14th ICCRTS: C^2 and Agility

Multi-level operational C^2 holonic reference architecture modeling for MHQ with MOC[*]

C^2 Approaches and Organization

Chulwoo Park [STUDENT]

Email: chp06004@engr.uconn.edu

Prof. David L. Kleinman

Naval Postgraduate School
Dept. of Information Sciences
Monterey, CA 93943, United States

E-mail: dlkleinm@nps.edu

Prof. Krishna R. Pattipati[**]

University of Connecticut
Dept. of Electrical and Computer Engineering
371 Fairfield Way, U-2157
Storrs, CT 06269-2157, United States

FAX: 860-486-5585
Phone: 860-486-2890

Email: krishna@engr.uconn.edu

[*] This work was supported by the Office of Naval Research under contract # N00014-09-1-0062

[**] To whom correspondence should be addressed: krishna@engr.uconn.edu

ABSTRACT

The purpose of this paper is to present a multi-level operational C^2 holonic reference architecture that is applicable to Navy maritime headquarters (MHQ) with maritime operations center (MOC) for assessing, planning and executing multiple missions and tasks across a range of military operations. The control architecture consists of three levels: strategic level control (SLC), operational level control (OLC) and tactical level control (TLC). In addition to coordination within each level, two specific coordination layers are identified at the SLC-OLC and the OLC-TLC interfaces. The SLC-OLC interface layer resolves evaluation issues associate with selecting DIME (diplomatic, information, military and economic) actions based on national priorities, while the OLC-TLC interface layer is used to resolve mission monitoring/planning issues associated with deciding on the courses of action based on outcomes of asset-task allocation at the tactical level. We employ semi-Markov decision process (SMDP) approach to decide on missions to be executed and their time sequence at the SLC-OLC layer (coordination of future plans), while a distributed SMDP approach to an action-goal attainment (AGA) graph for addressing the mission monitoring/ planning issues related to task sequencing and asset allocation at the OLC-TLC layer (coordination of future operations and current operations). The times between decision epochs at the SLC-OLC layer are determined by the mission completion times at the OLC-TLC layer, while the DIME actions and missions to be planned at the OLC-TLC layer are determined at the SLC-OLC layer.

Keywords: holonic reference architecture (HRA), maritime headquarters (MHQ), maritime operations centers (MOC), strategic level control (SLC), operational level control (OLC) and tactical level control (TLC), semi-Markov decision process (SMDP), action-goal attainment (AGA) graph, diplomatic, information, military and economic (DIME) actions

I. INTRODUCTION

Motivation

The term *maritime headquarters* refers generically to those Navy operational-level commands with the capability to assess, plan, and execute at the operational level of war and the term is inclusive of the commander, the staff and the facilities [1]. The Navy's new concept of incorporating MHQ with MOC emphasizes standardized processes and methods, centralized assessment and guidance, networked distributed planning capabilities, and decentralized execution for assessing, planning and executing missions across a range of military operations. The assessment is a continuous process, whose primary purpose is to provide the commander (CDR) with a comprehensive report of progress made with regard to the achievement of maritime objectives. This overall objective assessment combines the monitored outcomes of mission execution and the analyzed effects of operations (diplomatic, information, military or economic), with the situational awareness to inform future development of plans, prioritize ISR activities and allocate forces. The maritime planning process contributes to the development of the CDR's guidance, an executable plan and orders to tactical forces. The planning process is informed by guidance from higher headquarters and the assessment process, and should be highly collaborative both vertically, with higher headquarters and subordinates, and horizontally, with other MOCs and joint components. The maritime planning processes focus on the desired objectives and operational effects required by higher headquarters guidance. In execution, the CDR will command by directing, monitoring, assessing and re-directing forces. The primary tool for the MOC will be the collaborative information environment (CIE). Important to effective execution is operational environment awareness, horizontal and vertical integration with other commands and continuous assessment.

In this paper, we model the coordination issues inherent in the MHQ with MOC via a three-level architecture that links tactical, operational and strategic levels of decision making. Here, we seek to demonstrate that the C^2 coordination issues at the three levels, viz., strategic, operational and tactical levels, associated with DIME actions (future plans), and mission planning (future operations and current operations) can be modeled and addressed by using the proposed architecture.

Related research and new contributions

Our previous research on C^2 organizational design has included the modeling and synthesis of organizational structures at the tactical level to achieve a set of command objectives, such as maximizing the speed of command, minimizing coordination, balancing workload, and so on. Levchuk et al [2-4] developed the following three-phase process to design mission-congruent organizations:

Phase I: The first phase of the design process determines the task-asset allocation and task sequencing that optimizes mission objectives (e.g., mission completion time, accuracy, workload, asset utilization, asset coordination, etc.), taking into account task precedence constraints and synchronization delays, task-resource requirements, resource capabilities, as well as geographical and other task transition constraints. The generated task-asset allocation schedule specifies the workload of each asset. In addition, for every mission task, the first phase of the algorithm delineates a set of non-redundant asset packages capable of jointly processing a task. This information is later used for iterative refinement of the design, and, if necessary, for on-line strategy adaptation.

Phase II: The second phase of the design process combines assets into nonintersecting groups, to match the operational expertise and workload threshold constraints on available DMs, and assigns each group to an individual DM to define the DM-asset allocation. Thus, the second phase delineates the DM-asset-task allocation schedule and, consequently, the individual operational workload of each DM.

Phase III: Finally, Phase III of the design process completes the design by specifying a communication structure and a decision hierarchy to optimize the responsibility distribution and inter-DM control coordination, as well as to balance the control workload among DMs according to their expertise constraints.

Each phase of the algorithm provides, if necessary, feedback to the previous stages to iteratively modify the task-asset allocation and DM-asset-task schedule. Phase I of the design process essentially performs mission planning, while Phases II and III construct the organization to match the devised courses of action.

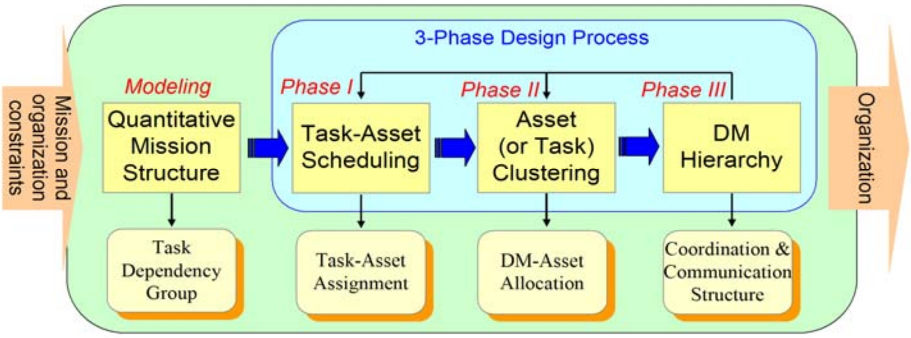


Figure 1. Three-phase organizational design process.

C^2 architecture can be organized as a hierarchy, heterarchy, or a holarchy. A hybrid organizational structure, termed the holonic structure or the holarchy, is proposed in order to overcome the drawbacks of both the hierarchy and the heterarchy. The term ‘holonic’ is derived from the word ‘holon’, and was introduced by Koestler in the context of social and living organisms [8]. This word is a combination of the Greek ‘holos’ meaning whole, with the suffix ‘on’ which, as in proton or neutron, suggests a particle or part. The holon, then, implies a combination of ‘wholes’ and ‘parts’. Thus, ‘holons’ refer to autonomous self-reliant units (“cells”), which hold a degree of independence and are able to manage local contingencies without interference from their superiors.

The holonic structure combines the best features of hierarchical and heterarchical structures, and addresses key requirements of C^2 organizational structures operating in dynamic and uncertain environments. It is a hierarchy of self-regulating holons ability to model and control very complex systems, high resilience to internal and external disturbances, and adapts to changes in the environment [9]. Within a holonic organization, holons can dynamically create and change hierarchies. They can be both autonomous, as well as cooperative. That is, holons can handle circumstances and incidents based on their own knowledge and information available without interference from superiors; at the same time, holons can still receive instructions or be controlled by their superiors. This combined hierarchical and heterarchical behavior ensures superior performance in complex C^2 operations.

Yu et al [5-6] employed concepts from group technology and nested genetic algorithms to solve holonic coordination problem in a two-level structure (operational and tactical levels) involved in planning and executing a single mission. The focus was on asset allocation and task scheduling problems for the expeditionary strike groups (ESG). Park et al [7] modeled three-level structures (viz., strategic, operational, and tactical levels) for MHQ with MOC facing multiple simultaneous or sequential missions.

Over the years, research in reinforcement learning (RL) has advanced to an extent that realistic partially observed stochastic control problems involving semi-Markov decision process (SMDP) models are solvable. In addition, hierarchical reinforcement learning (HRL) techniques have been proposed, including options [16], the hierarchies of abstract machines (HAMs) [17] and maximum Q-value (MAXQ) function decomposition [18]. The HRL techniques depend on the theory of SMDP to provide a formal basis. Sutton et al [16] formalized learning, planning, and representing knowledge at multiple levels of temporal abstraction. Parr [17] developed an approach to hierarchically structure MDP policies, termed HAMs. The emphasis is on simplifying complex MDPs by restricting the class of realizable policies rather than expanding the action choices. Dietterich [18] developed another approach to HRL, termed the MAXQ value function decomposition, which relies on the theory of SMDPs in a manner similar to options and HAMs; however, the MAXQ approach does not rely on reducing the entire problem to a single SMDP unlike options and HAMs. Instead, a hierarchy of SMDPs is created whose solutions can be learned simultaneously. Rohanimanesh and Mahadevan [19] investigated a model for planning under uncertainty with temporally extended actions, where multiple actions can be taken concurrently at each decision epoch.

The present work extends the work in [7] on multi-level coordination problems in MHQ with MOC by developing a two level SMDP process to decide on missions to be executed and their time sequence at the SLC-OLC layer (coordination of future plans), while a SMDP approach to an action-goal attainment (AGA) graph [14] for addressing the mission monitoring/planning issues related to task sequencing and asset allocation at the OLC-TLC layer (coordination of future operations and current operations).

The contributions of this paper are four fold. The three-level architecture gives a solution to the C^2 coordination problem involving a higher level authority's intent (e.g., desired effects at the strategic level), and mission sequencing, mission planning, mission monitoring and mission execution at the SLC-OLC-TLC levels. The second contribution is the coordination mechanism between the SLC-OLC interface layer and the OLC-TLC interface layer that enables the two layers to share the results of individual SMDP problems being solved at each layer, viz., DIME action selection and mission sequencing at the SLC-OLC layer and mission planning and mission monitoring at the OLC-TLC layer. The third contribution of the paper is the use of distributed SMDPs at the OLC-TLC layer to solve individual mission planning problems. The final contribution of the paper is that it provides a framework on how multi-level organizational structures may be employed for the USN's complex and distributed coordination problems involving MHQ with MOC.

Organization of the Paper

This paper is organized as follows. Section II described our three level C^2 organizational design model, and introduces a holonic reference architecture (HRA). Section III shows how two layer SMDP is applied at the SLC-OLC and the OLC-TLC layers. An application example of the approach to sequence and plan multiple missions is discussed in section IV. Herein, the processes of centralized assessment and guidance, distributed and collaborative planning, and decentralized execution are evident in that it employs centralized decision making at the strategic level via a SMDP, collaborative planning at the operational level using distributed SMDPs in terms of specifying the alternative task paths for missions and delineating mission phases, and negotiation mechanisms at the lower level to resolve scheduling conflicts. Finally, the paper concludes with a summary of key findings and future research directions in section V.

