Execution-time Decision Aids for Command and Control of Robotic Entities

Rick Ordower
SAIC
Red Bank, NJ

Lee Dixon
SAIC
Orlando, FL

ABSTRACT

Unmanned Systems are rapidly increasing in numbers and capabilities. From 2002 to 2008 the number of robots in the Pentagon arsenal has grown from 200 to 6,000, many of these supporting the soldier in the close battle. Contrasting this growth in unmanned assets, the Army's future force tactical organization reduces soldier numbers and, by extension, the number of robotics officers. Other soldiers in the force structure take on the robotics officer role, increasing their workload and effectively hindering their primary warfighting role.

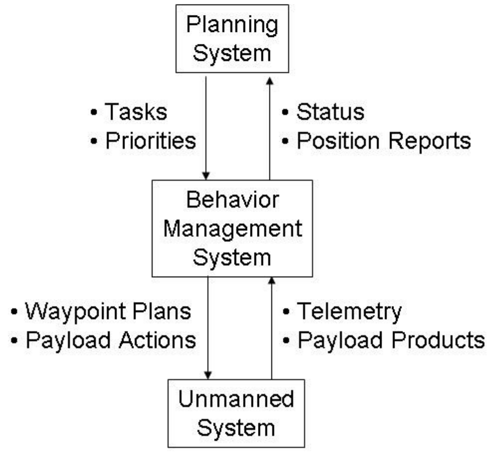
SAIC has been working with CERDEC C2D over the last three years (and AMRDEC prior to this) to introduce Battle Command decision aids supporting (semi) autonomous execution and collaboration of unmanned assets. This paper takes a qualitative/conceptual style to describe a behavior management approach to command and control of multiple unmanned assets.

INTRODUCTION

I will tell you that a commander without the proper C2 assets commands nothing except a desk. You must have the ability to communicate with the forces under your command. You must have the ability to exchange information with them freely, frequently, and on a global basis. It's one thing to have highly technical, sophisticated observation platforms, but if you can't use the information in a timely manner, it's wasted. General (ret) Ronald R. Fogleman

Army RDECOM Communications-Electronics Research, Development and Engineering Center (CERDEC) Command and Control Directorate (C2D) is the R&D arm of the Army addressing C2 technology objectives. CERDEC C2D manages the Army Technology Objective (ATO) Program entitled Command and Control of Robotic Entities (C2ORE) which focuses specifically on providing planning and execution tools to aid guidance and communication from the commander to the operator in the trench. The program goals included development of two software services enabling teaming and tactical control of unmanned air and ground assets as a risk mitigator for the Future Combat Systems Battle Command System.

For the C2ORE program, SAIC held responsibility for development of execution-time decision aids to reduce warfighter workload for unmanned systems (US) control. In the context of this document, Execution-time decision aids refer to scheduling, tasking and routing of the asset while performing maneuvers. These capabilities were implemented in the field tested system, called the Behavior Management System (BMS). Contextually, in its simplest form, the BMS sits between a planning system and a set of unmanned systems.



The BMS implements a set of activities or decision aids that accept plans and immediate taskings and turns them into actionable waypoint and payload inputs for unmanned systems. On the northbound interface, the BMS accepts telemetry and payload products and provides Situational Awareness and other systems access to the status of plans/tasks.

Decision aids supporting US execution have taken on new importance for the Army in the last decade. DOD had more than 3,000 unmanned aircraft as of February 2006, compared to fewer than 50 unmanned aircraft in 2000. As of January 2006, more than 2000 of these aircraft were supporting ongoing operations in Iraq. Over 88 percent of the unmanned aircraft currently in inventory are small unmanned aerial systems (SUASs), those launched by hand or bungee with a weight less than 10lbs [1]. Unlike theatre level UAS, SUAS missions are conducted with limited to no planning. These small unmanned aircraft systems are allocated to the operators in the trenches. The mechanisms to facilitate communication and status of asset progress or for coordination of plan modifications are limited. This same growth trend and use of decision aids for UGVs follows very closely the track of UAVs from a control and status perspective.

