Investigating Alternative Concepts of Operations for a Maritime Security System of Systems

Brian Mekdeci, Adam M. Ross, Donna H. Rhodes, and Daniel E. Hastings
Massachusetts Institute of Technology
77 Massachusetts Avenue, E38-576
Cambridge, MA 02139-4307
617-324-0473

Abstract. For complex systems of systems, such as those required to perform maritime security, system architects have numerous choices they may select from, both in the components and in the way the system operates. Component choices, such as the length of a wing or the number of ground control stations, are often considered in tradespace studies, but this paper highlights the operational choices that are often overlooked. Using an unmanned vehicle system of systems as an example, the importance of considering operational choices as well as the highly interdependent nature of operational and component choices is demonstrated, further strengthening the case for careful consideration of operational variables early in the concept phase of the design process. Finally, a high-level methodology for generating and evaluating operational choices is introduced.

Introduction

Tradespace evaluations allow decision makers the ability to compare and contrast thousands of system designs across numerous contexts to determine the designs that continue to provide value, during nominal and abnormal conditions (Richards 2009). In the past, tradespace evaluations primarily focused on variables relating to the components themselves, such as gain on the antennas or the wingspan, since these choices were typically more difficult to change after implementation, and thus were important to get "right" during conceptual design. In hindsight, the fact that physical design parameters are often fixed early in the lifecycle, is the reason operations choices should be considered. The way components or constituent systems behave and interact with each other, that is, the Concept of Operations (CONOPs)[1], should be examined – they are often the most feasible, or least costly choices stakeholders have that allow the system to adapt to changing contexts. Unfortunately, operations choices are often more difficult to identify than physical design choices, and thus can be easily overlooked.

There are three goals of this paper; (1) demonstrate the importance of CONOPs choices by showing that they can have a significant impact on system effectiveness, (2) show that component and CONOPs changes are highly interdependent, so voluntary and involuntary changes need to be tested early in order for decision makers to understand their impact, and (3) highlight a methodology by which component and CONOPs choices can be identified and evaluated by decision makers. To accomplish these goals, a system of systems (SoS)

---

[1] A *CONOPs* refers to a document that describes how a system works, often from a user's perspective (Software Engineering Standards Committee of the IEEE Computer Society 1998). In this paper, a Concept of Operations (CONOPs) refers to the *concept* (itself) of how the system works, i.e. how the components operate and interact with each other to provide value to the stakeholders.

consisting primarily of unmanned vehicles (UVs) performing maritime security will serve as a case study.

Unmanned Vehicle System of Systems

The primary task of an unmanned vehicle is surveillance/reconnaissance, although a few vehicles have payloads that allow other functionality such as engaging hostile targets or performing a rescue operation. A typical unmanned vehicle system of systems might be used to provide maritime security. In this type of system, the main tasks would be search & rescue, as well as identifying ships that pass through some particular Area of Interest (AOI), and engaging those that fail to comply with the various laws and regulations that the SoS is in charge of enforcing. Maritime security systems typically consist of multiple patrol boats and planes (manned or unmanned), as well as supporting components such as radar towers, ground control stations, satellites, and command centers. Ships and/or victims usually appear in the AOI randomly, and common metrics of performance are the likelihood and time it takes to detect, locate, identify, engage and/or aid targets. At a minimum, the SoS requires three capabilities to perform these tasks (1) the ability to *detect* targets that are in the AOI, (2) the ability to *identify* targets, and (3), the ability to *affect* the targets in some way, either by aiding them (if they are victims) or engaging them (if they are non-compliant ships in the AOI). Each one of these tasks typically requires specific payloads; some weapon capability is likely to be required to effectively engage hostile ships, whereas an electro-optical/infrared (EO-IR) camera of some sort might be used to identify unknown ships. Since each payload has a limited range of effectiveness and the AOIs are usually very large, multiple unmanned vehicles are used as platforms to carry the payloads around. The need for multiple payloads results in a network of UVs that coordinate, cooperate and combine their efforts for some overall set of goals and mission objectives.

The Impact of CONOPs Choices on System Effectiveness

An engineered system is effective if the value it provides is satisfactory to stakeholders. Value is typically assessed by how well the system performs, how much it costs, and other attributes.

Performance. Systems are designed to perform specific missions under certain conditions, and how well that system performs the required tasks provides a measure of performance to the stakeholders. In maritime security, for example, the probability of detecting ships and how quickly targets are identified are both measures of performance. Performance is typically the most common way to assess the effectiveness of a particular system design.

Cost. Choices often incur costs, both in initial acquisition, building and other non-recurring engineering (NRE) costs, as well as operation, maintenance, disposal and other miscellaneous expenses that make up the full life-cycle cost (LCC). The objective of most engineered systems is to maximize performance and minimize cost. During operations, cost is sometimes easier to assess than performance, although cost is sometimes very difficult to accurately assess during the design phase.

