INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL.12, NO.2 JUNE 2007 97-106

A Distributed Intelligent Tactical Sensor Management System

Abderrezak BENASKEUR, Peter MCGUIRE, Robert BRENNAN,
Geoffrey LIGGINS and Peter WOJCIK

Abstract- **In this paper, we report on the project initiated by Defense R&D Canada - Valcartier that is intended as a vehicle to assess holonic control as a means of improving tactical sensor management for distributed military surveillance operations. Three levels of sensor management are considered: sensor, platform, and group. The proposed design is used to develop a simulation using a military scenario in which the holonic control system is employed in the sensor management role. The results of this simulation are presented.**

***Index Terms*—Sensor Management, Holonic Systems, Simulation**

1. INTRODUCTION

The military typically operate in demanding, dynamic, semi-structured and large-scale environments. The nature of this operating environment makes it difficult to detect, identify and monitor all targets in the Volume Of Interest (VOI) [1]. A key challenge facing the military is how to make the most effective use of the available scarce sensing resources when they are distributed across a large area. Military platforms, such as ships, planes and helicopters, are generally outfitted with surveillance sensors that provide a wealth of data when properly managed. Historically, interpreting this data and managing the sensors was done manually, however, this has become difficult, if not impossible, due to the complexity of modern sensory systems.

Sensor Management (SM) determines the utilization of the sensing resources in a manner that improves the quality of the acquired data. While SM is not a necessary part of the fusion process, it aids the fusion performance by improving the quality of data provided. This ultimately leads to an improvement of situation analysis.

The recursive and hierarchical structure of Holonic Control (HC) and its ability to generate dynamic linkages to form an impromptu command structure to perform a task [11] make it a very promising approach to military tactical SM. In this paper we investigate how the HC methodology can be applied to SM.

In this paper, we report on the design and testing of a holonic control architecture for tactical SM. To evaluate the performance of this approach, a scenario was developed involving a group of military platforms, located off the coast of Canada, that are tasked with conducting surveillance operations for force protection in the port of Victoria, B.C. A control strategy is employed that maintains high quality tracks for targets that pose an actual threat, while lowering quality for all other tracks.

The paper begins with background on sensor management in military settings and on holonic control. Next, we provide a general overview of our proposed HC architecture in Section 3. Section 4 provides a summary of the tactical SM scenario used to assess HC in this domain, which is then followed by our experimental results in Section 5. The paper concludes with a summary of our work in this area as well as the next steps in this project.

2. BACKGROUND

Modern military platforms are generally outfitted with a set of sensors that provide a wealth of data when properly managed. A key challenge of the management of these sensors however is the focus of attention and effort: *i.e.*, how one makes the most effective use of the available, but scarce, resources to gather the most relevant information from a dynamic environment. In this section, we begin with some background on the military SM problem then introduce holonic control as a potential solution to this problem.

2.1 Sensor Management

The objective of any surveillance mission is to gather information about the presence and activity of all objects within the VOI. The information gathered is used to build a representation of the situation of interest. SM aids the surveillance process by directing sensing resources in a manner to acquire data that is the most relevant to mission objectives. The military organization by its very nature has a hierarchical command structure to manage its resources, men, and equipment. SM, a subset of this command structure, is not only hierarchical but also recursive. An example of such a recursive hierarchy is shown in Figure 1 for a typical naval Task Force configuration.

Figure 2 summarizes the different SM problems independently of the level they belong to. These problems are described in more details below. Note that this list is not intended to be exhaustive.

- Allocation – This is concerned with the determination of which sensing resource, or set of resources, to use

Manuscript received September 29, 2006; revised January 29, 2007. This work was supported by DRDC under PWGSC contract W7701-042527/001/QCA. This paper is extended from “The application of holonic control to tactical sensor management” published at *IEEE 2006 Workshop on Distributed Intelligent Systems Collective Intelligence and its Applications,* Prague, Czech Republic, 15-16 June, 2006.

A. Benaskeur is with Defence R&D Canada, Valcartier, Quebec, Canada (e-mail: Abderrezak.Benaskeur@drdc-rddc.gc.ca); P. McGuire and P. Wojcik are with C-Core, St. John’s, Newfoundland, Canada (e-mail: peter.mcguire@c-core.ca and peter.wojcik@c-core.ca); R. Brennan is with the Schulich School of Engineering, University of Calgary, Calgary, Canada (e-mail: rbrennan@ucalgary.ca); G. Liggins is with Orica Mining Services, Dever, CO, USA (e-mail: Geoff.Liggins@orica.com).

to achieve the sensing objectives. By breaking these objectives down into a series of tasks, the SM needs to determine the most suitable resource to allocate to each task.

- Coordination/Cooperation – If a sensing resource when in operation is in conflict with other resources then the SM must determine which resource is more important and prevent the others from operating or must allow for some schedule to allow one resource to operate for a period of time and then the other. This defines the coordination, or conflict resolving, problem. Dual to this problem is the cooperation, where synergy among complementary resources is maximized by the SM module.