This paper, nor the research and development that supports it, does not aim to address a particular algorithm or a particular function. Rather its aim is to explore the multitude of activities, and corresponding decisions, carried out across warfighter roles and echelons. The paper continues by extrapolating how (semi-) automation of some of these decisions better support warfighter focus in conduct of reconnaissance and surveillance. The reader should come away with the appreciation that the Army's future problem is an issue of automating commander-to-foxhole command, control and communications for a brigade of reconnaissance and surveillance assets (as much as it is automation of a single asset).

The following section addresses the operational need ofr a BMS. In Operational Architecture, the context for the BMS is identified by comparing/contrasting related activities with existing Army operational architecture. In System Architecture, the developed system architecture and driving architectural principles are discussed. Pulling from both the operational and system discussions, the Operational Benefits dialogue presents how the BMS provides a new paradigm for the warfighter.

Operational Need

TRADOC is the Architect of the Army, and "thinks for the Army" to meet the demands of a Nation at war while simultaneously anticipating solutions to the challenges of tomorrow. TRADOC's Army Capabilities Integration Center (ARCIC) leads the development and Integration of DOTMLPF for the Army. The ARCIC Warfighter Outcomes (WFOs) product identifies operational requirements traceable to Army Science & Technology priorities. C2ORE development and experimentation works in direct support of some these ARCIC 2010 WFOs [2] as outlined here:

- Battle Command Network Integrated Warfighter Outcome
- Increase Control of Unmanned Systems
- UGV Autonomous Tactical Behaviors.
- RSTA and Attack Operations
- C2 of Battlespace Awareness Assets
- Running Estimates
- Army Airspace Command and Control

.

In addition to these WFOs, a GAO report on Unmanned Aircraft systems [1] states that ground control stations can not easily exchange data and hence interoperability remains a challenge.

In [3], MAJ Charles W. Innocenti, Brigade S2 Trainer, NTC develops a "common scenario" providing an operational context for requiring some of the behaviors above.

The lead opposing force (OPFOR) battalion crosses the RL. Soon afterward, the BLUFOR's UAV picks up a mobilized rifle battalion (MRB) in a deep named area of interest (NAI). The UAV picture is displayed on a monitor inside the brigade TOC next to the battle map. The BLUFOR determines that the UAV has found the OPFOR's advance guard main body (AGMB) and proceeds to take actions to engage the AGMB with artillery. The fire support officer (FSO) tells the UAV operators to have the UAV lock onto the lead T-80 and stay with the AGMB to provide target data and battle damage assessment (BDA).

The air liaison officer (ALO) announces that close air support (CAS) is on station; however, the UAV is now outside its restricted operating zone (ROZ), and CAS cannot engage the AGMB until the UAV repositions. The Brigade XO directs that the UAV move back to locate an enemy dismounted reconnaissance team (DRT) that it had previously located. The DRT seems to be the observer for fires that are falling on the BLUEFOR infantry strongpoint. The UAV is redirected to find the DRT.

Shortly thereafter, the S2 calls for the UAV to return to a deep NAI to look for the OPFOR Main Body, but the Brigade XO denies his request. The battle staffs in the TOC are glued to the UAV monitor as the UAV locates the DRT and artillery falls on its position. They are oblivious to the scout report of a large body of vehicles entering the battle space. The UAV operators are reacting to the latest demand by the battle staff and have totally forgotten the task of identifying the OPFOR main body. The OPFOR main body begins to engage the southern BLUFOR battle position and is attempting to breach. The close fight is now at hand, and UAV requests by the battle staff have dropped off. The UAV operators, with no guidance, simply fly the UAV all over the battlefield.

As complex as this scenario is, it addresses management of a single UAV asset only. "Hasty" operations requiring multiple assets exponentially impedes upon synchronization and coordination amongst warfighters. The SOURCE ATO Operational Concept [4], for example, identifies a scenario addressing multiple collaborating UAV/UGV assets.

Major Innocenti continues after the scenario, noting **lack of focus** as a major shortcoming plaguing units. This is the underlying theme of all the research performed on the C2ORE program. In a sentence:

How can commanders and operators control growing numbers of unmanned assets while maintaining focus on intent and collection products?

