Published in International Workshop on Harbour, Maritime & Multimodal Logistics Modelling and Simulation (HMS), 65-70, 2000
which should be used for any reference to this work

RESOURCE MANAGEMENT IN COMPLEX SOCIO-TECHNICAL SYSTEMS: A MULTIAGENT COORDINATION FRAMEWORK

Peter Kropf* and Brahim Chaib-draa†

*Dép. d'informatique et de rech. op.
Université de Montréal
Montréal, Canada H3C 3J7
kropf@iro.umontreal.ca

†Dép. d'informatique
Université Laval
Québec, Canada G1K 7P4
chaib@ift.ulaval.ca

Keywords. Resource management, Multiagent systems, Coordination, Planning, Resource allocation, Teamwork, Command and Control Systems.

ABSTRACT

Resource management in complex socio-technical systems is a central and crucial task. The many diverse components involved together with various constraints such as real-time conditions make it impossible to devise exact optimal solutions. In this article, we present an approach to the resource management problem based on the multiagent paradigm to be applied in the context of a shipboard command and control (C2) system. A general architecture for multiagent planning and scheduling for achieving a common shared goal together with a real-time simulation environment as well as a simulation test-bed using the agent teamwork approach is described.

INTRODUCTION

Socio-technical systems (STS) are becoming increasingly complex. Often this complexity arises from the multitude and variety of relationships that are involved among the resources to be deployed or used to achieve system goals. Additional complexity is further introduced when system behavior requiring human intervention and interaction forms an integral part of the system. Examples of such systems include transportation logistics, management and control (road, rail, sea, air), industrial engineering systems (process control, flexible manufacturing, and others), nuclear power plant control, communication management and control, shipboard command and control (C2), electric power management, reactive systems such as commercial aircraft control systems, etc. In these systems, tasks are performed in a highly dynamic, complex environment and call for a high degree of coordinated activity among actors, planners and decision makers to occur in a timely and responsive manner.

In the case of an industrial engineering system for example, the common goal of every entity involved in the production process is to produce manufacturing goods as efficiently and effectively as possible. There are multiple resources to be considered here: manufacturing components, assembly components (e.g. robots), human resources in the manufacturing process, resources at the engineering and marketing levels, as well as sensors for automated control, and humans responsible for monitoring and controlling the functioning of the whole process. In the same context, an air-traffic control system is characterized by the goal of ensuring passenger and crew security during all phases of a flight (take-off, flight, landing). Finally, shipboard C2 systems must assure adequate response to external threats while making the most effective use of its resources for tactical picture compilation and defensive measures.

The management of the resources involved constitutes a central and crucial task for such systems to achieve their goals. The multiple resources may be of many different kinds, such as computational equipment, communication channels, technical equipment, and personnel. In some cases, the scenarios to manage, the actions to take and the resource allocation strategies to employ are fairly deterministic or at least predictable. This is the case for instance with some applications of manufacturing. Other more open systems are potentially subject to large not anticipated variations and tend to be more reactive. This is due to the occurrence of non-deterministic arising events, which require implementing dynamic resource allocation strategies. Some systems show a further complication in that very often conflicting situations arise, be it conflicting or imprecise information for taking resource allocation decisions, be it conflicting or overlapping goals. Such situations may for instance arise in railways (or other transportation systems), where load capacity, delivery time, routing, etc., compete for transportation resources.

The diverse characteristics of the resources controlled and managed by such systems, as well as the diverse characteristics of the information available and the associated interaction environment, require new methods and techniques to find solutions. Moreover, the complexity of the resource management problem for STS do not allow for exact solutions, because the computational effort is very large even when using high performance computing systems (Kropf *et al.* 1999). Therefore, we rather envisage a *Decision Support System* (DSS) to help operators to take accurate resource allocation actions. While the allocation of a CPU to processes or the allocation of take off slots in air traffic control might use a simple round robin scheduling technique combined with a priority scheme, a transportation or shipboard C2 system should instead be viewed as a *Multiagent System* (MAS) where human or software agents provide decision support for dispatching and engaging resources. In MAS, knowledge, action and control are distributed among software entities (called agents) which may cooperate, compete or coexist depending on the context. MAS technology is becoming one of the most important and exciting areas of research and development in computer science today (Chaib-draa 1995). For these reasons, we have adopted the MAS paradigm by considering *Resource Management* (RM) as a coordination process involving goals, agents or actors (i.e., worker/operator/human entity or automated entity) and resources; a process which is viewed as the act of managing inter-dependencies between agents' activities as shown in Table 1.

