A TEAMWORK TEST-BED FOR A DECISION SUPPORT SYSTEM

Brahim Chaib-draa[1], Peter Kropf[2], Sébastien Paquet[1]

[1]Laval University, Québec, Canada
{chaib,paquets}@ift.ulaval.ca
[2]University of Montreal, Montréal, Canada
kropf@iro.umontreal.ca

ABSTRACT

Resource management in complex socio-technical systems (e.g. management and control (road, rail, sea, air), industrial engineering systems, transportation logistics, etc.) is a central and crucial process. 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 multi-agent paradigm to be applied in the context of a shipboard command and control (C2) system. A general architecture for multi-agent 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.

KEYWORDS

Control–Command (C2) Systems, Multi-agent systems, Resource Management, Coordination, Planning, Resource allocation, Teamwork.

1 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 in these systems 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 unanticipated variations and tend to be more reactive. This is due to the occurrence of non-deterministically 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 different characteristics of the resources controlled and managed by such systems, as well as the different characteristics of the information available and the associated interaction environment, require necessarily different 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. 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 *Multi-agent System* (MAS) where ”autonomous” software agents provide decision *support* for dispatching and engaging resources. Notice that it is just a support and the final decision is under control of human. In MAS, knowledge, action and control are distributed among 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.

One particular STS is shipboard Command and Control (C2) for the combat system, where operator activities involve a number of data and information processing tasks which must be continually performed in real time as part of tactical decision making. These tasks include: drawing a picture of the tactical situation using both real-time and non real-time data from a variety of sources; using this picture to monitor the tactical situation and assess and comprehend its elements; and responding to perceived or potential threats in a manner that complies with various rules of engagement.

In general, we try to address this problem by conceiving a Command Control System (CCS) which assists human operators in best utilizing the fighting capabilities of the ship. Based on the growing complexity of naval warfare, an inevitable conclusion is that future shipboard CCSs must provide increased or new kinds of tactical decision support if human performance limits are not to be exceeded. Unfortunately, current operational systems generally provide little decision support in complex, highly dynamic scenarios. For example, among support capabilities, one can envisage computer-based tools that automate tracking to speed up reactions; provide context dependent cues to help focus the operators' attention; provide threat analysis tools to assist in decision making; present a common force-level tactical picture; and assign weapons under human veto.

Our group has for several years investigated methods to augment or enhance CCS capabilities. We are now broadening the scope of this work by exploring concepts turning around multi-agent techniques for the design, development, implementation, and evaluation of a computer-based, real-time decision support system (DSS) that can be integrated into the ship's CCS to assist operators in conducting tactical Command and Control (C2). The main reasons that sustain our choice for the multi-agent techniques are: (1) the C2 is a complex process; (2) the shipboard C2 is a distributed application; (3) the shipboard C2 application needs coordination and negotiation between different entities in order to manage its resources.

2 THE SIMULATED REAL-TIME ENVIRONMENT (SRTE)

The basic SRTE testbed (Duquet, Bergeron, Blodgett, Couture, Chalmers, and Paradis 1998) has constituted a first step towards a Decision-Aid System for naval command and Control systems, and as such, it has been evolving into a more sophisticated testbed. The components of the SRTE are displayed in Figure 1. These include (1) a target and scenario generator, (2) an ownship simulator which emulates the ship's sensors, (3) a Multi-Sensor Data Fusion system (MSDF) which acts as a tracker and attribute fusion engine, and also provides target identity propositions, (4) a Command and Control System (CCS), (5) a simulated real-time controller, and (6) a Performance Evaluation module. In the CCS, the Situation and Threat Assessment (STA) and Resource Management (RM) functionalities are represented as agents on top of a Knowledge Based System (KBS) shell implemented on a blackboard architecture. Here is a brief description of the main SRTE's components:

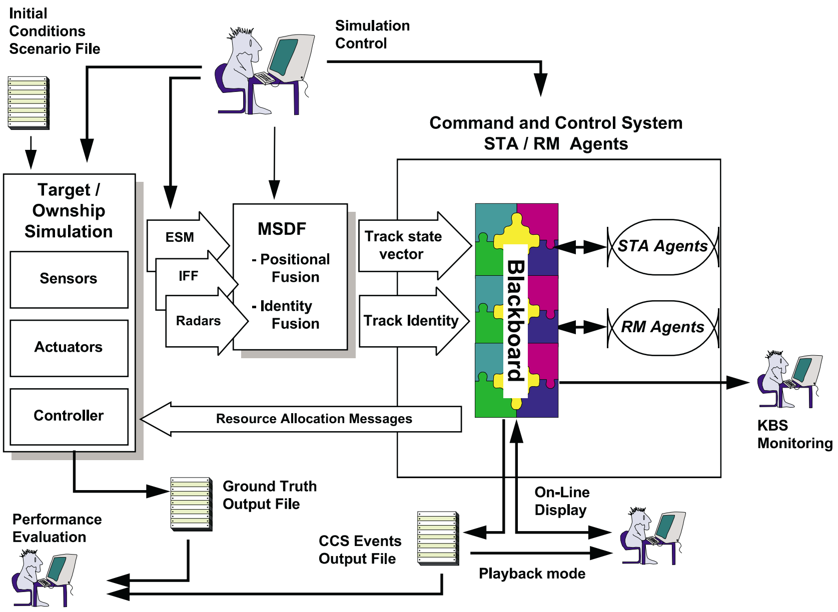

Figure 1. Generic architecture for the SRTE.