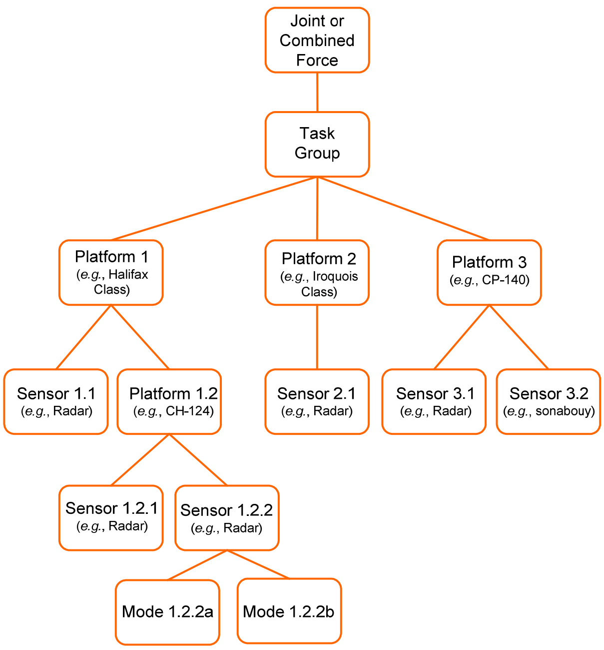

Fig.1. Hierarchy of naval sensing resources

- Scheduling – Scheduling is the designation of time segments to specific tasks or activities, the nature of which is defined during the allocation or coordination stages. Scheduling typically uses time as its base variable; tasks are expected to start at a specified time and to execute for a fixed time interval.
- Mode Control – In case of sensors offering multiple modes, the SM should make use of the most optimal mode for the tasks being done provided that there is no other overriding reason not to.
- Mode Switching Control – While changing sensor modes, the data stream may be halted during the transition. The SM must address whether it is more important to maintain operation in possibly a sub-optimal mode while maintaining a live data stream or to change to a more optimal mode.

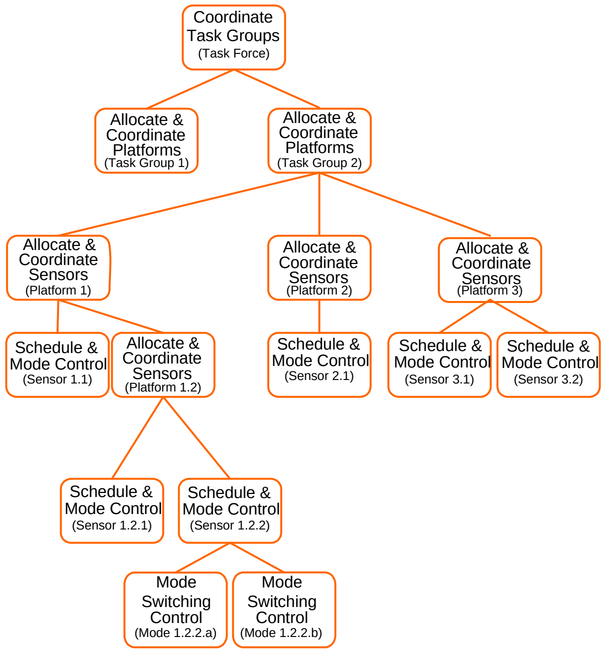

Fig.2. Hierarchy of sensor management problems

- Others Problems – Other potential issues in tactical surveillance, for which strategies within SM are required at all levels, would include: emission control: (SM system must trade off the gathering of more complete information using active sensor over self-security); failure recovery (SM must alter the sensing allocation and schedule in case of disabled or diminished sensing capability); and contingency handling (SM must address when and how to make the necessary changes if situation/objectives change).

Currently, SM is performed using a centralized architecture. SM requires a control architecture that matches the underlying command structure (see Figure 1). At the highest level, decision-making is based on very high-level information. As one descends the tiers in the hierarchy, the decision-making becomes more focused. As data moves up the hierarchy, it is transformed into information necessary for high-level decision-makers. The selection of the appropriate control architecture, to address the SM problems, is discussed the next section.

2.2 Holonic Control

In distributed control schemes, each node in the architecture has a controller that allows it to works collectively with its neighbours to achieve some overall goal. There are several organizational structures that allow nodes and their controllers to work together. Given the hierarchical and recursive nature of the SM problems, the desirable characteristics of a control architecture to address them are: 1) hierarchy to account for a clear chain of command; 2) adaptability to the current situation; 3) sufficient autonomy of each node to perform its function without being encumbered from actions taken at the top level; 4) sufficient robustness to maintain operations even if (elements of) the network are incapacitated; and 5) recursiveness: where each node could be composed of one or more nodes of a lower abstraction level. For the tactical SM problem, five architectures were considered as summarized in Figure 3.

The centralized approach was felt to be unsuitable because it requires that a central node be kept intact at all times: this is a significant risk in the military context. It has the advantage of relatively simple control; however, if the situation for which it is configured changes then it requires a massive effort to reconfigure it. Centralized architectures are characterized by high communications requirements, a high computational burden at the central node, and a lack of general robustness and flexibility.

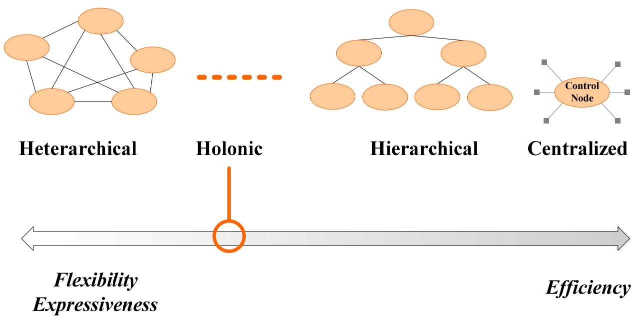

Fig.3. Spectrum of control architectures

For military applications, the fully decentralized, or heterarchical [5], approach was also felt to be unsuitable because of its lack of structure. This makes the heterarchical architecture relatively robust because there is very little to break, but also makes it very difficult to control, which can lead to an undesirable chaotic behaviour.

Hierarchical architectures enforce a top-down decomposition of tasks and division of labour approach. This structure is good in that it forces an expected behaviour; but it is inflexible and branches can become uncontrollable if an intermediate element is incapacitated. The degree of autonomy of an element in a hierarchy is quite limited. The top-down approach is convenient for planning purposes and the dissemination of instructions and goals. However, if the situation changes significantly then a new entire plan must be derived, which can be a significant computational and communications burden.

Federated architectures (e.g., [7]) provide a compromise between the hierarchical and the heterarchical structures. Like the heterarchical approach the nodes have a high degree of autonomy but form a structure through communications and the use of specialized middle nodes. This approach has improved robustness and flexibility over the other architectures but does not allow for the dynamic restructuring that is an integral part of holonic architectures, described below.

