Effective and Efficient Coordination Strategies for Agile Crisis Response Organizations

J.B. van Veelen
DECIS Lab
bernard.vanveelen@decis.nl

P.P.A. Storms
Y'All
patrick@yall.nl

C.J. van Aart
Y'All
chris@yall.nl

ABSTRACT

Agile crisis response organizations can be seen as actor-agent communities, where artificial coordination strategies are applied to manage activities. This paper provides a classification of artificial coordination strategies, specified in terms of the Rasmussen's three-level model for supervisory control: skill, rule and knowledge based. Three distinct strategies to artificial coordination based on Rasmussen's levels are described. These approaches are applied in a small case study related to the problem of medic-casualty allocation in the crisis response domain. In terms of effectiveness and efficiency, the knowledge-level coordination strategies seem to be the most effective, where the skill-level strategies are the most efficient. Concerning flexibility there is a reverse trade-off with efficiency. Opposed to skill-level strategies, knowledge-level strategies easily adjust to changing operational requirements. On all aspects, the performance of rule-level strategies is in-between knowledge-level and skill-level strategies.
The results of this work can be used to improve the performance and effectiveness of actor-agent communities for mission critical applications.

Keywords

Coordination, Autonomous Systems, Actor-Agent Communities, Self-Management, Multi Agent-based System, Quality of Service, Crisis Response Organization and Management.

1. INTRODUCTION

Crisis response organizations are shifting from hierarchical, regionally organized structures to dynamic assemblies of resources. In a strategic vision document (NIBRA, 2002), The Dutch NIBRA stresses that modern information and communication technology enables the dynamic creation of ad-hoc networked organizations. By forming agile organizations tailored to the situation, resources are used more efficiently (Harrald, 2005). These agile organizations allow sharing of information on different levels and between different disciplines, which increases *situational awareness* (Endsley, 2000) and effectiveness.

Similar to this trend is the shift from local to network centric organizations in the military (Alberts, Garstka and Stein, 1999; Krygiel, 1999). Both the crisis response and the military domains set similar requirements to organizations and infrastructure regarding information-sharing and flexibility. In both domains an evolution from rigid hierarchical to agile ad-hoc organizations takes place. Similar problems have to be faced in both cases, regarding avoidance of information overload and loss of organizational awareness (Oomes, 2004) and overcoming cultural differences. For background reading on agile organizations we refer to (Atkinson and Moffat, 2005).

This work focuses on the process of coordinating resources in a networked organization. Coordination is "*managing interdependencies between activities, enabling all resources to work together harmoniously in achieving a common goal*" (Malone and Crowston, 1990). In a fixed organizational structure, coordination is accomplished by predetermined roles, responsibilities and procedures. In an agile organization coordination has to be tailored to changing operational requirements. The agility gained in a networked organization comes at the cost of an increased coordination effort, especially when the coordination is performed by people. To prevent loss of efficiency there is a need for automated approaches to coordination.

The field of Distributed Artificial Intelligence (DAI) has extensively studied coordination for distributed problem solving systems (Hannebauer, 2002; Jennings, 1996; Malone and Crowston, 1994; Omicini, Zambonelli, Klusch and

Tolksdorf, 2001). A crisis response organization is typically a hybrid distributed problem-solving system, containing multiple actors and agents engaged in multiple tasks. In this paper we study the applicability of emerging new technologies from the DAI domain, to establish effective and efficient coordination in agile crisis response organizations.

This paper is organized as follows. Section 2 introduces the topic of coordination. In section 3 three approaches to artificial coordination are discussed. Next, section 4 illustrates how these approaches can be applied in a crisis response context, using a case study. Section 5 gives a qualitative evaluation of the coordination strategies and their applicability to the crisis response domain. Finally, section 6 states conclusions and future work.

2. BACKGROUND

2.1. Crisis response organizations viewed as actor-agent communities

Agile crisis response organizations can be seen as a collection of actors that are linked together in a network by means of intelligent agents. Agents are automated entities (machines, software processes) that have some degree of autonomy in pursuing a set of goals (Wooldridge, 2002). Agents are able to perceive their environment and respond to changes by adjusting themselves or manipulating their environment. These actions are either reactive or proactive.

Agents can interact with other agents, humans and non-agent systems to offer their services, represent others (*agency*) and cooperate to achieve a set of goals. These goals can be common or conflicting. The term *multi-agent system* (MAS) denotes a network of interacting agents. Hybrid networks of people and agents are called *actor-agent communities* (AAC). In AAC the social and technical are treated as inseparable and any actor, whether person, object (including computer software, hardware, and technical standards), or organization, is equally important to a social network. As such, societal order is an effect caused by the smooth running of AAC. For background reading on the underlying theory, we refer to the work of (Tatnall and Gilding, 1999) on information systems and actor-network theory,

We envision agile crisis response organizations as AAC in which actors and agents communicate as peers. The interests of actors are reflected by goals of agents. Through the agent of an actor, the actor's information, services and capabilities can be accessed by other actors. The agents manage resource availability and actor access policies. Standardization in communication shields agents from each other's specific technical and behavioral characteristics.

