

Command and Control for Mixed Manned and Unmanned Security Forces

Daniel Carroll, Gary Gilbreath, Kelly Grant (CSC), Laura Day

Space and Naval Warfare Systems Center San Diego (SSC San Diego)

Code 2371, 53406 Woodward Road, San Diego, CA 92152-7383

619-553-7636, FAX: 619-553-6188, Daniel.Carroll@navy.mil, <http://www.spawar.navy.mil/robots/>

Computer Sciences Corporation (CSC), San Diego, CA

ABSTRACT

The Multiple Resource Host Architecture (MRHA) performs Command and Control (C2) for security robots. The MRHA allows a single human guard to oversee and monitor up to 255 unmanned vehicles and/or sensors. Lessons learned from user testing have led to C2 improvements that optimize the integration of the Unmanned Vehicle Systems (UVS) with the manned guard force.

The Mobile Detection Assessment Response System (MDARS) is a joint Army-Navy development effort to field mobile robots at DoD sites for physical security and automated inventory missions. Autonomous platforms patrol inside warehouses (Interior) and outside of storage facilities (Exterior), carrying payloads for intruder detection, inventory assessment, and barrier assessment. The MDARS Control Station is based upon the MRHA.

This paper provides an overview of the concept of operations for MDARS with emphasis on the coordination between the manned and unmanned forces. Detail is provided on the recent MRHA development including integration with legacy fixed sensors systems, seamless communication of “eyes and ears” data of the response force to the dispatcher, and resource management to handoff control of an unmanned vehicle from a remote dispatcher to an on-scene commander.

1. Concept of Operations (CONOPS)

The MRHA is a distributed host architecture geared towards a single operator controlling up to 255 unmanned vehicles and/or unmanned sensors. The MRHA is a key component of the Mobile Detection Assessment Response System (MDARS) program. MDARS employs multiple robotic security platforms operating under the high level control of a remote host, with the direct supervision of a human operator. MDARS uses unmanned security platforms for physical security. The Space and Naval Warfare Systems Center San Diego (SSC San Diego) is the system developer for the MRHA and the technical director for MDARS.

A control station directs the platforms on pre-planned patrols, random patrols, or user-directed patrols. The platforms carry payloads for intruder detection and assessment, barrier assessment, and inventory assessment. The objective is to field a supervised robotic security system that basically runs itself until an exceptional condition is encountered that requires human intervention. **Figure 1** diagrams the Concept of Operations for Physical Security.