For this application, a holonic architecture [3],[11],[13] was chosen primarily because it is a hybrid approach that takes the best of different architectures and avoids many of their pitfalls. The holonic architecture takes advantage of the distributed capabilities from classical Multi-Agent Systems (MAS) [12] while incorporating the benefits of the hierarchical command structure that allows for strong goal orientation. For example, agents can be used to implement recursiveness when organized in teams [6]. This architecture is discussed in more details in the next section.

Holonic systems have their origins in the work of Arthur Koestler [8]. To explain complex biological and social systems, he makes two key observations: 1) these systems evolve and grow to satisfy increasingly complex and changing needs by creating stable intermediate forms which are self-reliant and more capable than the initial systems; and 2) in living and organizational systems it is generally difficult to distinguish between wholes and parts: almost every distinguishable element is simultaneously a whole (an essentially autonomous body) and a part (an integrated section of a larger, more capable body). To explain this concept, Koestler suggested a new term: holon, from the Greek *holos* meaning whole and the suffix *on* implying particle as in proton or neutron.

A Holonic System (HS) consists of autonomous, self-reliant units, called holons that co-operate to achieve the overall system objectives [2]. Some key properties of a HS developed from Koestler's model are [9]: 1) autonomy - the capability of a holon to create and control the execution of its own plans and/or strategies (and to maintain its own functions); 2) cooperation - the process whereby a set of holons develop mutually acceptable plans and execute them; 3) self-organization - the ability of holons to collect and arrange themselves in order to achieve an overall system goal; and 4) Reconfigurability - the ability of the function of a holon to be simply altered in a timely and effective manner. Another important holonic concept is the notion of functional decomposition. The complexity of dynamic systems can be dealt with by decomposing the systems into smaller parts. A consequence of this is the idea that holons can contain other holons (*i.e.*, they are recursive). Problems are solved by holarchies (hierarchies of holons), or groups of autonomous and co-operative basic holons and/or recursive holons that are themselves holarchies. The recursive and hierarchical structure of holonic architecture and its ability to generate dynamic linkages to form an impromptu command structure to perform a task make it well suited to the above described military SM problems.

The common thread that runs throughout the work on HSs is the close link between MAS and HSs. Given this link, various co-operation, communication and organizational techniques from the MAS world can be used to implement autonomous, co-operative and recursive agents (*i.e.*, holons). HS can be considered a general paradigm for distributed intelligent control, whereas MAS are regarded as software technologies that can be used to implement HSs. There are similarities and differences between holons and agents. One of the major difference concerns the recursiveness. A holon may be composed of other holons, while there is no recursive architecture as such in MAS.

3. The Holonic Control Architecture

The proposed HC architecture for SM in military settings is decomposed into three main levels: sensor, platform, and group. The levels are related to each other in a recursive hierarchical manner typical of holonic systems [11]. The sensors represent the lowest level of the hierarchy. Each platform coordinates the sensors that are located aboard it, but does not control the sensors aboard other platforms. Likewise, the group level manager coordinates sensing activities between platforms but does not directly manage the sensors aboard those platforms.

In this section, we begin with a general overview of the HC architecture then look more closely at how the specific issues of timeliness and task negotiation can be addressed in a military setting.

3.1 The Tactical SM Architecture

At the lowest level of the SM holarchy, each sensor is assigned a local controller that manages the utilization of that particular resource. These sensor controllers interface with the platform level SM, which in turn, interfaces with the group level SM.

The platform SM's coordinate sensing tasks aboard each platform and take on most of the management responsibilities. So far, the responsibilities have been limited to include: searching for and tracking of targets (sensor allocation), sensor configuration adjustments (mode control), and intra-platform cue and hand-off (cooperation and coordination).

Despite being at the top of the holarchy, the group level SM role is not so much to control the sensing activities of the platforms but rather to assemble tracking data gathered by the platforms and coordinate inter-platform sensing activities such as cueing and hand-off of tracks. The group level SM may be thought of as a military command center and may be located aboard one of the platforms or may be in a separate location. In the design presented here we concentrate on the second option.

The control strategy is designed with as a main objective to maintain tracks on all targets in the area of interest while searching for new targets. Special attention needs to be given to those targets that are judged to be threatening to High Value Units (HVU) and to those areas where new targets are expected to appear.

Each level in the SM holarchy consists of a single holon, itself containing a number of holons (*i.e.*, a recursive architecture). The general form of resource holon architecture is illustrated in Figure 4.

The Service Interface Command Holon (SICH) acts as the negotiator between holarchy levels and is also responsible for Situation Analysis (SA), while the Resource Holons (RHs) represent the services available within the level. RHs may consist of a subordinate holarchy, such as a sensor aboard a platform, or a platform as a member of a group. Task Holons (THs) are generated by the SICH and are responsible for utilizing the resources to meet level-specific sensing objectives. RHs are responsible for implementing the requested tasks.

As an illustration, consider a platform level holon that represents an entire platform (such as a frigate). The associated RHs represent the resources aboard the platform such as sensors, communications, power, etc. THs act as autonomous agents aboard the platform and are charged with a specific task such as "track this target" or "perform a search operation in this sector". To perform their respective tasks, THs must negotiate with the resources. They are created by the SICH and typically exist only while they are needed. For example, a holon tracking a single target exists while the target is in the sensing domain of the platform and can be tracked. Once the target leaves this region, the associated tracking holon is no longer useful and is therefore terminated.