2.2. Coordination of actor-agent communities

A loose definition of coordination is: *the act of working together harmoniously* (Malone and Crowston, 1990). If we apply this definition to an AAC, the act of 'working together' implies that agents perform interdependent activities, which are part of a larger plan. The predicate 'harmoniously' implies that emerging conflicts are resolved. This requires some kind of management process (van Aart, 2005; Malone and Crowston, 1990). We refer to this management process as *coordination strategy*. Coordination is an essential activity in distributed systems. It permits participants to perform complex composite tasks and achieve (common) goals by interaction (Corkill and Lander, 1998).

There are various approaches to coordination in AAC, each with its own properties and characteristics. Based on criteria described in literature (Jennings, 1996; Ossowski, 1999; Willmott, 2002), we distinguish four major dimensions to classify coordination strategies:

Implicit versus explicit

> In implicit (communication-less) coordination strategies, there is little or no explicit inter-agent communication related to coordination. Either agreement on coordination is shared by all participants in an AAC, or agents operate under local sensing and control. In the latter case system-level behavior emerges from indirect interaction between participants through the environment. An example of such a mechanism is stigmergy which was originally used to explain the flow of information in social insects and is currently utilized to explain the implicit coordination in multi-agent systems (Bonabeau, Dorigo and Theraulaz, 1999; Mason, 2002).
>
> In explicit (communication-extensive) coordination strategies, participants explicitly communicate to coordinate. Explicit coordination has many forms, such as market-based and organization-based approaches (Chevaleyre, Dunne, Endriss, Lang, Lemare, Maudet, Padget, Phelps, Rodruez-Aguilar and Sousa, 2005; Giorgini, Kolp and Mylopoulos, 2005; Jennings, Faratin, Lomuscio, Parsons, Sierra and Wooldridge, 2001).

Dynamic versus static

Dynamic coordination allows altering the coordination strategy at runtime. This can be done by either fine tuning the configuration of a specific strategy, or replacing the current coordination strategy by another.

With static coordination, the coordination strategy is determined and configured *a-priori*, for instance at design time.

Cooperative versus competitive

In a cooperative AAC the participants join together to carry out an activity of mutual benefit. Individual preferences or goals are of secondary priority.

In a competitive AAC participants are self-interested and will primarily pursue their individual goals. The overall performance of the group is of secondary interest. The goal of a coordination strategy in a competitive environment is to ‘persuade’ agents to cooperate in performing a task, by satisfying individual goals or preferences. In a competitive AAC agents may have conflicting goals and might not be willing to share all information.

Centralized versus decentralized

In an AAC where coordination is centralized, a separate group of participants can be distinguished that is occupied with handling coordination. All the other participants are only allowed to inform the central coordination mechanism about their state and obey its instructions. Note that centralized coordination can also be implemented as a fully distributed system.

In an AAC with decentralized coordination each agent has the capability to coordinate, as well as functional (problem solving) capabilities.

2.3. Positioning of coordination strategies

To classify coordination strategies we introduce an additional characteristic, based on Rasmussen's three-level model of human thinking in supervisory control (Rasmussen, 1983).

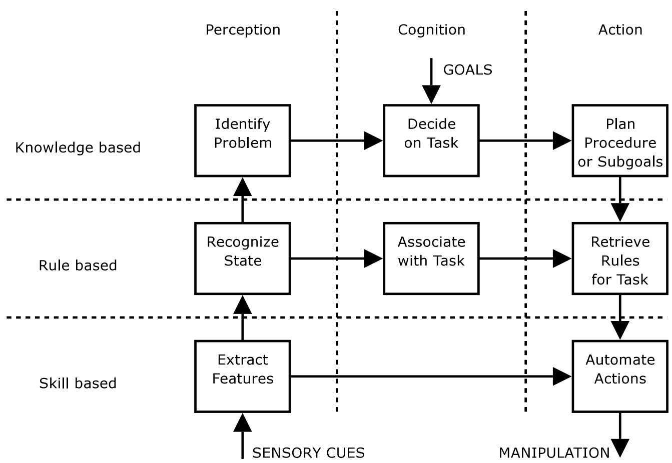

Figure 1: Rasmussen's three-level model

This model integrates stimulus-response (skill based), rule-based and first-principles (knowledge based) decision making, see Figure 1. According to Rasmussen, human decision makers try to minimize cognitive effort because this is difficult and takes time. They prefer reasoning at the skill-based level, where there are fast and direct relations between stimuli and responses. Skills are learned by repetition, training and experience. If skill-based reasoning fails, decision makers will apply rule-based reasoning to tackle the problem. This requires more cognitive effort than skill-based reasoning, but is still fast. Rules can be seen as standardized procedures, resulting from cognitive analysis. If rule-based reasoning fails, the decision maker reverts to the knowledge-based level. Here, the problem is analyzed 'from scratch', which takes the most cognitive effort. Solutions obtained from knowledge-based reasoning can be captured in procedures, to be reused at the rule-based level if a similar problem occurs in the future.