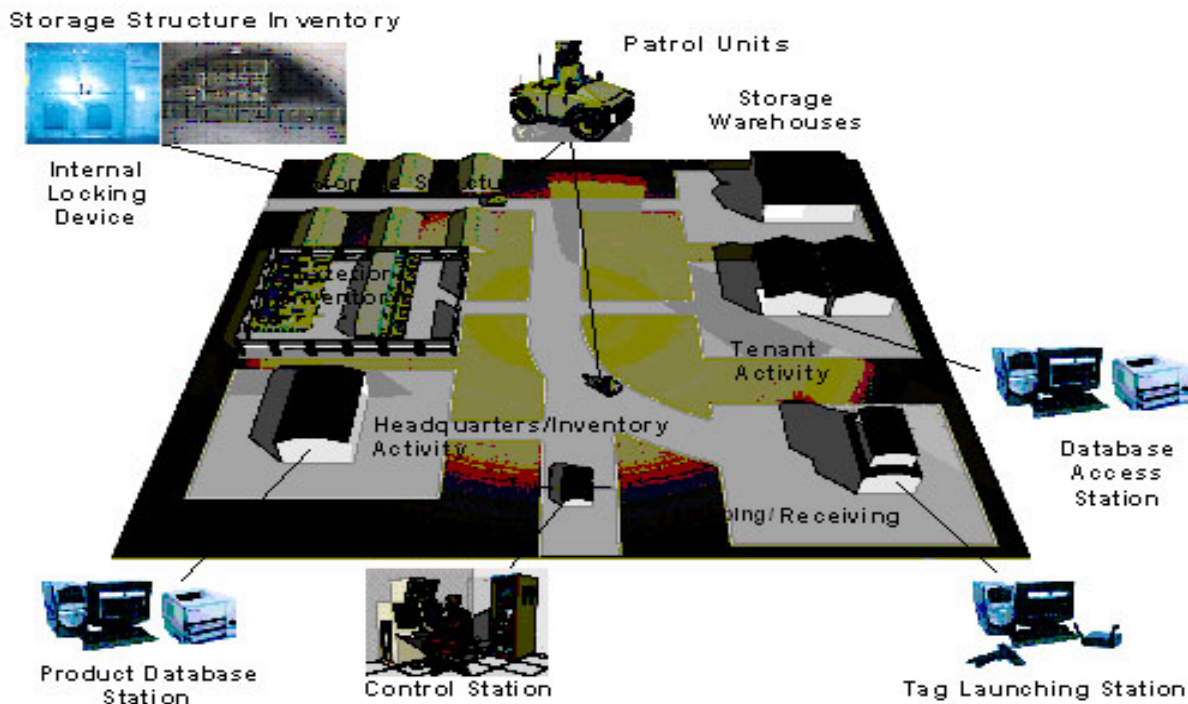


Figure 1. Concept of Operations (CONOPS) for Physical Security

The MRHA is designed to run automatically with minimal human supervision. Operator intervention is only required when a resource encounters an *exceptional condition* such as an environmental hazard or a security breach. Exceptional conditions are prioritized as events and displayed in simple terms to the user for action. The Control Console human-interface provides high-level system information (for all resources) via the *Supervisor* display, and detailed operational/diagnostic information (for a single resource) via the *Operator Station* display. For the production MDARS system, the MRHA is hosted on several desktop and rack computers with multiple screens and is permanently installed inside a typical security facility such as a guard shack. **Figure 2** shows the MRHA Components and Console.

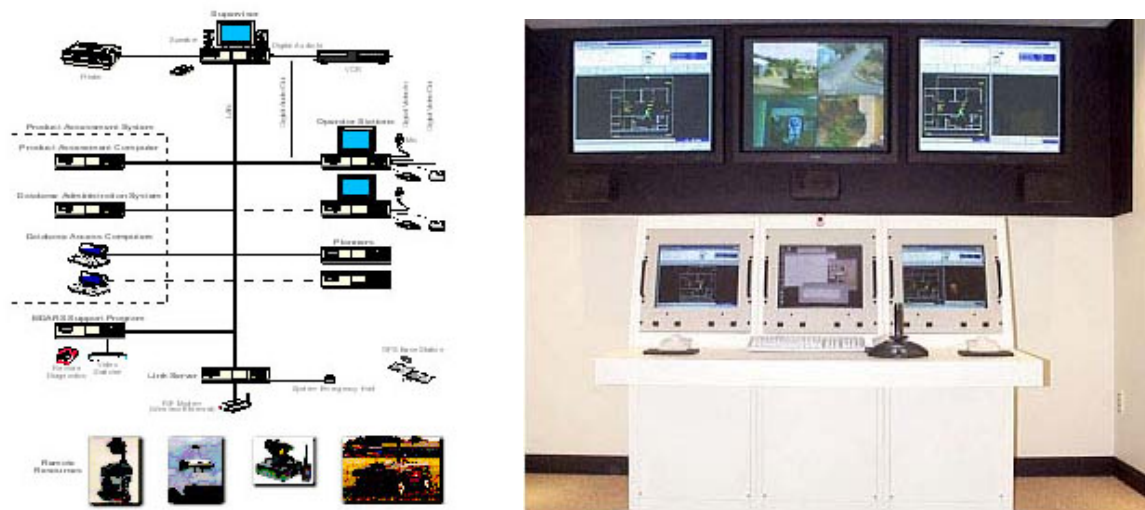


Figure 2. MRHA Components and Console

The MRHA is a scalable design; all software components can be dispersed across networked computers or execute simultaneously on a single laptop personal computer. SSC San Diego has written an Interface Design Document (IDD) for standard messages between the MRHA and the remote resources. The MRHA supports both wireless Ethernet and wireless serial modems to communicate with resources.

MDARS is a joint Army-Navy development effort and is managed by the U.S. Army Office of the Product Manager for Physical Security Equipment (PM-PSE). The U.S. Army Military Police School (USAMPS) is the target user for MDARS and the MRHA. MDARS completed Technical Feasibility Testing (TFT) in May 2000 during the Program Definition and Risk Reduction (PDRR) acquisition phase. The MDARS System Development and Demonstration (SDD) acquisition phase is currently ongoing with a 12 month Early User Appraisal scheduled for 2004 at Hawthorne Army Depot, NV. **Figure 3** shows the MDARS PDRR and SDD Vehicles.



Figure 3. MDARS PDRR and SDD Vehicles

2. C2 Requirements for Manned and Unmanned Forces

Based upon feedback from the operational users and testers, below is a partial list of enhancements to improve the coordination between the manned and unmanned forces.

2.1 Expand Situational Awareness

A single C2 architecture is desired for all of the available assets including unmanned vehicles, remote sensors, and communications to manned patrol forces. The dispatcher at the security headquarters requires a Common Operational Picture including a map to display the current locations of all manned units and unmanned units in the field. For example, after an unmanned security vehicle reports an exceptional condition, the dispatcher may send a nearby manned patrol team to investigate. While the

patrol team responds, the dispatcher follows their progress while monitoring the area around the unmanned vehicle.