The Operational Architecture

Expanding on (and in support of) this question, C2ORE has several operational objectives:

- Autonomously implement a plan
- Support modifications to the plan in real time
- Support collaborative asset operations, reducing operator communications and mistaskings
- React to battlefield events
- Provide sufficient plan status to allow commanders at each echelon to manage intent, while removing the need for them to micromanage the actual product collection

Decomposing these objectives, several desired capabilities are identified:

- Distributed and collaborative warfighter operations
- Task Automation
- Synchronization of ISR assets
- Running estimates of current/future operations
- Responsive operations
- Airspace Control
- Interoperability across platforms

These objective capabilities were identified as much through iterative requirements and development spirals as much as they were through top-down decomposition.

The operational activities defined below are specifically designed to address these operational objectives and capabilities.

Operational Activities

Operationally, C2ORE software building blocks are primarily driven by the Collection Management Process. Other processes such as airspace management play important roles as well. According to [5], Collection Management encompasses 6 steps:

- Develop Requirements
- Develop Collection Plan
- Task or Request Collection
- Disseminate
- Evaluation Reporting

The C2ORE execution software focuses primarily on the step *Task or Request Collection*, which supports the following activities:

- Determine tasking
- Execute and implement
- Collect and exploit

In addition to these activities, the execution system supports *development of the collection plan*, *dissemination* and *evaluation reporting*. The trigger to begin execution processing is at the point where the R&S tasking matrix is handed to his scout platoon leader.

The C2ORE Behavior Management System is designed to provide decision logic, reducing soldier cognitive workload, on an equivalency with collection management processes identified above. These activities are grouped, described in subsequent sub-sections and related back to functions in the Collection Management Process.

Note that Dissemination is an implicit aspect of the Behavior Management System. All activities below are coupled to other processes to facilitate dissemination, simplifying the operator's involvement in dissemination of both plan and results.

Control Region Management

This set of activities performs simple propagation of control region constraints (UAV airspace, UGV operation area, keep-out zones) to each team of unmanned assets as defined by external planning activities (such as an Air Liaison Officer). The control region data is utilized by other activities including *Dynamic Scheduling* and *Task Planning* for route generation constraints.

Adherence to air and ground control measures directly support the *Execute and implement* function in the Collection Management process.

Task Allocation

This set of activities is responsible for automating the allocation of tasks to teams of unmanned assets. As task events (additions/changes/deletions) and asset events (asset launch/recovery) are received, it interacts with the "Task Planning" activity to determine which team of assets is capable of performing each task. If multiple teams are capable, then the activity must utilize some heuristic to determine which is best. The current heuristic utilizes a load-balancing approach. Bids are requested from each team; the resulting bid includes a task load based on the number of capable assets within the team, and the number of outstanding tasks they are responsible for. Using composite data such as "task load" keeps this high-level activity from requiring time-critical or asset-specific knowledge.

Task Allocation directly supports the *Development of the Collection Plan.*

Asset Management

This activity group is responsible for several major operations related to assets management:

- Generate asset system events (asset launched/recovered)
- Provide idle tasking (such as a UAV loiter)
- Monitor asset energy reserve (fuel/battery) **[not yet implemented]**
- Generate task-load composite data, respond to bid requests from "Task Allocation" activity
- Request task scheduling, deliver task schedules to "Asset Command & Control" activity

Asset Management supports both *Execute and Implement* as well as *Evaluate Reporting* functions.

Dynamic Scheduling

This activity is responsible for selecting the assignment and ordering of tasks for assets. The logic for selecting particular schedules can be implemented in a variety of methods, but the current implementation has demonstrated three types of scheduling:

- **Operator Mandated** – An operator explicitly specifies a schedule for each asset; only the return to loiter/idle task is automated
- **Automated Fixed Purpose** – A management task may mandate a predefined schedule (ex: Three Pass Attack: a target acquisition snapshot, followed by a munition task, followed by a BDA snapshot)
- **Fully Automated General Purpose** – A pool of tasks scheduled using a task-agnostic, cost-based algorithm to minimize average time-to-complete, including task priority and time-window constraints

Other scheduling methods can easily be imagined and implemented because the architecture defines a very clear mechanism for pairing tasks and assets, without needing intimate knowledge of the task or the asset. Resulting asset schedules are then delivered to the "Asset Command & Ctrl" activity for implementation.