Other Attributes. Aside from performance, other system attributes provide utility to stakeholders as well. For instance, stakeholders may be interested in the size and appearance of a cellphone, even if these aspects have nothing to do with how well the phone makes calls or sends text messages. Stakeholders may be interested in other system attributes as well, such as the "-ilities." Examples include survivability (Richards et al. 2007), testability (Voas and Miller 1995), modularity (Huang and Kusiak 1998), and scalability (Jogalekar and Woodside 2000). When possible, these other attributes can be translated into utility so that they may be

evaluated in tradespace studies, but this conversion to utility can be very difficult to operationalize, particularly if these attributes are poorly defined, are functions of utility themselves, or are not properly understood by the stakeholders (McManus et al. 2007).

Case Study: Mechanistic vs. Organic CONOPs

The importance of considering alternative CONOPs was demonstrated in an empirical study and simulation that consisted of nine unmanned vehicles (UVs) that perform a search and rescue operation in a hostile environment. There are three tasks, three types of vehicles (each type specializing in each of the tasks), and three vehicles of each type. The tasks were (1) identifying targets as either friendly or hostile, (2), aiding friendly targets, and (3) engaging hostile targets. There was an order to the tasks, as an unidentified target has to be identified as being either friendly or hostile before it is aided or engaged. Three human operators control the nine vehicles. Performance in this type of scenario is typically determined by such metrics as the time it takes to "complete" an objective, meaning the time it takes to rescue a friendly or engage a hostile, from the moment it appears as an unidentified target in the AOI. As with any system, architects have numerous choices they can make that will impact its effectiveness. The obvious ones involve component choices. The system architects can vary the types and specifications of the vehicles and payloads being used. One would expect that varying the top speeds of the vehicles, the accuracy of the weapons and the optical quality of the cameras would have a significant effect on performance. Of course, simply adding or removing more operators or vehicles should also impact cost and performance as well, although like component upgrades, performance and cost are often not linearly related. A classic tradespace study might look at determining the ideal number of UVs and operators along with component upgrades that maximize performance for a particular budget. However, in order to assess the performance and cost of a system, a particular CONOPs has to be assumed.

A simple CONOPs would be to have each operator take control of all of the vehicles of a particular type, and specialize in one of the tasks in what is referred to as a *mechanistic* team (Drach-Zahavy and Freund 2007). In this CONOPs, operator A controls the vehicles that perform identification, operator B controls the vehicles that aid friendlies and operator C controls the vehicles in charge of engaging hostile targets. If, instead of having each operator control all three vehicles of one type, each operator controlled one vehicle of each type, an alternative CONOPs emerges. This type of CONOPs, in which roles and responsibilities are shared among constituent members, is referred to as an *organic* team (Reigle 2001). Figure 1 shows both the mechanistic and organic team structures for a UV SoS that has identical components. In both teams, there are exactly three operators and three UVs of each type. The only difference is that the mechanistic team has specialized roles, whereas the organic team shares roles. In both an empirical study and in a stochastic model, the mechanistic SoS performed better than the organic SoS under nominal conditions (i.e. the rate of targets appearing in the AOI was a constant, as seen in Figure 2) (Mekdeci and Cummings 2009). The difference in performance between the mechanistic and organic teams could be explained by the fact that the organic team had to share roles and responsibilities and had to coordinate their efforts appropriately. This additional communication and coordination took a finite length of time, which impeded the system of systems' ability to respond quickly to emerging targets, thereby decreasing overall performance. Additionally, if the coordination was not done properly, task errors occurred, such as an operator trying to identify a target that was previously identified by another operator, or all the operators neglecting a target due to social loafing (Latane, Williams, and Harkins 1979) or simple miscommunication.
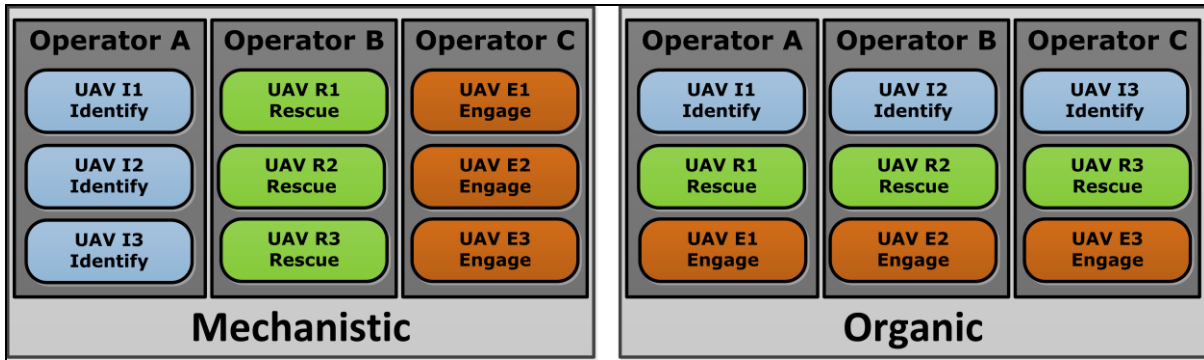
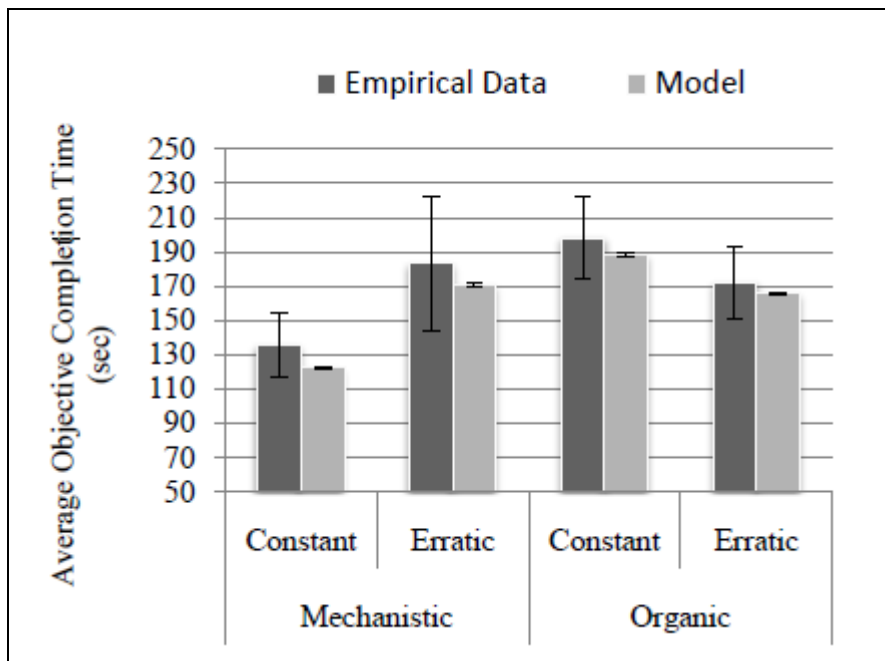Figure 1: Mechanistic and Organic Team Structures for a UV SoS