2.2 Hand-off Control

En route, the response team needs access to all available information including apriori data and real-time sensor data from the unmanned vehicle. After the response team arrives, the remote dispatcher needs to share control of the unmanned vehicles with the on-scene commander to deploy the assets to their best possible advantage. If necessary, the dispatcher can hand-off control to the on-scene team.

During excursion testing, one military policeman drew an analogy of the relationship between the response team and the unmanned vehicle to a police K9 (dog) unit. The response team may station the unmanned vehicle to perform sentry duty at an entrance like a guard dog at a gate; the response team may use the unmanned vehicle as a scout to enter and secure an unexplored area ahead of the team similar to when the leash is removed from the K9.

2.3 Task Unmanned Forces as Manned Forces

Users require resource management methods above teleoperation and autonomous patrol. Similar to how manned forces are tasked, users want to define scripted sets of actions for navigation and execution (e.g. payload operations) for unmanned vehicles and to define schedules of activities for unmanned vehicles to perform particular tasks at designated times. For maneuvering, the users requested control techniques beyond using a joystick to drive the unmanned vehicle or sending the autonomous vehicle to a pre-defined location.

An example of schedule is on every Tuesday at 2100 hours one unmanned vehicle performs a random patrol in the northeast quadrant of a base while another is performing a methodical inventory covering the entire inventory stored in the southwest quadrant. At 2400 hours, the first vehicle is tasked by the C2 architecture to inventory in the northwest quadrant and the second vehicle is randomly patrolling the southeast quadrant. On Wednesday at 0430 hours, both vehicles are commanded to return to their maintenance areas for refueling/recharging to prepare for their next shift.

Schedules of activities and scripted set of actions can be overridden by user-directed actions. For example, the user could stop a scheduled script to perform an inventory action and instead direct the unmanned vehicle to fall-in behind the response team and follow them.

2.4 Integrate with Unattended Sensors

Remote sensors need to feed into the C2 architecture in order for their associated data to be shared among unmanned systems. Legacy security systems have an established network of fixed sensors installed. Man-portable tactical sensors can be rapidly deployed to provide early warning of possible intrusions into a protected area. Alarms from remote sensors are reported to the C2 console that can dispatch unmanned vehicles to investigate and to provide additional sensor data to the manned response team.

3. Technical Approach

The MRHA is being developed in a phased rapid-prototyping approach that maximizes across-the-board progress while minimizing technical risk. An iterative "build-test-evaluate" approach has been incorporated to allow user and developer feedback to continuously influence the design, while the operational requirements are systematically identified and matched to the technological needs. Recent modifications to MRHA have been implemented to address the above requirements.

3.1 Expand Situational Awareness

The types of remote resources controlled by MRHA were expanded from two different unmanned ground vehicles (interior and exterior) to an extensive variety of unmanned systems as well as manned platforms by using site-configurable elements. Command and control of the following remote resources have been demonstrated using the MRHA:

- MDARS Unmanned Ground Vehicle (UGV)
- Urban Robot (URBOT) UGV
- ROBART III Research and Development (R&D) UGV
- All-Terrain Robotic Vehicle (ATRV) R&D UGV
- Guard Truck monitored by a Network-Enabled Resource Device (NERD)
- Fixed Sensor/Fire Door Actuator controlled by a NERD
- Less-Than-Lethal Unattended Munitions controlled by a NERD
- Platoon Early Warning Device II Unattended Ground Sensor (UGS)

SSC San Diego designed the Network-Enabled Resource Device (NERD) to interface remote devices to the MRHA. The NERD converts any electro-magnetic device to an IP address, allowing any auxiliary control by either the MRHA console or unmanned vehicles. The NERD send/receives MRHA IDD messages. It allows interchangeable hardware and software between applications and locations. A NERD was attached to a Guard Truck to allow a dispatcher to monitor the location and video from a patrol team. Each security vehicle has a NERD equipped with a Global Positioning System (GPS), one or more digital cameras, and a wireless communications radio. **Figure 4** shows a guard truck outfitted with a NERD and MRHA display of its location on the map and the video from the dashboard.