Dynamic Scheduling directly supports *Determine Tasking*.

Task Planning and Execution

This activity implements planning and execution operations specific to particular task types. To support planning, this activity processes cost requests from "Dynamic Scheduling" which include a predicted starting condition of the asset, which usually is the final predicted state from the cost estimation of the preceding task in the order permutation. Sensor planning tools are used to select appropriate platform locations for collecting sensor products, and the "Route Planning" activity is used to plan paths from the starting condition to the desired resulting position/orientation.

Task Planning and Execution supports various aspects of all Collection Management processes.

Modular Task Architecture

One goal of the Task Planning architecture is to provide a programmer's interface for easily adding new task types. New platforms, sensors, and methods for using them can easily be added to the system by implementing three major sets of interfaces:

- Allocation Planning (is an asset capable of performing the task?)
- Schedule Cost Estimation (given an asset with a specific starting condition, what is the cost to complete the task?)
- Task Execution (perform direct communication with the Asset Command & Control Activity)

Sensor Planning

Available to each Task P&E module is a library of sensor planning tools. This library can be used to compute a set of platform locations/orientations to meet task requirements (resolution, area coverage, relative aspect) given the payload characteristics (field-of-view settings, detector geometry, articulation constraints). The library can also be used to determine if a sensor target may be viewed from within the control region constraints, used during Task Allocation".

Route Planning

For UAV assets, a route planner is available for generating routes which reflect flight characteristics (turn rate, climb rate, airspeed limits, etc) and operational boundaries (restricted operation zone, keep-out zones). For UGV assets, a common interface to external route generation systems is defined, allowing each UGV platform type to utilize a vendor-specific route generation service.

Asset Command and Control

This activity implements all direct interactions with the assets. Beyond acting simply as a translation layer to communicate in the asset's vendor-specific messaging structure, the activity performs advanced features to offload work from other parts of the system, as well as defining interfaces to simplify asset scheduling and management.

Asset Command and Control supports the *Execute and Implement* function.

Asset Virtualization

A common level of interfacing and functionality is defined for each asset type to create a type of virtualization.

Functions commonly required by tasks are performed either in the embedded asset's hardware, or may be implemented locally. For example, tasks may generate long, intricate paths through a maze of keep-out zones or to perform sensor scan patterns. The Raven UAV in particular can only process 4 waypoints at a time. This activity virtualizes the ability to process the full path by queuing waypoints to the asset as progress is made.

To date, the list of virtualized functionalities include:

- Provide vehicle properties, including flight characteristics, payloads, and optional capabilities (target tracking, loiter about a point, etc)
- Emit telemetry in a common format
- Accept and execute a <u>schedule</u> of tasks; each task is composed of a list of waypoints and payload actions; the schedule is composed of the ordered list of tasks
- Accept simple imaging payload actions: snapshot at default angles (strap-down camera), snapshot at inertial angle, snapshot at GPS point
- Platform-specific modes: a simple pass-through mechanism allows for the creation of tasks which utilize platform-specific capabilities

Asset Time Sharing

This activity also defines a time-sharing mechanism for tasks. Incoming schedules from *Dynamic Scheduling* define the ordering of tasks with defined start and end conditions. Each task connects at run-time to an "asset session", which is activated at the appropriate time within a schedule. During this session, the task's pre-defined or real-time commands are authorized to control the path and payload(s).

Task Progress and Completion Estimation

As each "asset session" begins, ends, or makes intermediate progress, the estimated timeline for the current and subsequent tasks is continually updated. These timeline updates are broadcast for monitoring by operators, or even to be used by *Dynamic Scheduling* to make real-time logical decisions.

Team Scheduling Authorization Transfer

Each schedule installed into this activity may also define an authorized team which may then take control at the end of the schedule. This allows a team to transfer an asset to a handoff location, perhaps at an operational boundary, for subsequent usage by another team.

The System Architecture

This section addresses how the operational architecture is translated into a system architecture. Architectural system principles, services, communication and deployment for PM C4ISR OTM are discussed in here.