- **Target and Ownship Simulation:** This simulation part relies on concept analysis and simulation environment for automatic target and identification testbed. It provides a scenario builder and target generator, as well as realistic models for the ownship perceptors and actuators, and in particular a high-fidelity simulation of the ship's sensors.

- **Multi-Sensor Data Fusion and Situation Assessment:** The Multi-Sensor (or Multi-Source) Data Fusion (MSDF) subsystem receives positional and attribute data from sensors (radars, electronic support measures, and identification measures) and combine them to automatically extract an optimal estimate of the position, kinematic behavior, and identification of all objects surrounding a single ship.

- **Command and Control System:** The main tasks of the Command and Control System are to understand the tactical situation taking place, to evaluate the threats in the environment, and to take appropriate actions against each threat by allocating the ownship resources. As this system is considered as the heart of SRTE, we review it in the next section in more details

3 A MULTI-AGENT APPROACH TO THE COMMAND AND CONTROL

As previously stated, the tasks of the command and control system are to understand the tactical situation taking place, to evaluate the threats and to take appropriate actions. In SRTE, these tasks are accomplished by the Situation and Threat Assessment (STA) and Resource Management agents. The C2 part in SRTE has been implemented using a Knowledge-Based System (KBS) in which the STA and resources management (RM) functionalities are represented as agents (Duquet, Bergeron, Blodgett, Couture, Chalmers, and Paradis 1998). This KBS shell has been developed on a blackboard architecture which serves as a global database through which agents can exchange and process data. With the blackboard, knowledge is encapsulated into small and modular entities (called agents), each of them acting as an expert on a very specific aspect of the problem. This characteristic is particularly well suited for STA, whose goal is to refine target identity and situation interpretation using several types of knowledge sources and inference rules. The STA agents act on a data type of any object as follows: once a Track object is instantiated on the blackboard, it is assigned specific pieces of information, either through its attributes or through functional relations (identity, position, engaging parameters and so on). This data then triggers the activation of a sequence of agents who can either modify some elements of the track, or act on another data type. The objective here is to evaluate threats and to rank them. Once the threat list has been produced through STA agents, it is given to Resource Management (RM) agents which takes care of the engagement itself, and whose role is weapon assignment.

We view here RM as a coordination process and this process is considered as the act of managing interdependencies between agents' activities as specified by Malone (Malone and Crowston 1994). We propose in the context of RM, five components of coordination and coordination processes that are associated to these components : the situation, goals, plans, agents and resources (see Table 1). All five components are necessary for a situation to be analyzed in terms of coordination.

Components of coordination	*Associated coordination process*
Situation Assessment	Multi-source data fusion, situation and threat assessment
Goals	Identifying goals
Agents	Mapping goals to agents (goal allocation and negotiation)
Plans	Mapping goals to plans (planning)
Interdependencies between plans	*Managing* interdependencies (resource allocation, sequencing, and synchronization)

Table 1. Components of coordination

As previously stated, STA assesses the current situation and consequently they also determine the goals to achieve since each situation is sustained by some specific goals. In conformity with components of coordination as specified in Table 1, it remains to elaborate plans that match those goals. We achieve that by a set of planning agents on the blackboard. Agents here are entities reflecting "expertise" in the sense of a blackboard and not really of "autonomous agents" as is the case for classical multi-agent systems. These agents have in charge deliberative planning, reactive planning, plan selection and plan instantiation. For the deliberative planning, the blackboard system uses parameters provided by the STA as well as physical constraints on the ownship resources (weapon speed and range, blind zones) to compute a list of possible engagements for a subset of the targets on the threat list. This list is then refined into an incompatibility graph or contingency plan, resulting from accepted doctrine (e.g. shoot-look-shoot) that forbids some concurrent engagements. Once this tree is produced, the next steps are the computation of an initial plan over a given time horizon, and eventually the optimization of this plan using a search algorithm together with some quality criteria for a given plan.

Because this procedure is very demanding in terms of CPU resources, the plan cannot be continuously recomputed. Therefore additional monitoring processes must be put in place to evaluate if the plan is still valid with respect to the evolving situation, or if it has to be dropped and replaced by the always reactive plan. This ensures that the system is always able to provide ”anytime” answer to the current situation. Clearly in this case, a good answer now is often preferred to a perfect answer later.