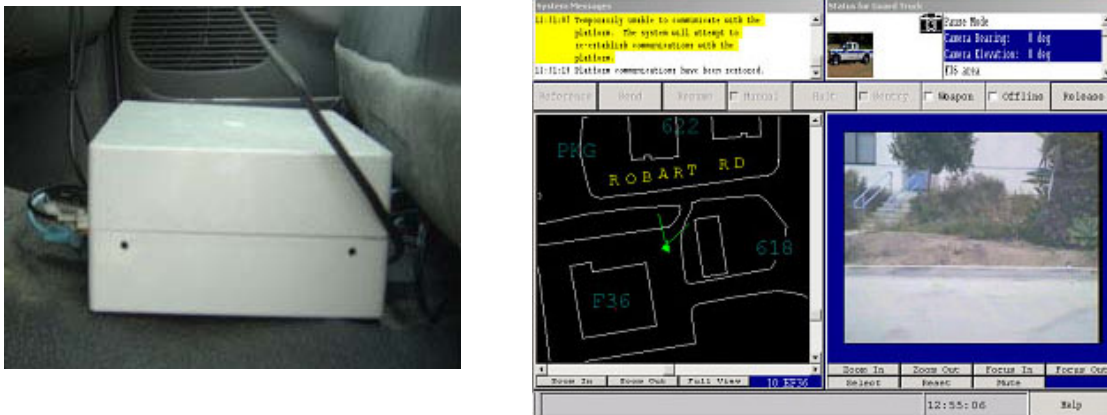


Figure 4. Guard Truck NERD and MRHA Display

Development and testing is underway to integrate Unmanned Surface Vehicles (USV) and Unmanned Aerial Vehicles (UAV). The theoretical number of remote resources handled by the MRHA was increased from 32 to 255. Computational resources and communications bandwidth are limiting factors.

The architecture is designed for a remote resource to have multiple subsystems where subsystems may be associated with one or more devices. A remote resource may also be virtual or simulated. For example, a remote resource could be a System Monitor software application that pings each of the communication devices on the wireless infrastructure (e.g. repeaters); if a device fails to respond, the System Monitor sends a diagnostic message to the C2 console to notify the user of the problem.

The maps on the MRHA User Interface and the MRHA IDD were augmented to display multiple intruders from the Intruder Detection System on the unmanned vehicles. The user can click on an intruder displayed on the map and select the state of the intruder as either acknowledged, rejected, or authorized. The user can designate a primary intruder that the Intruder Assessment cameras will automatically track and display video to the C2 console. A Video Server software component was added to the MRHA to simultaneously display videos from four unmanned vehicles on the console. Again, computational resources and communications bandwidth are limiting factors. **Figure 5** shows the MRHA Operator Station with multiple intruders and the MRHA Video Server.



Figure 5. MRHA Display for Multiple Intruders and Multiple Videos

3.2 Hand-off Control

Since the architecture is scalable and distributed, all MRHA applications can be run on a single computer, however the display real estate is optimized for the applications to run across multiple computers with dedicated screens. The Multi-robot Operator Control Unit (MOCU) was designed to control multiple resources using a single laptop computer. The requirements for MOCU are:

- Command and control multiple unmanned vehicles using MRHA IDD protocol
- Display status, video, and map (static or moving) associated with a resource
- Support teleoperation driving using joystick
- Support waypoint navigation using flexible drag-n-drop interface

A key feature for the user to draw new paths for the unmanned vehicle is for the map to display and manipulate high-resolution, ortho-rectified digital maps. **Figure 6** is a screen snapshot of the Multi-robot Operator Control Unit (MOCU) with a digital map using 6-inches/pixel resolution.

