Title: **Integration of Holarchy and Holonic Scheduling Concepts for C2 Organizational Design** *

(Student Paper Submission)

Suggested Track: **C4ISR/C2 Architecture**

Feili Yu
Graduate Student, ECE Dept., UCONN
Storrs, CT
Fax: 860-486-5585
Phone: 860-486-5965
e-mail: yu02001@engr.uconn.edu

Candra Meirina
Graduate Student, ECE Dept., UCONN
Storrs, CT
Fax: 860-486-5585
Phone: 860-486-5965
e-mail: meirina@engr.uconn.edu

Sui Ruan
Graduate Student, ECE Dept., UCONN
Storrs, CT
Fax: 860-486-5585
Phone: 860-486-5965
e-mail: sruan@engr.uconn.edu

Fang Tu,
Ph.D., Qualtech Systems, Inc.
100 Great Meadow Rd., St. 603
Wethersfield, CT 06109
Ph: (860)257-8014 x120
e-mail: fang@teamqsi.com

Correspondence:

Prof. Krishna R. Pattipati
Dept. of ECE, The University of Connecticut
Dept. of Electrical & Computer Engineering
371 Fairfield Road, Unit 2157
Storrs, CT 06269-2157
Fax: 860-486-5585
Phone: 860-486-2890
e-mail: krishna@engr.uconn.edu

* This work was supported by the Office of Naval Research under contract # N00014-00-1-0101.

Integration of Holarchy and Holonic Scheduling Concepts for C^2 Organizational Design

Feili Yu[1], Candra Meirina[1], Sui Ruan[1], Fang Tu[2], and Krishna Pattipati[1]

***Abstract*— Based on the concept of autonomous co-operating holons, this paper presents a C^2 holonic reference architecture (HRA). The control architecture consists of two levels: operational level control (OLC) and tactical level control (TLC). The authority and control are highly distributed among holons belonging to different levels in order to empower the edges, while the integration of system is ensured to achieve overall mission objectives. The architecture has a high degree of self-similarity, which reduces the complexity in integrating new components and enables easy reconfiguration. An operational model for C^2 HRA application shows that it exhibits robustness in the face of disturbances, adaptability and flexibility in the face of environmental changes.**

Keywords: command and control (C^2), decision maker (DM), operational level control (OLC), tactical level control (TLC), holonic reference architecture (HRA), tactical level control unit (TU)

I. Introduction

Keys for Command and Control (C^2) organizations to meet the challenges stemming from the increasingly dynamic and unpredictable environments are the following: (a) dispersing the organizational center of gravity among the war-fighters, as well as transferring organizational control to the lowest level; (b) shifting from a rigid centralized control/decentralized execution to a more flexible decentralized control/decentralized execution; and (c) broadening the war-fighters' operational autonomy, while maintaining the overall mission objectives [2].

Innovative concepts and technologies, made possible in the current information age, offer tremendous increases in military's precision, reach, and connectivity, ushering in a new era of joint operational effectiveness. This provides a conducive setting for a more integrated C^2 concepts of managing the sea, land, air, space, and cyberspace power to a greater extent than ever [1].

In the information age, the organizational C^2 refers to the ability to conduct dynamic action synchronization, achieve organizational agility, and increase speed of command over a robust, decentralized architecture. The empowerment of the 'edges' of an organization encourages appropriate interactions between and among members in the organization [2]. The approach, suggested in [2], is to un-couple the organizational command and control. Here, *command* refers to the overall intent, while *control* defines the individual decision maker's attributes. The objective of this C^2 decentralization is to disperse the gravity of control across the organization, thereby providing effective military power at the furthest edge.

Traditional C^2 hierarchy keeps authority and information at the center. Arthur Koestler [5] observes that most of the complex systems are organized hierarchically: the control flow is typically top-down and the feedback information is bottom-up. The decision makers at the upper level define tasks and coordinate lower level units, while decision makers at the lowest level execute the tasks. One of the many merits of a hierarchical C^2 structure is that it provides $unity\ of\ command$, which refers to the principle that a subordinate should have one and only one superior to whom he or she is directly responsible. Because military power is the product

[1]Electrical and Computer Engineering Department, University of Connecticut, Storrs, CT 06269-1157, USA. E-mail: [yu02001, meirina, sruan, krishna]@engr.uconn.edu, [2]Qualtech Systems, Inc., 100 Great Meadow Rd., St. 603, Wethersfield, CT 06109, USA. Email: Fang@teamqsi.com. This work is supported by the Office of Naval Research under contract No. 00014-00-1-0101.

of multiple capabilities, a centralized C^2, as an embodiment of the principle of $unity\ of\ command$, is essential to effectively fuse these capabilities. However, hierarchical control assumes deterministic behavior of the components [14]. Studies indicate that a fixed, vertically integrated hierarchy has the following drawbacks [18]: (a) limited ability for reconfiguration to novel situations; (b) slower response and limited immediate intelligent actions in the face of a major disturbance due to the rigidity of a hierarchy; and (c) limited propagation of bottom-up information due to a multi-level bureaucratic structure.