Components of coordination	*Associated coordination process*
Goals (G)	Identifying goals
Planning (P)	Mapping the complex multiagent plan to the goals (G)
Agents	Task allocation or mapping parts of the plan to agents
Coordination	Managing the inter-dependencies between agents (resource allocation, sequencing, synchronization, etc.)

Table 1: Components of coordination.

RESOURCE MANAGEMENT

The aim of resource management, simply stated, is to manage inter-dependencies between activities. We have selected the complex socio-technical system of tactical C2 (Chalmers 1998) on board a navy frigate as a concrete environment in which to study the characteristics of such coordination processes and to identify and generate appropriate coordination mechanisms for these processes, including algorithms and architectures, for these processes. Operators in this type of environment perceive and interpret information available from own-ship sensors or data-linked from other cooperating platforms, and plan and conduct mission operations. In highly dynamic scenarios with a large number of constraints, handling such a large amount of information could quickly overwhelm human capabilities. This suggests a need for tools to support and complement operators by matching their perceptual and cognitive resources to the demands of the environment, and by supporting, when necessary, their mental strategies used to deal with complexity and perform decision making activities. The *Simulated Real-Time Environment* (SRTE) (Chalmers and Blodgett 1998) provides the necessary simulation infrastructure for our C2 application in which to accomplish planning and re-planning. The SRTE shown in figure 1 simulates targets and own-ship features (sensors, actuators, etc.), performs multi-sensor data fusion (MSDF), and has a KBS shell and agents for *Situation and Threat Assessment* (STA) and a multiagent planning architecture for resource management (RM). The Human-computer Interface allows to monitor and evaluate the system and to communicate with the KBS.

APPROACH

Multi-Sensor Data Fusion and Situation Assessment

For our purposes, the principal functions related to these processes are: (1) threat detection based on data from several sensors; (2) target tracking based on data fusion; (3) contact discrimination which consists of separating threats from friends,

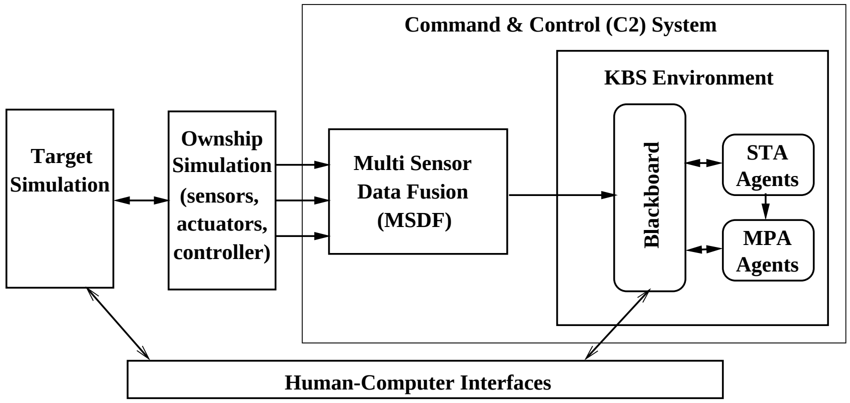

Figure 1: Generic architecture for the SRTE.

neutrals, and decoys; and finally (4) identification which consists of establishing further identity information on threats, friends and neutrals. These functions are part of multi-sensor data fusion (MSDF) and situation and threat assessment (STA) processing (Duquet *et al.* 1998).

Goal Selection and Planning in a MPA

The concept of coordination that we have adopted implies multiple activities related to some goal(s). Therefore, in order for coordination to occur, these goals and activities must somehow be identified. The most commonly analyzed case of this process occurs when an individual or group decides to pursue a goal, and then decomposes this goal into subgoals (or plans), which together will achieve the original goal. We call the process of choosing the goal, *goal selection*, and the process of choosing the activities goal, decomposition or *planning*.