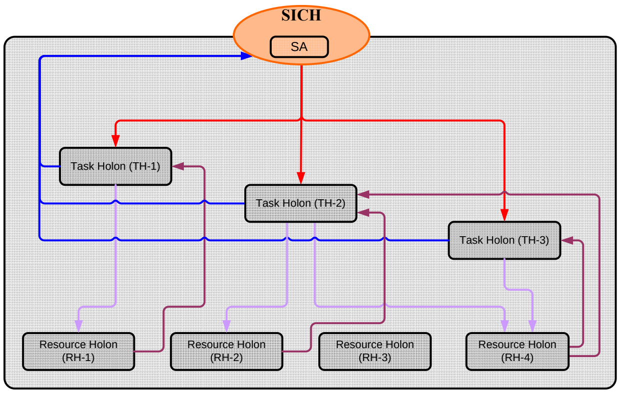

Fig.4. Generic resource holon structure

The SICH and the THs work together to implement SM. More specifically, the SICH acts as a task planner and the THs perform task allocation and scheduling. Typically the SICH will plan tasks that address sensing objectives and each task plan will lead to the creation of a TH. Utilization of the resources (sensor allocation) is left up to the TH negotiation process. The SICH plans are devised based on a SA and the predicted evolution of the situation. Clearly the predictions will not always be accurate and a revised task plan will be generated. In the interim, THs attempt to implement existing task plans using available resources.

Communication between THs and SICH is assumed to take place over a very high bandwidth medium and is therefore not modeled. Additionally, communication limitations between THs at the platform level and the sensors are neglected. However, the communication limitation between the platforms and the group level SICH is addressed by introducing a Communications Holon (CH) that manages the (typically low-bandwidth) wireless connection between them.

3.2 Control Sensor Intervals

Given the nature of practical SM, resources may be limited in their capacity to serve the THs and may not be able to service all of the jobs requested of them. In particular, two of these resources, the radar sensors and the wireless communications are fundamentally limited to serve one task at a time (*e.g.* update track, transmit message). Using a time-sharing strategy though, these resources can seemingly service many tasks simultaneously. For example, the time it takes to update a track is generally much shorter than the time between track updates; therefore by scheduling updates intelligently the sensor can support the tracking of multiple targets simultaneously.

To simulate this, the time-limited resources accept a number of tasks that they will sequentially execute over a predefined time interval. The time interval is chosen to be long in comparison to the typical task execution time (*e.g.* time to update a single track), but short in comparison to the task revisit time (*e.g.* time between track updates). In this way, the resource can seemingly service multiple tasks simultaneously while still being able to quickly address new tasks as they arise (*i.e.*, adapt to the changing environment).

This control interval concept is a departure from the holonic strategy of event-driven processes, in particular the IEC 61499 based holonic architecture proposed by Christensen [4] but is adopted here mainly due to the project schedule constraints. It is assumed that the sensors are capable of scheduling tasks over short time periods while the HC is used to plan, allocate and schedule tasks over longer time periods. This simplification is a realistic one that demonstrates that HC can be implemented using resources that are not designed with a holonic or event-driven control strategy in mind.

The imposed control time interval is limited to wireless communication resources and the sensors. A control interval is not imposed on the generation of THs at the group or platform level. As a result of the time interval at the sensors, platform level THs can only interface with the sensors once during each control interval. Likewise, tasks at the group level can only access the wireless communications once per control interval. The control interval of the communications and the sensors are not linked. In fact, each resource may have a unique control interval.

3.3 Task Negotiation

As noted previously, THs negotiate with the RHs available to them to achieve their task goals. This negotiation is most significant between platform level THs and the sensors. THs must also negotiate with the communications resource (transmitter) to send messages between the group and platform levels.

The basic negotiation strategy for THs is as follows. At each time instant the TH evaluates its need for access to resources. If needed, THs request a service quality estimate from each resource. RHs then return Quality Of Service (QOS), a measure of how well the service can be performed (if at all). This is a quality of service measure, not a measure of resource availability. THs then request service from the resource that provides the best QOS. Finally, RHs return an accept/decline indication. Acceptable QOS limits are currently pre-set in the simulation.

The negotiation process is complicated when there are multiple THs and multiple sensors. In this case the holons attempt to get service from the sensors offering the best QOS while the sensors attempt to service the tasks in order of priority. Since the sensors are limited in the number of tasks that can be serviced during a control interval a number of iterations may be required to achieve the best matching of tasks with resources. This iterative process may be trivial (no iterations) if the sensors are underutilized and available to service all jobs requested of them. On the other hand, when sensor loads are high, it may take much iteration to derive an optimal solution. The negotiation process occurs each time a resource control interval begins. The negotiations will be assumed to occur instantaneously, though in practice negotiations consume a finite amount of time and may in some situations affect SM performance.

It should be noted that the result of the negotiation is the allocation of sensing tasks to individual resources. This is a one-to-one pairing (*i.e.*, the THs only get service from one sensor at a time) and does not preclude the use of multiple sensors by individual THs since the pairing is renegotiated at the beginning of each sensor control interval.

3.4 Tactical SM Control Levels

In the sensor control model presented here, the platforms perform most of the sensing activities, which includes detecting targets, tracking targets, and modifying sensor configurations based on analysis of the local situation. In this model the group level SM role is to selectively acquire data from the platforms, and modify platform sensing-operations based on a group level situation analysis. The sensors themselves are responsible for managing the various tasks specified by the platform. In this section, we summarize how each holon, illustrated in Figure 4, is used to perform this functionality.

The sensor level SM is concerned with scheduling the use of sensor hardware. At this level, the SICH is responsible for negotiating with the platform THs for task allocations as well as creating, managing, and destroying THs at the sensor level.

SA, at this level, involves converting raw sensor data into measurements representing target location. In order to accomplish this at the sensor level, three types of THs are used: search, track-update, and configuration. Search holons deliver control instructions to the sensor hardware holon (*i.e.*, the RH) and pass measurement data back to the SICH. In this case, the result will be one of detection, no-detection, or task failed to execute (*i.e.*, sensor malfunction).

Track-Update Holons perform the same basic functionality for tracked targets and result in a track initiation, track update (*i.e.*, converted to update form at SICH), missed update (only model-based prediction), or a task failed to execute message (*i.e.*, sensor malfunction). Finally, the configuration holon delivers control instructions to sensor hardware holon and status information back to the SICH, resulting in a coverage adjustment complete, or sensor inoperable (*i.e.*, sensor malfunction) response.