II. STRUCTURE OF HOLONIC C^2 REFERENCE ARCHITECTRUE

Three-level Control Architecture

Within the scope of decentralized C^2 requirements, the control architecture should be distributed, abstract and generalized. The control is *abstract* in the sense that the assumptions on the internal structure and the behavior of other DMs should be least restrictive. The *generalized* control requires that a holon be cloned from certain basic structures. The distributed control should also be both *reactive* and *self-organizing*, i.e., control is able to respond to environmental disturbances and adapt to changes during the mission execution process. We categorize the C^2 architectural concepts into the strategic, operational and tactical levels as shown in Fig. 2.

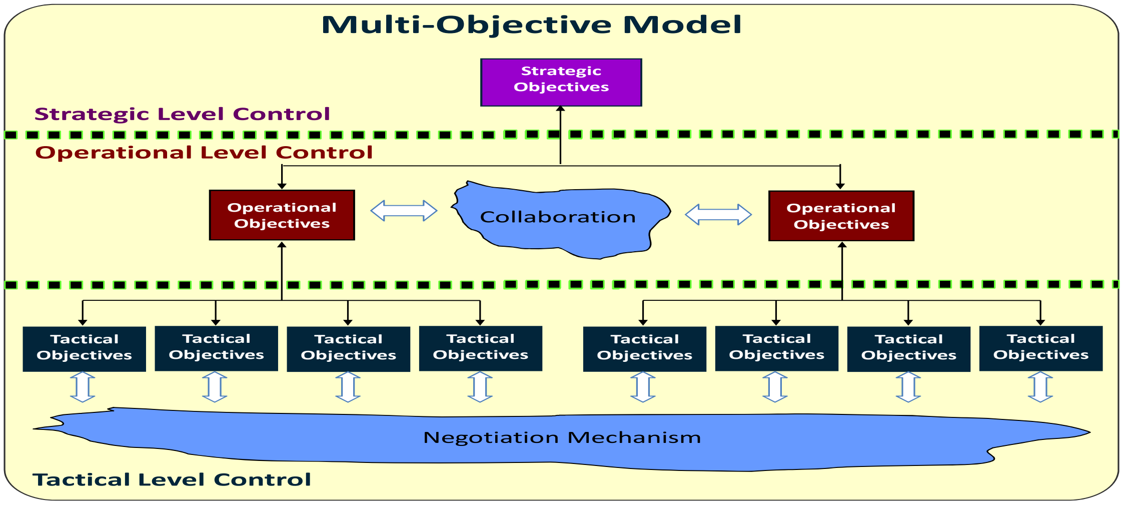


Figure 2. Three level holonic reference control architecture.

Strategic Level Control (SLC) Architecture

The SLC architecture provides a structure for establishing mission objectives and guidance for future plans. At this level, the process is focused on national/international objectives. It gives strategic-level guidance to MHQ commanders in the form of potential DIME actions for various missions, available assets and mechanisms for resolving mission conflicts as they arise during subsequent planning and operations. This level also decides on the time sensitivity of multiple missions and ensures that the missions meet the strategic objectives. We model the strategic guidance using SMDP. The SMDP decides on the sequence of DIME actions for missions with the national level constraints (i.e., diplomatic, information, military and economic). That is, the SMDP decides which DIME actions should be executed for various missions to be planned at the operational level (future plans).

Operational Level Control (*OLC*) *Architecture*

The OLC architecture provides facilities for mission decomposition (i.e., generating the task graph), deliberate planning (future operations), command, and inter-holon coordination/negotiation. At this level, the process is focused on meeting the strategic guidance of the SLC by integrating and synchronizing key objectives at all levels of war. It seeks to produce an initial force structure that places the subordinate units at the right place and at the right time prior to mission execution. During the current operations, it monitors the real-time mission execution and its effects, and adjusts the initial plan, if needed, to ensure that the mission is successfully completed. It also has a collaboration mechanism to resolve conflicts among multiple missions based on the selected DIME actions at the SLC. We model the operational objectives using goal-action graphs involving OR nodes (that represent alternate paths to accomplish the end goals of the mission), AND nodes (representing sub-goals that are necessary to accomplish the end goals), and Exclusive OR (XOR) nodes (representing actions and/or goals that are in conflict or at odds with each other) [14].

Tactical Level Control (*TLC*) *Architecture*

The TLC architecture encapsulates the functional holons that execute the assigned sub-missions or tasks (current operations). This tactical process involves local task scheduling, battlefield pattern recognition, and negotiation mechanism. It also provides an interface to the physical assets. The TLC architecture can have more than one TLC instance (TLC unit); the numbers of instances are decided by deliberate planning at the OLC level. The TLC units can be dynamically added or deleted according to the perceived mission environment. A negotiation mechanism is provided for the TLC units to resolve conflicts among themselves, or to provide coordination as needed.

Coupling the three-level architecture, there are two coordinating decision layers at the SLC-OLC and the OLC-TLC interfaces. The first decision layer (the SLC-OLC layer) is used for deciding on DIME action sequences for multiple missions, and the second decision layer (the OLC-TLC layer) solves the mission planning problem under asset constraints. Task status reports from subordinate holons at the TLC are sent up to holons at the OLC. The monitoring and supervision of the overall progress of the mission and adjustment of tactical actions are promulgated to lower level holons. If missions are in conflict at the OLC, the OLC requests the SLC for strategic guidance to resolve the conflict(s) and yet achieve long-term strategic objectives.

III. TWO LAYER SMDP HIERARCHY

The hierarchical decision (learning) process spanning the two layers of coordination is shown in Fig. 3. At the SLC-OLC layer, we model the optimization problem of selecting the DIME actions to achieve the desired effects as a semi-Markov decision process (SMDP). At the OLC-TLC layer, the problems of planning for each mission are modeled as distributed SMDPs. A discrete-time SMDP is a generalization of MDP, in which the actions have a variable amount of time to complete. The SMDP to solve the DIME action selection problem at the SLC-OLC layer is denoted by $< X, U, P(T), R(T) >$. Here X is the state space, U is the action set, $P(T)$ is the matrix of action and time-dependent state transition probabilities, and $R(T)$ is the action and state-dependent reward structure that is also a function of the (random) time between decision epochs T. The time between decision epochs, T at the SLC-OLC layer is an output of the mission planning problem; each mission planning problem is solved using another SMDP model at the OLC-TLC layer, denoted by $\sim <\Xi, \mathrm{Y}, \Pi(\mathrm{T}), \mathrm{P}(\mathrm{T})>$. Thus, multiple SMDPs are running concurrently at the OLC-TLC layer. The SMDP at the OLC-TLC layer models each mission as a goal-attainment graph [14] and the time between decision epochs of this SMDP, denoted by T, is an output of TLC level as the completion time of a single task (sub-goal) of the goal-attainment graph.

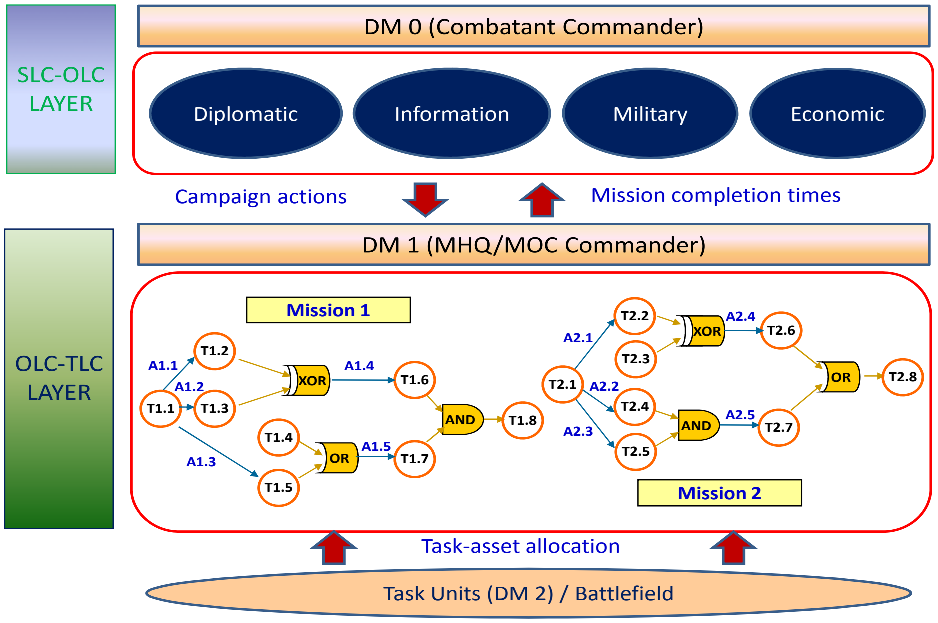


Figure 3. Two layer coordination architecture.

SMDP at the SLC-OLC layer $\sim <X, U, P\ (T), R\ (T)>$

Given a MDP and a set of concurrent temporally extended actions defined on it, the decision process that selects only among multi-actions and executes each one until its termination according to a given termination condition forms a SMDP. The SMDP at the SLC-OLC layer is formulated by learning hierarchically the concurrent action plans over temporally extended actions, which are the variable amount of times to complete individual missions at the OLC-TLC layer. There are hierarchical concurrent actions which are both the courses of action of individual missions at the OLC-TLC layer and goal-attained actions of parallel missions at the TLC. Here, the goal-attained action is the completion time of a goal-attained task and is captured in deciding the courses of action at the OLC-TLC layer. In the SMDP of <*X*, *U*, *P*(*T*), *R*(*T*)>, the time between decision epochs, *T* is defined as the duration of time that any of the missions corresponding to a state in the *X* at the SLC-OLC layer being completed and directly affected by the courses of actions at the OLC-TLC layer.

The SMPD at the SLC-OLC layer is formalized as follows (mathematical details are included in the Appendix).

- The state here represents the mission space. We consider a scenario where the SLC-OLC layer needs to dynamically select a state-dependent policy that decides on the DIME action sequences for multiple missions that are to be planned at the OLC-TLC layer. The combination of military missions, such as peacekeeping, HA/DR (Humanitarian Assistance and Disaster Relief), stability operations, and major combat operations, constitute the states of SMDP (see Table 1).
- The actions represent feasible paths (courses of action) in a directed acyclic network of DIME action sequences from a source node to a destination node as illustrated for a hypothetical scenario in Fig. 4. Note that each mission has a different network graph consisting of feasible action sets.
- The state transition probability is defined as the probability of being in the next state at a decision epoch that is *T* time steps ahead, given the current state, and an action.
- The reward function represents the expected national level resource usage costs, given current state, and an action.