Figure 2: Performance Difference between Mechanistic and Organic Systems of Systems (Mekdeci and Cummings, 2009)

At first, it appears that assuming a mechanistic CONOPs is justified. As the proverb goes, "too many cooks spoil the soup", and thus assigning each function to just one component follows the principles of good systems design (Bahill and Botta 2008). However, as the experiment demonstrated, the performance benefits of a mechanistic team can be erased under off-nominal conditions. In fact, when arrival of targets was erratic (i.e. the targets appeared in the AOI in clusters), there was no significant difference in performance between the organic and mechanistic CONOPs (Figure 2). This is because the mechanistic CONOPs suffers from a bottleneck when the single operator performing the identification task is overwhelmed and the other two operators do not have the capability to assist. Even though there are three vehicles capable of performing identification, the task itself requires a non-trivial amount of operator intervention. The lone operator cannot perform the identification task with all three vehicles simultaneously. This bottleneck is not as severe in the organic CONOPs, since each operator can perform the identification task and thus each vehicle can be used simultaneously.

Even though performance under off-nominal conditions is about the same between mechanistic and organic teams, other factors, such as survivability, may make the organic SoS more attractive to stakeholders. There are a number of disturbances that could cause one of the

operators in the SoS to be unavailable. Operators get sick, hardware in the ground control station may fail, communications may be jammed, etc. Since each function is only assigned to one operator in the mechanistic team, then that becomes a single point of failure. If Operator B in the mechanistic team shown in Figure 1 is not available, for example, then the SoS will be unable to aid any victims for that length of time. However, if Operator B is unavailable in the organic SoS, then performance would decrease overall, but all of the required tasks would still be completed to some degree. Under this disturbance, we can say that the organic SoS is more survivable than the mechanistic SoS and depending on stakeholder preferences, a better design overall. The most interesting aspect of this, however, is that the two designs are identical in form, but different only in their modes of operation, showing just how important it is to explicitly consider CONOPs choices for determining system effectiveness.

Relationship between CONOPs and Component Choices

CONOPs and components choices can be independent, but more often than not, they are interdependent to some degree. Any non-passive capability added by a component, has to be considered in the CONOPs in order for it to be effective. Simply adding an automatic landing feature to a SoS does not impact effectiveness if the CONOPs isn't changed to take advantage of it. However, if instead of adding this feature, the automatic landing replaces an existing manual landing capability, then the CONOPs may be forced to change to accommodate this modification, otherwise the SoS may actually fail as a result of this particular component change. Similarly, a CONOPs change may require a component change as well. To be effective, an organic CONOPs will likely require a communication component through which constituent systems can coordinate their efforts. A mechanistic CONOPs may not require such a communication component to be effective, or at least not as sophisticated as one required in an organic CONOPs, since the roles and responsibilities are more clear.

It is important for system architects to realize that more often than not, CONOPs changes are easier to implement after a system has been implemented if the form of the system supports it. As an example, it might be trivial for a mechanistic SoS with a communication component to switch to an organic CONOPs to effectively handle an unusually high level of target arrivals, or missing operator. However, without such a communication component in place, the only way a mechanistic team might handle the additional demands of unusually high volume of targets would be to add additional operators and/or UVs, which is something that is typically much harder to do. This asymmetry of transitioning between alternative CONOPs means that system architects should explore not only additional CONOPs choices, but also how systems can change from one design into another through component and/or CONOPs choices, in order to identify designs that are more likely to be value robust during long lifecycles across varying contexts.