The platform level holon represents a military asset (*e.g.*, frigate), and as such, contains sensor holons as resources (RHs). Although the basic structure is the same as that at the sensor level, the functions of the THs in particular are much different. As well, one additional RH is included at this level, a Communication Holon (CH) that is used to simulate a wireless communication link (*e.g.*, link 11 [10]) between the platform and the group.

At the platform level, SA involves analyzing the incoming data and making assessments of the evolving situation. This requires the SICH to balance resource usage amongst the tracking tasks. For example, targets that are judged to be threatening to the platform are to be tracked more closely than non-threatening targets (*e.g.*, a fast approaching target would be deemed threatening while a retreating target would be deemed non-threatening). To achieve SA, the SICH manages four types of THs at this level: tracking, searching, configuration, and message.

Configuration holons are generally the highest priority holons since they are responsible for establishing the platform's sensor modes for the current sensing objectives. Based on these objectives, search holons are created to detect new target's in the platform's area of operations. For example, a platform participating in a military operation may want to continually look for targets approaching from ahead; alternatively, a particular target may be expected in a predetermined time and location, in which case the corresponding search holon would be created for just the expected time window.

Once a target is detected, a tracking holon is created that subsequently acquires updates in order to maintain the specified track quality. The tracking calculations are performed by the tracking holon and the results (*i.e.*, position, velocity) are reported to the platform SICH as they are available. Finally, the SICH communications within the group level require the creation of a message holon. These holons interface with the communication resource where they are scheduled on a priority basis.

Group level SM is the most abstract level of SM considered here. The group level SM acts as a coordinator between multiple platforms, where each platform implements its own SM. SA at the group level is based on information provided by the platforms, creating a broader picture of the environment than is available to any of the platforms individually. For example, the target threat assessments from the platform level SA's are used to determine the overall tracking and hand-off strategy between platforms; they may also be used to assign positions for the platforms in the area of operations. Due to the limitations in the group-platform communication link (*i.e.*, wireless) sensory data is analyzed at the group level and inter-platform coordination is achieved by the issuing of tasks to the platforms. With target tracking for example, data sent from one platform may indicate a track threatening a second platform. The group level SM would recognize this situation and issue a search task to the second platform.

Three main tasks at the group level are searching, tracking, and configuration adjustments. The nature of these tasks differs from their counterparts at the platform level. In searching, for example, the group level SM can use tracking information from one platform to predict the arrival of new targets at a second platform. With this knowledge a search to acquire tracks can be initiated on the second platform in advance of their arrival. Once a search is initiated the group level SM can only periodically monitor the platform as it conducts the search.

Subsequent to any track initiation at the platform level, a message will be sent to the group SM with the track information. In response, the group level SM will begin a process of periodically monitoring and requesting new track data from the platform that initiated the track. The group requests these updates with a frequency that is based on an assessment of the track importance (*e.g.*, threat) relative to the other tracks it is aware of.

Platform sensor configuration is primarily the responsibility of the platforms themselves. However the group level SM may be in a position where information gathered by one platform is used to suggest a configuration adjustment at another platform. In these cases, the platform must balance its own configuration assessment with that of the group and reconfigure the sensors accordingly. One example of this is when targets are approaching a platform sensing domain but beyond the range of the sensors in their current configuration. In this case, sensory data from a second platform may be used as the basis for requesting a configuration adjustment.

In addition to monitoring track and search tasks, the group level SM must respond to the situation analysis provided by the platforms and use this information in its own assessment of the larger situation. This primarily comes in the form of messages sent by the platforms that indicate some critical situation arising. Most notable among these would be the loss of a track or the prediction of an impending cue/hand-off event. In this case, the tracking holon responsible for a transiting target usually can handle the hand-off by simply acquiring service on a different sensor as the situation warrants.

4. Evaluating the Holonic SM

4.1 The SM Scenario

In order to demonstrate the holonic control, the scenario “Surveillance/Control of Canadian Territory and Approaches”, from the Departmental Force Planning Scenarios, was identified as a suitable candidate. The scenario has been modified to better illustrate elements of the SM design and force protection problematic.

The area of interest is limited to the port of Victoria, British-Columbia, and surrounding sea approaches. The surveillance region of operation is limited to the triangular section of sea bounded by the coastline and the predefined limits of responsibility as depicted in Figure 5. This area will be monitored with a group of platforms and a single ground station located at Victoria. The platforms perform a search and cue the ground station upon detection. In order to reduce its signature, the ground station is set not to perform target search itself, but rather rely on the referral (cue/hand-off) from the platforms. A scenario with more coverage could have been used (*e.g.*, overlapping sensor ranges and increased numbers of platforms at critical areas), however a more limited arrangement was used to test the limits of the system with resource constraints (not atypical of military operations).

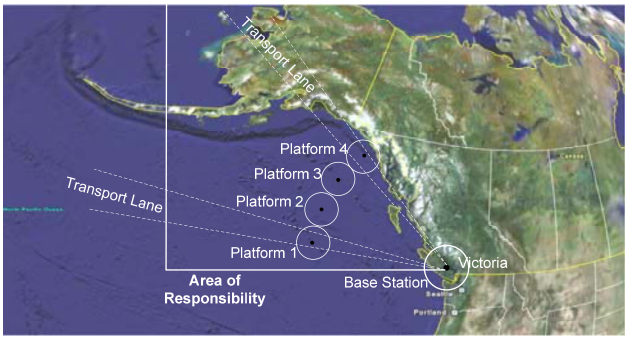

Fig.5. Tactical sensor management scenario

One of the main difficulties with this mission is the presence of a large number of spurious objects. Objects are constantly entering and leaving the sensing domain of the platforms, and it may become difficult or impossible to track all of them simultaneously while maintaining a search capability for new objects. The proposed SM aids this process by tailoring the use of the sensing resources based on an assessment of the likelihood of the target being a threat. Without this feedback, all targets would be given equal attention by the sensing resources. With feedback, those targets that are deemed unimportant to the force protection mission are tracked less closely providing greater sensing resources for the important ones (*i.e.*, focus of atention). This provides significant advantage as the number of targets to be tracked increases.