The triad *goals, problems and opportunities* is principally motivated by the meaning and significance of potential problems imposed and opportunities provided by an external environment from an operator-centered perspective. For our purposes, we define a *problem* to be a feature of the situation that has the potential to negatively impact the achievement of one or more *goals* or which should at least alert a decision maker to consider a change in the way these goals are being, can be, or should be achieved. A problem therefore represents an important goal-relevant property of the environment in that it can shape some aspect of an operator's behavior. The detection of a problem signals a possible need for corrective measures to avoid or resolve the problem. A second important type of goal-relevant property for an operator interacting with a complex, dynamic environment is related to opportunities. An *opportunity* is defined as a feature of the situation that represents a possibility to achieve one or more goals, or to accelerate their achievement, or to resolve the obstacles to their achievement. Whereas a problem can be thought of as a behavioral constraint, recognition or identification of an opportunity is an event that offers potential for enlarging the degrees of freedom for that behavior. For example, a particular geographical or environmental feature may offer an opportunity for concealing detection from an enemy. In some cases, there may be a cost attached to taking advantage of an opportunity. This cost may need to be estimated as a precursor to a decision.

The planning problem addressed here is a multiagent planning problem in which the decision agents (or planning agents) are comprised of human decision makers, computer-based agents and decision aids. The decision agents can be geographically dispersed due to the distributed nature of the operational environment (task force, task group), the nature of sensors, and physical limitations of weapon systems. To capitalize on the benefits of distributed computing architectures which sustain our distributed decision agents, we propose here a *multiagent planning architecture* (MPA). MPA provides protocols to support the sharing of knowledge and capabilities among (software) agents involved in cooperative planning and problem solving. Such software agents share a well-defined, uniform interface specification, so that we can explore ways of reconfiguring, re-implementing, and adding new capabilities to the planning system.

MPA also provides an environment to create intelligent agents, called plan managers (PM's) associated with each planning agent. PM's communicate with each other during the planning process,

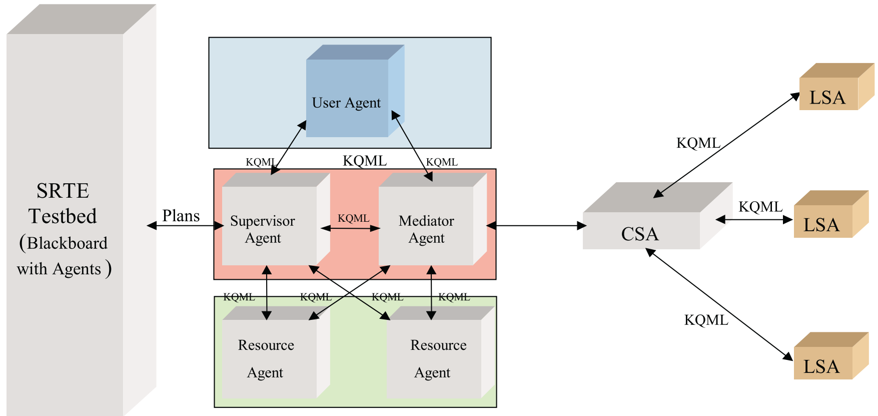

Figure 2: Architecture for Multiagent Planning and Scheduling.

notifying other plan managers of changes to the plan (taking into account: temporal constraints, operational constraints, resource utilization, opportunities, problems, etc.) that may cause conflicts among the partial plans. Once the conflict is identified, the plan managers use a set of conflict resolution strategies to repair the conflict, with involvement of human planners where necessary.

Task Allocation and Negociation

The previous planning process provides as output a complex plan (i.e., a set of activities) which addresses the target C2 situation. How to decompose this plan and how to allocate sub-plans to actor agents, constitutes the task allocation problem. To solve this problem, we will analyze three alternatives in terms of coordination costs and benefits.

The first one uses two specific software agents: the supervisor agent which *knows* how to decompose a given multiagent plan, and the mediator agent which *knows* which actor agents are capable of performing which tasks. These two agents are parts of the Laval University multiagent architecture, called *Networked Software Agents* (NetSA) (Côté and Troudi 1998). Other agents of NetSA have the following characteristics: (1) *User agents*: they contain mechanisms to select an ontology, support a variety of interchangeable user interfaces, maintain models of other agents, etc., (2) *Mediator agent*: it implements a 'yellow pages' and 'white pages' directory services for locating appropriate actor agents with appropriate capabilities. It also manages *Agent Name Server* (ANS) and resources services, and may have the ability to store and forward messages, and locates message recipients. It also functions as a communication aid, by managing communications among the various other agents. Communication is based on the standard language KQML (Chaib-draa and Vanderveken 1998; Lizotte and Chaib-draa 1997); (3) *Resource agents*: they come in a variety of common types, depending on which resource they are representing. In the context of the C2 application, a resource designates target, weapon, soft-kill weapon, hard-kill weapon, sensor, ship, a computer system, a database, etc. and a resource agent means the software agent which constitutes the interface to this specific resource; (4) *Supervisor agent*: it decomposes a complex plan and allocates the different activities to actor agents with help from the mediator. It also provides the SRTE simulation test-bed with all the information regarding a conflict or a failure in order to help it to re-plan. Figure 2 shows the design of an architecture based on NetSA for the C2 application.