System Archtiecture Principles

Activity organization within the BMS are designed with a common theme: <u>maintain a loose coupling between activities</u>. A real-world analogy is to keep activities from "micro-managing" other activities. There are several benefits from this goal:

- Like their human echelon counterparts, automated processes can work autonomously from higher level management processes without constant oversight.
- Decisions made closer to the data source allow for faster responses. Data latencies are reduced.
- Less data transmissions are required between activities, likely resulting in lower transmission bandwidth requirements
- Loss of communication link does not render a lower echelon useless. The node can still implement it's latest set of tasking or contingency plan.
- The resulting software architecture supports modular replacements and/or expansion of computational or decision-making operations.

As the chain of command is followed from bottom to top, higher-level decisions are made using less data, composed of data composites from lower levels.

System Services

The C2ORE Behavior Management System (BMS) implements the operational activities defined in *Operational Architecture*. Using the principles mentioned above, the BMS has three major services which collaborate during a mission. Each of these services shadow a soldier role (or set of soldiers) to automate or augment their activities.

Vehicle Interface

This service is instanced once for each unmanned asset. Its main job is to implement the *Asset Command & Control* activities. As such, it contains all the platform-specific logic necessary to implement the virtualized capabilities required by other services. This service acts much like a plug-in, requiring new software development for each new asset type. To date, interfaces have been built for a simple simulation-only UAV, the Raven UAV, and for the ARL eXperimental Unmanned Vehicle (UGV).

Team Commander

In the C2ORE operational architecture, a “Team” is defined as a set of assets which can be used to perform close collaboration of tasks. The team is operationally defined by the commander and could consist of squad, platoon or company level assets.

Team operations are closely tied together, and can therefore be planned to execute multiple tasks which are geographically close, and whose relative timing is sensitive. The assets also share the same airspace via altitude assignments. Due to these types of operations, the Team Commander requires a moderate level of monitoring of assets, playing an up-to-the-minute role in their operations.

The Team Commander implements *Asset Management*, *Dynamic Scheduling*, and *Task Planning & Execution* activity groups.

Group Commander

A “Group” is a grouping of “Teams” which are organized to complete tasking in a larger geographical area. Group Commander capabilities would most likely be configured for Brigade or Battalion level command.

Tasks are not allocated to teams simply by area assignment, but rather using the *Task Allocation* (defined in Operational Activities) bidding process to determine which assets are capable for reasons related to payload properties (like sensor zoom). It requires a low level of asset monitoring, being more concerned with the activity of a team as a whole rather than single assets.

System Communication Description

The objective System Communications Description (SV-2) for the C2ORE OTM demonstration is shown on the following page.

Group Commander (GC) software resides at the BDE/BTN TOC to support mission planning. Team Commander (TC) software resides at the CP/PLT/SQD echelons to support collaborative operations within a restricted operational zone.

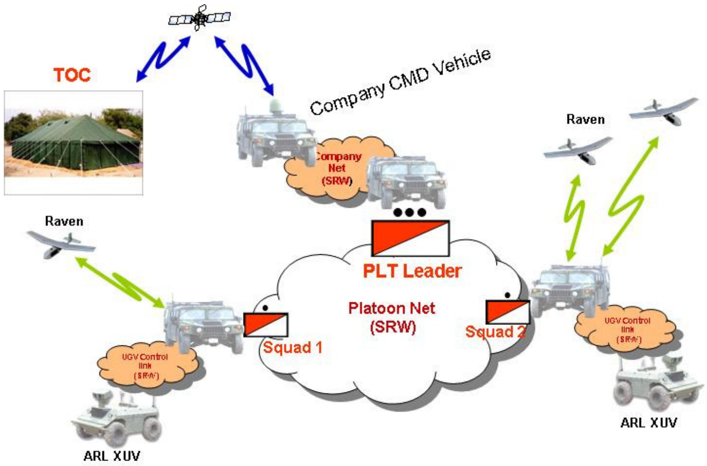
As an objective, system design supports distribution of the TC and VI. Distribution of VI, each instance supporting a different unmanned asset, allows the TC to manage multiple assets, hosted off different ground platforms across the echelon. Present implementation does not fully support distribution of TC/Vehicle Interface (VI) software. Hence, Team Commander must be collocated with the ground platform hosting ground control stations (GCSs) /operator control units (OCUs) associated with the assets under the TCs control.

