization the agents initially use the decentralized mechanism but when the simulation starts they are able to switch between the centralized and decentralized coordination mechanisms.

In the experiment, each simulation starts with a group of 60 agents which is split into 3 groups of $n = 20$ at some time during the simulation. The timing of the split event was varied between $t = 20$ and $t = 50$ with a 10 step interval. Results of the experiment are obtained by running 1000 simulations for each $t$ for each of the three versions and measuring the time it took for the agents to complete all tasks. The average score and standard deviation for each $t$ is shown in Figure 51.

Our first observation in Figure 51 is that the performance of the static centralized organization decreases linearly as the split event occurs later in the simulation. This is because the first phase in the centralized interaction pattern is the collection of world information by the coordinator agent. This
Figure 51: Simulation results; average time to perform all tasks for three different organizations.

Stage will take at least 60 time steps because each agent needs to send its world view to the manager agent. Whenever a split event occurs during this phase, the organization will be split into 3 separate static centralized organizations. Again, these organizations first have to go through the phase of collecting world information. For these smaller organizations, each consisting of 20 agents, this will take at least 20 time steps before they can start performing tasks. Therefore, the later the split event takes place, the time needed to complete all tasks increases. Because the overall workload and the size of the organization is the same for all simulations, the time needed to perform all tasks is also the same for each organization. The very small variation that is seen in the data is caused by the unbalance in workload between the agents. This causes some agents to take longer than one time step to communicate their task information to the coordinator agent.

Our second observation in Figure 51 is that there is only a small influence of the split events on the performance of the static decentralized organization. This influence is small (as compared to the centralized organization) because the operational activities of the organization are not interrupted by a split event. This is because the communication needed to coordinate task execution and the task execution itself all takes place in the same time step. When a split event occurs it is actually beneficial for the performance of the decentralized organization because one single large organization is now split into three.
smaller organizations that are working in parallel. This explains why the performance of the decentralized organization is better when the split event occurs early in the simulation. We also see that the standard deviation in the data for the decentralized organization is larger than for the static centralized organization. This is because the performance of the decentralized organization, due to its communication scheme, is more sensitive to unbalance in workload between agents. If an agent has a large workload that requires more than one time step to coordinate the execution of its tasks, the agent has to wait for its turn to communicate again for the second time.

Finally we observe in Figure 51 that the dynamic organization outperforms the two static organizations. We should note that although the observed difference between the dynamic organization and the static decentralized organization is small, the overall difference for all \( t \) is statistically significant with \( p < 0.01 \). This is because the agents are now able to select the most effective coordination mechanism depending on the organization size, the remaining workload for each agent and the available communication bandwidth. In the dynamic organization, the agents start using the decentralized coordination mechanism but once they have acquired sufficient information to make an informed decision to switch, they can switch to the centralized mechanism if this is faster. As we saw from the results of the static centralized organization, the weak point of the centralized mechanism is that it takes time for the coordinator agent to gather all information needed to make a plan for task execution. The task execution phase is then very fast because the coordinator agent has all the information to make an optimal plan. The information that is exchanged when using the decentralized mechanism can also be used by a coordinator agent to make a plan for centralized task execution.

Concluding, we circumvent the time consuming information gathering phase of the centralized coordination mechanism by using the decentralized mechanism when starting a simulation. The disadvantage of the decentralized mechanism is that it is sensitive to unbalanced workload causing the agents to wait for their turn to communicate and coordinate the execution of tasks. In those cases, it is beneficial for the agents to switch to the centralized mechanism once sufficient information has been gathered to perform all tasks.

### 6.6 Application to Other Approaches

In this Section we discuss existing approaches on dynamic selection of coordination mechanisms. The goal of this discussion is to discover to which degree our decision making framework is applicable to other approaches. Therefore we provide a description of these other approaches in terms of our decision making framework. Some aspects of these approaches can be directly described in terms of one of the four steps in our decision making framework. However, other decisions of our framework are not always described explicitly in these approaches. In these cases we have re-interpreted these decisions by