Rasmussen's model is useful for understanding human decision-making, but also forms a basis for designing decision support systems (Sheridan, 1988). Coordination is a decision making process and as such we use this model to classify coordination strategies. But instead of human cognitive effort we use the required computational effort and domain knowledge as indicator to classify an approach.

3. APPROACHES TO ARTIFICIAL COORDINATION

In this section we discuss three approaches for allocating resources in actor-agent systems. We discuss a knowledge based reasoning approach called SMDS (3.1), a negotiation-based approach (3.2) and a greedy approach (3.3). Each approach must be viewed as a *class* of coordination strategies, since there are many possible specific implementations for each approach. Strategies that belong to the same class share the same properties. The properties of each approach are expressed in terms of the four dimensions (2.2) and the position in Rasmussen's model (2.3). We choose these particular approaches for two reasons. First, each of these approaches has been proven to result in effective and efficient coordination for different types of problems. Second, the approaches have very distinct properties, together covering nearly the full spectrum of coordination strategies.

3.1. Knowledge based coordination: SMDS

The first class we consider is coordination by Self Managing Distributed Systems (SMDS). The SMDS concept, developed by European defense industry, is aimed at providing multi-platform middleware technology featuring autonomous and dynamic configuration capabilities in network centric systems.

An SMDS-based system is a distributed system comprised of agents that have managerial responsibility for a system under control. We make a distinction between SMDS agents that perform coordination tasks (planning, delegation, monitoring, etc.) and the agents that are controlled by the SMDS agents.

Within SMDS, managerial responsibilities are grouped in segments of similar functionality. Tasks that need to be performed by the AAC are presented to a *planning segment*. The planning segment creates a global plan for task allocation. The plan is inferred from a *distributed knowledge based system* (KBS) containing knowledge on all available actors and agents, their capabilities, properties and interrelations. The *instantiation segment* takes care of contracting the right actors and agents that have to perform subtasks in the global plan. Next, a *monitoring segment* is responsible for guarding plan execution and quality of service management. Finally, a *federation segment* deals with multi-party collaboration of systems.

Since we can clearly distinguish between agents that are responsible for coordination and agents that are coordinated, SMDS is a *centralistic* approach to coordination. Coordination is *explicit*, since the agents in the planning segment send out ‘orders’ and configuration messages to subordinate agents. A prerequisite for SMDS is that all agents in the system respect the authority structure and will be honest about their capabilities and operational status. Therefore, SMDS-based coordination is suitable for *cooperative environments*. Coordination is *static* since SMDS does not change the coordination strategy at run-time. Because SMDS continuously infers task allocation plans, using the facts on actors and agents stored in the KBS, coordination is positioned at the *knowledge-level* of Rasmussen's model.

3.2. Rule based coordination: negotiation

The second class concerns coordination strategies that mimic real-world market mechanisms. These strategies exploit the competitive behavior of agents to optimize some global utility function. In a virtual market *buying agents* may request or place bids for a common set of objects such as services, information or access to resources. *Selling agents* or *auctioneer agents* are responsible for processing bids and determining the winner (Lesser and Horling, 2005). Allocation of objects to agents is either facilitated by a central auctioneer agent or a by a sequence of (bi-lateral) local negotiations between buyers and sellers.

Market-based coordination mechanisms borrow concepts from economics and trade, such as the notion of *auctions* and *negotiation*. Various auction-based mechanisms exist that vary along a number of dimensions. An alternative approach for allocation of resources in a competitive environment is by local bi-lateral negotiations. Typically, this involves a series of exchanges of messages. Each participant assigns a value (utility) to the set of resources it owns, and interacts with a peer to see if a deal can be made such that both parties benefit. An agent might offer side-payments to its peer, to compensate for possible loss of utility. Ultimately, the goal is to reach an allocation such that some global utility function is optimized. For in-depth discussions of negotiation and auctions, see (Jennings et al., 2001; Chevaleyre et al., 2005).

Although all agents implicitly agree to the rules of the virtual market, negotiation-based coordination is *explicit*

since agent exchange information about bids and preferences. Negotiation-based coordination is best suited for systems of *competitive* agents. Coordination is *decentralized* since agents have to interact with their peers to negotiate. Typically the coordination strategy in a market will not change at run-time, so coordination is *static*.
The rules of the market and protocols for negotiation are set at design-time. Therefore we position negotiation-based coordination at the *rule-level* of Rasmussen's model.

3.3. Skill based coordination: greedy

As the third class of coordination strategies we consider a greedy strategy. In such an approach, each agent has a simple evaluation function to select its next job. A greedy approach to multi-dimensional assignment problems can yield acceptable solutions, (Storms and Spieksma, 2003). Greedy algorithms focus on local optimization. Each agent makes its own local plan, without consulting others.