The mission employs four platforms, each of which is equipped with a single Electronically Scanned Aperture (ESA)-type sensor. Figure 5 illustrates the placement of the platforms within the covered area, beyond the sensing range of the base station. The sensing ranges of these platforms do not overlap. The platforms are positioned such that any object that approaches Victoria within the area of responsibility will be likely to cross their detection region. This arrangement of platforms provides early detection of potential threats, without presenting a large radar signature from the ground station. The ground station only tracks the most significant objects, while the platforms will track all objects within their sensing range. To reduce emissions, these platforms will use only 20% of their sensing capacity to perform their tasks.

4.2 Simulation Design

The simulation is implemented in the MATLAB programming environment. Although the holonic design relies on agents (holons) that are independent, the simulation does not employ object-oriented programming. Rather, the simulation is programmed using a functional style, where the independent agents are themselves simulated. This is the approach that many video game developers use due to its low computational overhead as compared to object-oriented programming.

The simulation proceeds in increments of Δt. This time increment is set to be very small: much smaller than the time it takes to complete even a simple sensor scan. Although it is a user configurable parameter, for the purposes of this simulation it is set to 0.01 seconds. In contrast, a sensor control interval will take 0.1 seconds and can perform several tasks in this time.

At each of the time increments, the full simulation loop shown in Figure 6 is performed. The platform’s service interface command holon (SICH) algorithms are run, the platform’s task holons are executed and negotiate, through a broker, for time on the sensors (if they are ready to negotiate), the sensor simulation proceeds by Δt (including updating results of finished scans) and the positions of platforms and targets are updated. At each Δt, this cycle repeats until a preset total simulation time is reached.

Aside from the main simulation loop, there are three main entities in the simulation. These are the targets, the platforms and the sensors. The targets can vary in size and speed, as can the platforms. The sensors, however, are restricted to one type: Electronic Sensor Arrays (ESAs). This simplifies the programming, while not limiting the simulation functionality.

The simulation is restricted to single target type, namely aircraft. Targets are grouped into two categories, incoming (headed toward Victoria) and outgoing (headed away from Victoria). Incoming targets not in the transport lanes will be rated highly likely to be threats, and need to be tracked closely by the platforms. Outgoing targets and targets in the transport lanes have their threat level set to zero: they serve as a complicating factor in the mission as they must also be tracked though they are not as important to the mission. They also increase the sensor loading, thereby reducing the amount of sensor time available for searching. An additional important incoming target designated "the actual threat" is simulated with a heading directed towards Victoria.

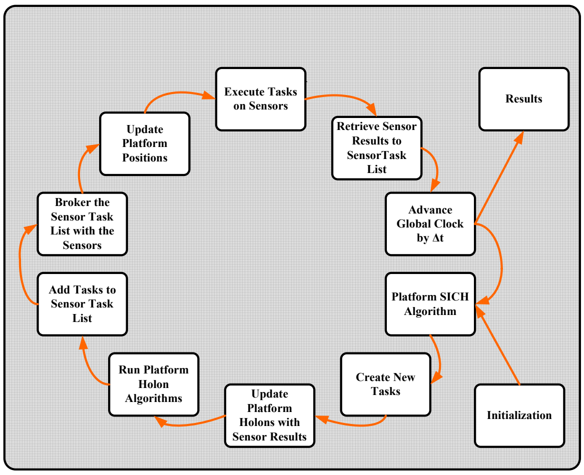

Fig.6. Simulation program flow

Each target is given a heading that sends it directly towards or away from Victoria; however, each heading is also subject to a random perturbation of up to +/- 10 degrees. The simulation is run for a time that is sufficient for targets to pass completely through the platform sensing domains and provides a rapid increase in tracking load as the targets enter the sensing domains. The general approach employed is to simulate relatively few incoming targets and gradually increase the number of outgoing targets in each experiment. Different simulations, with varying incoming/outgoing threat ratios (5/10, 5/21, 5/39, 5/75, 5/95, and 5/195) have been run. This setup provides suitable configurations for evaluating mission performance as the number of objects is increased.

5. Experimental Results

Several differing methods were identified in order to evaluate the holonic control. Comparison of the holonic architecture in operation with, and without, threat assessment feedback further illustrates the utility of such closed-loop control. In addition, this comparison demonstrates the capability of the holonic architecture to utilize high-level situation analysis for resource management purposes. The experimental results reported in this section were performed to assess the utility of HS in these two modes (feedback and non-feedback). HS performance was assessed along typical SM metrics: platform load, tracking performance, search performance, surveillance performance, and actual threat tracking performance.

5.1 Resource Usage (Platform Load)

Resource usage is measured in terms of the fraction of the available sensing resources available at any given time. In the results reported here, platform load is averaged over several platforms to provide a composite measure. Figure 7 shows the platform load, averaged over the lifetime of the simulation, as a function of the number of outgoing targets. The feedback configuration maintains a lower average load and its load increases less quickly than in the non-feedback configuration.