Prior to deployment, it was well understood that digital communication between the Group Commander and Team Commander service could suffer from data loss across the packet radio links. Therefore, a robust asynchronous messaging infrastructure was implemented to handle communication exceptions. The infrastructure included automatic error handling such as message retries and detection of message duplicates. But during outright failures, the operator was given choices such as “cancel”, “retry”, or “assume worked”. This allowed the operator to play a role when determining how the system should handle communication outages.

System Deployment at Ft. Dix

In live experimentation at CERDECs PM C4ISR OTM, in both ’07 and ’09, the C2ORE program demonstrated the activities/capabilities described above.

During experimentation, two Aerovironment Ravens and one ARL XUV OCU were controlled via a single instance of the BMS Team Commander. After launch of the assets, operators at the C2ORE GUI were able to disseminate an R&S plan or individual R&S tasks.



TC software resided at the Squad vehicle at Ft.Dix. Group commander software resided at the TOC (communicating over a SATCOM or SRW link) or Company command vehicle.

Various scenarios were conducted to validate the C2ORE software. A sub-set of scenarios included:

- tasking numerous individual snapshots: demonstrated scheduling and asset allocation
- adding priority snapshots: demonstrated schedule updates

- adding map regions: demonstrated collaborative tasking of two air assets.
- UGV Detections: detections triggered UAV surveillance of the area and demonstrated cross cueing.
- ROZ development: tested airspace deconfliction.

Operational Benefits

The *Operational Architecture* introduction discussed a set of capabilities which guided C2ORE BMS architecture. How C2ORE supports these capabilities and the supported benefits are discussed here.

Distributed/Collaborative Warfighter Operations

Distributed/Collaborative Warfighter Operations pertains primarily to dissemination of plans, tasks, position reports, status information and sensor data (e.g., imagery). As mentioned in *Operational Architecture*, dissemination is abstracted from the operator.

In typical computing systems, dissemination is seen as such a fundamental element of the system, its importance is often overlooked. While the Army has mechanized to a great extent, operational distribution is still largely facilitated by Voice and VMF messaging. As discussed in [6], actions for close air support (CAS), for example, “still rely, to a high degree, on passing critical data through radio-based voice communications nets”. This article continues, “In addition to creating information overload for users, the potential for lost, misinterpreted, or misdirected data produces an inefficient and potentially dangerous situation.”

In the case of plans and tasks, as commanders/leaders lay out operations, dissemination is accomplished by a button click. Who and where plans are published is a product of configuration.

For position reports and status information, commanders no longer need to rely on voice as their primary means of communication to understand the situation on the ground. Position and status is autonomously collected from the BMS and the asset and forwarded to subscribed echelons. Seeing task assignment and status allows all echelons to **focus** on tasks at hand.

For sensor data, all products are correlated back to the initiating task. Commanders no longer have to guess as to how or why a sensor product came about.

Dissemination is the base enabler of all other capabilities. Due to distribution of the decision makers, compounded by limited data pipes, robust dissemination of information is critical. Following sections refer back to dissemination where applicable.

Task Automation

The automation provided by the BMS allows a warfighter to stay focused on high-level monitoring, analyzing task results, and producing additional tasking to meet mission goals. The automation provided affects several levels of activities.

At the asset level, a soldier no longer needs to spend time planning routes which adhere to dynamically changing airspace constraints. Sensor planning automation results in consistent acquisition of products which meet task requirements. Task requirements can now be more dynamic. For example, iterative requests for increased pixels-on-target would require a soldier to consider the altitude and zoom levels of a sensor. The BMS sensor planning handles this automatically.

Multi-asset tasks are also automated to account for the complexities involved with planning collaborative operations. By introducing collaborative asset automation, more complex tasking can now be achieved; mission planners can have a higher level of confidence when planning complicated missions. Multi-asset tasks require quick dissemination of position and status to allow for collaboration.