Greedy coordination is *implicit* because agents don't interact. There is no central management group, so coordination is *decentralized*. Agents have a fixed behavioral model, yielding *static* coordination. This type of strategy is of *cooperative* nature, since greedy coordination can only function when all participating agents pursue collective goals. Greedy coordination is positioned at the *skill-level* of Rasmussen's model, because agents are programmed to display typical stimulus/response behavior.

3.4. Summary

The properties of each approach are summarized in Table 1.

Property/approach	knowledge based	negotiation	greedy
Explicit	●	●	
Implicit			●
Dynamic			
Static	●	●	●
Cooperative	●		●
Competitive		●	
Centralized	●		
Decentralized		●	●
Rasmussen level	knowledge	rule	skill

Table 1: Properties of coordination strategies

4. FIRST AID CASE

This section discusses the measures and experiments to compare the coordination strategies. For this comparison, we need an environment that hosts *actors*, carrying out *jobs*. Carrying out a job means navigating to the job-location and executing a series of *tasks* that are required to complete the job. A task consists of a number of coherent *actions*, which may require *tools* and *goods*.

As an example, consider the job of aiding in a car-crash. One of the tasks could be extinguishing a fire. This task would require aiming the spout of the fire-extinguisher at the fire, pressing the lever on the extinguisher to release the foam and retracting the extinguisher. In this example the extinguisher is the tool and the foam the good that is required to complete the task. We see tools and goods as *resources*.

Each actor possesses a limited set of resources. It is possible a job requires multiple actors to complete, because it requires synchronized action or multiple types of resource to complete. For example, two actors lifting a stretcher or one actor cutting open the car door while another applies a neckbrace.

Every emergency situation involves injured people that need to be treated or evacuated. Our case handles coordination of first aid provision. The case revolves around coordinating medics (actors) in aiding a number of casualties (jobs) in a virtual disaster area. Each casualty has a specific injury and a triage-state. The triage-state indicates the severity of the injury, and how quickly a casualty needs to receive medical attention. The triage-state degrades over time, until the casualty eventually dies if left untreated. Each medic has a set of skill-values indicating how well she can treat a specific injury, and some medical resources. The case is limited to allocating casualties to medics; other tasks are not (yet) considered. The First Aid Case enables us to conduct comparative experiments, but also allows us to illustrate how coordination strategies can help AAC deal with complex, dynamic crisis situations.

4.1. Measures of performance

We use the First Aid Case to compare coordination strategies by offering identical scenarios to each strategy. Each strategy has to continually determine an allocation of medics to casualties. For the comparison we use two measures of performance:

- The *efficiency* in terms of the amount of effort required to determine an allocation
- The *effectiveness* in terms of the amount of jobs that are successfully completed

4.2. Effectiveness

The effectiveness of a coordination strategy is the capability to divide jobs between actors, given a set of conditions. Intuitively, a mechanism is effective when it's able to determine an allocation which completes all the jobs. However, in the experiments we intend to conduct the mechanisms will encounter situations, which cannot be solved without failing some jobs. In that case, the fewer jobs remain unattended, the more effective we regard the mechanism.

We define effectiveness to be the quality to complete as many jobs as possible. The effectiveness of an allocation at time t is computed according to:

$$\text{Effectiveness}(t) = \frac{\sum_{i=1}^{N_c(t)} C_i(t)}{N_c(t)},$$

where $C_i(t) = 1$ when casualty i is healed at time t and 0 otherwise. $N_c(t)$ is the total number of casualties at time t. Hence, we rate effectiveness by counting the amount of jobs completed in ratio to the total number of jobs.

4.3. Efficiency

To compare the efficiency of the coordination strategies, we need a measure to establish the volume of the coordination effort for each strategy. In the greedy and negotiation-based coordination strategies, the coordination activities are part of the processing of the actor. This means we have to distinguish between the effort spent on coordination, and the effort spent on executing another activity.

Therefore, we introduce two states for each processing thread in the systems under evaluation:

- A *coordination* state *C*, in which a thread resides whenever it is performing coordination activities.
- A *stable* state *S*, in which a thread resides whenever it is not performing coordination activities.

To determine the efficiency of a coordination strategy, we trace the time each thread spends in coordination state given a specific problem size. We will define a function of the number of actors and the number of jobs `CoordinationEffort' as:

$$\text{CoordinationEffort}(x, y) = \frac{\sum_{i=1}^{T} C_i(x, y)}{\sum_{i=1}^{T} \left(C_i(x, y) + S_i(x, y)\right)},$$

where x is the number of resources, y is the number of casualties, T is the total number of threads, $C_i(x, y)$ is the amount of time thread i spent in coordination state and $S_i(x, y)$ is the amount of time thread i spent in stable state.