This is how the process works. The SLC-OLC layer is provided the results of previous courses of action by the OLC-TLC layer (e.g., completion times for various course of action). Evidently, the completion times of missions constitute the decision epochs of the SMDP at the SLC-OLC layer. The results from OLC-TLC layer (an output of local SMDPs at the OLC-TLC layer) affect the state transition probabilities of the SMDP process at the SLC-OLC layer. Formally, we define the holding time distribution function $F(T(k) \mid x(k), u_j(k))$ at time k as the probability that any of the missions corresponding to state $x(k)$ at the SLC-OLC layer finishes at time $T(k)$, i.e.,

$$F(T(k) \mid x(k), u_j(k)) = 1 - \prod_{i=1}^{N} (1 - \omega_i(x(k+1), T(k)) z_i(k)), \qquad (1)$$

where $z_i(k)$ denotes the status of mission i at time k in Table 1. Here, z_i =1 denotes the presence of a mission and 0 its absence (see the details at the Appendix). The right hand side of eq. (1) denotes the probability that at least one of the missions terminates in state $\{x(k+1), T(k)\}$ according to its termination condition ω_i (see the details at the Appendix). The SMDP at the SLC-OLC layer selects the DIME actions to minimize the total expected national level resource usage costs given the current mission state. The SLC-OLC layer affects the reward functions of local SMDPs at the OLC-TLC layer by providing DIME policy-dependent mission weights to each of the missions. For a given DIME policy π (i.e., a path in the DIME action network), mission weight is defined as the mean of individual action weights in each phase $\{b_m^{\pi}\}_{m=1}^{M}$ at the SLC-OLC layer:

$b^{\pi} = (1/M)\sum_{m=1}^{M} b_m^{\pi}$, where M is the number of DIME action phases. For example, for a phase *m,* if one selects $b_m^{\pi} = 1$ for a military action, 0.9 for a diplomatic action, and 0.8 for information or economic action, mission weight will have a larger value for a policy having more military actions than other actions.

X	Peacekeeping	HA/DR	Stability Ops.	Major Combat Ops.
x_1	1	1	1	1
x_2	1	1	1	0
x_3	1	1	0	1
x_4	1	1	0	0
x_5	1	0	1	1
x_6	1	0	1	0
x_7	1	0	0	1
x_8	1	0	0	0

Table 1. State space denoting combinations of military missions (operations).

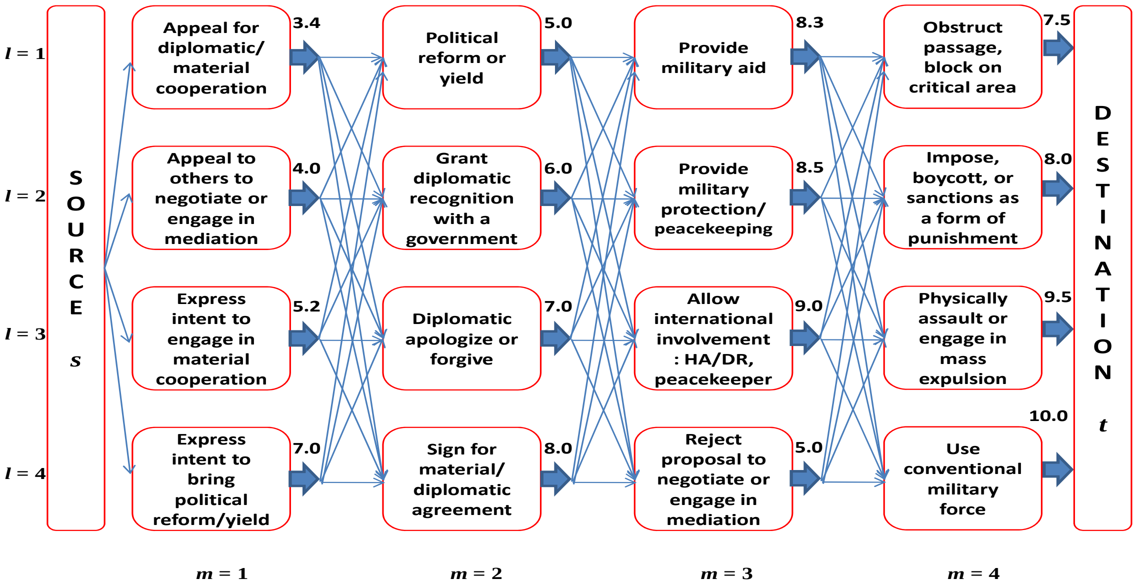

l = 1
l = 2
l = 3
l = 4
SOURCE
s
Appeal for diplomatic/ material cooperation
3.4
Appeal to others to negotiate or engage in mediation
4.0
Express intent to engage in material cooperation
5.2
Express intent to bring political reform/yield
7.0
Political reform or yield
5.0
Grant diplomatic recognition with a government
6.0
Diplomatic apologize or forgive
7.0
Sign for material/ diplomatic agreement
8.0
Provide military aid
8.3
Provide military protection/ peacekeeping
8.5
Allow international involvement : HA/DR, peacekeeper
9.0
Reject proposal to negotiate or engage in mediation
5.0
Obstruct passage, block on critical area
7.5
Impose, boycott, or sanctions as a form of punishment
8.0
Physically assault or engage in mass expulsion
9.5
Use conventional military force
10.0
DESTINATION
t
m = 1
m = 2
m = 3
m = 4

Figure 4. A DIME action network: the numbers between actions denote the required resources for action *l* in phase *m* of the mission.

(Distributed) SMDP at the OLC-TLC layer ~ <Ξ, Υ, Π(T), P(T)>

We convert a mission, represented as an acyclic action-goal attainment (AGA) graph [14], to a SMDP ~<Ξ, Υ, Π(T), P(T)>. Here Euclid letters <Ξ, Υ, Π(T), P(T)> denote the SMDP attributes at the OLC-TLC layer and the time between decision epochs, T at the OLC-TLC layer is a stochastic output of the tasks (sub-goals being executed) at the tactical level. In our previous work [14], we formulated and solved the problem of planning actions to achieve desired end goals (states) subject to resource and time constraints by employing a Markov decision process (MDP)-based method. It addresses the problem of optimally selecting a sequence of actions to transform the mission environment from an initial state to a desired state. It begins with a method to explicitly map an AGA graph to an MDP graph, and develops a dynamic programming (DP) recursion to solve small-sized MDP problems, and limited search AND/OR graph search techniques to solve large-scale MDP problems [14].

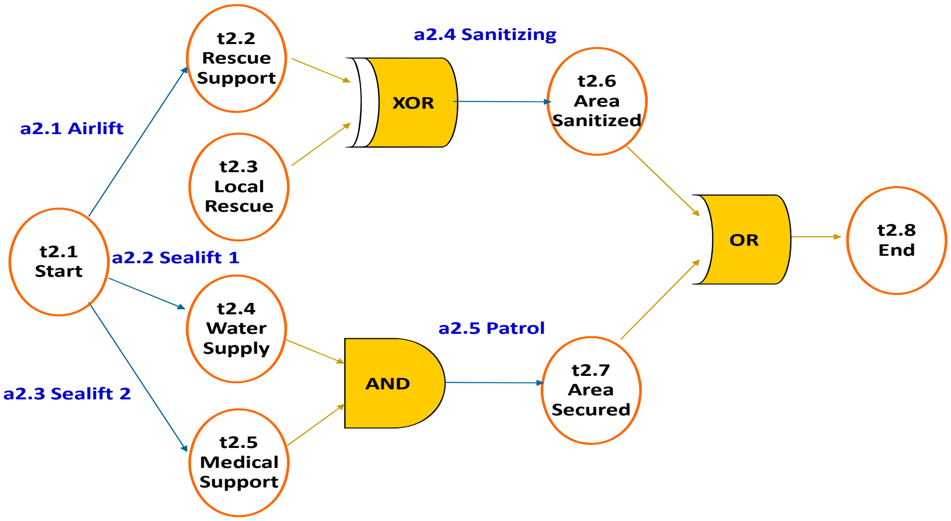


Figure 5. An AGA graph for a HA/DR operation.

In this paper, we transform the AGA graph (representing operational objectives or commander's intent in the form of DIME actions constituting a mission) into a SMDP. The purpose of SMDP is to decide on alternative options to complete missions (developing task graphs), as well as sequencing tasks (see Fig. 5). The distribution of mission completion time, an output of OLC-TLC SMDP, is shared with the SLC-OLC layer to be used to determine the time between decision epochs *T* at the SLC-OLC layer, and the state transition probabilities. The AGA graph consists of OR nodes (that represent alternative paths to accomplish the mission goals), AND nodes (representing sub goals that are necessary to accomplish the mission goals), and XOR nodes (representing actions and/or goals that are in conflict with or at odds with each other). These AGA graphs are transformed into SMDPs and solved via a DP recursion or its approximate variants.

The distributed SMPD for each mission at the OLC-TLC layer is as follows (mathematical details are included in the Appendix).

- The state here represents the combined status of sub goals in the AGA graph: each sub goal is accomplished or not (see Table 2).
- An action represents an option in a given state. Following the same line of reasoning in [14], a set of control availability conditions determines whether a combination of actions is allowed (see Table 3).

- The state transition probabilities is defined as the probability of being in the next state at a decision epoch T time steps ahead (i.e., holding time at the TLC), given current state, and an action.
- The rewards are related to task difficulty and task accuracy of alternative paths in the AGA graph.

Ξ	$t_{2.1}$	$t_{2.2}$	$t_{2.3}$	$t_{2.4}$	$t_{2.5}$	$t_{2.6}$	$t_{2.7}$	$t_{2.8}$
1	1	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0
3	1	0	1	0	0	0	0	0
4	1	1	1	0	0	0	0	0
5	1	0	0	1	0	0	0	0
6	1	1	0	1	0	0	0	0
7	1	0	1	1	0	0	0	0
8	1	1	1	1	0	0	0	0
9	1	0	0	0	1	0	0	0
10	1	1	0	0	1	0	0	0
11	1	0	1	0	1	0	0	0
12	1	1	1	0	1	0	0	0
13	1	0	0	1	1	0	0	0
14	1	1	0	1	1	0	0	0
15	1	0	1	1	1	0	0	0
16	1	1	1	1	1	0	0	0
17	1	1	0	0	0	1	0	1
⋮								
17	1	0	1	1	1	1	1	1

Table 2. The state space representing the combined status of sub goals in the AGA.

Y	$a_{2.1}$	$a_{2.2}$	$a_{2.3}$	$a_{2.4}$	$a_{2.5}$
u_1	1	0	0	0	0
u_2	0	1	0	0	0
u_3	1	1	0	0	0
u_4	0	0	1	0	0
u_5	1	0	1	0	0
u_6	0	1	1	0	0
u_7	1	1	1	0	0
u_8	0	0	0	1	0
u_9	0	1	0	1	0
u_{10}	0	0	1	1	0
u_{11}	0	1	1	1	0
u_{12}	0	0	0	0	1
u_{13}	1	0	0	0	1
u_{14}	0	0	0	1	1

Table 3. Action set.

This is how the process works. The OLC-TLC layer is provided the results of task completion times by the DMs at the TLC. Evidently, the task completion times at the TLC constitute the decision epochs of the SMDP at the OLC-TLC layer. The task-level results from TLC (an output of task scheduling algorithm) affect the state transition probabilities of the SMDP process at the OLC-TLC layer. The holding time distribution function $\Phi(\mathrm{T}(x)|\ x(k),\ u_j(k))$, probability of mission being completed within $\mathrm{T}(x)$ time units of decision epoch k at the OLC-TLC layer by any of the paths in the AGA graph, is given by:

$$\mathrm{F}\ (\mathrm{T}\ (k)\,|\,x(k),u_j(k)) = 1 - \prod_{p=1}^{\mathrm{N}p}(1-\varpi_p(x(k+1),\mathrm{T}\ (k)), \tag{2}$$

where N_p is the number of different paths through which the mission can be completed, and the termination condition is defined in terms of the make span of tasks on a path at the tactical level. The right hand side of eq. (2) denotes the probability that at least one of the paths terminates in state $\{x(k+1),\mathrm{T}(k)\}$ according to its termination condition ϖ_p (For details, the reader is refer to the Appendix). The goal of SMDP at the OLC-TLC layer is to find a policy at each state (best state-dependent action path in the AGA graph) using the task completion conditions provided by the DMs at the TLC, and mission weight provided by the SLC-OLC layer.