The conventional alternative to hierarchy is heterarchy. A heterarchial structure has the attributes of distributed intelligence, diversity, self-organiztion, and lateral accountability [15]. In a heterarchical architecture, decision makers (DMs) communicate as peers. There are no fixed supervisor/subordinate relationships. Each DM has equal right of access to resources and independent modes of operation. The coordination among DMs is realized by using a market mechanism, such as the 'contract net protocol' or the 'request for bid', etc. Most heterarchical systems have no central controller. Some inherent capabilities of heterarchy include self-configuration, flexibility, fault tolerance, reduced complexity, and emergent behaviors [16]. However, except for a few applications, heterarchical architecture is not widely used due to the following drawbacks: (a) limited performance due to the absence of global information; (b) unpredictability of organizational behaviors; (c) sensitivity of system performance to coordination protocols; (d) the low efficiency of market-based negotiation mechanism resulting in a slow decision process; (e) limited emergent behaviors; and (f) potential for chaotic behaviors.

A hybrid organizational architecture, termed holonic structure or holarchy, is proposed in order to overcome the drawbacks of both hierarchy and heterarchy. The holonic structure combines features of these two structures, and addresses key requirements of C^2 organizational structures operating in dynamic and uncertain situations. The term 'holonic' is derived from the word 'holon', and was introduced by Koestler in the context of social and living organisms [5]. This term means the combination of 'wholes' and 'parts'. Thus, 'holons' refer to autonomous self-reliant units, which hold a degree of independence and are able to manage contingency without interference from their superiors. Accordingly, a holarchy refers to a hierarchy of self-regulating holons with the following advantages [6]: (a) ability to model very complex systems; (b) high resilience to internal and external disturbances; (c) adaptability to changes in the environment. 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 behavior ensures effectiveness in complex C^2 operations.

The focus of this paper is to translate the concepts of holon and holarchy into a set of appropriate models and algorithms for C^2 organizational design. The goal is to realize the benefits that these concepts provide, viz., robustness in the face of disturbances, as well as adaptability and flexibility in the face of changes.

Accordingly, this paper is organized as follows. Section II reviews our previous research work on C^2 organizational design and introduces the holonic reference architecture (HRA). An operational model for holonic scheduling is discussed through an illustrative example in Section III. Here, the processes of operational level scheduling, tactical level scheduling, reactive scheduling, and negotiation mechanism are discussed. Finally, the paper concludes with a discussion on key findings and future work.

II. Structure of Holonic C^2 Reference Architecture

A. Previous Research on Organizational Design

Our previous research on C^2 organizational design has included the modeling and synthesis of organizational structures to achieve a set of command objectives: maximize speed of command, minimize coordination, balance workload, and so on. The mission modeling and three-phase organizational design methodology allowed one to overcome the computational complexity by synthesizing a command structure via an iterative solution of a sequence of smaller and well-defined optimization problems [7], [8]. In [9], the C^2 organizational structure is conceptualized as a capacitated topological design problem to account for average information delay. What we found in this work is that hierarchical structures suffer from link and nodal overheads, and the resulting information delays degrade their performance. A methodology to design heterarchical organizational structures that considers information/command transfer and processing activities was presented in [10]. A related methodology, employing concepts from group technology and nested genetic algorithms, to synthesize heterarchical organizational structures was proposed in [11]. Despite these research efforts, models for robust, adaptive, and flexible organizational structures are still in short supply.

B. Two-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 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 following two levels (see Fig. 1):

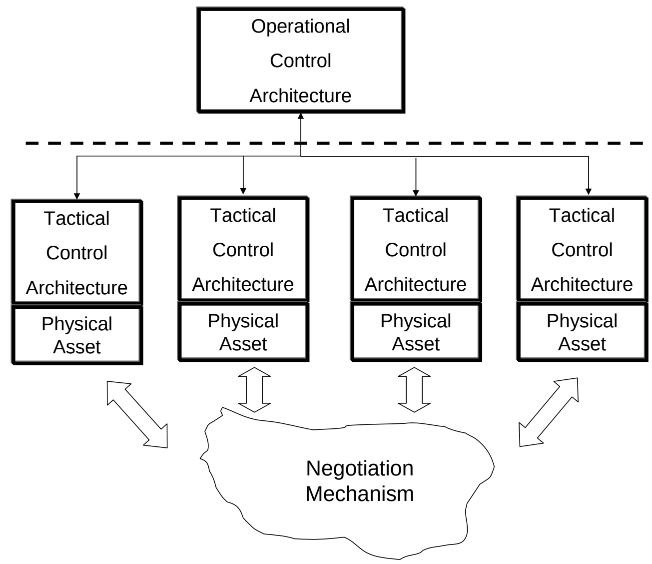

Fig. 1. Two Level Holonic Reference Control Architecture

(i) Operational Level Control (OLC) architecture, which provides a facility for mission decomposition, deliberate planning, command, inter-holon coordination. At this level, the process is focused on overall mission objectives ("commander's intent"). 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 mission, it monitors the real-time mission execution and adjusts the initial plan, if needed, to ensure that the mission is successfully completed.