4.4. SMDS based approach to First Aid Case

For the SMDS-based coordination strategy we will use a prototype COMPASS (Configuration, Organization and Management Prototype for Autonomous Systems of Systems). The coordination mechanism of COMPASS is based on a distributed knowledge base, distributed inference facilities and a distributed global allocation manager. The knowledge base contains facts about actors and their capabilities and domain specific rules. Figure 2 depicts the knowledge base and inference facility as a conceptual box, containing a knowledge base and inference facility. This 'box' that takes a situation (i.e. a set of jobs, or casualties) as input, infers an allocation plan and contracts the actors (medics) accordingly. The actors provide feedback about their status.

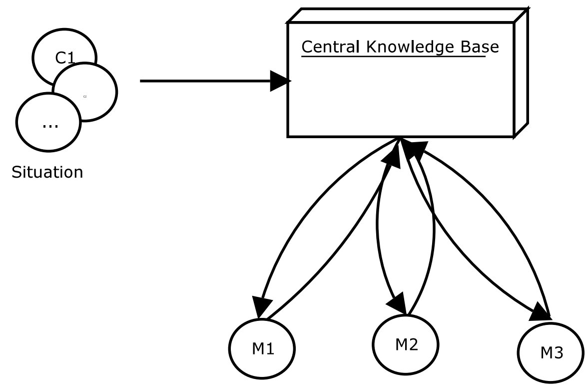

Figure 2: SMDS coordination strategy

The reasoning facilities in COMPASS are composed of a number of multi-threaded inference engines, each engine trying to allocate jobs for a set of actors. The algorithm of an inference thread is based on the GRAPHSEARCH algorithm documented in (Nilsson, 1982). Each engine will tentatively allocate a job to each of its actors in a partial allocation, immediately discarding the unsatisfactory allocations. Satisfactory partial allocations are shared between inference engines. This process is repeated until allocations without unallocated jobs emerge.

When searching for a feasible allocation, COMPASS will evaluate many partial solutions. The quality of an allocation is determined by cost/benefit calculation, where actions represent costs and casualties treated represent profit.

4.5. Negotiation-based approach to First Aid Case

In the negotiation-based approach medics (actors) negotiate with other medics in their proximity. Each actor has a job-list, containing the set of casualties that have to be treated. Actors assign a value to their job-list and will try to exchange jobs with other actors. A deal between two actors is made if both actors benefit from the trade, i.e. their individual value of the job-list increases. In the negotiation process the preferences of individual casualties are taken into account, e.g. when treatment of a casualty requires special skills. Figure 3 shows how a number of medics negotiate about who is including a specific casualty in their job list.

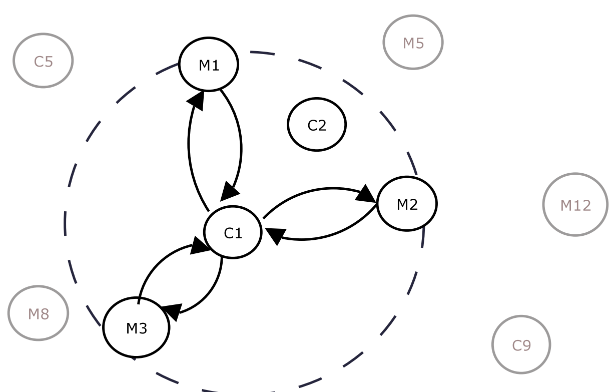

Figure 3: Negotiation-based coordination strategy

4.6. Greedy approach to First Aid Case

The greedy strategy is an intuitive approach to coordination. Experimental results show that selecting the nearest casualty yields somewhat better performance than selecting the casualty in worst condition first. Therefore, in this case, medics will choose the nearest casualty, regardless of triage level.

.

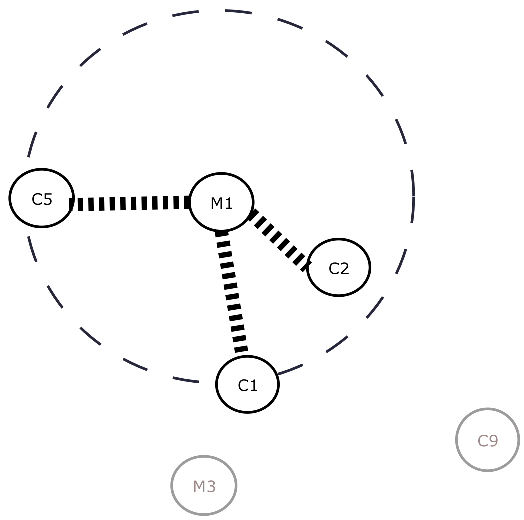

Figure 4: Greedy coordination strategy

Figure 4 shows how an individual medic selects the nearest casualties for treatment. This way, each medic generates an individual routing plan. The individual plans are not cross-checked between the medics, so the approach allows for conflicts (e.g. multiple medics are heading for the same casualty).