The overall coordination process is formalized as shown in Table 4.

1. Initialize SMDP at the SLC-TLC layer $\sim\ < X(0), U(0), P(T(0)), R(T(0)) >$ and SMDPs at the OLC-TLC layer ~ $\{<\Xi(0),\ \Upsilon(0),\ \Pi(\mathrm{T}(0)),\ \mathrm{P}(\mathrm{T}(0))>_i\}_{i=1}^{N}$, where N is the number of missions.

2. Decide on the DIME action policy by solving the SMDP problem at the SLC-OLC layer with current information on mission completion times form the OLC-TLC layer. Transmit mission weights, b^{π} to the OLC-TLC layer.

3. The take-asset assignment results with task completion conditions are provided to the SMDPs at the OLC-TLC layer by the DMs at theTLC.

4. Using the information from steps 2 and 3, each OLC-TLC layer mission planner decides on state-dependent action path in the AGA mission graph and the mission completion time. Transmit the mission completion time to the SLC-OLC layer.

5. Repeat steps 2 to 4 until the policies at the each layer has converge.

Table 4. The overall coordination process

IV. OPERATIONAL MODEL FOR HIERARCHICAL HOLONIC PLANNING

In our previous work [7], the three-level (SLC-OLC-TLC) model for the C^2 holonic reference architecture (HRA) for planning and executing multiple missions was considered. The model included the mission and its decomposition into a task graph, asset allocation, and task scheduling. Those elements of the model are also used in this work, with a focus on mission planning issues involving DIME actions. We consider the following example for illustrative purposes.

Missions: MHQ with MOC is assigned to complete two military missions, which occur in geographically separated areas, e.g., mission 1: capturing a seaport to allow an introduction of follow-on forces (major combat operation), mission 2: rescue activity after a hurricane in the homeland (HA/DR). Fig. 6 shows the geographical situation in this area [10]. We assumed that a single mission has multiple alternative paths of completing it [7]. The mission state 3 is the initial state where MHQ with MOC is tasked to execute a major military operation and an HA/DR.

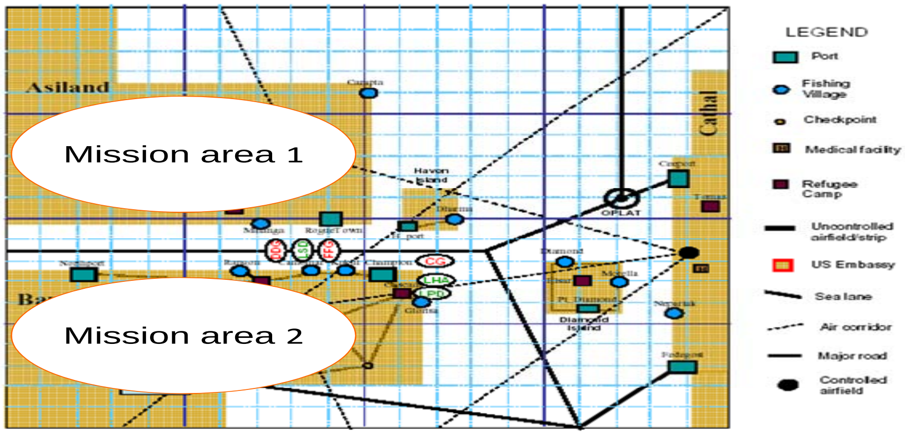
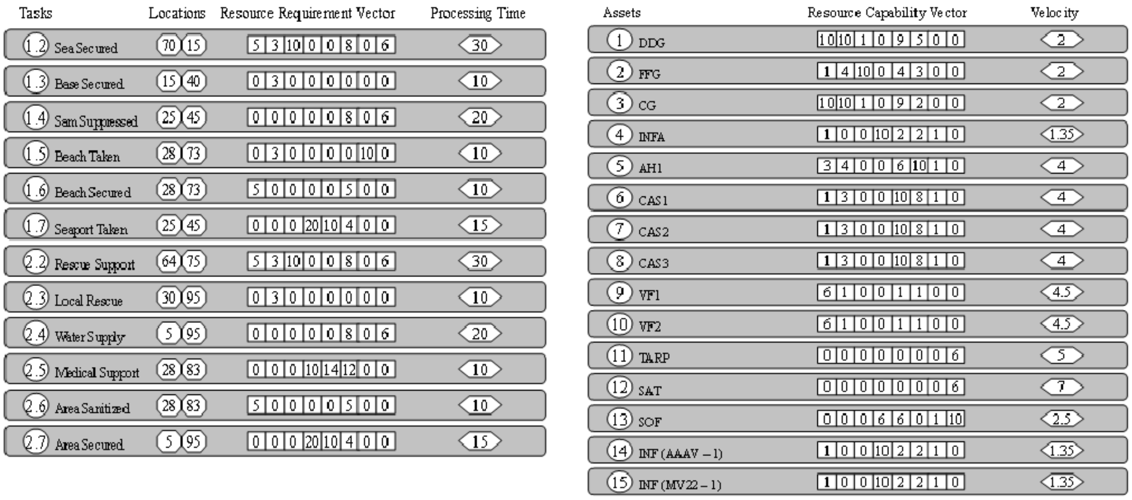


Figure 6. Notional mission areas.



Tasks	Locations		Resource Requirement Vector								Processing Time
1.2 Sea Secured	70	15	5	3	10	0	0	8	0	6	30
1.3 Base Secured	15	40	0	3	0	0	0	0	0	0	10
1.4 Sam Suppressed	25	45	0	0	0	0	0	8	0	6	20
1.5 Beach Taken	28	73	0	3	0	0	0	0	10	0	10
1.6 Beach Secured	28	73	5	0	0	0	0	5	0	0	10
1.7 Seaport Taken	25	45	0	0	0	20	10	4	0	0	15
2.2 Rescue Support	64	75	5	3	10	0	0	8	0	6	30
2.3 Local Rescue	30	95	0	3	0	0	0	0	0	0	10
2.4 Water Supply	5	95	0	0	0	0	0	8	0	6	20
2.5 Medical Support	28	83	0	0	0	10	14	12	0	0	10
2.6 Area Sanitized	28	83	5	0	0	0	0	5	0	0	10
2.7 Area Secured	5	95	0	0	0	20	10	4	0	0	15

Assets	Resource Capability Vector								Velocity
1 DDG	10	10	1	0	9	5	0	0	2
2 FFG	1	4	10	0	4	3	0	0	2
3 CG	10	10	1	0	9	2	0	0	2
4 INFA	1	0	0	10	2	2	1	0	1.35
5 AH1	3	4	0	0	6	10	1	0	4
6 CAS1	1	3	0	0	10	8	1	0	4
7 CAS2	1	3	0	0	10	8	1	0	4
8 CAS3	1	3	0	0	10	8	1	0	4
9 VF1	6	1	0	0	1	1	0	0	4.5
10 VF2	6	1	0	0	1	1	0	0	4.5
11 TARP	0	0	0	0	0	0	0	6	5
12 SAT	0	0	0	0	0	0	0	6	7
13 SOF	0	0	0	6	6	0	1	10	2.5
14 INF (AAAV – 1)	1	0	0	10	2	2	1	0	1.35
15 INF (MV22 – 1)	1	0	0	10	2	2	1	0	1.35

Figure 7. Task resource requirements (left) and asset resource capabilities (right).

The resource requirements for each task and the resource capabilities of each asset are presented in Fig. 7. The resource vector consists of 8 attributes, which are AAW (Anti-Air Warfare), ASUW (Anti-Surface Warfare), ASW (Anti-Submarine Warfare), GASLT (Ground Assault), FIRE (Artillery), ARM (Armor), MINE (Mine Clearing) and DES (Designation). We note that each task needs to be processed by a combination of assets.

DIME action sequencing (Future Plans): The SLC-OLC layer manages multiple missions; it provides guidance for future plans by specifying the sequence of DIME actions to be planned and executed using SMDP. From the optimal SMDP action set, the feasible action-paths for missions 1 and 2 are computed by assuming the mission difficulty factor in eq. (4) in the Appendix to be $\alpha_i \in$ [0.7 0.9] for HA/DR, [0.9 1.1] for the stability operations, and [1.1 1.3] for the major combat operations.

The feasible actions-paths for missions 1 and 2 are shown in Fig. 8. There are 48 action-paths for the mission 1 with upper bound (resource requirements) q_1= 33.3, and 18 action-paths for the mission 2 with q_2= 22.88 (see eq. (5) in the Appendix).

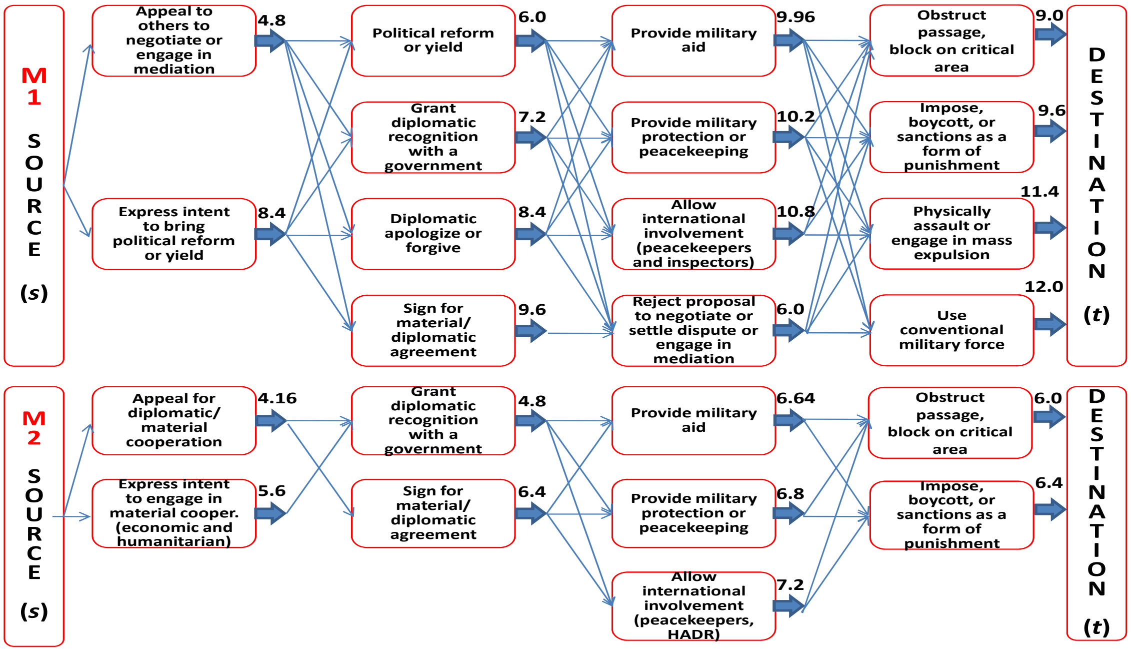

M1 SOURCE (s)
Appeal to others to negotiate or engage in mediation
4.8
Express intent to bring political reform or yield
8.4
Political reform or yield
6.0
Grant diplomatic recognition with a government
7.2
Diplomatic apologize or forgive
8.4
Sign for material/ diplomatic agreement
9.6
Provide military aid
9.96
Provide military protection or peacekeeping
10.2
Allow international involvement (peacekeepers and inspectors)
10.8
Reject proposal to negotiate or settle dispute or engage in mediation
6.0
Obstruct passage, block on critical area
9.0
Impose, boycott, or sanctions as a form of punishment
9.6
Physically assault or engage in mass expulsion
11.4
Use conventional military force
12.0
DESTINATION (t)
M2 SOURCE (s)
Appeal for diplomatic/ material cooperation
4.16
Express intent to engage in material cooper. (economic and humanitarian)
5.6
Grant diplomatic recognition with a government
4.8
Sign for material/ diplomatic agreement
6.4
Provide military aid
6.64
Provide military protection or peacekeeping
6.8
Allow international involvement (peacekeepers, HADR)
7.2
Obstruct passage, block on critical area
6.0
Impose, boycott, or sanctions as a form of punishment
6.4
DESTINATION (t)

Figure 8. The feasible DIME action-paths as computed at the SLC-OLC layer.