The second alternative for task allocation supposes that our mediator does not have all the knowledge to do a complete task allocation. In this case, the supervisor will first decompose the complex plan and then send a description of each activity to be performed to the mediator. The mediator attaches to each description the qualifications and capabilities required and makes an announcement. The potential resource agents then use this information to decide whether to submit a bid for the task. A bid includes a description of qualifications of the bidder and its availability for performing the task. The mediator uses these bid messages to decide which resource agent should perform the task and then send an award message to notify (1) the resource agent that is selected and, (2) the supervisor which actor agent it has selected.

Finally, the third alternative turns around a simulated market where actor agents compete on the prices of goods and services. In this context, the allocation task is viewed as a market and the 'best' strategy for this type of competition according to the analysis will be implemented.

Coordination with Resource Scheduling

Because of the diverse characteristics of the resources involved in a socio-technical system, as well as the many different methods used to manage those resources, we are faced with a situation where resources, their use and management must be considered at different levels in order to cope with complexity. We enrich our NetSA architecture with many *Local-Scheduler-Agents* (LSA) (one for each level) that communicate with a *Central-Scheduler-Agent* (CSA) as depicted in Figure 1. For a scheduling problem raised by the mediator, the CSA considers this problem as a *Distributed Constraint Satisfaction Problem* (DCSP), formulated in terms of variables, domains and constraints. The CSA decomposes the DCSP and contacts LSA's for partial solutions. If there are some conflicts between LSA's, it tries to solve them in collaboration with the concerned LSA's (by relaxing some constraints, for example). Finally, it combines the partial results to the DCSP and submits the solution to the mediator. For the local CSP, a number of general purpose (and powerful) techniques can analyze the problem and find a solution (Tsang 1993).

A TEAMWORK TEST-BED

Figure 2 illustrates the general design of the decision support system (DSS) test-bed we apply to the shipboard C2 system. This DSS supports operators at least in: (1) the identification and selection of actions; (2) the management of resources; and (3) action implementation. In order to better understand the specific characteristics of the resource management problem in the context of the C2 shipboard system and to deliver a proof of concept for our approach, we have developed a test-bed architecture applying the concept of *teamwork*. This prototype environment for C2 planning and scheduling includes the simulation of the target, own-ship and MSFD features of the SRTE test-bed as agents as well (see figure 1). This allows us to easily test and evaluate the planning and scheduling algorithms before their integration into the architecture shown in Figure 2, the SRTE simulation environment and ultimately into the real DSS system.

Tasks in complex technical systems with human interaction are in general too difficult to be executed by individual units. In the context of our application of naval combat, many different individual members of the defense system must work together as a team towards common objectives with dynamic shared plans and information. For example, in order to efficiently manage the defense resources of a ship, they can be considered as a teamwork trying to achieve the common goal : intercepting enemy missiles (or other weaponry). Unfortunately, in implemented multi-agent systems, team activities and the underlying model of teamwork are often not represented explicitly (except Tambe's work (Tambe 1997)). In fact, to simulate, test and evaluate operation strategies using multiagent systems, the agents must be provided an explicit model of teamwork.

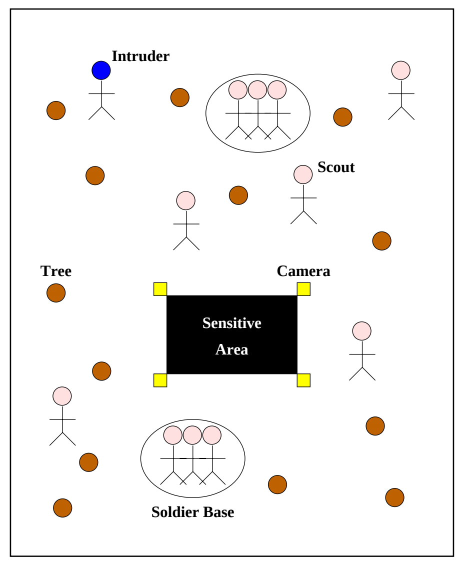

Figure 3: Teamwork test-bed environment

To achieve a multiagent module that supports teamwork between an arbitrary number of agents is necessary. A generalized environment realizing a prey-predator scheme has been designed. In the context of a military scenario, there are a number of agents patrolling the area which try to intercept intruders who want to reach a sensitive area which must be drastically defended.