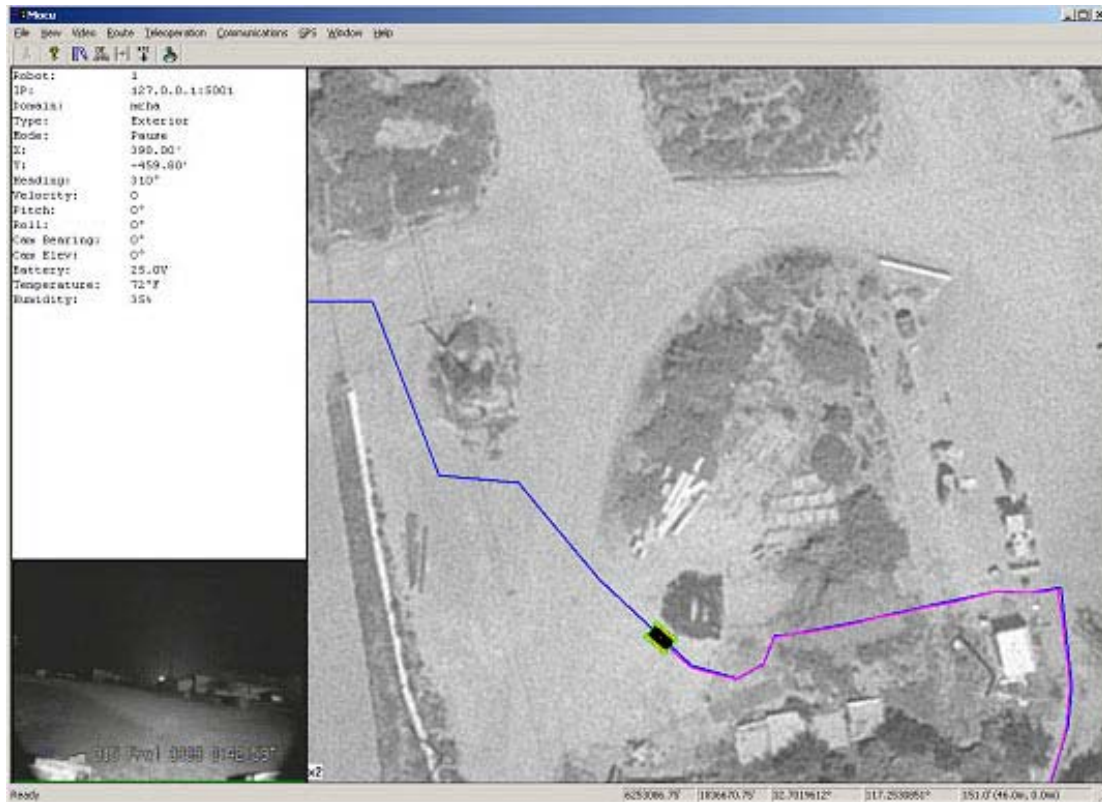


Figure 6. Multi-robot Operator Control Unit (MOCU)

The MOCU software can be executed from either a laptop or a handheld controller. The handheld version uses an embedded processor, a LCD touch screen display, audio hardware (microphone, speaker, and amplifier), and swappable hand controllers. **Figure 7** shows the design and prototype for the handheld controller for MOCU.



Figure 7. MOCU Design and Prototype for Handheld Control Unit

MOCU allows an on-scene commander to use either a laptop with a wireless modem PC card or a handheld control unit to direct an unmanned vehicle. The dispatcher notionally hands off control of the unmanned vehicle to the response team like an aircraft is passed between air traffic controllers. Similar to ATC both users monitor the vehicle, communicate with it, and are responsible for deconfliction.

3.3 Task Unmanned Forces as Manned Forces

The original design of the MRHA tasking stated that a remote resource was basically in one of three states: offline (unavailable), controlled by user on the Operator Station, or performing random patrols. When the Operator Station was assigned to an unmanned vehicle, the user could direct the vehicle to go on a pre-planned path or to operate a payload. When the vehicle was released from the Operator Station, it would begin to perform random patrols on pre-defined paths. After reaching a destination, also called node, the MRHA would randomly determine a different node and send the unmanned vehicle to its next location. The random patrols would continue until an exceptional event occurred and an Operator Station would automatically be assigned for a human guard to act, or until the human guard manually assigned an Operator Station to take over control.

Missions and Duty Rosters were designed for additional control hierarchy above directed one-on-one control and fully autonomous random patrols. A mission is a scripted set of actions for an unmanned vehicle. Missions are text files that contain a sequence of commands using conditional controls. The user can start, stop, or view a mission for any resource. During the execution of a mission, if the Operation Station is assigned, the user has one-on-one control of the vehicle and can override the current mission. The mission automatically resumes when the vehicle is released. Examples of mission commands are:

DEVICE <device id> <OPEN|ON|CLOSE|OFF> [timeout interval]

Toggles a device state.

RANDOM [duration in seconds for random patrols]

RANDOM UNTIL <time>

RANDOM UNTIL NEXT [ACTION]

Sends the platform on Random Patrols for the given time period.

SEND virtual_node_name

Sends to the specified node.

SENTRY [IDS] [INVENTORY] [LOCK] [duration in seconds for survey mode]

SENTRY [IDS] [INVENTORY] [LOCK] UNTIL <time>

SENTRY [IDS] [INVENTORY] [LOCK] UNTIL NEXT [ACTION]

Puts the platform into SENTRY mode with the given submodes.

SLEEP [duration in seconds]

SLEEP UNTIL <24-hour time to wake up>

SLEEP UNTIL NEXT [ACTION]

Causes the script to pause for the requested period of time.

A Duty Roster is a schedule to specify the desired activities for all resources controlled by the MRHA. Duty Rosters are text files that associate actions with specific resources. Actions are specified to be performed at a given day and time. The day can be a specific date, alternating days starting from a specific date, a named day of the week (e.g. *Tuesday*), or more generally (*Weekday*, *Weekend*, *Any_Day*). The start time is specified in standard 24-hour notation. The end time is not specified; the end time is defined as the next action's start time. There are special rules for specific date actions; if any actions are specified for a specific date, then all actions to be run on that date must also be given as a specific date. Typically, specific date actions either represent holidays, or are days when processing is very different from the standard schedule.

Each action is given a priority. The priority indicates a task's relative importance to the person who wrote the schedule. A standard Duty Roster command is:

SCHEDULE <resource id> <priority> <day reference> <start time> <action> <action> <..>
Schedule actions for a facility. Each line is associated with a specific resource, and indicates one or more specific actions to occur at a given time on one or more days. At any time, the most specific day reference value will be executed for a resource

Investigations are ongoing to expand maneuver control for unmanned resources. A leader-follower concept was developed and demonstrated using the C2 architecture described above. A Guard Truck NERD reports the GPS coordinates of the guard's vehicle as the response team drives to investigate an area. MOCU relays the GPS waypoints to an All-Terrain Robotic Vehicle (ATRV) and the unmanned vehicle follows the manned vehicle.