Methodology for Identifying and Evaluating Choices in the System Architecture

Having established the importance of including CONOPs choices, there is a need for a methodology to assist system architects in recognizing and developing CONOPs alternatives. There are existing frameworks, such the Command, Control, Communications, Computers, Intelligence, Surveillance, and Reconnaissance (C4ISR) and its successor the Department of Defense Architecture Framework (DoDAF), that describe in detail various methods for expressing and assessing system architectures, including operational considerations, using different "views". These frameworks have been useful (Bindi et al. 2008) (Dickerson et al. 2004), in assisting in describing and evaluating complex systems and systems of systems (of

family of systems).   Using the various views and specific use cases, these frameworks can help determine how well the architecture performs and what the architecture actually allows the system to do.   However, a high-level step-by-step process can be useful can be used in conjunction with these frameworks, to discover other CONOPs variables that may have been overlooked otherwise. Such a process, shown in Figure 3, is introduced here.

First, the relevant contextual considerations, such as the driving stakeholder objectives and key environmental factors, are stated. Afterwards, the necessary capabilities to meet the objectives are identified. Then, components that provide those capabilities are identified. Next, relevant operational considerations are established that may make it easier or more difficult for the components already identified to achieve the objectives identified by the stakeholders. Often, these operational considerations are driven by operator experience, knowledge, and skills, which may guide or constrain the CONOPs or component choices. From the set of component and operational considerations, a system architecture consisting of various components and CONOPs choices is developed. The architecture is then sampled into a manageable set of systems designs that are then evaluated through modeling and simulation, and finally explored in a tradespace evaluation. From the tradespace evaluation, the most viable system designs are identified for further consideration. Using an unmanned vehicle SoS performing maritime security as an example, the various steps in this methodology is expanded below:
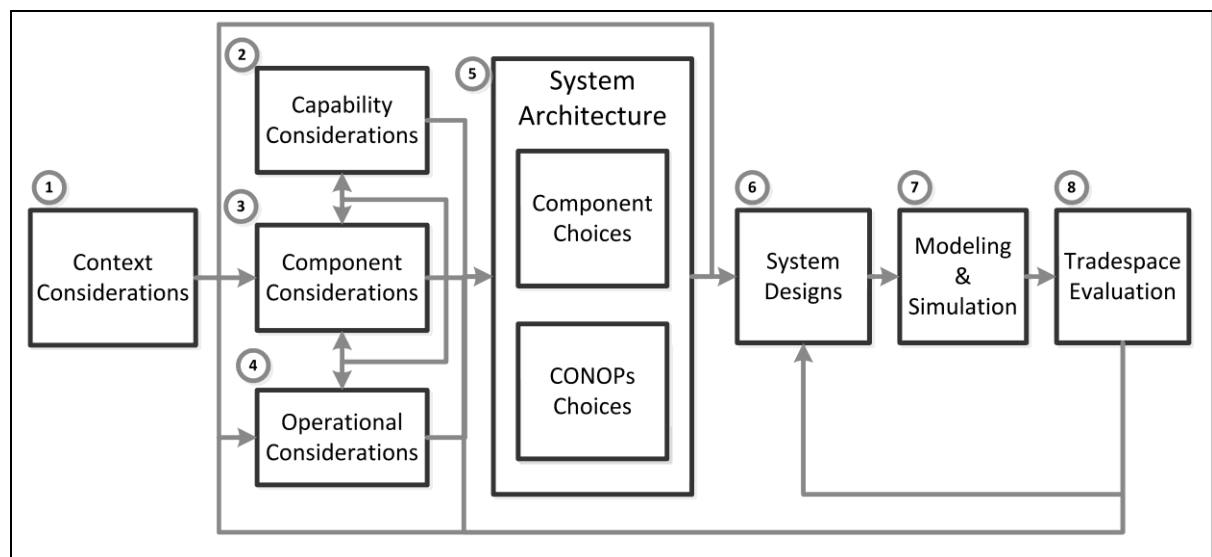

Figure 3: Methodology for Identifying and Evaluating Component and CONOPs Choices

1. Contextual Considerations

All systems operate under some context, which includes exogenous conditions, stakeholder needs (i.e. objectives) and various disturbances. Some contexts are fairly static, while others are dynamic. The first step in the methodology is to recognize the relevant exogenous data and constraints that make up the context under which the system is expected to function. Some important contextual considerations for a maritime security SoS include:

Location Dependence. Exogenous events occur at specific geographical locations, which may move over time.

Large Scale. The AOI and/or the number of exogenous events that the SoS is responsible for are so large, that numerous constituent systems with the same, or similar, capabilities have to work together to provide the necessary capacity.

Communication Disturbances. Communications via radio waves can be degraded , resulting in noisy channels and miscommunication. This can occur unintentionally (e.g. weather or amateur pirate radio stations) or intentionally (e.g. a jamming attack).