Figure 3 illustrates this environment as a two dimensional grid containing either trees, which serve as obstacles to the agents' vision, or flat land. A number of man-made constructs are also present:

bases, where soldier agents wait before they are sent to intercept an intruder, and the sensitive area.

Three different types of agents exist in this environment: scouts, soldiers and cameras. Scouts are the fastest agents and see very far but are inefficient in combat: their role is to patrol the area until they find an intruder, at which time they call for reinforcements and try to follow the intruder to update it's position to his other team-mates. Soldiers are slow and do not see far ahead but are the only agents able to intercept and possibly to kill intruders, their limited perception make them reliant on scouts for knowing the position of these intruders. When they are not out for an interception, the soldiers wait at small bases placed around the sensitive area. Finally, cameras are placed at the corners of the sensitive area. While they cannot move, the camera can turn and are able to see the furthest of all agents. Their role is only to inform other agents when an intruder is detected. Users of the program control the intruders themselves, as well as their arbitrary number in the environment. Obviously, the ultimate goal of intruders consists of reaching the sensitive zone to "bomb" it.

This generalized view can be extended to the complex shipboard system, which allows us to abstract from the many technical details of the very sophisticated equipment on board (sensors, radars, etc.) and the associated data (control system data, intruder speed, turning rate of a missile, etc.) Instead, this test-bed environment allows to concentrated on the resource management problem only, i.e. on the design and implementation of the agent teamwork. The agents in our test bed should have some sophisticated reasoning mechanism that allow them to achieve their goals and sometimes to operate as a teamwork, particularly for (1) perception and (2) interception of intruders. Our test bed is ideally suited for an advanced team-working module that, while being very efficient in the shipboard resource management problem, is robust and flexible enough to be used in other applications such as transportation systems or any other socio-technical system.

CONCLUSIONS

Real time decision support systems and resource management for such complex systems as the tactical C2 on board a navy frigate are a very demanding research and development challenge. The Multi-agent paradigm is well suited for our resource management problem. RM is considered as a coordination process involving goals, agents and resources. The use of agents is very suitable to generate plans and managing their inter-dependencies in the highly dynamic environment of the C2 system. The test-bed environment is currently being implemented and first experimentation results are expected in the very near future.

References

Chaib-draa, B. 1995. "Industrial applications of distributed AI." *Communications of the ACM 38*, no. 11: 49–53.

Chaib-draa, B. and D. Vanderveken. 1998. "Agent communication language: a semantics based on success, satisfaction and recursion." In *Proc. of ATAL'98*. Paris, France.

Chalmers, B. 1998. "On the design of a decision support system for data fusion and RM in a modern frigate ." In *Proc. of sensor DF and integ. of the human*, NATO Symposium. Ottawa, Canada.

Chalmers, B. and D. Blodgett. 1998. *Weapon engagement management for ship defence*. Technical Report R-9805, DREV, Valcartier, Canada.

Côté, M. and N. Troudi. 1998. "NetSA : Une architecture multiagent pour la recherche sur Internet." *L'Expertise informatique 3*, no. 3.

Duquet, J. R.; P. Bergeron; D. Blodgett; J. Couture; B. A. Chalmers; and S. Paradis. 1998. "Analysis of the functional and real-time requirements of a multi-sensor data fusion (MSDF)/ (STA)/ (RM) system." In *SPIE proceedings*, vol. 3376, 198–209.

Kropf, P.; B. Chalmers; and N. Pageau. 1999. *Parallel algorithmes for resource planning*. Technical Report DIUL-RR-9905, Laval University.

Lizotte, S. and B. Chaib-draa. 1997. "Coordination in CE systems: an approach based on the management of dependencies between activities." *Concurrent Engineering: Research and Applic. 5*, no. 4.

Tambe, M. 1997. "Towards flexible teamwork." *J. of Artificial Intelligence Research 7*, : 83–124.

Tsang, E. 1993. *Foundations of constraint satisfaction*. Academic Press, San Diego, CA.

We thank the National Science and Engineering Research Council, Canada (Nr. 222802-98) and Lockheed Martin Canada for their support.