5. EVALUATION

In this chapter we will give a qualitative comparison of the three coordination strategies, in terms of efficiency and effectiveness. Also, we discuss the applicability of the approaches for the crisis response domain.

5.1. Efficiency and effectiveness

The algorithm used by COMPASS has the characteristic that, if an allocation exists, the algorithm will find it. If multiple allocations exist, a feasible allocation will be selected and iteratively improved. COMPASS will find the optimal allocation, if there is one. In COMPASS the trade-off between effectiveness and efficiency is in favor of effectiveness. COMPASS will continuously search for the best global allocation, encompassing all casualties and medics. Consequently, the coordination effort is high and, relative to the other strategies, the efficiency is low. Computational feasibility is established by evaluating multiple allocations in parallel.

At the other end of the spectrum we find the greedy strategy. As documented in (Storms et al., 2003), this approach will very efficiently determine solutions to allocation problems. The coordination effort will be the lowest of all three strategies. This efficiency comes at the price of severely reduced effectiveness. Since the plans of other actors are not considered, it is possible that two or more medics plan to treat the same casualty. Such constraint violations can be resolved by central repair algorithm, at the expense of efficiency.

In-between the greedy and SMDS-based strategy we find the negotiation-based strategy. It consists of a series of localized negotiations, in a cluster of casualties and medics that are physically close to each other. Since not all medics and casualties are involved in the negotiation, this approach might not find solutions that are globally optimal. However, it is not likely that in a good allocation medics are sent to physically remote casualties. While the negotiation-based approach spends no coordination effort on evaluating these allocations, the allocations are expected to be of reasonable quality. Relative to the other strategies, this strategy will result in medium efficiency and effectiveness.

5.2. Escalation

The allocation algorithm of COMPASS can detect whether the amount of jobs is too large for the available actors, or that jobs cannot be completed due to missing skills or resources. In these cases, it can trigger an escalation mechanism to request more actors or resources.

The negotiation-based strategy has no integral trigger for detecting whether or not the situation is 'out of control'. Whenever an actor is no longer capable of performing a job, and it cannot be transferred to another actor, it has to be dropped. This fact in itself can be used to trigger an escalation mechanism. However, for jobs that are ignored by all actors, due to lack of time, skills or resources, no such mechanism exists in the negotiation strategy. This anomaly could be repaired by introducing actors that collect the ignored jobs, at the cost of introducing extra coordination overhead.

The greedy strategy will also ignore the jobs that cannot be completed due to lack of skills or resources. This implies the coordination strategy has no means of detecting a lack of skills or tools in the community of medics. Furthermore, there is no way the greedy strategy can guarantee all jobs are allocated at least once. This means there is no evident mechanism available to the greedy strategy to make use of an escalation mechanism.

5.3. Alternative approach: ant-based coordination

An alternative approach to the SMDS-based or negotiation-based strategy is stigmergy-based coordination. These strategies are inspired by coordination mechanisms found in communities of social insects. The idea is to establish *swarm intelligence* (Dorigo and Di Caro, 1999) in a community of simple agents that respond to the presence and characteristics of markers (pheromones) in their environment.

As an example, consider using stigmergy to solve vehicle routing problems by mimicking the foraging behavior of ants. In this metaphor mobile agents imitate the trail-laying behavior of ants. The first aid case can be perceived as a special class of vehicle routing problem, the Dynamic Vehicle Routing Problem with Time Windows (DVRPTW) (Ahuja, Magnanti and Orlin, 1993). In the First Aid Case each medic acts as a vehicle, trying to serve as many customers (casualties) as possible. Each ant tries to find the longest cycle which contains each medic and the maximum number of casualties. Each ant constructs a partial solution where each casualty is visited at most once.

The ant-based coordination strategy to this problem very much resembles the ant-based routing approach described in (Dorigo et al., 1999; Ellabib, Basir and Calamai, 2002; Guntsch and Middendorf, 2002). Effectiveness and efficiency of this approach are likely to be close the SMDS- and negotiation-based strategies. Future experiments will have to establish the exact qualities of this approach.

5.4. Applicability for crisis response

In addition to increased effectiveness, the SMDS approach has the benefit of flexibility. If the problem at hand slightly changes (e.g. new constraints or dependencies), an update of the SMDS knowledge-base at run-time is sufficient to adapt the coordination strategy for the new situation. Knowledge-level approaches, such as SMDS, are fairly generic but possibly less efficient.

Coordination strategies that are positioned at the skill-level lack the benefit of being generic. Behavior of the agents is based upon past experience or design-time analysis, and 'hard-wired' in the agents. Though extremely efficient, these strategies are tailored for one specific problem. If the problem changes, agent behavior needs to be tuned or redefined at design time, by means of simulation and analysis.

Market-based approaches, positioned at the rule-level, have some implicit knowledge, like the fundamental rules of the virtual market that have to be respected by all agents involved. Other knowledge, like agent preferences or the negotiation procedure, can be adapted at run-time. Flexibility of rule-level approaches is somewhere in-between knowledge-level and skill-level approaches.