The state transition probabilities associated with the SMDP at the SLC-OLC layer $P(x(k+1)|\ T(k),\ x(k),\ u_j(k))$ is obtained by assuming that it is related to the ratio of resources allocated, g_{hi} for a mission i in a state x_h to the resources required, q_i in eq. (7) (see Appendix):

$$P(x(k+1)\,|\,T(k),x(k),u_j(k)) = \frac{\displaystyle\prod_{\substack{i=1\\ z_i(k)(1-z_i(k+1))=1}}^{N} \frac{g_{hi}}{q_i} z_{hi}(k)(1-z_{hi}(k+1))}{1+\dfrac{N-2}{N-1}\displaystyle\sum_{\substack{h=1\\ h(k)\neq h(k+1)}}^{n_h} \prod_{\substack{i=1\\ z_i(k)(1-z_i(k+1))=1}}^{N} \frac{g_{hi}}{q_i} z_{hi}(k)(1-z_{hi}(k+1))}. \quad (3)$$

The numerator in eq. (3) is a function of the resource allocation ratio and whether the mission state for a mission i in a state x_h has changed from 1 to 0, i.e., $z_{hi}(k)(1\text{-}z_{hi}(k+1))$ is 1 only if $z_{hi}(k)$=1 and $z_{hi}(k+1)$= 0. For example, if current state x_1= $\underline{z}$= [1 1 1 1], and the next state is x_2= $\underline{z}$= [1 1 1 0], the state transition vector, $z_{hi}(k)(1\text{-}z_{hi}(k+1))$ is [0 0 0 1]. The holding time distribution function $F(T(k)|\, x(k), u_j(k))$ is calculated by eq. (1) once it is provided by OLC-TLC layer as the SMDP solution. The reward (cost) for a feasible action-path of a mission is calculated as $g_i = \sum_{(l,m)\in A_i} c_{ilm}$, $c_{ilm} \in C_i$, in eq. (4) and the reward (cost) for each state-action pair is computed as: $R(x(k), u_j(k)) = \sum_{i=1}^{N} R(z_i(k), u_j(k)) = \sum_{i=1}^{N} g_i(k)\, z_i(k)$ (see Appendix).

Mission Decomposition (Future Operations): The OLC-TLC layer provides plans for future operations; it devises plans for missions that include the mission decomposition and exploring alternative options (paths) in the AGA graph, such as those in Fig. 9, to select the best option.

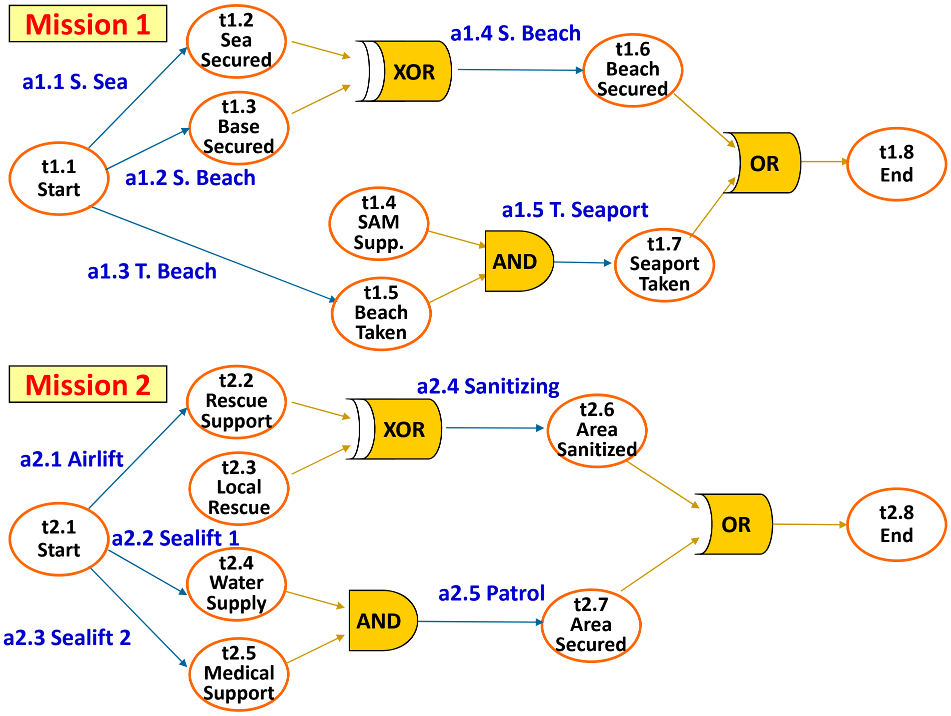
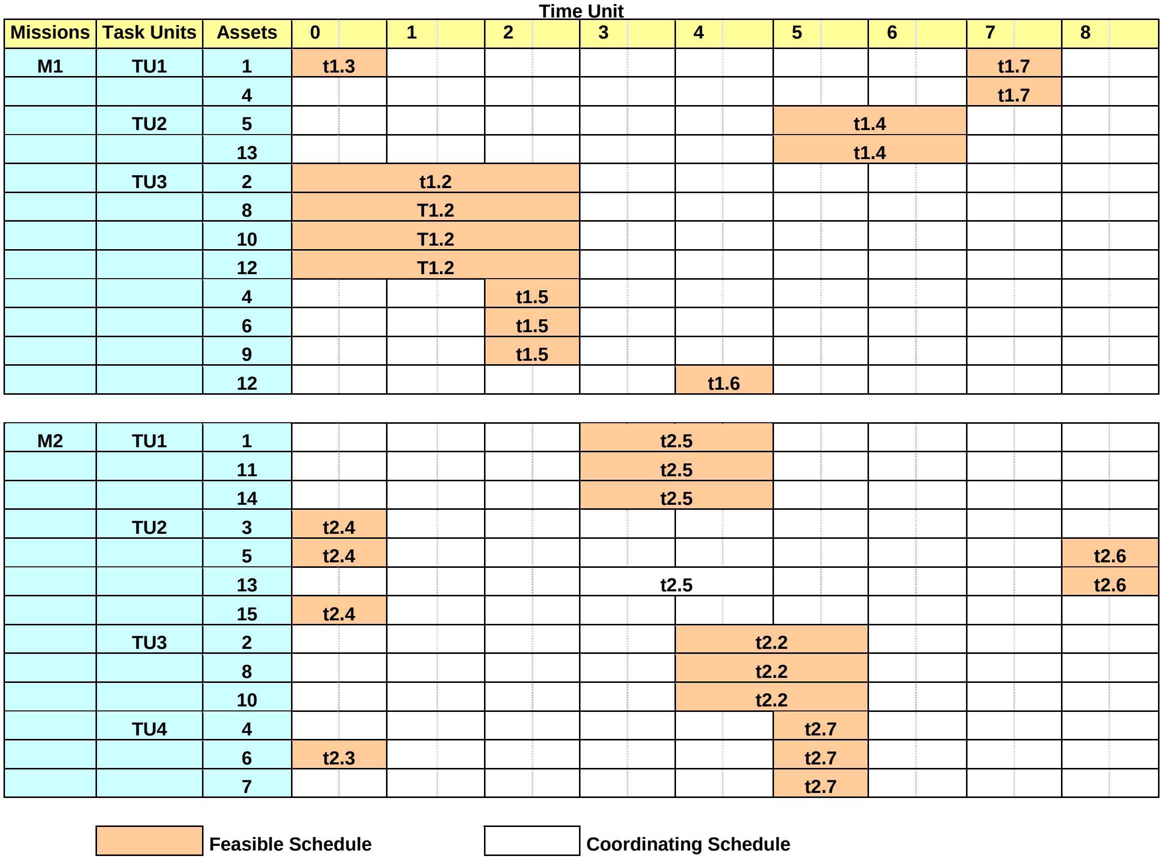


Figure 9. Alternative options on the AGA graph for the two missions.

In addition to this, we use the asset allocation plan and the mission scenario from our previous work [5, 7]. A set of tasks with specified resource requirements, locations, and precedence relations need to be processed by the organization. The tasks are assigned to DMs based on the fit between the resource requirements of tasks and the resource capabilities of DMs. The assigned DMs select and send their assets to the locations where tasks appear in order to execute them with minimum lead time and maximum accuracy. The probability of termination condition ϖ_p for each alternative path is calculated as the task success probability, $g_l\,(k)/\, q_l\,(k)$, which is the ratio of the resource capabilities of assets, g_l and the resource requirements of tasks, q_l, based on the asset allocation and task execution activities at the tactical level (see Fig. 10 in the Appendix) in [7]: $\varpi_p = (1/n_{t,tl=1})\sum_{l=1}^{n_t}[g_l(k)/\, q_l(k)]t_l(k)$, where n_t is the number of tasks and $t_l \in \{0, 1\}$ denotes the status of task l representing t_l =1 for the presence of a task and 0 for its absence.



Missions	Task Units	Assets	0	1	2	3	4	5	6	7	8
M1	TU1	1	t1.3							t1.7	
		4								t1.7	
	TU2	5						t1.4			
		13						t1.4			
	TU3	2	t1.2								
		8	T1.2								
		10	T1.2								
		12	T1.2								
		4			t1.5						
		6			t1.5						
		9			t1.5						
		12					t1.6				
M2	TU1	1				t2.5					
		11				t2.5					
		14				t2.5					
	TU2	3	t2.4								
		5	t2.4								t2.6
		13				t2.5					t2.6
		15	t2.4								
	TU3	2					t2.2				
		8					t2.2				
		10					t2.2				
	TU4	4						t2.7			
		6	t2.3					t2.7			
		7						t2.7			

Figure 10. The operational scenarios for two missions.

The task paths and their success rates for mission 2 in state 1 are shown in Table 5. We assume that the terminal condition ϖ_p is uniformly distributed over the holding time T(k).

p	ϖ_p	Max(T(k))	$t_{2.1}$	$t_{2.2}$	$t_{2.3}$	$t_{2.4}$	$t_{2.5}$	$t_{2.6}$	$t_{2.7}$	$t_{2.8}$	Logic
1	0.16	6	1	1	0	0	0	1	0	1	XOR
2	0.11	8	1	1	0	1	0	1	0	1	
3	0.08	10	1	1	0	0	1	1	0	1	
4	0.18	4	1	0	1	0	0	1	0	1	
5	0.12	6	1	0	1	1	0	1	0	1	
6	0.08	8	1	0	1	0	1	1	0	1	
7	0.09	8	1	0	0	1	1	0	1	1	AND
8	0.05	14	1	1	1	1	1	0	1	1	
9	0.06	14	1	1	0	1	1	1	1	1	XOR/
10	0.06	12	1	0	1	1	1	1	1	1	AND

Table 5. Task success rate for each alternative path of mission 2.