(ii) Tactical Level Control (TLC) architecture, which encapsulates the functional holons that execute the assigned sub-missions or tasks. This tactical process involves local task scheduling, battlefield pattern recognition, and negotiation mechanism. It also provides the interface to physical assets. The TLC architecture can have more than one TLC instance (TLC unit), the number of instances being decided by deliberate planning in the OLC architecture. TLC units can be dynamically added or deleted according to the perceived situation. A negotiation mechanism is provided for TLC units to resolve conflicts among TLC units, or to provide coordination as needed.

The two-level architecture is coupled: status/situation 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 activities is promulgated to lower level holons. The OLC and TLC architecture is discussed further in the following subsections.

C. Operational Level Control (OLC) architecture

The OLC architecture (illustrated in Fig. 2) is built around six types of basic holons: Operational holon (OPH), intelligence holon (INH), planning holon (PLH), coordinating holon (COOH), promulgating holon (PRH), and communication holon (COMH). Each of these holons is responsible for one aspect of operational level control. The different types of holons are described in the following paragraphs.

$Operational\ holon$ (OPH) plays the role of a superior agent, from which orders are sent to subordinate holons, such as the PLH and the COOH. It receives a mission from a higher level holon and decomposes the mission into many sub-missions or tasks. Based on its strategy, it populates the task graph with task precedence constraints and passes them to the PLH as well as the COOH to make initial plans and initial coordination patterns for the lower TLC units. During mission execution, it monitors the mission progress by retrieving intelligence information from the INH. If an adjustment to current plans and coordination is necessary, the concomitant orders are issued. The OPH is also responsible for resolving conflicts among subordinate holons from both the OLC and the TLC, if they cannot reach an agreement easily.

$Intelligence\ holon$ (INH) encapsulates the process of data collection from different sources, data fusion, and intelligence assessment. The objective is to fulfill the Operational holon's information and decision support requirements. It is also responsible for alerting both the OPH and the PLH when a significant event occurs. INH should have the highest information processing capability among all other holons.

$Planning\ holon$ (PLH) creates a plan that will enable the TLC units to accomplish the submissions assigned to them. Three aspects of work will be involved in the planning process: (a) determine how many TLC units are needed for mission execution; (b) optimally or near optimally assign resources and tasks to each TLC unit, given the objectives from OPH; and (c) suggest initial timing (start time, duration, end time) for each task. It is essential for the PLH to interact with the INH and the OPH to adjust plans in real-time in response to disturbances and unpredictable events. The PLH possesses strong computational capability, which enables it to create real-time plans efficiently.

$Coordinating\ holon$ $(COOH)$ is responsible for directing all TLC units in the target area and coordinating their efforts to achieve the most effective use of available resources. The COOH ensures the operations of TLC units to correspond to plans made by the PLH. It also resolves all the conflicts that are not resolvable among the TLC units through negotiation mechanism. It has authority to adjust the coordination patterns, when an unforeseen event occurs.