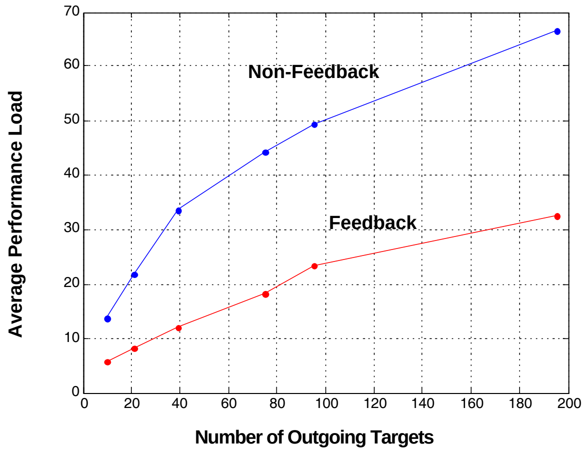

Fig.7. Platform loading

5.2 Tracking Performance

Track holons are created with a desired track quality attribute $Q_{desired}$. Their performance Q_{track} is computed according to the fractional difference in the desired and actual quality. For comparison purposes a sum, across all tracks (N) maintained by the group, is used.

$$TP(t) = \sum_{i=1}^{N} TP(t,i) = \sum_{i=1}^{N}\left(1+\min\left[\frac{Q_{track}-Q_{desired}}{Q_{desired}},0\right]\right)$$

From Figure 8, it can be observed that the non-feedback configuration performs slightly worse than the feedback configuration. The difference is most pronounced during the highest loading periods in the first half of the simulation. During this period, the sensors in the non-feedback configuration cannot adequately service all of the track holons, resulting in some decay in performance. The feedback configuration, on the other hand, is better able to service all of its tracks.

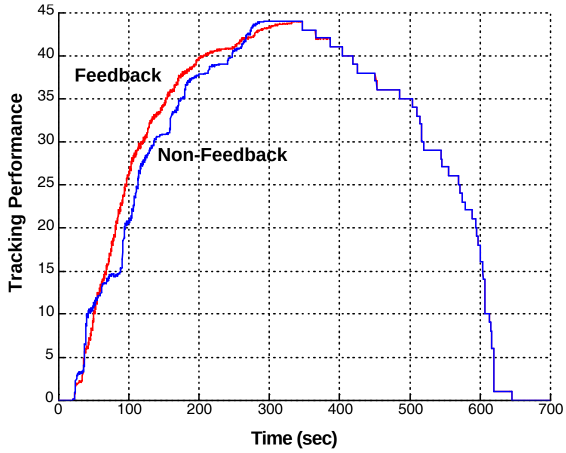

Fig.8. Tracking performance

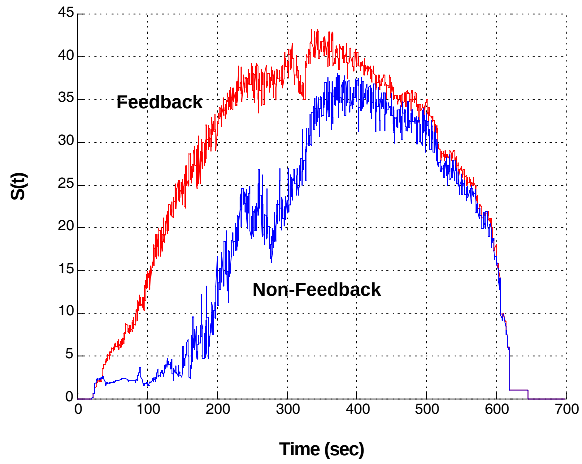

Fig.9. Scene global surveillance performance

5.3 Searching Performance

On all platforms an ongoing search task is established in order to detect new targets. The search holons will secure any sensor time not used up by tracking tasks. By subtracting the instantaneous platform load from the total sensing capacity of the platform, an instantaneous measure of search performance *RC(t,p)* is obtained. The latter is therefore inversely proportional to platform loading (see Section 5.1). The relative increase in search capacity in the feedback case results in targets being detected sooner than in the non-feedback case.

5.4 Global Surveillance Performance

A global surveillance performance, that combines the capacity to conduct target search *RC(t,p)* by a group of platforms with the tracking performance *TP(t)*, is defined as follows:

$$S(t)=\frac{1}{N}\sum_{p=1}^{N}S(t,p)=\frac{1}{N}\sum_{p=1}^{N}\left(RC(t,p)\bullet\sum_{i=1}^{N}\left[\left(1+\min\left[\frac{Q_{track}(i)-Q_{desired}(i)}{Q_{desired}(i)},0\right]\right)\bullet Z_b(t,p,i)\right]\right)$$

Here $Z_b(t,p,i)$ is the assessed threat level for track i aboard platform p at time t. The sum is conducted over the M tracks maintained by platform p at time t. An average over N platforms defines the instantaneous surveillance metric.

The increase in search capacity in the feedback configuration, leads to significantly better global surveillance score in the sensing domain of the platforms, as is illustrated in Figure 9. It is clear that the feedback configuration provides better surveillance, especially during periods of high platform load.

5.5 Actual Threat Tracking Performance

Since each simulation includes one target designated as "the actual threat", it is useful to examine the behaviour of the system with respect to this target. Figure 10 shows the track quality for "the threat" in both feedback and non-feedback configurations. Note, that while both configurations maintain a target track, the "threat" is detected much sooner in the feedback configuration. In has also been noticed that the penetration of "the threat" into the area of responsibility before detection increases as the number of spurious targets does.

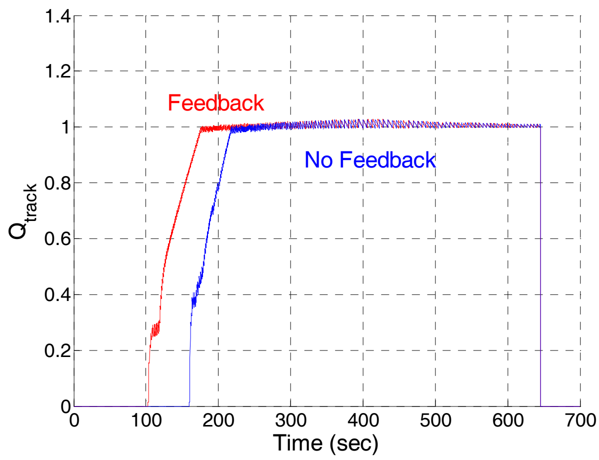

Fig.10. Actual threat detection & tracking performance

6. CONCLUSIONS

In this paper we presented a conceptual design for SM in military settings using a HC approach. Three levels of SM are considered, sensor, platform and group. Sensor level management involves the control and task scheduling for individual sensors. Platform level management involves the allocation of tasks to the sensors, control of sensor configurations, and coordinating sensing activities between sensors. Platform level SM is confined to individual platforms. Group level SM coordinates sensing tasks between platforms and controls platform sensing configurations.