2. Capability Considerations

The nature of the context under which it operates, affects its capability to provide value. In a maritime security UV SoS, some important capability considerations include:

Limited Sphere of Influence. Constituent systems have a Sphere of Influence (SOI), which is the geographic range within which the system has the ability to interact within its environment. The SOI is constrained by the effective range of the actuators and sensors of a particular system limit that system's ability to respond to those exogenous events at particular locations. In addition, the constituent systems may also be constrained by other factors. For example, a UAV may not be allowed to engage targets that have crossed into territory belonging to a foreign nation, even if it has the physical capability to do so.

Redundant Capabilities. The large-scale nature of the context far exceeds the SOI and the capacity of individual systems, and thus numerous constituent systems with the same or similar capabilities have to work together to provide the necessary SoS-level capacity. Homogenous constituent systems are those that are exactly the same, such as a squadron of MQ-1 Predator UAVs in a maritime security SoS. Heterogeneity exists when different constituent systems in a SoS have capabilities that overlap. An example of heterogeneity would be some UVs using Electro-Optical/Infrared (EO/IR) cameras to detect identities, while others use Synthetic Aperture Radar (SAR).

3. Component Considerations

Components have to be selected that provide the required capabilities of the system. Constraints on the capabilities need to be taken into consideration when selecting components. Some important considerations for the components of a maritime security UV SoS performing maritime security include:

Geographical Separation. Components and constituent systems are often not co-located, and are spread out over relatively large distances (Mekdeci et al. 2011). This almost always creates the problem of local information, which must be shared with the rest of the SoS via some type of communications.

Mobile Components. Given the fact that the components have a limited sphere of influence, to address the large-scale, location dependence nature of the context will require that some of the components move around the AOI as necessary to perform the tasks. Typically, this means that the capabilities have to be mounted on a moving platform, like UVs. Having vehicles in the SoS bring about their own issues that need to be considered, such as the location of supporting components and equipment, for example, landing pads, docks and ground control stations.

4. Operational Considerations

By looking at the characteristics of the context as well as the components and capabilities of the maritime security UV SoS, it seems that a viable system architecture will need to address at least two operational considerations; (1) a way to deal with the large geographical AOI relative to the small SOI of constituent systems along with the travelling required to handle exogenous events in a timely manner, (2) a way to allocate tasks in an efficient manner. If the task a particular constituent system needs to perform is not located within its SOI, then it must travel to an appropriate location to perform the task. Even if, under normal circumstances, the UV SoS has sufficient capability to be able to perform all tasks without any travel of the constituent systems, a disturbance may cause some of the constituent systems to be overloaded and/or fail. In that case, some of the constituent systems may be required to travel, either because the disturbance is localized, or because the disturbance has caused a geographic imbalance in capability. Typically, travelling should be kept to a minimum because of the following problems:

Time. The time it takes a vehicle to travel from where it is, to where it is needed, is included in the overall time it takes for the SoS to respond to exogenous events such as a victim suddenly appearing in the AOI. Thus, reducing travelling time is one of the most significant ways to improve the time-based metrics of performance, such as mean time to identify or mean time to intercept.

Task Sharing. CONOPs need to specify how tasks are shared any time there are multiple constituent systems performing tasks that are interdependent, or could overlap. For all other systems, task sharing has to be explicitly addressed; otherwise there is the risk of task errors, which may result in reduced performance and possibly reduced survivability

5. Identifying Choices in the System Architecture

A system architecture (SA) consists of components (or constituent systems) as well as their capabilities, and a CONOPs. When a particular system architecture is instantiated, that instance is a system, for example, the Nimitz class aircraft carrier is a system architecture whereas the USS Ronald Reagan is a system instance of that particular architecture (Mekdeci et al. 2011). Once the relevant component and operational considerations have been declared, choices in components and CONOPs for the system architecture can be identified. Typically, design choices for a UV SoS, do not allow much physical customization of the constituent systems. For instance, the size and shape of a wing on a particular UAV are not typical design choices. Rather, the system architects assemble their particular SoS by choosing the type and number of off-the-shelf components to be included in the system. For a maritime security UV SoS, available components choices include:

Component Payloads Each task is accomplished by a payload and each payload type has its own advantages and disadvantages, such as cost, accuracy and susceptibility to bad weather. Changing a camera from EO/IR to SAR may be advantageous in some situations, and not in others. Similarly, the platforms on which the payloads are mounted, in this case UVs, can be choices as well. Some UVs are faster or have longer endurance than others, for example.

Types of Vehicles. Payloads have to be carried by a platform, which in a maritime security SoS may include manned and/or unmanned vehicles. Different vehicles can carry different payloads and their specifications (endurance, top speed, etc.) can have a large impact on the overall value the SoS delivers.

Number of Vehicles. The number of each type of vehicle is a choice, subject to the minimum necessary to provide the required capabilities, as well as the maximum allowable as constrained by the context, component selection and operational considerations.