The holding time distribution function $\Phi(\mathrm{T}(k) \mid x(k), u_j(k))$ for an action $u_j(k)$ in state $x(k)$ is calculated using the distribution of termination condition ϖ_p in eq. (2). The state transition probability given current state $x(k)$, an action $u_j(k)$, $\Pi(x(k+1)| \mathrm{T}(k), x(k), u_j(k))$ is obtained by the task success probability. Using eq. (10) in the Appendix, the state transition probabilities at the OLC-TLC layer, $\{\Pi(x(k+1), \mathrm{T}(x)| x(k), u_j(k))\}$ are obtained as shown in Table 6.

x_h	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
x_1	0.35	0.65	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_5	0	0	0	0	0.4	0.6	0	0	0	0	0	0	0	0	0	0
x_9	0	0	0	0	0	0	0	0	0.27	0.73	0	0	0	0	0	0
x_{13}	0	0	0	0	0	0	0	0	0	0	0	0	0.34	0.86	0	0

Table 6. The state transition matrix for mission 2 (u_1 action).

The mission completion reward, $\mathrm{P}(\mathrm{T}(k)| x(k), u_j(k))$ for SMDP at the OLC-TLC layer is calculated by eq. (11) in the Appendix. Furthermore, the reward $\mathrm{P}(x(k), u_j(k))$ of an action is obtained by $\mathrm{P}(x(k), u_j(k))= q_l(k)/ \max(q_l(k))$ estimating mission difficulty in terms of resource requirements of tasks for a task, q_l. The additional reward over the holding time $\mathrm{T}(k)$ is obtained in terms of resource-redundancy for tasks, i.e., $r(\mathrm{T}(k))= r_d(\mathrm{T}(k))/ \max(r_d(\mathrm{T}(k))$, where $r_d(\mathrm{T}(k))$ is the resource-redundancy for tasks during the holding time $\mathrm{T}(k)$. The reward structure of SMDP at the OLC-TLC layer is shown in Table 7. Here, zero entries denote that there are no transitions between those states.

x_h	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}	u_{13}	u_{14}
x_1	0.59	0.63	1.22	0.54	1.13	1.17	1.76	0	0	0	0	0	0	0
x_2	0	0.63	0	0.54	1.13	1.17	0.00	0	0	0	0	0	0	0
x_3	0	0.63	0	0.54	0	1.17	0.00	0.32	0.95	0.86	1.49	0	0	0
x_4	0	0.63	0	0.54	0	1.17	0	0	0	0	0	0	0	0
x_5	0.59	0	0	0.54	0	0	0	0	0	0	0	0	0	0
x_6	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0
x_7	0	0	0	0.54	0	0	0	0.32	0	0.86	0	0	0	0
x_8	0	0	0	0.54	0	0	0	0	0	0	0	0	0	0
x_9	0.59	0.63	1.22	0	0	0	0	0	0	0	0	0	0	0
x_{10}	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0
x_{11}	0	0.63	0	0	0	0	0	0.32	0.95	0	0	0	0	0
x_{12}	0	0.63	0	0	0	0	0	0	0	0	0	0	0	0
x_{13}	0.59	0	0	0	0	0	0	0	0	0	0	0.63	1.22	0
x_{14}	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0
x_{15}	0	0	0	0	0	0	0	0.32	0	0	0	0.63	0	0.95
x_{16}	0	0	0	0	0	0	0	0	0	0	0	0.63	0	0

Table 7. The reward structure of mission 2 at the OLC-TLC layer.

The SMPD is solved via a DP recursion (value iteration) [13]. The optimal policies of each state are shown in Table 8.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
u^π	u_7	u_6	u_6	u_6	u_4	u_4	u_4	u_4	u_2	u_2	u_2	u_2	u_{12}	u_{12}	u_{12}	u_{12}
	$a_{2.1}$ $a_{2.2}$ $a_{2.3}$	$a_{2.2}$ $a_{2.3}$	$a_{2.2}$ $a_{2.3}$	$a_{2.2}$ $a_{2.3}$	$a_{2.3}$	$a_{2.3}$	$a_{2.3}$	$a_{2.3}$	$a_{2.1}$ $a_{2.2}$	$a_{2.2}$	$a_{2.2}$	$a_{2.2}$	$a_{2.5}$	$a_{2.5}$	$a_{2.5}$	$a_{2.5}$

Table 8. The optimal policy for mission 2 at the OLC-TLC layer.

Deliberate Planning (Current Operations): The OLC-TLC layer also provides current operations plans by selecting best options for a mission based on optimal policy (actions). For example, the DM decides on patrol action for securing the area (A2.5), thereby maximizing the operational level rewards while the mission 2 is being completed using path 8 on the AGA graph (see Table 5), i.e., $\underline{t}= x_{17}$ = [1 1 1 1 1 0 1 1] with the holding time T(k)= 14 and the termination condition ω_i= 0.05 (see Fig. 11).

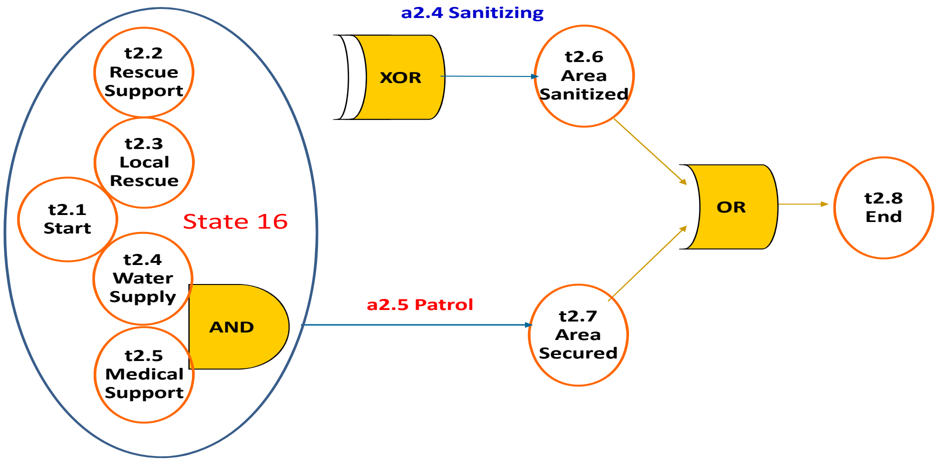


Figure 11. Optimal actions at a state 16 for mission 2.

Now the DM at the SLC-OLC layer is provided the operational information from each DM at the OLC-TLC layer to solve the SMDP problem at the SLC-OLC layer, i.e., the termination condition ϖ_p for alternative paths impacts the termination condition ω_i and the optimal value function at the OLC-TLC layer. These, in turn, are used as the rewards over the holding time $T(k)$ at the SLC-OLC layer. As defined in section IV, the terminal condition ω_i for a mission i is the terminal condition of alternative path having the maximum completion time (make span) for a mission at the OLC-TLC layer. The terminal conditions and the maximum completion time for missions, and their path are shown in Table 9.

i	ω_i	Max($T(k)$)	$t_{i.1}$	$t_{i.2}$	$t_{i.3}$	$t_{i.4}$	$t_{i.5}$	$t_{i.6}$	$t_{i.7}$	$t_{i.8}$	Path #
1	0.06	14	1	1	0	1	1	1	1	1	9
2	0.05	14	1	1	1	1	1	0	1	1	8

Table 9. The terminal conditions ω_i for missions 1 and 2.

The holding time distributions $F(T(k) \mid x(k), u_j(k))$ are obtained by eq. (1); using this in eqs. (3) and (6) in the Appendix, we obtain the overall state transition probability, $P(x(k+1)|T(k), x(k), u_j(k))$ at the SLC-OCL layer. There are n_j matrices of dimension $n_h \times n_h$, where n_j is 57,722 with 18 HADR actions, 61 stability operations and 48 major combat operations.

The reward structures, $R(T(k)| x(k), u_j(k))$ is obtained by eq. (8) in the Appendix, i.e. the sum of the usage cost at the SLC-OCL layer, $R(x(k), u_j(k))$ and the expected total reward $\varsigma^{\pi}(x)$, of an alternative option to complete any mission at the OLC-TLC layer. Instead of summing usage cost and expected reward directly, we normalize them in terms of desiring values: $R(x(k), u_j(k))= \exp[-R(x(k), u_j(k))/\max(R(x(k), u_j(k))]$ and $\varsigma^{\pi}(x) = \exp[\text{mean}(\varsigma^{\pi}(x))/ \max (\varsigma^{\pi}(x))]$. The result of the SMDP at the SLC-OLC layer using the state transition probabilities and reward structure is shown in Fig. 12 in terms of future plans for MHQ / MOC. The two layers may iterate until the decisions at the two layers are congruent.

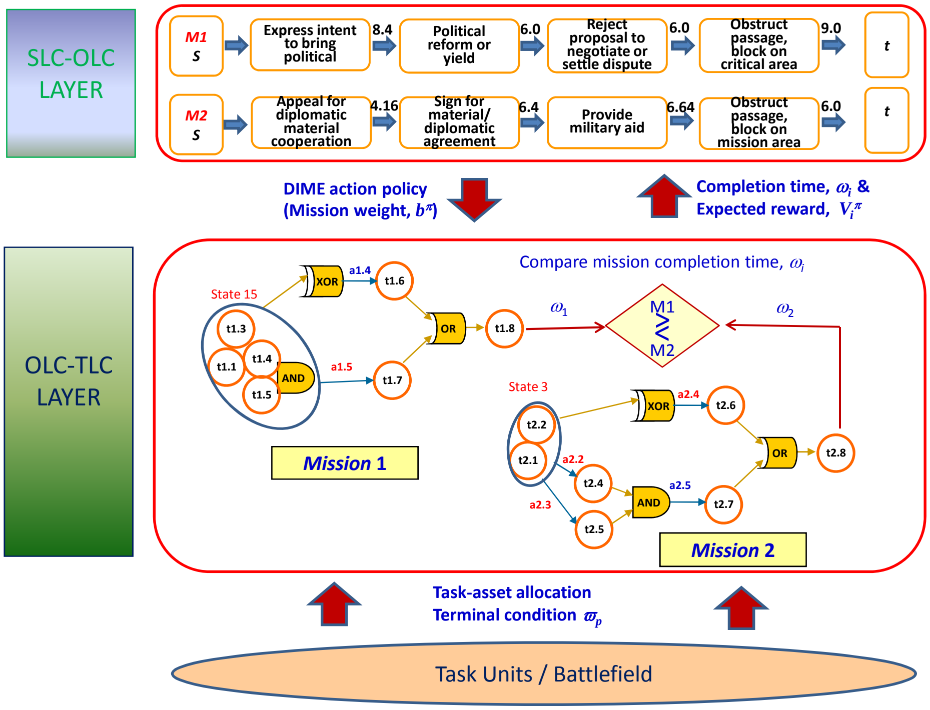

SLC-OLC LAYER
M1 S
Express intent to bring political
8.4
Political reform or yield
6.0
Reject proposal to negotiate or settle dispute
6.0
Obstruct passage, block on critical area
9.0
t
M2 S
Appeal for diplomatic material cooperation
4.16
Sign for material/ diplomatic agreement
6.4
Provide military aid
6.64
Obstruct passage, block on mission area
6.0
t
DIME action policy (Mission weight, b^{π})
Completion time, ω_i & Expected reward, V_i^{π}
OLC-TLC LAYER
Compare mission completion time, ω_i
State 15
XOR
a1.4
t1.6
t1.3
t1.1
t1.4
t1.5
AND
a1.5
t1.7
OR
t1.8
ω_1
M1
M2
ω_2
Mission 1
State 3
t2.2
t2.1
XOR
a2.4
t2.6
OR
t2.8
a2.2
t2.4
AND
a2.5
t2.7
a2.3
t2.5
Mission 2
Task-asset allocation
Terminal condition ϖ_p
Task Units / Battlefield

Figure 12. The information flow between the SMDP processes at the interface layers.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we developed the rudiments of a C2 holonic reference architecture that is applicable to Navy MHQs with MOC for assessing, planning and executing multiple missions and tasks across a range of military operations. We model the coordination issues inherent in the MHQ with MOC via a three-level holonic reference architecture that links tactical and strategic levels of decision making. We sought to demonstrate that the C^2 coordination issues at the three levels, viz., strategic, operational and tactical levels, associated with DIME actions (future plans), and mission planning (future operations and current operations) can be modeled using SMDP formalisms within the proposed holonic architecture. The two layers share the results of individual SMDP problems at each level while the distributed SMDPs at the OLC-TLC layer solve individual mission planning problems. The approach is illustrated using a representative scenario involving multiple missions.