Finally, task management itself is also automated which allows the planner to stay focused on results. Task products stay associated with the original task, even if the task required the use of multiple assets. Task specifications are defined in a concise manner, and the planner has a higher confidence that the system will execute the task to meet the task’s minimum requirements.

Synchronization of ISR Assets

“No battle plan survives contact with the enemy”, states a famous quote from 19th century Prussian Chief of Staff Moltke. Moltke's main thesis was that military strategy had to be understood as a system of options since only the beginning of a military operation was plannable. But synchronization must be maintained beyond planning.

Synchronization, in C2ORE, is achieved at two levels:

- Status - status updates provide the commander down to the asset operator the ability to visualize allocation, ordering and progression of tasks.
- Scheduling - Deliberate and hastily planned tasks support priorities in the C2ORE tasking interface. Commanding officers, through the C2ORE GUI, identify priority amongst tasks. The BMS

manages these priorities, along with other factors such as distance from tasks, and creates an ordered precedence of when tasks are scheduled.

Through status, all echelons have access to the same situational awareness (SA), minimizing communications. SA in this case is primarily not the typical position reports associated with SA. Higher level commanders do not typically require this level of detail. While they can gain access to this data, continuous updates have the potential to undermine the communications network. Rather higher level commander's gain access to asset and task status. At lower echelons, operators will typically access the same status info in addition to asset position reports to maintain finer grained control.

Asset scheduling during planning stages is somewhat of a black art. It is assumed that scheduling during planning stages follows well defined Rules of Engagement (ROEs). This scheduling likely takes variables such as distances from tasks, asset workload or asset fuel into account, for example. During "Hasty" operations, operators wouldn't have the facilities to redefine schedules in real time. The result is lack of focus similar to the scenario posed in the Operational Need section of this document. If a well defined set of ROEs are identified, these rules can be implemented by a scheduling algorithm, as was accomplished in C2ORE. Use of automated scheduling (supported by operator acknowledgements) reduces warfighter workload and allows them to focus attention on other activities.

Running Estimates

In conjunction with telemetry and status reports, as described above in *Operational Architecture: Asset Command and Control*, the estimated timeline for the current and subsequent tasks is continually updated. These timelines are provided locally at the operator level and can be published to higher echelons at their request.

Running estimates relieve operators from the need to status task progression, minimizing misinterpretation and miscommunication, especially during "Hasty" operations. It can also be argued that the algorithms for running estimates are more trusted than what an operator could deliver given their focus on other events.

Responsive Operations

There are numerous instances where battlefield events are deemed relevant to trigger new operation R&S responses. For example, fires on enemy targets should be followed by Battle Damage Assessment (BDA). Detection of a target may be cause for performing more detailed surveillance in support of recognition and identification.

C2ORE supports these responsive operations autonomously. Commander's intent for conduct of these operations is configurable, allowing ROEs to be defined during planning. If any condition identified by the commander is met during execution, the BMS autonomously triggers new tasks. These tasks are then allocated and scheduled in the same manner as pre-planned or execution time defined tasks.

Responsive operations prompts the commander when commander defined battlefield events occur during execution. Commander options are suggested and the commander is prompted to accept/reject the operation..

This capability both increases responsiveness of operations and offloads warfighter workload.

Airspace Deconfliction

"On a typical day, about 100 aircraft, about one-third of them unmanned, pass through the 30 mile square above Baghdad, about twice as many as last year. Up to 40 are in the air at any given moment…Since the 2003 invasion, there have been at least five collisions between UAVs and manned aircraft". [7]

C2ORE utilizes a combination of procedural and positive control measure to enhance airspace control. Through the VI, the BMS gains direct access to unmanned system telemetry. As Airspace control or other users require, position reports can be published. Because the network does not easily support the bandwidth necessary to transmit frequent position reports, C2ORE relies more heavily on procedural control, consistent with traditional Army airspace control methodology. [8]

Operational Activities: Route Planning develops routes constrained by air and ground control measures (e.g., Restricted Operating Zones), supporting horizontal deconfliction from other air or ground space users. Assets are not allowed to be routed outside of these measures. Route plans are subsequently enforced by the VI. Inside the air and ground control measures, C2ORE applies varying levels of altitude (air only), time or horizontal deconfliction.