Although components choices in a UV SoS may be limited, there may be many more choices on how the system can be operated, and these choices have a large impact on cost and performance. Additionally, once a system is implemented, operational changes are often the only way a system can actively respond to disturbances and disruptions, in order to improve survivability and robustness. Finally, the range of operational choices available is constrained by the components. For these reasons, it is particularly important that system architects explicitly include operational choices when they develop the system architecture. Some relevant operational choices to a maritime security UV SoS are:

Team Structure. First, the system architects select the payloads they are interested in using, and then they select an appropriate UV platform to support that particular payload. Certain UVs may support multiple payloads simultaneously, but they are typically larger and more expensive to acquire and operate, as one would expect. Additionally, while both payloads may be on the UV simultaneously, they may not be able to be operated simultaneously. Rather, they may share certain resources and operators may have to switch from one to another. We can say that systems cooperate if they have the same capabilities and share the same tasks, whereas if they have dissimilar capabilities, then they must coordinate their actions to ensure that tasks are performed in the correct order and in a timely manner (Jang et al. 2003). Another issue is authority, i.e. who decides who does what task and when? The answer depends on which constituent systems have the authority to make those decisions. Particularly for unmanned SoS, mixed manned and unmanned SoS, and/or constituent systems under the control of different stakeholders, there is often some level of autonomy that allows for independent decision making and action of the constituent systems. A central authority restricts the independence of the constituent system by requiring them to obey instructions.

Roles and authority are technically independent, but they very often are interdependent. In particular, they are often combined into some sort of *mechanistic* or *organic* type of organization. Mechanistic SoS typically have dedicated roles and central authorities who coordinate the execution of tasks, whereas organic systems of systems tend to not have central authorities or dedicated roles, but rather share the workload amongst independent constituents systems (Courtright, Fairhurst, and Rogers 1989). However, it is possible to have an organic team with a central authority, in the sense that all the vehicles have the same role, but a central authority controls the task handling.

At this stage, system architects can review the existing contextual, operational and component considerations and attempt to determine CONOPs choices that have not been thought of (Figure 4). Recognizing that there is a large problem of traversing the AOI, the choice of segregation becomes a variable to consider.
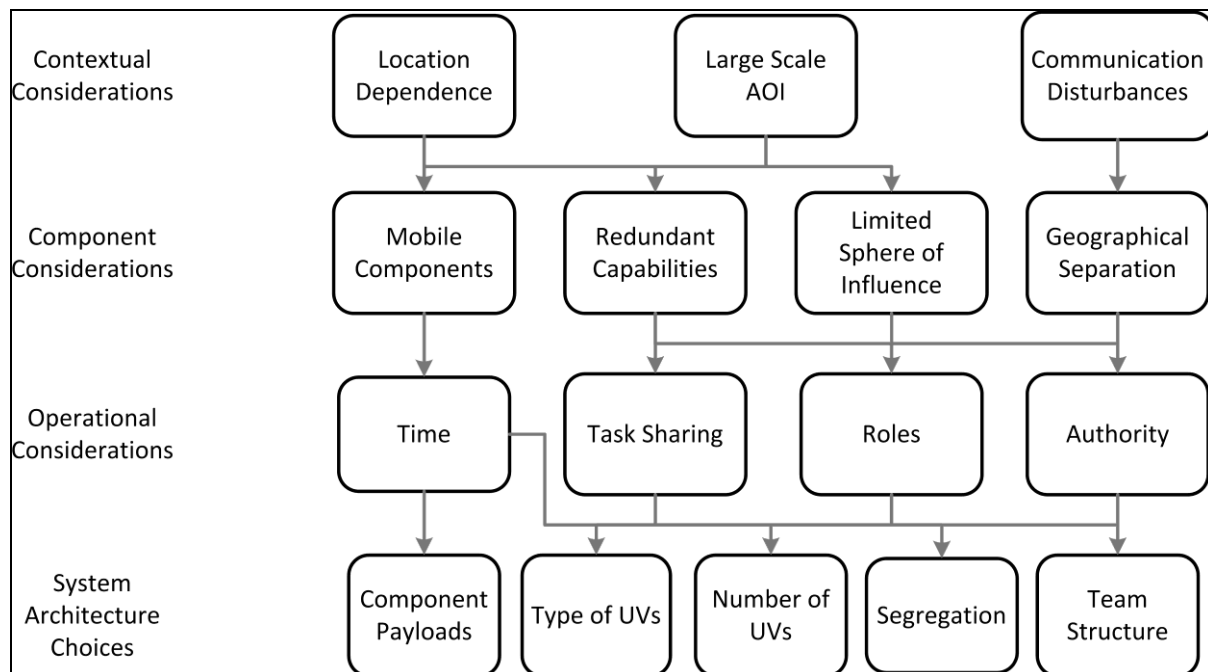
Figure 4: Methodology for Identifying and Evaluating Component and CONOPs Choices