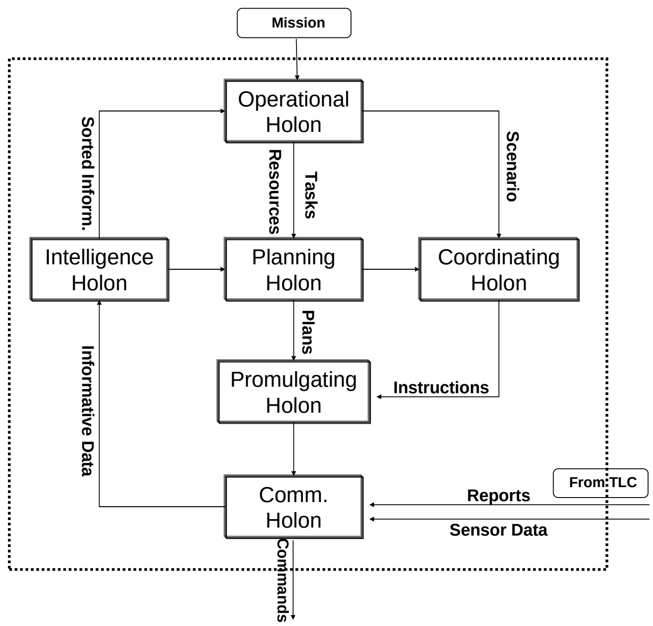

Fig. 2. Operational Level Control Architecture

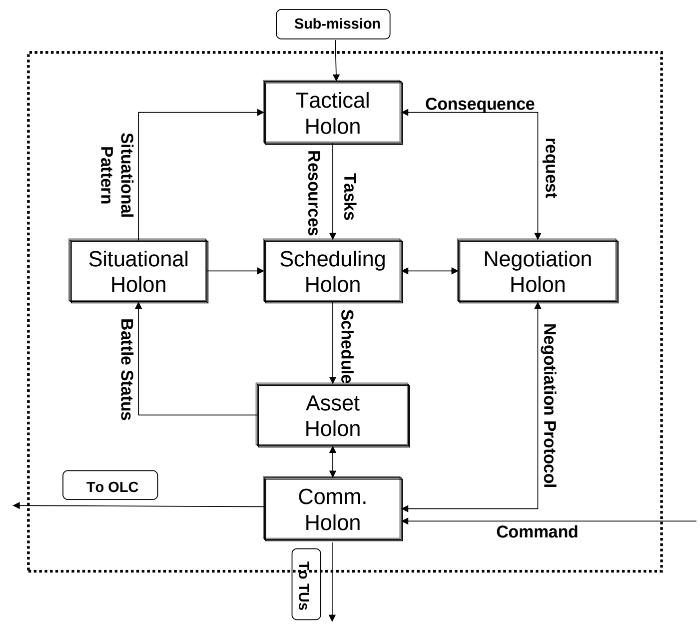

Fig. 3. Tactical Level Control Architecture

Promulgating holon (*PRH*) creates administrative messages of various types, including orders, status/situation reports and requests for information. The messages are passed to the COMH for transmission to appropriate TLC recipients.

Communication holon (*COMH*) provides communication facilities allowing the OLC holons to exchange various types of information (orders, reports and requests) with the TLC holons.

D. Tactical Level Control (TLC) architecture

The TLC architecture (illustrated in Fig. 3) is concerned with mission execution, given assigned resources by the OLC holons. The TLC architecture is comprised of six holons: Tactical holon (TAH), Situational holon (SIH), Scheduling holon (SCH), Negotiation holon (NEH), Asset holon (ASH), and Communication holon (COMH). These holons are described in the following paragraphs.

Tactical holon (*TAH*) has the highest authority within each local TLC unit. It coordinates local holons to work together to achieve local objectives related to the assigned sub-missions. It also realizes the autonomy of the TLC unit by responding to disturbances or unexpected incidents (such as an asset breaking down). The TAH in each TLC unit should have faster decision making processes than the OPH in the OLC architecture.

Situational holon (*SIH*) is a recognition-based holon that monitors the targeted area, retrieves a satisfactory response from its repertoire, and reports its solution to the TAH. The process can be viewed as pattern matching, where a perceived situation is compared with a set of stored references (which is accumulated by experience and training). The SIH should possess the capability to process information under high time pressure.

Scheduling holon (*SCH*) determines the sequence of activities for each asset under the authority of the TLC unit. The schedule should conform to the plan articulated in the OLC architecture. The schedule is also subject to revisions imposed by orders from the COOH in the OLC architecture, or necessitated by local disturbances.

Negotiation holon (*NEH*) is responsible for coordination with other TLC units, or for achieving agreement with other TLC units when conflicts arise. If the requested coordination is not available, or an agreement cannot be reached, a request for instructions from higher authority will be issued.

$Asset\ holon$ (ASH) is the interface between other holons and the physical assets. The ASH transforms various orders into operational instructions for controlling physical assets. It is also responsible for collecting status data from physical assets and reporting it to the SIH.

$Communication\ holon$ ($COMH$) in the TLC architecture has the same functionality as the COMH in the OLC architecture, except that it has lower communication capacity.

E. Characteristics of C^2 holonic reference architecture (HRA)

Koestler observes that [5] (a) biological and social systems evolve and grow to satisfy increasingly complex and changing needs by creating stable “intermediate” forms that are self-reliant and more capable than the initial systems; and that (b) it is generally difficult to distinguish between ’wholes’ and ’parts’: almost every distinguishable element is simultaneously a whole and a part. The proposed C^2 HRA is an “intermediate” form between hierarchy and heterarchy: it is a two-level hierarchy with abundant lateral liaison; it is a ‘whole’ and ‘part’ structure, but each ‘part’ is a ‘whole’ by itself, i.e., each ‘part’ has an integrated internal structure. These characteristics enable C^2 HRA to inherit the following characteristics.

· Autonomy - authority and control are distributed among holons or units (clusters of holons) enabling them to have local recognition, decision-making, action-selection capabilities. Therefore, they are able to behave reactively and pro-actively.

· Integration - authority and control flow from the highest to the lowest. The units or holons with lower authority have to follow orders from units and holons with higher authority, which ensures the “$unity\ of\ command$”.

· Co-operation - mandatory coordination orders and negotiation mechanisms allow units and holons to flexibly interact with other units or holons with a minimum degree of disorder.

· Self-Organization - whenever environmental conditions change or a major disturbance occurs, the units or holons can re-organize operations either automatically or via orders from a higher authority.

· Reconfigurability - the TLC units can be cloned or deleted according to the mission plan evolving with environmental conditions. Each unit and holon can reconfigure itself by planning and scheduling differently.

The C^2 holonic reference architecture (HRA) presented here appears promising in the sense that it aims to achieve a number of network-centric C^2 requirements.

III. Operational Model for holonic scheduling

The operational model for C^2 HRA application includes mission, mission decomposition into a task graph, deliberate planning, and holonic scheduling. The model is illustrated via the following example. For a detailed description of this example, please refer to our previous work in [17].

$Mission$: A joint group of Navy and Marine Forces is assigned to complete a military mission that includes capturing a seaport and airport to allow for the introduction of follow-on forces. There are two suitable landing beaches designated “North” and “South”, with a road leading from the North Beach to the seaport, and another road leading from the South Beach to the airport. From intelligence sources, the approximate concentration of the hostile forces is known, and counter-strikes are anticipated.