APPENDIX

SMDP formulation at the SLC-OLC layer ~ <X, U, P (T), R (T)>

State space, ***X***: The mission environment is assumed to have n_h states. Each state, $x(k)$, defined at the beginning of a decision epoch k, denotes a combination of missions; it is assumed to belong to a set, $X = \{x_h\}_{h=1}^{n_h}$. If there are N missions, and if we let $z_i \in \{0, 1\}$denote the status of mission i, where $z_i = 1$ denotes the presence of a mission and 0 implies its absence, the state x_h is represented by an N-dimensional binary vector $\underline{z} = [z_1 \; z_2 \ldots z_N]$ and the number of states, n_h can be at most 2^N (in practice much less, see Table 1). Table 1 shows the different states (one for each row) of missions, along with the operational attributes (presence, absence) characterizing them. For example, state $x_3 = [1 \; 1 \; 0 \; 1]$ corresponds to $z_1 = 1$, $z_2 = 1$, $z_3 = 0$ and $z_4 = 1$. That is, this state is characterized by {Peacekeeping, HA/DR, Major combat operations}.

Action set, ***U***: An action $u_j(k)$, defined at the beginning of a decision epoch k, in state x is assumed to belong to the set $U_k(x) = \{u_j(x)\}_{j=1}^{n_j}$, where n_j is the number of feasible action sequences. The feasibility of action sequences is determined by solving a shortest-path problem and a longest-path problem using Dijkstra's algorithm [11] based on the DIME resource requirements of a mission in the network. We employed a normalized CAMEO scale [12] to obtain link costs in this network.

Let c_{ilm} denote the required resources for action l in phase m of the network for mission i. This can be represented as a DIME resource usage cost matrix C_i, where C_i is an L *by* M matrix:

$$C_i = \alpha_i C \square F_i = \begin{bmatrix} c_{i11} & \cdots & c_{i1M} \\ \vdots & \ddots & \vdots \\ c_{iL1} & \cdots & c_{iLM} \end{bmatrix}, \qquad c_{ilm} \geq 0. \tag{4}$$

Here, α_i is the mission difficulty factor and L is the maximum number of actions (rows of C_i matrix) in a phase l and M is the number of phases (columns of C_i matrix).

In addition, C is a (mission-independent) DIME resource usage cost matrix including all possible DIME actions in an area, and F_i is a matrix of feasible action sets for each mission:

$$F_i = \{f_{ilm}\}_{l=1}^{n_l}, \qquad f_{ilm} = \begin{cases} 1, & \text{if an action } l \text{ can be used for a mission } i \text{ in phase } m \\ 0, & \text{otherwise.} \end{cases}$$

In Eq. (4), • denotes Hadamard (Schur) product, i.e., $c_{ijk} = \alpha_i \, c_{jk} \, . f_{ijk}$. Let $\rho(t)$ and $\gamma(t)$

denote the shortest and longest distances (minimum and maximum resource costs) of any path from a source node s to a destination node t, computed using, for example, the Dijkstra's algorithm[1] [11]. We assume that these costs specify an upper bound on the path costs that strategic level decision maker is willing to commit to a mission. Thus, by letting q_i be an upper bound on the resource requirement for a mission i, we assume the following generalized mean with exponent a for q_i:

$$q_i = f(\rho(t), \gamma(t)) = M_a(\rho(t), \gamma(t)) = \left(\frac{(\rho(t))^a + (\gamma(t))^a}{2} \right)^{1/a} . \qquad (5)$$

The value of a allows us to model a variety of behaviors at the strategic level. When $a \rightarrow -\infty$, q_i is the minimum resource cost (shortest distance); when $a = -1$, q_i is the harmonic mean of the minimum and maximum resource cost $2\rho(t)\gamma(t) / [\rho(t)+\gamma(t)]$; when $a = 1$, q_i is the arithmetic mean (average) of the minimum and maximum resource cost $[\rho(\mathrm{t})+\gamma(\mathrm{t})] / 2$; when $a = 2$, q_i is the root mean square value of the minimum and maximum resource cost; and when $a \rightarrow \infty$, q_i is the maximum resource cost (longest distance). In the simulations below, we assume $a = 1$. Given an upper bound on resource usage q_i , all paths with length $\leq q_i$ comprise the feasible action paths A_i for a mission i. Letting $L_i = |A_i|$, the cardinality of action set $|U_k(x_h)| = |U_k(\underline{z})| = \prod_{i=1 \ni z_i=1}^{N} L_i$.

State transition probabilities, **$\{P(x(k+1)|T(k), x(k), u_j(k))\}$**: given current state $x(k)$, and an action $u_j(k)$, the probability of $\{x(k+1), T(k)\}$ being the next state at decision epoch $T(k)$ time steps ahead (i.e., holding time) is denoted by $P(x(k+1)|T(k), x(k), u_j(k))$. Evidently,

$$P(x(k+1), T(k) \,|\, x(k), u_j(k)) = P(x(k+1) \,|\, T(k), x(k), u_j(k)) F(T(k) \,|\, x(k), u_j(k)) \quad (6)$$

where $P(x(k+1) \mid T(k), x(k), u_j(k))$ denotes the state transition probability given current state $x(k)$, an action $u_j(k)$, and the holding time $T(k)$ prior to transition to state $x(k+1)$ at the SLC-OLC layer. $F(T(k) \mid x(k), u_j(k))$ is the holding time distribution function that the next decision epoch occurs within $T(k)$ time units of the current decision epoch k, given current state $x(k)$, and action $u_j(k)$ (see eq. (1)).

[1] Since the graph is acyclic, both the shortest and longest path lengths can be computed using the Dijkstra's algorithm.

We define $P(x(k+1) \mid T(k), x(k), u_j(k))$ as the probability that action $u_j(k)$ is initiated in state $x(k)$ at decision epoch k, without terminating in state $x(k+1)$ until $T(k)$ units. Thus,

$$P(x(k+1)\,|\,T(k),x(k),u_j(k)) = \sum_{x'\in X}[P(x'\,|\,T(k)-T_0,x(k),u_j(k)) \\ \prod_{i=1}^{N}(1-\omega_i(x',T(k)-T_0)z_i(k))P(x(k+1)\,|\,T_0,x',u_j(k))] \tag{7}$$

where T_0 is the single time unit and $\omega_i(x', T(k) - T_0)$ is the termination probability of mission i. The first term denotes the probability of executing missions $T(k) - T_0$ time units and reaching an intermediate state x'; the second term denotes the probability that none of the missions terminates in state $\{x', T(k) - T_0\}$ according to its termination condition ω_i; and the last term denotes the probability that at least one of the missions is completed in a single time step T_0 so that the state transitions to $x(k+1)$. We obtain the distribution of the termination condition ω_i for a mission i in terms of the maximum completion time (make span) of alternative options (paths) for a mission at the OLC-TLC layer.

Reward (Cost) structure, $\{R(T(k)|\ x(k),\ u_j(k))\}$: The reward function $\{R(T(k)|\ x(k), u_j(k))\}$ denotes the expected reward being the next state within time $T(k)$ time units of the current decision epoch k, given current state $x(k)$, and an action $u_j(k)$. The reward (cost) structure $R(x(k),u_j(k))$ is defined as the sum of the usage cost at the SLC-OCL layer, and the expected total reward $\varsigma^{\pi}(x)$, of an alternative option to complete any mission at the OLC-TLC layer. The reward at the OCL-TLC layer is provided as the expected cumulative reward obtained by solving the SMDP problem at the OLC-TLC layer; thus the reward structure for taking an action $u_j(k)$ in state $x(k)$ during $T(k)$ is defined as:

$$R(T(k)\,|\,x(k),u_j(k)) = R(x(k),u_j(k)) + \mathrm{V}^{\pi}(x) = \sum_{i=1}^{N} R(z_i(k),u_j(k)) + \mathrm{V}^{\pi}(x). \tag{8}$$

The first term in eq. (8), $R(x(k), u_j(k))$, refers to the sum of rewards received by SMDP at the SLC-OLC layer for performing action $u_j(k)$ in local state $x(k)$. The second term, $\varsigma^{\pi}(x)$, completes the sum by accounting for rewards earned for completing a mission at the OLC-TLC layer.

Discount rate, β: the relative weight of future rewards, $0< \beta \le 1$.

The objective of SMDP model at the SLC-OLC layer is to determine an optimal policy, i.e., a mapping from states to actions, such that the value function (expected total cost) is minimized. The value function of an initial state $x = x(0)$, for policy π is denoted as:

$$V^{\pi}(x) = E^{\pi}[\sum_{k=0}^{K-1}\beta^k R(T(k)\,|\,x(k),u_j(k)) + \beta^K R(T(K)\,|\,x(K))], \tag{9}$$

where K is the number of decision epochs (planning horizon).

SMDP formulation at the OLC-TLC layer ~ <Ξ, Y, Π(T), P(T)>

State space, Ξ: The goal model is best visualized as a network of action alternatives and their respective outcomes via a directed acyclic graph, termed the AGA graph, Γ(T, A), a task set T = $\{t_l\}_{l=1}^{n_t} \cup \{\upsilon_i\}_{i=1}^{n_\upsilon}$, $t_l \in \{0, 1\}$, $\upsilon_i \in$ {OR, AND, XOR}, and an action set A = $\{a_j\}_{j=1}^{n_a}$, $a_j \in \{0, 1\}$, where t_l is a task node, υ_i is a logic node, a_j is an action, and n_t, n_υ and n_a are the numbers of task nodes, logic nodes and action nodes, respectively (see Fig. 5). Let the binary representation of a goal state $x_h \in \Xi$ be an n_t-dimensional vector $\underline{t} = [t_1 \ldots t_{nt\text{-}1}\ t_{nt}]$, whose l^{th} bit is 1 or 0, depending on whether the l^{th} goal has been successfully achieved or not. However, not all $\underline{t}$: $\sum_{l=1}^{n_t} t_l 2^{l-1} \le 2^{n_t} - 1$ are valid goal states. For example, the state $x_h =$ [11010111] = 235 is not valid for an AGA graph in Fig. 5, because $t_{2.7}$ cannot be 1 if either $t_{2.4}$ or $t_{2.5}$ are unattained.