Segregation. The large-scale nature and location dependence of the maritime security UV SoS, as well as the limited sphere of influence of the constituent components, ensure that colocation of all the components within these SoS is not practical. In most of these systems, time is of the essence, and the response time to a particular event will be directly proportional to the distance between the event location and the location of the constituent subsystems with the particular capabilities necessary to respond to that event. Additionally, the signal strength of RF communications is inversely proportional to the distance between sender and receiver. Therefore, to cope with the large distance, a typical design choice is to segregate the constituent systems into zones in order to minimize distance between systems and potential targets. An effective SoS design that segregates its components, will ensure that each zone has enough resources and capacity to be effective without requiring assistance from other components outside its zone. A drawback to zoning is that mobile capabilities may not be shared across zones. This means that in a mechanistic team, for instance, there must be at least three vehicles in each zone; one vehicle capable of detecting, one vehicle capable of identifying, and one vehicle capable of intercepting. This may actually be a waste of resources, if the total number of vehicles across all zones is actually more than the total number of vehicles necessary for the entire AOI if zoning was not used. This quantification requires an understanding of the size of the AOI and the capacity and resources required so that the SoS is divided into proper zones.

6. *Sampling the Tradespace for System Designs*

The goal of a tradespace evaluation is to attempt to enumerate and evaluate as many different design choices and contexts as possible, so that decision makers can have as complete of a picture as possible of the range of performance and costs associated with particular system choices across uncertain futures. Often, it is infeasible to enumerate and evaluate every possible combination of choices in components and CONOPs, so a subset of the "full" tradespace should be selected to evaluate.

7. Modeling & Simulation

To determine the impact of various CONOPs choices on performance and survivability and to generate a tradespace against cost and stakeholder utility, modeling and simulation is required. The outputs of the models and simulations quantifies the impacts of the various component and CONOPs choices, although some of the more qualitative performance and other attributes of interest (such as modularity) need to be assessed by other methods. To explore the full context of dynamic environments, both stochastic inputs and specific scenarios (or use cases) should be used where various system designs are tested against nominal and exception circumstances. A discrete-event simulation is currently being developed that will allow researchers the ability to test and evaluate different maritime security SoS designs (with both component and operational variables) against dynamic contexts. Future papers will explore the results of these simulations.

8. Tradespace Evaluation and Feedback

The final step in the methodology is where the value of the various system designs, as determined by modeling and simulation along with sets of stakeholder preferences, is plotted against their cost in a tradespace evaluation. Here, various designs can be compared against each other in terms of the value they provide to various stakeholders.

It is worth noting that the methodology described above is not strictly linear. Rather, it is an iterative process where knowledge gained in one step can be iteratively applied to previous steps. If the actual system designs tested were just a sample of the possible designs allowable by the system's architecture, then the results of the tradespace evaluation can suggest other viable designs to test. Similarly, information gained from generating the system architecture choices may cause system designers to realize other component and/or operational considerations. As an example, during the development of the system architecture it might be discovered that an organic team might be a good CONOPs choice to deal with some of the issues identified in the operational considerations (step 3). However, an organic team structure might require a communication system that was not originally considered when looking at required components (step 2). Thus, step 2 can be revisited and the methodology can iterate accordingly.

Discussion and Future Work

This paper presents the results of research in three areas. First, the paper demonstrates the importance of considering CONOPs variables in addition to component variables. Second, the paper highlights the highly interdependent nature of single CONOPs considerations with the rest of the system architecture. This is important because changes in CONOPs can often be much easier to accomplish than changes in components, and therefore it is very important that system architects consider CONOPs early in the process of designing complex systems, particularly systems of systems where they may not have as much influence over component selection. After the system has been developed, however, due to the interdependency, these changes may have limited effect, or unintended consequences, and care must be exercised. Finally, using a maritime security UV SoS as an example, this paper presents a methodology for thinking about and generating CONOPs choices, that may not have been obvious.

The methodology introduced in this paper still needs to be verified and validated. One of the best ways to do this would be to conduct an experiment, where practitioners develop system designs with and without the methodology introduced here. The methodology would be assessed according to the number of generated designs, as well as the quality. Quality of the

designs would be determined by how much value the designs would be expected to deliver to stakeholders, determined though modeling and simulation, as well as stakeholder preferences. If using the methodology generates high quality designs that otherwise would have been overlooked, then the methodology would be useful as it increases the likelihood of finding a design that would satisfy all of the stakeholder preferences and constraints.

Future work is focused on applying the methodology to other systems of systems cases, to generate a comprehensive list of context and SoS characteristics, as well as components and CONOPs considerations, that may be used in future empirical use via modeling and simulation. The eventual goal is to design principles for performance and survivability for similar systems of systems that share these contextual, component and CONOPs considerations.

References

Bahill, A.T., and R. Botta. 2008. Fundamental principles of good system design. *Engineering Management Journal* 20 (4):9-17.

Bindi, V.D., J. Strunk, J. Baker, R. Bacon, M. Boensel, F. III, and R. Vaidyanathan. 2008. Littoral undersea warfare: a case study in process modelling for functionality and interoperability of complex systems. *International Journal of System of Systems Engineering* 1 (1):18-58.

Courtright, J.A., G.T. Fairhurst, and L.E. Rogers. 1989. Interaction patterns in organic and mechanistic systems. *Academy of Management Journal*:773-802.