Given the constraints inherent to this project, a number of aspects of SM have been simplified or not fully addressed in this design. In particular, we focus on the role of SM in acquiring and maintaining target tracks across multiple platforms, but avoid the more involved SA of target associations, track separation, and sensor emission control. It is important to note however that these issues can be addressed within the SM design proposed here and may be part of some future follow-on work.

Currently, we are working on the implementation of the HC system described in this paper with the goal of demonstrating the applicability of HC to SM. The simulation will model sensing resources of a naval group spread over several platforms (*i.e.*, ships and aircraft) as described in the scenario.

ACKNOWLEDGEMENT

This work was funded by Defence Research and Development Cansada (DRDC) through Public Works and Government Services Canada (PWGSC) contract: W7701-042527/001/QCA. The intellectual property described in this paper is the property of Her Majesty the Queen as represented by the Canadian Minister of National Defence. Any use of this material must be approved by the DRDC Scientific Authority.

REFERENCES

[1] A. Benaskeur, "Sensor management in command and control", *Proceedings of the 7th International Command and Control Research and Technology Symposium*, September 2002.

[2] S. Bussmann, "An agent-oriented architecture for holonic manufacturing control", *First Open Workshop on Intelligent Manufacturing Systems Europe*, 1998, pp. 1-12.

[3] J. Christensen, "Holonic manufacturing systems: initial architecture and standards directions", *Proceedings of the 1st European Conference on Holonic Manufacturing Systems*, Hannover, Germany, 1994.

[4] J. H. Christensen, "HMS/FB architecture and its implementation," in *Agent-Based Manufacturing: Advances in the Holonic Approach*, S. M. Deen, Ed. Berlin: Springer, 2003, pp. 53-88.

[5] Duffie, N.A., and R.S. Piper, "Non-hierearchical control of a flexible manufacturing cell," *Robotics and Compututer Integrated Manufacturing*, Vol. 3, 1986, pp. 175-179.

[6] M. Fletcher, D. McFarlane, A. Lucas, J. Brusey, D. Jarvis, "The Cambridge packing cell: A holonic enterprise demonstrator" *Lecture Notes in Artificial Intelligence*, v 2691, Multi-Agent Systems and Application III, 2003, p 533-543.

[7] B.R. Gaines, D.H. Norrie and A.Z. Lapsley, "Mediator: An intelligent information system supporting the virtual manufacturing enterprise," *Proceedings of the 1995 IEEE International Conference on Systems, Man and Cybernetics*. Part 1 (of 5), Oct 22-25 1995, 1995, pp. 964-969.

[8] Koestler, A., *The Ghost in the Machine,* Arkana, 1967.

[9] McFarlane, D. and Bussmann, S., "Developments in holonic production planning and control," *Production Planning and Control*, vol. 11, pp. 522-36, 2000.

[10] Simensen, T.A., "LINK-11 communications", *M.Sc. Thesis*, Naval Postgraduate School, Monterey, CA., 1992.

[11] Valckenaers, P., Van Brussel, H., Bongaerts, L. and Wyns, J. "Holonic manufacturing systems", *Integrated Computer-Aided Engineering*, 4, 1997, pp. 191-201.

[12] Weiss, G., *Multi-Agent Systems*, Cambridge, MA: MIT Press, 1999,

[13] Wyns, J., "Reference architecture for holonic manufacturing systems," *Ph.D. Thesis*, K.U. Leuven, 1999.

Abderrezak Benaskeur is a Defence Scientist at Defence R&D Canada – Valcartier. His reseach activities cover: Military Command & Control, Distributed Combat Power (Sensor & Weapons) Management in Naval Operations, Planning & Scheduling, Control Systems, and Adaptive Data Fusion. He holds a PhD degree in Control Theory from Université Laval, Quebec City, MSc in Computer Engineeering from Univesity o Paris XII, and BSc degree in Control Theory from University of Constantine.

Peter McGuire holds a Ph.D. from the University of Toronto in the field of aerospace engineering with a specialty in computer vision. Working at C-CORE (www.c-core.ca) he has participated in a number of image processing projects as a software designer and hardware integrator. Projects include: automatic determination of falling woodchip sizes using optical imagery, and automated inspection of hoist ropes using high-speed imagery. Peter has also collaborated on a project involving signal processing of radio channel distress calls for the Canadian Coast Guard. Recent projects completed for the Canadian military (DRDC) involve target tracking from a video stream and sensor management using Holonic control technology.

Robert Brennan is an Associate Professor of Mechanical and Manufacturing Engineering and Associate Dean (Academic and Planning) at the Schulich School of Engineering, University of Calgary, Alberta, Canada. His research interests include distributed real-time control of manufacturing systems, modelling and analysis of manufacturing systems, and manufacturing control architectures. He holds BSc and PhD Degrees from the University of Calgary. He has over seven years of industrial experience in project management and control systems and is a professional engineer and a member of SME, ISA, and IEEE.

Geoffrey Liggins is a senior research engineer with Orica Mining Services, with technical focus on system architectures for managing and automating heavy mobile equipment. He recently left his role as Technical Lead in Automated Systems at C-CORE where he had contributed to the work being presented. His work in intelligent systems and hydrodynamics has seen him work on a variety of complex control system architectures. He holds BASc and MASc Degrees from the University of British Columbia in Mechanical Engineering. He has over ten years of experience in applied research and development of intelligent systems, with a focus on field service robotics and automation.

Peter Wojcik holds M.Sc. and Ph.D. degrees in electronic Engineering from Gdansk University of Technology, Poland. He has over 25 years of diverse experience in R&D developing of advanced engineering technologies to solve problems in several industries. His research interests include control systems, signal processing and systems engineering. He is a professional engineer and a member of IEEE.