Procedural control automation reduces the potential for operator error and allows the operator to focus attention on other activities.

Interoperability Across Platforms

As discussed in [1], "ground stations cannot easily exchange data because they were not designed to interoperable communications standards". "When communication systems are incompatible, operating forces may be required to operate their own UAS to accomplish a mission, rather than using UAS that are already operating in the same area".

C2ORE itself has not focused on use of standards in its development, such as JAUS. This is clearly an area of future effort that C2ORE or its successor needs to pursue. C2ORE has however worked to bridge interoperability gaps with other systems. Through the documented C2ORE vehicle interface, C2ORE has adapted other non-standard unmanned systems into the C2ORE infrastructure. Notably, the Aerovironment Raven and the ARL XUV have been integrated. C2ORE is now integrating with Procerus' Kestrol autopilot, which can be used to support control of various UAVs including the Maverick and the Buster.

Conclusion

As the DoD pushes forward with increased robotics battlefield usage in conjunction with decreased force structure, robotic autonomy needs to take a larger role to support these operations. Robotic autonomy includes autonomy in command and control an not just autonomy of the asset alone. This is affirmed by the TRADOC Warfighting Outcomes discussed in the introduction.

In support of autonomy, it is clear that doctrine on robotic usage will have to change. The simple fact is that there just aren't any requirements for how this autonomy should be conducted. There are numerous questions that are asked when providing this autonomy:

- How is air/ground space managed?
- What is the nature of tasking for these assets?
- How are the assets scheduled to perform these tasks?
- How reactive (i.e., autonomous) do we allow the assets to be and what is the nature of the reactiveness?
- How situationally aware should warfighters be at each echelon?

C2ORE research touches all of these qualitative questions. Research has revolved around existing doctrine, including field manuals pertaining to robotic assets, reconnaissance & surveillance, and collection management. To achieve WFOs, however, deviations from doctrine must occur.

In the absence of requirements, C2ORE has set an operational context for how increasing asset numbers can be commanded and controlled from a smaller number of warfighters. The question is whether C2ORE provides the correct operational views. How does the warfighter really want to interact with robotic assets? The only way to answer these qualitative questions is to put the capabilities in the hands of the warfighter. Warfighers are just beginning to get hands-on experience using C2ORE in the field at Ft. Dix, PM C4ISR OTM. Their feedback is critical to address future directions/efforts for C2ORE.

In its present state, if shown to provide the correct operational context, C2ORE software components could be hardened and delivered for formal operational tests. Given the early stage in operational development, the more useful outcome of C2ORE may be as a requirements development tool. Put the system in front of commanders/operators. Let them use it in their environment and with their vignettes. Through operational test and system development iterations, the capabilities can be morphed into validated requirements and in the end, a validated system.

REFERENCES

[1] United States Government Accountability Office; *Improved Planning and Acquisition Strategies Can Help Address Operational Strategies*, Statement of Sharon Pickup and Michael Sullivan, April 6, 2006.

[2] TRADOC ARCIC, FY10 Warfighter Outcomes

[3] MAJ Charles W. Innocenti, "Applying the Unmanned Aerial Vehicle (UAV) to Brigade Reconnaissance & Surveillance (R&S) Operations", GlobalSecurity.org

[4] TARDEC SOURCE ATO Effort, SOURCE ATO Operational Concept, June 5, 2009, http://contracting.tacom.army.mil/research/tardecsourceatoeffort/tardecsourceatoeffort.htm,

[5] US Army, Field Manual 34-2, Collection Management and Synchronization Planning

[6] Dusseau, D. and Brock, C. (2003), "Network Centric Interoperability – Using a Variable Message Format (VMF) Based Data-Link to Improve Situational Awareness and Close Air Support (CAS)", Proceedings of the 22nd Digital Avionics Systems Conference, pp. 9A3.1 to 9A3.7.

[7] Gail S. Putrich, "Unmanned and Dangerous: How UAV-plane collisions are changing US air control". Defense News. June 11, 2007.

[8] "Airspace Control in the Combat Zone". Air Force Doctrine Document 2-1.7. July 13, 2005.