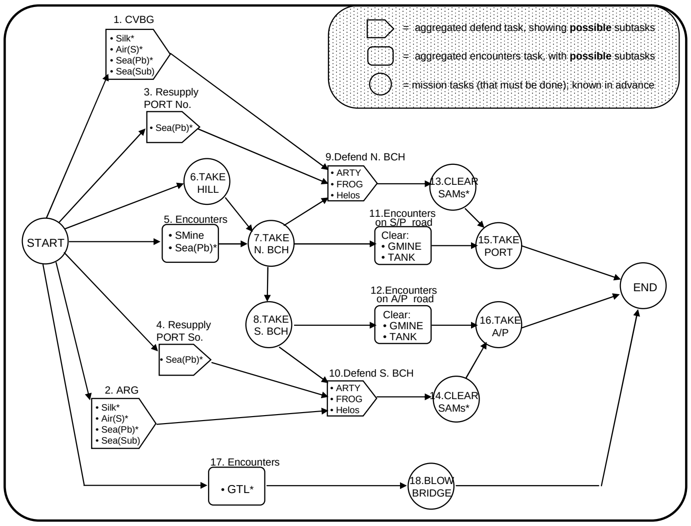

Fig. 4. Task graph for the decomposed mission from the OPH

Mission Decomposition and *Task Graph*: The OPH in the OLC architecture devises a plan for the mission that includes the processing of tasks shown in Fig. 4. The task graph describes the task procedure, constraints, and preferences. While the task procedure comprises the task execution logic, the constraints represent task-dependencies and precedence restrictions, preferences specify the task authority structure. The task decomposition knowledge is often held by the OPH, but in other cases, it might be necessary to negotiate with other holons before the criteria for decomposition are established.

Deliberate Planning: The PLH is responsible for addressing the following planning issues: (a) how many TLC units are need for the mission? (b) how to assign tasks to each TLC unit?, and (c) how to allocate available assets to each TLC unit? In the work presented in [11], a nested genetic algorithm was developed to solve these problems with the objective of minimizing both the internal and the external workloads of the system. The resulting plan is shown in Fig. 5. The mission plan provides the following information: (a) optimal number of TLC units for this mission is four; (b) task assignment to each TLC unit; (c) asset allocation to each TLC unit; and (d) start time, duration, and completion time for each task.

Holonic Scheduling: A holonic scheduling approach differs from a conventional one primarily in terms of the distribution of computation and decision making functions, and the interactive (and largely co-operative) nature of the communication between the holons [18]. The holonic scheduling process involves interactive coordination and communication between holons from both the OLC and the TLC architecture, i.e. the COOH in the OLC architecture and the SCH in each TLC unit. Accordingly, the holonic scheduling is comprised of four elements: Operational level scheduling, Tactical level scheduling, Reactive scheduling, and Negotiation.

(i) Operational level scheduling is carried out by the COOH in the OLC architecture. To be more precise, the COOH is not involved in the scheduling process directly. The primary responsibilities of the COOH are: (a) distributing the central plan to each local TLC unit; (b) monitoring the schedule of each TLC unit and ensuring that the global schedule is feasible; and (c) responding to each disturbance and, if warranted, directing the TLC units to change the schedule.

The COOH first decomposes the central plan and the task precedence graph, and then distributes them among the TLC units. The task precedence graph is decomposed based on task priority, that is, firstly, it calculates the priority value of each task (the algorithm is shown in Fig. 6); secondly, it orders the tasks according to their priority values (shown in Fig. 7); finally, it distributes the tasks and task priority information among the TLC units.