Flexibility is a key requirement to crisis response organizations. Actors and agents operate in unpredictable, even chaotic, environments. Operational requirements might change unexpectedly. Artificial coordination strategies have to be able to cope with these dynamics. However, the strategies that have sufficient flexibility and effectiveness are the least efficient, and vice versa. To fully exploit the benefits of all approaches, we recommend an eclectic approach to artificial coordination, as depicted in Figure 5. In this approach a selection mechanism chooses the best allocation, based on the valuation criterion of effectiveness, for example. Furthermore, this combined strategy includes the capability to detect the shortage of skills or resources and trigger some escalation mechanism.

In a layered coordination strategy, fast skill-level strategies come up with an initial (heuristic) allocation of actors and jobs. In this allocation some constraints might even be ignored. Next, if time allows it, rule-level and knowledge-level strategies can improve the initial allocation, and take other aspects of the problem into account.

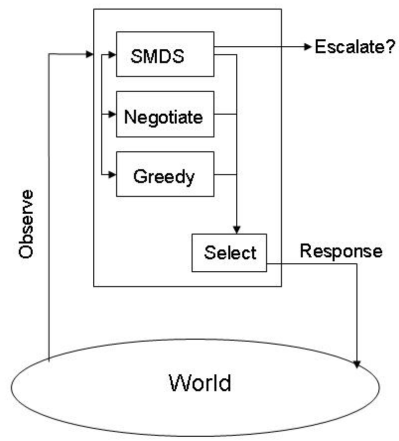

Figure 5: Combined, layered coordination engine

6. CONCLUSIONS

In this work we argued discussed AAC as a metaphor for agile crisis response organizations and discussed the need for artificial coordination for managing activities. Coordination strategies can be classified in terms of four dimensions and their position in Rasmussen's model for supervisory control. This model has served as a design guideline for three artificial coordination strategies. We described three distinct approaches to artificial coordination, one for each of Rasmussen's levels. By means of a case study we have shown how these approaches can be applied in the crisis response domain. Next, we compared the approaches in a qualitative manner. In terms of effectiveness and efficiency, the knowledge-level coordination strategies are the most effective, and the skill-level strategies are the most efficient. Concerning flexibility there is a reverse trade-off with efficiency. Opposed to skill-level strategies, knowledge-level strategies easily adjust to changing operational requirements. On all aspects, rule-level strategies are in-between knowledge-level and skill-level strategies.

In order to give a more insightful verdict on the applicability of the coordination mechanisms, we have to take into account other relevant issues like the robustness of the coordination mechanism. The robustness of a coordination mechanism would express how well the mechanism is able to recuperate in case of unexpected events, for example when the period of time a medic is available to the coordination mechanism is determined by the scenario, but unknown to the mechanism.

Moreover, the analysis of this work is limited to a descriptive and qualitative comparison. Future research will focus on simulation and computational experiments, in order to establish quantitative results on efficiency and effectiveness.

7. ACKNOWLEDGEMENTS

This work is supported by the Dutch Ministry of Economic Affairs, grant number: TSIT2003, a Senter ICT Breakthrough project. The research in this paper is part of the COMBINED Systems Project, conducted at the DECIS Lab in Delft, a joint venture of Thales, University of Amsterdam, Technical University of Delft and TNO.

8. REFERENCES

1. Aart, C.J. van (2005) Organizational Principles for Multi-Agent Architectures. *Whitestein Series in Software Agent Technologies*. Birkhäuser, Basel, Switzerland.

2. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. (1993) Network Flows: Theory, Algorithms, and Applications. Prentice Hall, New Jersey.

3. Alberts, D.S., Gartska J.J., Stein F.P. (1999) Network Centric Warfare: developing and leveraging information

superiority. DoD CCRP. Available at http://www.dodccrp.org

4. Atkinson, S.R. and Moffat, J. (2005) The Agile Organization: From Informal Networks to Complex Effects and Agility. DoD CCRP. Available at http://www.dodccrp.org

5. Bonabeau, E., Dorigo, M. and Theraulaz, G. (1999) Swarm Intelligence: From Natural to Artificial Systems. Santa Fe Institute Studies in the Sciences of Complexity. Oxford University Press, New York, NY, USA.

6. Chevaleyre, Y., Dunne, P. E., Endriss, U., Lang, J., Lemare, M., Maudet, N., Padget, J., Phelps, S., Rodruez-Aguilar, J. A., and Sousa P. (2005) Issues in Multi-Agent Resource Allocation, *Informatica* (Accepted for publication), IOS Press, Amsterdam, The Netherlands.

7. Corkill, D. and Lander, S. (1998) Agent Organizations. *Object Magazine*, Vol. 8(4), pages 41-47.

8. Dorigo, M. and Di Caro, G. (1999) The ant-colony optimization meta-heuristic. In: *New Ideas in Optimization* (Corne, D and Dorigo, M. ed.), 11-32, McGraw-Hill.