Figure 8. Leader-Follower (ATRV perspective and MOCU perspective)

3.4 Integrate with Unattended Sensors

With the expansion of the resource kinds and numbers, a variety of unattended sensors have been integrated with the MRHA for proof of concept demonstrations.

For legacy applications, prototype interface between the MRHA and the Joint Services Interior Intrusion Detection System (J-SIIDS) was demonstrated for the user at the MRHA console to arm and disarm J-SIIDS volumetric sensors. J-SIIDS was designed to protect small arms, ammunition, and sensitive materials in storage and is a legacy security system at many Department of Defense sites. The various components of JSIIDS are: (1) the control unit with its sensor components and (2) the display equipment. The control unit is installed at the area to be monitored; the display equipment is typically located at the security headquarters. The display equipment is connected to one or more control units; each control unit is connected to one or more sensors.

A NERD was modified to manage the interface between a J-SIIDS control unit and its sensors. Before an unmanned vehicle entered an area monitored by J-SIIDS sensors, the user could manually disarm the sensors. If the sensors were not disarmed, then every time the vehicle would trigger an alarm at the display equipment which would be a nuisance to the user. Figure 9 shows the hardware involved for the proof-of-concept demo for the MRHA to control legacy J-SIIDS fixed sensors.

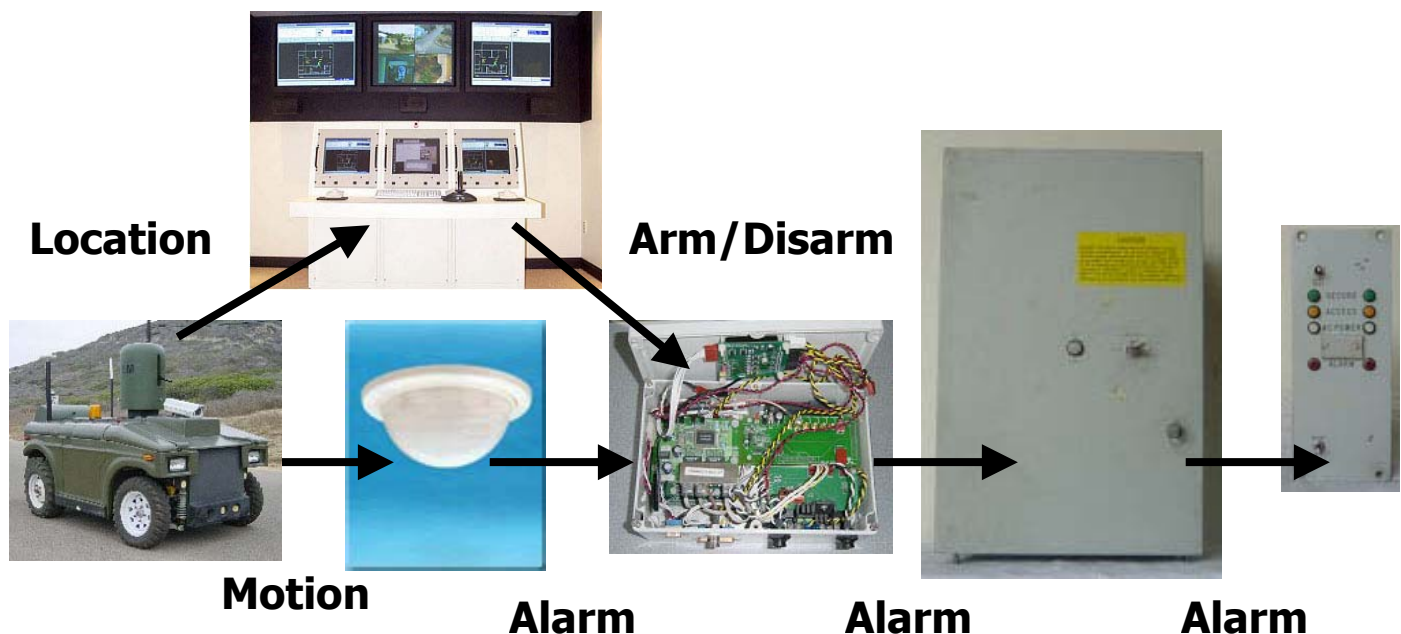


Figure 9. J-SIIDS Sensor, NERD, Control Unit, and Annunciator

For tactical applications, the C2 Architecture has been integrated with the Army's Battlefield Anti-Intrusion System (BAIS), a Tactical Security System designed for use of small unit operational forces deployed in a variety of environments and missions. **Figure 10** is the complete set of equipment for the BAIS.