(ii) Tactical level scheduling is a distributed

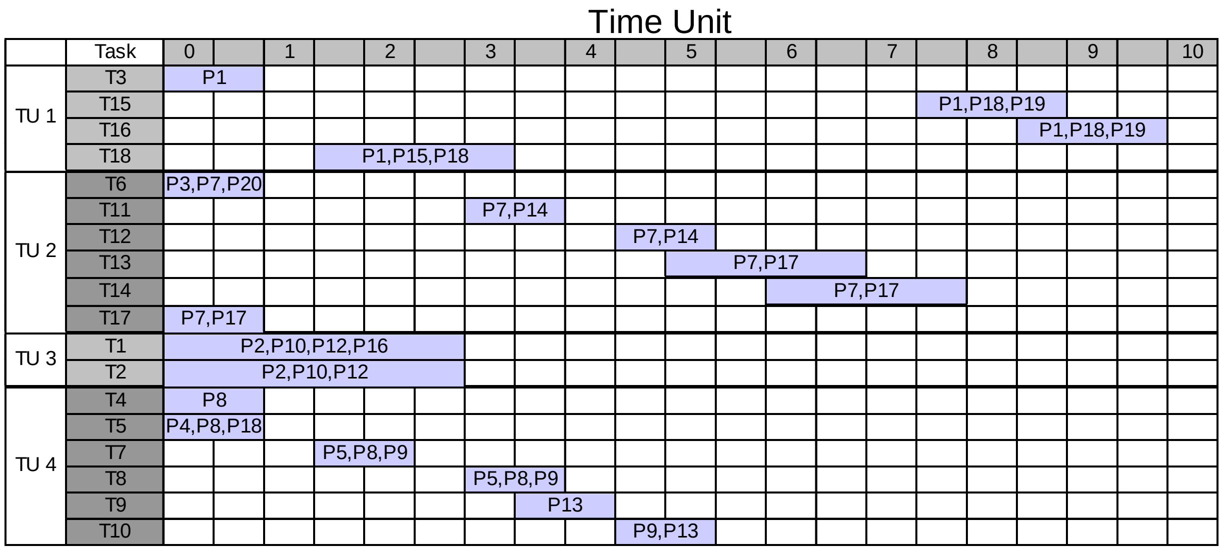

Fig. 5. The central plan created by the PLH in OLC architecture

Algorithm for calculating task priority

While $i \neq number\ of\ total\ tasks$, do
  $T_i \rightarrow Priority = 1$
  If T_i has precedent tasks $\{T_k\}$
    $T_i \rightarrow Priority = max\{T_k \rightarrow Priority\} + 1$
  Else
    $i = i + 1$
  end if
end do;

Fig. 6. Algorithm for calculating task priority

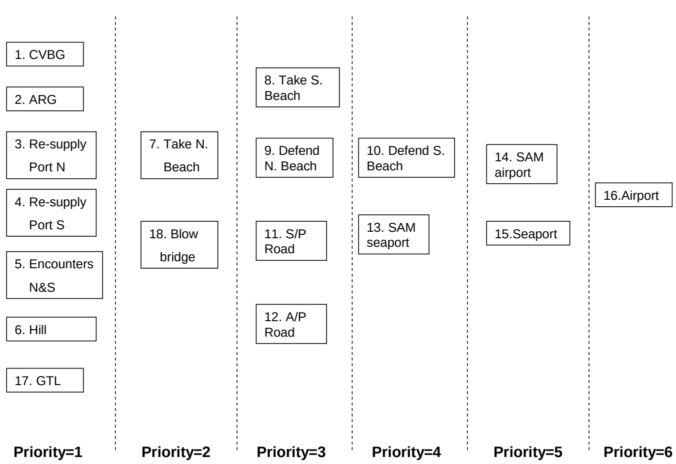

Fig. 7. The task priorities calculated by the COOH

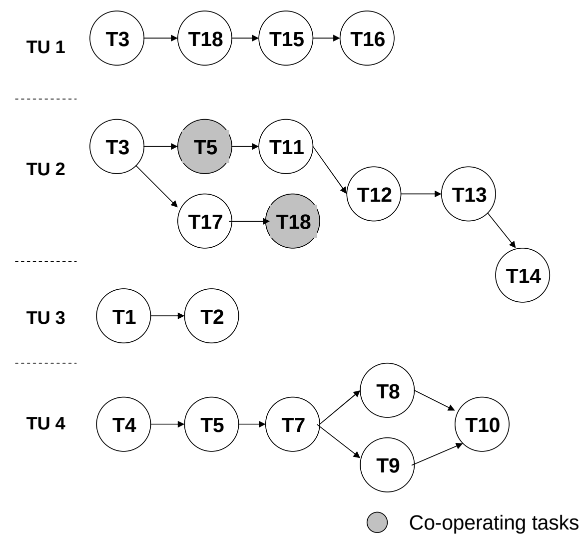

Fig. 8. The distributed task execution sequence for each TLC unit

process, in which each TLC unit makes the sequencing decisions based on local information, local objectives (e.g. minimize makespan of the sub-mission, or evenly distribute workload among assets), and constraints. In this paper, we assume that each TLC unit seeks to find a schedule that minimizes the makespan of the sub-mission. The task execution sequence for each TLC unit is illustrated in Fig. 8; the distributed schedule is shown in Fig.