Dickerson, C.E., SM Soules, MR Sabins, and PH Charles. 2004. Using Architectures for Research, Development, and Acquisition. Washington, DC: Office of the Assistant Secretary of the Navy (Research, Development and Acquisition).

Drach-Zahavy, A., and A. Freund. 2007. Team effectiveness under stress: A structural contingency approach. *Journal of Organizational Behavior* 28 (4):423-450.

Huang, C.C., and A. Kusiak. 1998. Modularity in design of products and systems. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on* 28 (1):66-77.

Jang, M.W., S. Reddy, P. Tosic, L. Chen, and G. Agha. 2003. An actor-based simulation for studying UAV coordination. In *15th European Simulation Symposium*. Delft, The Netherlands.

Jogalekar, P., and M. Woodside. 2000. Evaluating the scalability of distributed systems. *Parallel and Distributed Systems, IEEE Transactions on* 11 (6):589-603.

Latane, B., K. Williams, and S. Harkins. 1979. Many hands make light the work: The causes and consequences of social loafing. *Journal of Personality and Social Psychology; Journal of Personality and Social Psychology* 37 (6):822.

McManus, H., M. Richards, A. Ross, and D. Hastings. 2007. A Framework for Incorporating" ilities" in Tradespace Studies.". Paper read at AIAA Space 2007, at Long Beach, CA.

Mekdeci, B., and ML. Cummings. 2009. Modeling Multiple Human Operators in the Supervisory Control of Heterogeneous Unmanned Vehicles. In *9th Conference on Performance Metrics for Intelligent Systems (PerMIS'09)*. Gaithersburg, MD.

Mekdeci, B., A.M. Ross, D.H. Rhodes, and D. Hastings. 2011. System Architecture Pliability and Trading Operations in Tradespace Exploration. In *IEEE International Systems Conference 2011*. Montreal, PQ.

Mekdeci, B., A.M. Ross, D.H. Rhodes, and D.E. Hastings. 2011. Examining Survivability of Systems of Systems. In *INCOSE International Symposium 2011*. Denver, CO.

Reigle, R.F. 2001. Measuring organic and mechanistic cultures. *Engineering Management Journal* 13 (4):3-8.

Richards, M, D Hastings, D Rhodes, and A Weigel. 2007. Defining Survivability for Engineering Systems. Paper read at 5th Conference on Systems Engineering Research, March 2007, at Hoboken, NJ.

Richards, Matthew G. 2009. Multi-Attribute Tradespace Exploration for Survivability, Engineering Systems Division, Massachusetts Institute of Technology, Cambridge, MA.

Software Engineering Standards Committee of the IEEE Computer Society. 1998. IEEE Guide for Information Technology - System Definition - Concept of Operations (ConOps) Document. New York, NY: The Institute of Electrical and Electronics Engineers, Inc.

Voas, J. M., and K. W. Miller. 1995. Software testability: the new verification. *Software, IEEE* 12 (3):17-28.

Biography

Brian Mekdeci completed a B.A.Sc. (2002) and a M.A.Sc. (2005) in systems design engineering at the University of Waterloo in Canada. Afterwards, he worked at CDL Systems Ltd in Calgary, Alberta as a systems engineer in charge of designing ground control station software for unmanned aerial vehicles. Currently, Brian is researching survivability and methods of operations for systems of systems as part of his doctoral studies at MIT.

Adam M. Ross is a Research Scientist in the MIT Engineering Systems Division (ESD) and a cofounder of the MIT Systems Engineering Advancement Research Initiative (SEAri). His research focuses on managing unarticulated value, designing for value robustness, and dynamic tradespace exploration for complex systems. Dr. Ross received his Ph.D. from MIT ESD and has published over 50 papers on changeability in system design and analysis. He has work experience with government, industry, and academia including NASA Goddard, JPL, the Smithsonian Astrophysical Observatory, Boeing Satellite Systems, MIT, Harvard, and Florida State Universities, performing both science and engineering research.

Daniel E. Hastings is a Professor of Aeronautics and Astronautics and Engineering Systems at MIT. Dr. Hastings has taught courses and seminars in plasma physics, rocket propulsion, advanced space power and propulsion systems, aerospace policy, technology and policy, and space systems engineering. He served as chief scientist to the U.S. Air Force from 1997 to 1999, as director of MIT's Engineering Systems Division from 2004 to 2005, and is a former chair of the Air Force Scientific Advisory Board. Dr. Hastings was elected a Fellow of the International Council on Systems Engineering (INCOSE) in June 2007.

Dr. Donna H. Rhodes is co-founder and director of SEAri and a Senior Lecturer in Engineering Systems at MIT, where she is also a Principal Research Scientist. Her research interests include predictive architecting of socio-technical systems and anticipatory capacity in enterprises. Dr. Rhodes has 20 years of experience in the aerospace, defense systems, systems integration, and commercial product industries. Prior to joining MIT, she held senior level management positions at IBM Federal Systems, Lockheed Martin, and Lucent Technologies in the areas of systems engineering and enterprise transformation. Dr. Rhodes is a past president and Fellow of INCOSE, and the 2005 recipient of the INCOSE Founders Award.