Figure 10. Battlefield Anti-Intrusion System (BAIS)

The synergism provided by the coupling of these technologies will enhance the protection of forces in both structured and semi-structured environments. The BAIS is a small, rapidly deployable, man portable set of sensors that can provide early warning of possible intrusions into a protected area, and at the same time queue the unmanned vehicle to deploy to a pre-programmed point to provide real time assessment of the BAIS alarm using on board sensors and imagers. In addition, the system provides an unmanned first responder that will reduce troop vulnerability. Future enhancements will fuse these technologies with Area/Battle Command Situational Awareness and Battle Management resources. Further, the addition of both lethal and less-than-lethal force capabilities to the unmanned vehicle will provide commanders with the ability to locally project force without introducing ground forces in confrontational situations. These and other enhancements will provide field commanders with opportunities to reduce levels of force structure presently committed to providing local security in both fixed and contingency operations. **Figure 11** is the Concept of Operations for Tactical Force Protection.

4. Denouement

The Concept of Operations for Physical Security to leverage unmanned systems involves Command and Control (C2) of up to 255 unmanned vehicles and/or sensors by a single user. Recent development and testing for the Mobile Detection Assessment Response System (MDARS) and the Multiple Resource Host Architecture (MRHA) have led to advancements for C2 of mixed unmanned and manned security forces. MDARS and the MRHA address new user requirements for coordination between the manned and unmanned forces.

The MRHA provides a single C2 Console to monitor all base activities including the location and video from manned guard vehicles including sending “eyes and ears” data of the response force to the dispatcher. Maps and video displays provide simultaneous tracking of multiple intruders from several unmanned security vehicles. The security shift dispatcher stationed at the headquarters can hand off control of an unmanned vehicle to the on-scene commander. The response team uses a portable, hand-held Multi-robot Operator Control Unit (MOCU) to direct the security robots.

Unmanned security forces are tasked by the MRHA using a hierarchy of Duty Rosters and Missions to coordinate their schedules and activities with the manned forces. In a response situation, a robot can be directed to follow a manned security vehicle to the region. Integration of MDARS and MRHA with legacy security systems and man-deployable tactical sensors provide a force multiplier to extend the Concept of Operations to Tactical Force Protection.