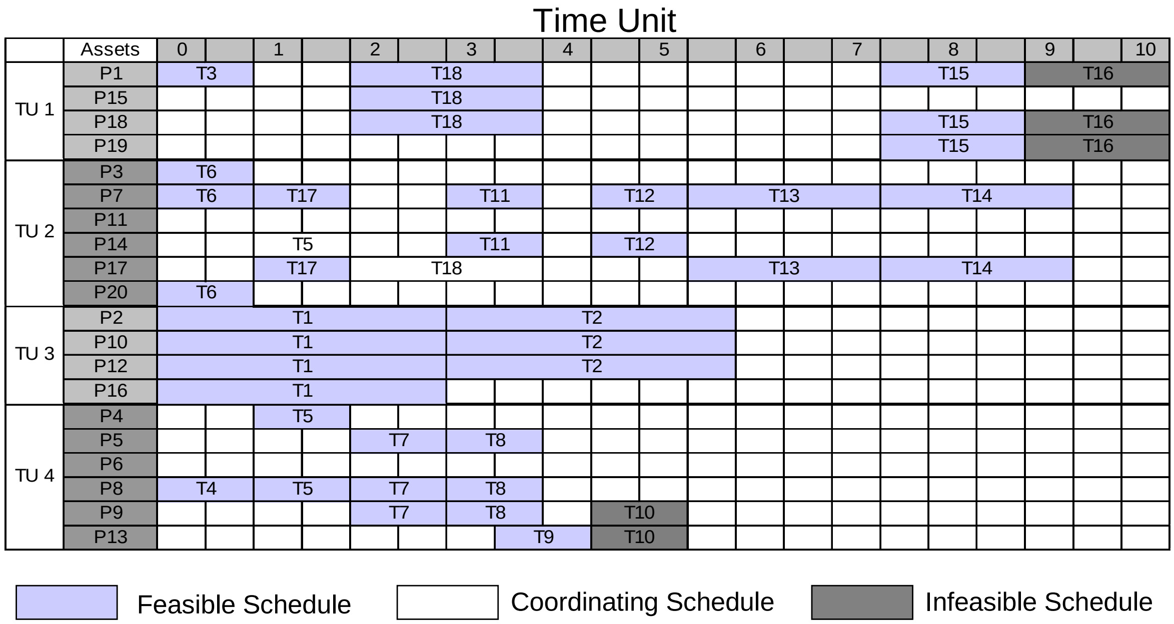

Fig. 9. The infeasible distributed scheduling created by TLC units

9.

(iii) Reactive scheduling combines the autonomy of local scheduling with the co-operative tendencies needed to maintain the feasibility and optimality of a centralized schedule. Each TLC unit schedules its operations based on its own information, objectives, and constraints. Consequently, schedule may potentially conflict with the schedule made by other TLC units. For example, in Fig. 9, we may notice that the schedule for task 10 and task 16 is infeasible because task 10 is planned to start after task 2 completes, and task 16 should start after task 14 is accomplished. The infeasibility of the global schedule can be detected by the COOH in the OLC architecture. It then starts the re-scheduling procedure: firstly, it generates a feasible global schedule; secondly, it advises the related TLC units to adjust their local schedules by changing the constraints, for example, the constraint for the starting time of task 10 would be changed from 4.5 to 6; finally, each related TLC holon regenerates the schedule based on new constraints. Fig. 10 shows a feasible schedule after the reactive scheduling process.

(iv) Negotiation mechanism provides communication among TLC units when coordinating on the execution of a task. Information and commands are exchanged by the use of a negotiation protocol, in which the schedule of certain assets executing co-operative tasks can be determined by negotiation. In the example, the TLC unit TU2 will negotiate with TU1 and TU4 before it sends its assets P14 and P17 to coordinate on tasks T5 and T18. Under certain circumstances, TU2 may refuse to provide resources to support other TLC units. In this case, the OPH will intercede and resolve the conflict.

IV. Conclusions

In this paper, a holonic reference architecture (HRA) to represent the Information Age C^2 processes was presented. This HRA is comprised of two levels: Operational level control (OLC) architecture and Tactical level control (TLC) architecture. The OLC architecture provides a facility for mission decomposition, deliberate planning, command

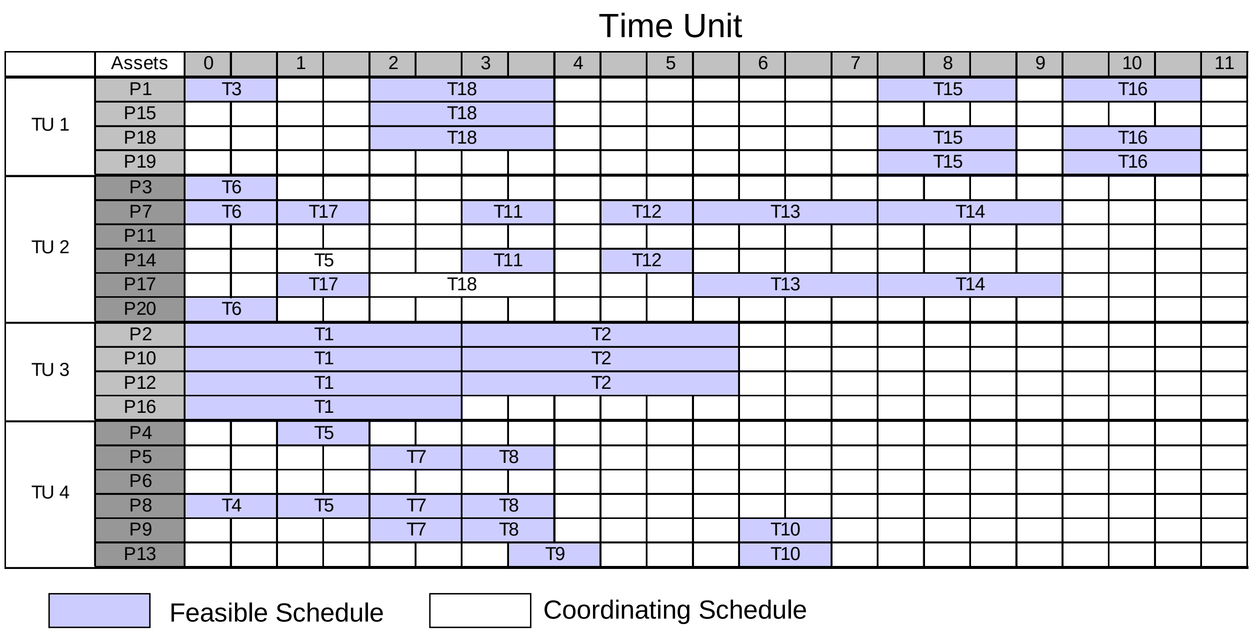

Fig. 10. The feasible distributed scheduling created through holonic reactive scheduling scheme