The validity of a goal state $x_h = \underline{t}$ is established via a set of logical functions $\{g_{tl}\ (\underline{t})\}$ defined in [14]. The logical functions $\{g_{tl}(\underline{t})\}$ are as follow: $g_{t2.1} = t_{2.1} = 1$; $g_{t2.2} = g_{t2.3} = g_{t2.4} = g_{t2.5} =$ 0 or 1; $g_{t2.6} = t_{2.2} \oplus t_{2.3}$; $g_{t2.7} = t_{2.4} \cdot t_{2.5}$; $g_{t2.8} = t_{2.6} + t_{2.7}$. Due to $g_{tl}(\underline{t})$ requirements, the cardinality of Ξ, n_h, depends largely on the size of the unconstrained goals (e.g., $t_{2.2}$, $t_{2.3}$, $t_{2.4}$ and $t_{2.5}$), rather than n_t. Thus, instead of 2^8, n_h can be as small as 17, in this case. Moreover, the absorbing states, i.e., the set of valid goal states representing the desired terminal goal states (e.g., all valid goal states with $t_{2.8}$=1), can simply be absorbed into a single state. This reduces n_h even further. For example, a subset of such states is highlighted in Table 2. Consequently, n_h is reduced from the original 256 to 17. The list of all valid goal states (one for each row) of tasks, along with the tactical attributes (successfully achieved, not achieved) characterizing them is shown in Table 2. For example, state x_3 = [1 0 1 0 0 0 0 0] corresponds to $t_{2.1}$= 1, $t_{2.2}$= 0, $t_{2.3}$= 1, $t_{2.4}$= 0, $t_{2.5}$= 0, $t_{2.6}$= 1, $t_{2.7}$= 0, and $t_{2.8}$= 0. That is, the successfully achieved tasks of this state are tasks 2.1 and 2.3.

Action set, Y: The validity of the control action Y = $\{\underline{a}: a_j \in g_{aj}(\underline{a}), \forall j = 1, \ldots, n_a\}$, n_u = |Y| is established via a set of functions $\{g_{aj}(\underline{a})\}$ as in [14]. The logical functions $\{g_{aj}(\underline{a})\}$ are as follow: $g_{a2.1} = g_{a2.2} = g_{a2.3} =$ 0 or 1; $g_{a2.4} = \overline{a}_{2.1}$; $g_{a5} = \overline{a}_{2.1} \square \overline{a}_{2.2}$. The symbol '$\overline{\bullet}$' (over bar denotes) a logical complement to the argument. The function $g_{a2.4}$ specifies that $a_{2.4}$ is only allowed if $a_{2.1}$ is not. Also, the function $g_{a2.5}$ restricts that the inclusion of $a_{2.5}$ necessitates the exclusion of $a_{2.2}$ and $a_{2.3.}$ The list of all valid control functions are shown in Table 3. Table 10 lists all reachable goal states from each valid goal state via an application of a feasible control action [14].

State transition probabilities, **{Π(*x*(*k*+1), T(*x*)| *x*(*k*), *u*$_j$(*k*))}**: given current state $x(k)$, an action $u_j(k)$, the probability of $\{x(k+1), \mathrm{T}(x)\}$ being the next state at decision epoch of holding time T(k) at the OLC-TLC layer is denoted by Π(x(k+1), T(x)| x(k), u_j(k)):

$$\mathrm{P}\,(x(k+1), \mathrm{T}\,(k) \mid x(k), u_j(k)) = \mathrm{P}\,(x(k+1) \mid \mathrm{T}\,(k), x(k), u_j(k))\mathrm{F}\,\,(\mathrm{T}\,(k) \mid x(k), u_j(k)) \quad (10)$$

where Π(x(k+1)| T(x), x(k), u_j(k)) denotes the state transition probability given current state x(k), an action u_j(k), and holding time T(k) at the OLC-TLC layer. Here Π(x(k+1)| T(x), x(k), u_j(k)) is defined as the task success probability, based on the asset allocation and task execution activities at the tactical level.

Reward (Cost) structure, **{P(T(*k*)| *x*(*k*), *u*$_j$(*k*))}**: The reward P(T(k)| x(k), u_j(k)) of an action u_j(k) is represented by the intensity score function which the probability of answering correctly a particular response category [15]. The intensity score function defined as the cumulative form of the logistics function:

$$\mathrm{R}\,(\mathrm{T}\,(k)\,|\,x(k),u_j(k)) = 1/[1+\exp(-b^{\pi}(\mathrm{R}\,(x(k),u_j(k)) - r(\mathrm{T}\,(k))))]. \qquad (11)$$

where b^{π} is the mission weight of the policy (action) π at the SLC-OLC layer, P(x(k), u_j(k)) is estimated reward of mission difficulty in terms of resource requirements of tasks, and r(T(k)) is the accrued reward of assigning resources to tasks is calculated in terms of excess resource allocation.

The value function of an initial state x = x(0), for policy π is written as:

$$\mathrm{V}^{\pi}(x) = E^{\pi}[\sum_{k=0}^{K-1} \beta^{k}\mathrm{R}\,(\mathrm{T}\,(k)\,|\,x(k),u_j(k)) + \beta^{K}\mathrm{R}\,(\mathrm{T}\,(k)\,|\,x(K))] \qquad (12)$$

where β is discount rate and K is the number of decision epochs at the OLC-TLC layer.

Ξ(k)	Y(x)	{Ξ(k+1)| Ξ(k), Y(x)}	Ξ(k)	Y(x)	{{Ξ(k+1)| Ξ(k), Y(x)}	Ξ(k)	Y(x)	{{Ξ(k+1)| Ξ(k), Y(x)}
x_1	u_1	x_1, x_2	x_3	u_{10}	x_3, x_{11}, x_{17}	x_9	u_2	x_9, x_{13}
	u_2	x_1, x_5		u_{11}	x_3, x_7, x_{11},	x_{11}	u_3	x_9, x_{13}, x_{14}
	u_3	x_1, x_2, x_5, x_6			x_{15}, x_{17}	x_{10}	u_2	x_{10}, x_{14}
	u_4	x_1, x_9	x_4	u_2	x_4, x_8	x_{11}	u_2	x_{11}, x_{15}
	u_5	x_1, x_2, x_9, x_{10}		u_4	x_4, x_{12}		u_8	x_{11}, x_{17}
	u_6	x_1, x_5, x_9, x_{13}		u_6	x_4, x_8, x_{12}, x_{16}		u_9	x_{11}, x_{15}, x_{17}
	u_7	x_1, x_2, x_5, x_6, x_9, x_{10}, x_{13}, x_{14}	x_5	u_1	x_5, x_6	x_{12}	u_2	x_{12}, x_{16}
x_2	u_2	x_2, x_6		u_4	x_5, x_{13}	x_{13}	u_1	x_{13}, x_{14}
	u_4	x_2, x_{10}		u_5	x_5, x_6, x_{13}, x_{14}		u_{12}	x_{13}, x_{17}
	u_6	x_2, x_6, x_{10}, x_{14}	x_6	u_4	x_6, x_{14}		u_{13}	x_{13}, x_{14}, x_{17}
x_3	u_2	x_3, x_7	x_7	u_4	x_7, x_{15}	x_{14}	u_{12}	x_{14}, x_{17}
	u_4	x_3, x_{11}		u_8	x_7, x_{17}	x_{15}	u_8	x_{15}, x_{17}
	u_6	x_3, x_7, x_{11}, x_{15}		u_{10}	x_7, x_{15}, x_{17}		u_{12}	x_{15}, x_{17}
	u_8	x_3, x_{17}	x_8	u_4	x_8, x_{16}		u_{14}	x_{15}, x_{17}
	u_9	x_3, x_7, x_{17}	x_9	u_1	x_9, x_{10}	x_{16}	u_{12}	x_{16}, x_{17}

Table 10. {Ξ(k+1)| Ξ(k), Y(x)} reachability.

REFERENCES

[1] United State Fleet Forces Command. 2007. *Maritime Headquarters with Maritime Operations Center Concept of Operations (CONOPS).*

[2] Levchuk, G. M., Y. N. Levchuk, J. Luo, K. R. Pattipati, and D. L. Kleinman. 2002. Normative Design of Organizations-Part I: Mission Planning. In *IEEE Trans. Syst., Man, Cybern. A, vol. 32*: 346-359.

[3] Levchuk, G. M., Y. N. Levchuk, J. Luo, K. R. Pattipati, and D. L. Kleinman. 2002. Normative Design of Organizations-Part II: Organizational Structure. In *IEEE Trans. Syst., Man, Cybern. A, vol. 32*: 360-375.

[4] Levchuk, G. M., Y. N. Levchuk, C. Meirina, K. R. Pattipati, and D. L. Kleinman. 2004. Normative Design of Organizations-Part III: Modeling Congruent, Robust, and Adaptive Organizations. In *IEEE Trans. Syst., Man, Cybern. A, vol. 34*: 337-350.

[5] Yu, F., F. Tu, and K. R. Pattipati. 2006. A Novel Congruent Organizational Design Methodology Using Group Technology and a Nested Genetic Algorithm. In *IEEE Trans. Syst., Man, Cybern. A, vol. 36*: 5-18.

[6] Yu, F., F. Tu and K. R. Pattipati. 2008. Integration of a Holonic Organizational Control Architecture and Multi-Objective Evolutionary Algorithm for Flexible Distributed Scheduling. In *IEEE Trans. Syst., Man, Cybern. A, vol. 38*: 1001-1017.

[7] Park, C., D.L. Kleinman and K.R. Pattipati. 2008. Holonic Scheduling Concepts for C^2 Organizational Design for MHQ with MOC. 13th ICCRTS. June 17-19, Bellevue, WA.

[8] Koestler, A. 1968. *The Ghost in the Machine*. New York: The Macmillan Company.

[9] Giret, A. and V. Botti. 2004. Holons and agents, 2004. In *Journal of Intelligent Manufacturing*, 15, 645-659.

[10] Hutchins, S.G., S. Weil, D. L. Kleinman., S. P. Hocevar, W. G. Kemple, K. Pfeiffer, D. Kennedy, H. Oonk, G. Averett, and E. Entin. 2007. An investigation of ISR Coordination and Information presentation strategies to support expeditionary strike groups. 12th ICCRTS, June 19-21, Newport, RI.

[11] Dijkstra, E.W. 1959. A note on two problems in connexion with graphs. *Numerische Mathematik*, Vol. 1: 296-271.

[12] http://web.ku.edu/keds/index.html

[13] Puterman, M. 1994. *Markov decision process: discrete stochastic dynamic programming*. New York: John Wiley & Sons.

[14] Meirina, C., Y. N. Levchuk, G. M. Levchuk, and K. R. Pattipati. 2008. A Markov Decision Problem (MDP) Approach to Goal Attainment. In *IEEE Trans. Syst., Man, Cybern. A,* Vol. 37: 116-132.

[15] Baker, F.B. and S-H Kim. 2004. *Item response theory*. New York: Marcel-Dekker.

[16] Sutton, R., D. Precup and S. Singh. 1999. Between MDPs and Semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning. In *Artificial Intelligence*, Vol. 112: 181-211.

[17] Parr, R. 1998. Hierarchical Control and Learning for Markov Decision Process. PhD Thesis, University of California, Berkeley.

[18] Dietterich, T. 2000. Hierarchical Reinforcement Learning with MAXQ value function decomposition. In Journal of *Artificial Intelligence Research*, Vol. 13: 227-303.

[19] Rohanimanesh, K. and S. Mahadevan. 2001. Decision-Theoretic Planning with Concurrent Temporally Extended Actions. 17th Conference on Uncertainty in Artificial Intelligence. August 2-5, Seattle, WA.