In our system, the operator is given the choice to apply or drop the proposed plan. This plan can propose early engagement of nearby, lower-threat targets, or assign fire channels depending on expected position of intercept rather than on the current position.

3.1 Acting as a Teamwork

The ship’s command structure is organized hierarchically. Although the Commanding Officer (CO) is responsible in all respects, he normally delegates control and charge of the ship to personnel of his choice to allow the most efficient deployment of the ship. Effective tactical C2 is the result of coordinated team effort and communication among its members is critical in sharing information relevant to the mission and the decision-making tasks involved. To achieve that, operators must: (i) continuously scan consoles and monitor communication nets for significant events and alerts; (ii) exchange information among themselves or pass information up the chain of command, (iii) issue or respond to orders depending on an operator’s position and role in the chain of command; and (iv) focus attention at any given moment among several competing stimuli and divide attention between several competing or complementary multiple tasks in response to operator-specific goals.

The SRTE enhanced by considerations on planning helps to coordinate teamwork onboard since it supports operators at least in: (i) the integration or fusion of data from the ship’s sensors and other sources; (ii) the formulation, maintenance and display of an accurate dynamic situation picture, leading to enhanced situation awareness; (iii) the identification and selection of courses of actions in response to anticipated or actual threats to the mission; and (iv) action implementation once a decision to act has been made and is being carried out.

Finally, our system provides the ultimate time, i.e. the critical time window of engagement, where the ship should be viewed as a tightly coulped team with one head (possibly the commander or the system itself) and many “reactive” arms (representing the resources of the ship). A view of this specific team is illustrated in Figure 2. Preliminary results of simulation of this specific teamwork are promising particularly with regard to real-time constraints.

Figure 2. C2 Teamwork in the ultimate time.

4 RUN-TIME RESULTS

Even though complete results about the performance of this approach are not yet available, it is worth presenting here an overview of its potential as a promising approach for the present C2 application. The basic version turning around STA/RM system as described here comprises about 40 agents permanently residing on the blackboard, and acting on each incoming data (track update) when the appropriate context is present. A typical scenario has been used to validate the basic STA/RM approach. This scenario features 6 air targets : 2 of them are commercial and the remaining are hostile and flying towards the ownship. One of the hostile targets also launches a missile during the scenario. These targets are seen through 4 types of sensors (2 radars, ESM, IFF), and are continually evaluated through STA and RM agents, and eventually engaged when closing on the ownship. Those targets, plus the missiles fired by the ownship, generate about 2.5 track updates per second per track during the whole scenario, which simulates a 500 second engagement. After completion of this scenario, some 5500 track updates have activated blackboard agents about 250 000 times. To isolate the run-time behavior of the STA/RM portion of SRTE, the scenario has been executed in open-loop mode on a single UltraSparc processor, taking its input from files recorded during a previous (closed-loop) run. Under these conditions, the STA/RM system has shown, that it can process the 500 second scenario in about 30 seconds. Similar experiments were performed with 30 target, the processing speed being about twice as fast as real-time in this case (i.e. 100 second scenario processed in about 50 seconds). No deviation from real-time output or significant slow-down was observed at any point in these scenarios, at least on time scales larger than 1 second.

The enhancement of SRTE by deliberative planning, task allocation and intelligent scheduling shows efficiency and flexibility, but only if the target is detected very early (generally by communication coming from friend and reliable sources). Otherwise, the enhanced SRTE can provide: (1) an anytime plan and scheduling for intermediate situations where real-time constraints are not so hard; (2) reactive plan and scheduling for hard real-time constraints.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we briefly described a simulated real-time environment (SRTE), a testbed for the development and validation of intelligent approaches for naval Command and Control Systems. We have explained how this environment deals with the Situation and Threat Assessment and Resource Management using a real-time KBS shell as an Engine. To achieve that, we have argued for anytime planning and scheduling and a task allocation based on “yellow pages” for increasing reliability and flexibility.

Current results clearly demonstrate the potential of this system built on a blackboard architecture for the assessment and planning on one side, and considering resources as agents interacting each with other through a supervisor and yellow page agent for task allocation and distributed scheduling on the other side.

Future versions of this testbed should include advanced functionality, including softkill management, as well as more sophisticated RM planning and scheduling, which really optimize the resource allocation, and which will therefore allow to take into account more real-time constraints.

ACKNOWLEDGEMENTS

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

REFERENCES

Chaib-draa, B. (1995). Industrial applications of distributed AI. *Communications of the ACM 38*(11), 49–53.

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*, Volume 3376, pp. 198–209.

Malone, T.W. and K. Crowston (1994). The interdisciplinary study of coordination. *ACM Computing Survey 26*(1), 87–120.