promulgation, and inter-holon coordination. At this level, the C^2 process is focused on the overall mission objectives. The tactical level control (TLC) involves local task scheduling, battlefield pattern recognition, and negotiation. It also provides an interface to physical resources. The TLC architecture can have more than one instance (TLC unit), which together fulfill the objectives of the mission. After illustrating the holonic scheduling process via a realistic mission model, it is concluded that the C^2 holonic reference architecture is an integration of centrality and autonomy; rigidity and flexibility; doctrine and adaptability, which enables a C^2 organization to possess the capability of handling mission changes, while ensuring the "$unity\ of\ command$" during mission execution.

Future work needs to focus on: (a) establishing analytical models for hierarchy, heterarchy, and holarchy as well as of the mission environment; (b) exploring performance measures for different structures based on the organizational structure and mission models; and (c) application to existing and future systems (e.g., Expeditionary Strike Group (ESG) and FORCEnet).

References

[1] V. Clark *Sea Power 21 Series: Part I Projecting Decisive Joint Capabilities*, Proceedings, October 2002.

[2] D. S. Albers, R. E. Hayes *Information Age Transformation Series: Power of the Edge,* CCRP Publication.

[3] L. Gou, P. B. Luh, and Y. Kyoya *Holonic manufacturing scheduling: architecture, cooperation, mechanism, and implementation,* Computer in Industry 37, pp 213-231, 1998.

[4] S. S. Heragu, R. J. Graves, B. Kim and A. S. Onge *Intelligent agent based framework for manufacturing systems control,* IEEE Tran. on SMC, Part A, Vol. 32, No. 5, 2002.

[5] A. Koestler, *The Ghost in the Machine,* Arkana Books, Carnegie-Mellon University.

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

[7] G. M. Levchuk, Y. Levchuk, J. Luo, K. R. Pattipati, and D. L. Kleiman *Normative Design of Organizations - part I: Mission Planning,* IEEE Transactions on Systems, Man, and Cybernetics, Vol. 32, No. 3, May 2002, pp. 346-359.

[8] G. M. Levchuk, Y. Levchuk, J. Luo, K. R. Pattipati, and D. L. Kleiman *Normative Design of Organizations - part II: Organizational Structure,* IEEE Transactions on Systems, Man, and Cybernetics, Vol. 32, No. 3, May 2002, pp. 360-375.

[9] G. M. Levchuk, F. Yu, K. R. Pattipati, and Y. Levchuk, *From Hierarchies to Heterarchies: Application of Network Optimization to Design of Organizational Structures.,* Proceedings of the 2003 CCRTS, Washington DC, June, 2003.

[10] G. M. Levchuk, F. Yu, Y. Levchuk, and K. R. Pattipati *Networks of Decision-making and communicating Agents: A New A NovMethodology for Design and Evaluation of Organizational Strategies and Heterarchical Structure,* Proceedings of the 2004 CCRTS, Dan Diego, CA, June, 2004.

[11] F. Yu, G. M. Levchuk, M. Candra, S. Ruan, and K. R. Pattipati *Congruent Organizational Design Methodology Using Group Technology and a nested genetic Algorithm,* Proceedings of the 2004 CCRTS, Dan Diego, CA, June, 2004.

[12] U. Aickelin and K. Dowsland, *An Indirect Genetic Algorithm for a Nurse Scheduling Problem* Computers and Operational Research, 2002

[13] D. Montana, S. Moor, and G. Bidwell, *Genetic Algorithms for complex, Real-time Scheduling*, IEEE conference on SMC 1998

[14] H. V. Brussel, J. Wyns, P. Valckenaers, and L. Bongaerts*Reference architecture for holonic manufacturing systems: PROSA,* Computers in Industry, vol. 37, pp. 255-274, 1998

[15] D. Stark, *Heterarchy: Distributing Intelligence and Organizing Diversity,* The Biology of Business: Decoding the Natural Laws of Enterprise, pp. 153-179 San Francisco: Jossey-Bass.

[16] A. D. Baker, *A Survey of Factory Control Algorithms which Can be Implemented in a Multi-Agent Heterarchy,* Journal of Manufacturing Systems, vol. 17, pp.279, 1998

[17] G. M. Levchuk, Y. Levchuk, J. Luo, F. Tu and K. Pattipati, *A Library of Optimization Algorithms for Organizational Design,* 2000 Command and Control Symposium, Monterey, CA, June 26-28, 2000

[18] D. C. McFarlane and S. Bussmann, *Developments in Holonic Production Planning and Control,* Int. Journal of Production Planning and Control, vol. 11, No. 6, pp. 522-536, 2000.