9. Ellabib, I., Basir, O. A., Calamai, P. (2002) An Experimental Study of a Simple Ant Colony System for the Vehicle Routing Problem with Time Windows. Ant Algorithms, pages 53-64, Springer Verlag, Lecture Notes in Computer Science 2463.

10. Endsley, M.R. (2000) Theoretical Underpinning of Situation Awareness. In: *Situation Awareness Analysis and Measurement*, (Endsley, M.R. and Garland. D.J. ed.), LEA, Mahwah, NJ, USA.

11. Giorgini, P., Kolp, M. and Mylopoulos, J. (2005) Multi-Agent Architectures as Organizational Structures. *Journal of Autonomous Agents and Multi-Agent Systems.* (in press).

12. Guntsch, M., Middendorf M., (2002): Applying Population Based ACO to Dynamic Optimization Problems. Ant Algorithms, pages 111-122, Springer Verlag, Lecture Notes In Computer Science 2463.

13. Hannebauer, M. (2002) Autonomous Dynamic Reconfiguration in Multi-Agent Systems: Improving the Quality and Efficiency of Collaborative Problem Solving, *Lecture Notes in Computer Science*, Vol. 2427, Springer-Verlag, Heidelberg, Germany.

14. Harrald, J.R., (2005), Supporting agility and discipline when preparing for and responding to extreme events, ISCRAM 2005, keynote speech, available at http:\\www.iscram.org

15. Jennings, N.R. (1996) Coordination techniques for distributed artificial intelligence. In *Foundations of distributed artificial intelligence*, John Wiley & Sons, Inc, New York, NY, USA, pages 187-210.

16. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Sierra C. and Wooldridge, M. (2001) Automated Negotiation: Prospects, Methods and Challenges. *International Journal of Group Decision and Negotiation*, Vol.10(2).

17. Krygiel, A.J. (1999) Behind the Wizard's Curtain: An integration Environment for a System of Systems, DoD *C4ISR Cooperative research Program*, DoD CCRP. Available at http://www.dodccrp.org/html2/pubs_pdf.html

18. Lesser, V. and Horling B. (2005) A Survey of Multi-Organizational Paradigms. *The Knowledge Engineering Review*, Vol. 19(4), pages 281-316.

19. Malone, T.W. and Crowston, K. (1990). What is coordination theory and how can it help design cooperative work systems? *Proceedings of the 1990 ACM conference on Computer-supported cooperative work (CSCW'90)*, 357-370.

20. Malone, T.W. and Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Computing Surveys*, Vol 26(1), pages 87-119.

21. Mason, Z. (2002). Programming with stigmergy: using swarms in construction. *Proceedings of the Eight International Conference on Artificial Life (Artificial Life VIII)*, MIT Press, Sydney, Australia, pages 371-374.

22. NIBRA (Nederlands Instituut voor Brandweer en Rampenbestrijding) i.c.w. VNG Commisie Brouwer (2002). Studie NIBRA: Krachten bundelen voor veiligheid. Available (in Dutch) at http://www.brandweer.nl

23. Nilsson, N.J., (1982). Principles of Artificial Intelligence, Springer Verlag.

24. Omicini, A., Zambonelli F., Klusch, M., Tolksdorf, R. (2001) Coordination of Internet Agents: Models Technologies and Applications. Springer Verlag, Heidelberg, Germany.

25. Oomes, A.H.J. (2004). Organization Awareness in Crisis Management, *Proceedings of the International Workshop on Information Systems on Crisis Response and Management (ISCRAM),* Brussels, Belgium.

26. Ossowski, S. (1999). Co-Ordination in Artificial Agent Societies: Social Structures and Its Implications for Autonomous Problem Solving Agents. Springer Verlag New York Inc., Secaucus, NJ, USA.

27. Rasmussen, J. (1983). Skills, Rules and Knowledge: signals signs and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 3(3), pages 257-267.

28. Sheridan, T.R. (1988). Task Allocation and Supervisory Control. In *Handbook of Human-Computer Interaction* (Helander, M. ed.), pages 159-172, Elsevier Science Publishers BV.

29. Storms, P. P. and Spieksma, F. C. (2003). An LP-based algorithm for the data association problem in multitarget tracking. Comput. Oper. Res. 30, 7 (Jun. 2003), 1067-1085.

30. Tatnall, A. and Gilding, A. (1999). Actor-Network Theory and Information Systems Research. In *Proc. 10th Australasian Conference on Information Systems*. Available at http://www.vuw.ac.nz/acis99/Papers/PaperTatnall-069.pdf

31. Wilmott, S. (2002). Coordination Structures for the Intelligent Control of Dynamic Distributed Systems. PhD-thesis, Ecole Polytechnique Fédérale de Lausanne.

32. Wooldridge, M. (2002). An introduction to MultiAgent Systems. John Wiley & Sons, Chichester, UK.