In this Section we discuss existing approaches on dynamic selection of coordination mechanisms. The goal of this discussion is to discover to which degree our decision making framework is applicable to other approaches. Therefore we provide a description of these other approaches in terms of our decision making framework. Some aspects of these approaches can be directly described in terms of one of the four steps in our decision making framework. However, other decisions of our framework are not always described explicitly in these approaches. In these cases we have re-interpreted these decisions by
In this approach, Excelente-Toledo and Jennings (2004) use agents that are able to dynamically select a coordination mechanism for each collaborative task they encounter. When an agent decides to adopt this task as its goal it will send out a proposal to other agents to work on this task. The application domain is an abstract environment in which agents walk around in a grid world and encounter collaborative tasks for which they need the help of other agents to perform these tasks. The 4 domain dependent decisions are the following:

1. Each agent that encounters a new collaborative task is allowed to make a decision to switch between coordination mechanisms.

2. The selection criteria are the setup time of a coordination mechanism, the probability of success, and the expected reward after the collaborative task is finished. These are operationalizations of the time-to-goal-achievement, probability-of-success and solution-quality criteria respectively.

3. Based on the setup time and the probability of success, the expected bids of other agents are calculated and deducted from the reward that will be given to the agent in charge when the task is completed.

4. Because selection of a coordination mechanism takes place when a collaborative task is encountered the change procedure is not really about changing to a different coordination mechanism. Instead the change procedure describes how the agents agree on the coordination mechanism to use for a collaborative task. The agent sends a proposal to other agents and then waits for bids from other agents. It then accepts the best bids.

In this research, Martin and Barber (2006) give agents the capability to switch between coordination mechanisms. Based on different levels of autonomy of agents, several different coordination mechanisms are constructed. Four different levels of autonomy are defined: local autonomous, master, true consensus, and command driven. The application is a radar domain where agents have to coordinate to minimize radar interference.

1. All agents that have the capability to adapt the coordination mechanism are allowed to make the decision to switch to a new coordination mechanism. However agents that are command driven will not make this
decision. All other levels of autonomy are allowed to make the decision to switch.

2. Only one selection criterion is used, which is the normalized average signal interference of the radars. This selection criterion is an operationalization of the solution-quality criterion.

3. Cost calculation in this radar domain is done by calculating the interference level for each state in the state space and for each coordination mechanism.

4. A negotiation protocol is used to reach agreement on the new coordination mechanism to be used (Barber et al., 2001). The coordination mechanisms are defined in terms of the levels of autonomy of agents so a change in coordination mechanism automatically results in a change of autonomy level.

6.6.3 Rosenfeld, Kaminka, Kraus and Shehory

In this research by Rosenfeld et al. (2008) robots try to avoid collisions while performing search and foraging tasks. Each robot has several types of coordination mechanisms to its disposal to coordinate collision avoidance with other robots.

1. Each robot is allowed to make decisions whether to switch to another coordination mechanism.

2. Selection criteria are fuel consumption (an operationalization of resource-consumption) and/or time needed to complete the task of the robots (which is an operationalization of time-to-goal-achievement).

3. The costs of the coordination mechanisms are determined heuristically by either estimating the costs of coordination mechanisms or estimating a parameter within one type of coordination mechanism which results in different variations of that type of coordination mechanism.

4. When a robot has decided to switch to a different coordination mechanism no communication with other agents is required because the coordination mechanism is determined only by each robots own behavior.

6.7 Conclusions

In this chapter we have elaborated on our decision making framework that enables agents to dynamically select the most appropriate coordination mechanism in a given situation. This has resulted in factors that influence the strategy
for initiating the decision making process, a number of domain independent
selection criteria to determine the performance of coordination mechanisms,
and a number of mechanisms to transfer from one coordination mechanism to
the other.

The applicability of the framework was demonstrated by its operationaliza-
tion and implementation in an abstract task environment. Experiments in this
environment show that the dynamic selection of coordination mechanisms
enables agents to exploit the advantages and avoid the disadvantages of two
coordination mechanisms and increase their performance.

In this chapter we not only use our decision making framework as a basis for
the design of a dynamic MAS organizations. We also show in this chapter that
this framework can be used as a basis for a structured analysis and comparison
of different approaches to this problem. In general, a comparison of these
different approaches is difficult because they all focus on different aspects
of selecting coordination mechanisms and authors might not even be aware
of some of the important design decisions they make. By using the decision
making framework we identify these major design decisions and separate the
sometimes complex algorithms into different steps to make them easier to
understand and compare.