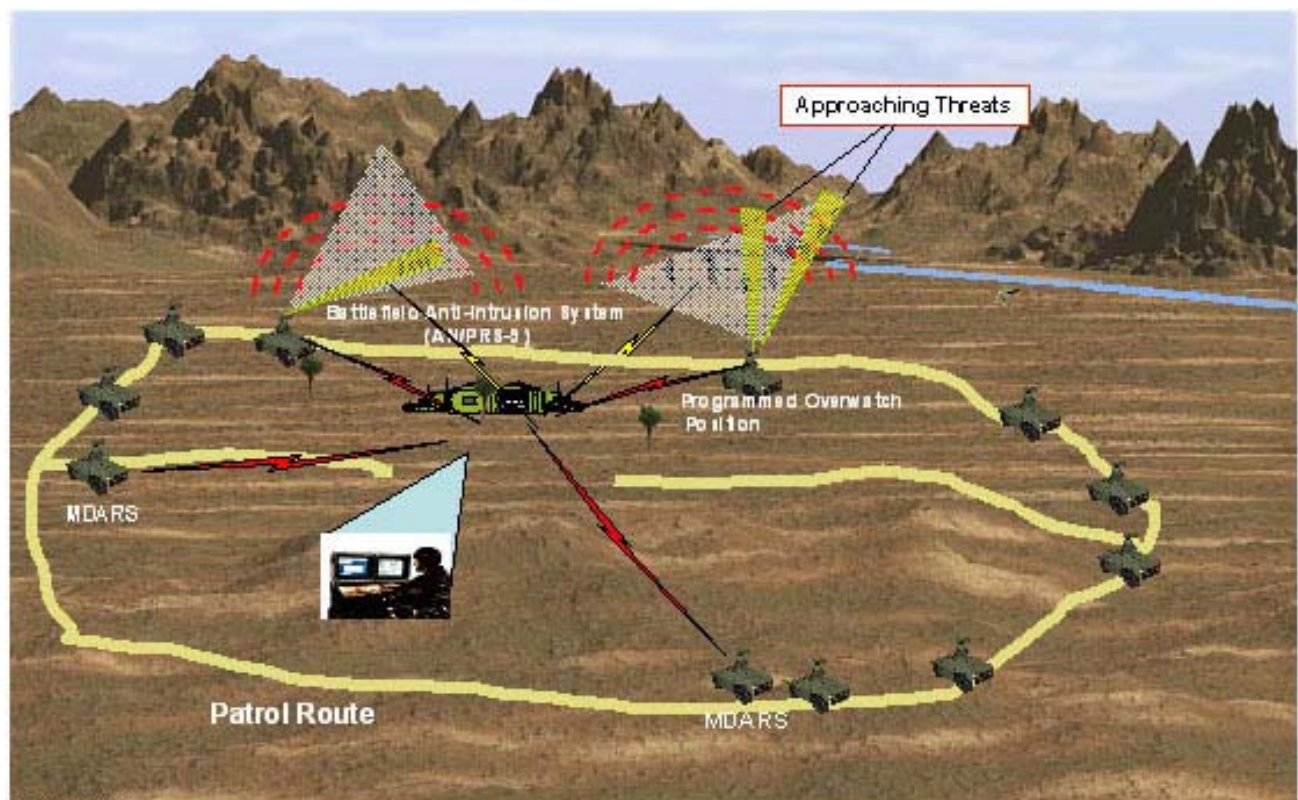


Figure 11. Concept of Operations for Tactical Force Protection

5. References

- Bruch, M.H., Gilbreath, G.A., Muelhauser, J.W., and J.Q. Lum, "Accurate Waypoint Navigation Using Non-differential GPS," AUVSI Unmanned Systems 2002, Lake Buena Vista, FL, July 9-11, 2002.
- Carroll, D., Gilbreath, G.A., and H.R. Everett, "Extending Mobile Security Robots to Force Protection Missions," AUVSI Unmanned Systems 2002, Lake Buena Vista, FL, July 9-11, 2002.
- Everett, H.R. and D.W. Gage, "From Laboratory to Warehouse: Security Robots Meet the Real World," *International Journal of Robotics Research*, Special Issue on Field and Service Robotics, volume 18, number 7, July 1999, pp. 760-768.

Heath-Pastore, T.A., H.R. Everett, and K. Bonner, “Mobile Robots for Outdoor Security Applications”, American Nuclear Society 8th International Topical Meeting on Robotics and Remote Systems (ANS'99), Pittsburgh, PA, 25-29 April 1999.

VanSeeters, C.A., “Abbreviated Test Report for the Technical Feasibility Test (TFT) of the Mobile Detection Assessment Response System - Exterior (MDARS-E),” Report No. ATC-8287, US Army Aberdeen Test Center, Aberdeen Proving Ground, MD, 21005-5059, July 2000.

Appendix

Complete list of mission commands:

comment

Ignored (as are blank lines) as a comment.

DEVICE <device id> <OPEN|ON|CLOSE|OFF> [timeout interval]

Toggles a device state.

DIAGNOSTIC [timeout]

Places the platform in *Diagnostic* mode.

IDS [ON/OFF]

Toggles the Intruder Assessment payload.

INVENTORY [ON/OFF]

Toggles the Product Assessment payload.

LOCK [ON/OFF]

Toggles the Barrier Assessment payload.

LOG <“message text”>

Writes the double-quoted “message text” to the printer log.

LOW_BATTERY <IGNORE | PROCESS>

Either ignores or processes the Low Battery condition.

LOW_FUEL <IGNORE | PROCESS>

Either ignores or processes the Low Fuel condition.

MESSAGE<“message text”>

Displays the double-quoted “message text” in a popup window with an OK button.

OFFLINE

Causes the platform to go off-line.

ONLINE

Causes the platform to return to on-line status.

RANDOM [duration in seconds for random patrols]

RANDOM UNTIL <time>

RANDOM UNTIL NEXT [ACTION]

Sends the platform on Random Patrols for the given time period.

SAY [ONLY]<sound name>

SAY [ONLY]<“text string to be spoken”>

Plays the sound associated with the given sound name or audibly announce the text string.

SEND virtual_node_name

Sends to the specified node.

SENTRY [IDS] [INVENTORY] [LOCK] [duration in seconds for survey mode]

SENTRY [IDS] [INVENTORY] [LOCK] UNTIL <time>

SENTRY [IDS] [INVENTORY] [LOCK] UNTIL NEXT [ACTION]

Puts the platform into SENTRY mode with the given submodes.

SLEEP [duration in seconds]

SLEEP UNTIL <24 hour time to wake up>

SLEEP UNTIL NEXT [ACTION]

Causes the mission to pause for the requested period of time.

SUBSYSTEM [subsystem id #] ENABLE

SUBSYSTEM [subsystem id #] RESET

SUBSYSTEM [subsystem id #] DISABLE [TEMPORARILY <seconds to disable>]

Performs subsystem administration function for the supplied subsystem identifier.

Complete list of duty roster commands:

SCHEDULE <resource id> <priority> <day reference> <start time> <action> <action> <..>

Schedule actions for a facility. Each line is associated with a specific resource, and indicates one or more specific actions to occur at a given time on one or more days. At any time, the most specific day reference value will be executed for a resource

SHARE <first resource id> <second resource id>

Indicate the resources that share a patrol area and can be safely swapped for one or more duty cycles.

SHIFT <START | END>

Denote the start and end of a shift.