IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received December 27, 2019, accepted January 9, 2020, date of publication January 21, 2020, date of current version February 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2968348

Battlefield Mapping by an Unmanned Aerial Vehicle Swarm: Applied Systems Engineering Processes and Architectural Considerations From System of Systems

CHAD KERR, (Member, IEEE), RAED JARADAT, (Member, IEEE), AND NIAMAT ULLAH IBNE HOSSAIN
Industrial and Systems Engineering Department, Mississippi State University, Starkville, MS 39759, USA
Corresponding author: Raed Jaradat (jaradat@ise.msstate.edu)

ABSTRACT For many years, Unmanned Aerial Vehicle Swarm (UAVs) have been implemented as stand-alone systems for various applications. Advances in technology necessitate the need for the complexity of the systems to rise to a level higher than ever. The capability needs today require the use of multiple systems to act together to accomplish a common goal that is unachievable with a single system. The use of multiple UAVs in an autonomous, or semi-autonomous, nature to accomplish this higher-level goal is known as a UAV Swarm system of systems. In designing a system with the complexity of a UAV Swarm there are many aspects that must be taken into careful consideration. This article will identify the SoS principles and considerations for developing and deploying a UAV Swarm for Battlefield Mapping using System Modeling Language. The design of the developed battlefield SoS will utilize standard Systems Engineering processes, adapted Systems Engineering processes for Systems of Systems, and architecture representation models. Each of these key utilities will be analyzed to provide guidance and considerations for SoS Engineering of the UAV Swarm.

INDEX TERMS Systems engineering, system of systems engineering, unmanned aerial vehicles (UAV), emergent behaviors, systems thinkers.

I. INTRODUCTION

The Systems Engineering process is well established and widely used in the development of systems. Organizations have adopted the systems engineering process from the International Council on Systems Engineering (INCOSE) and tailored the approach for their specific application area [1]. Although the process may be tailored, the key steps are the same throughout the systems engineering domain.

The advances in technology and the variety of systems available today, provide the means for creating even more complex systems and Systems of Systems (SoS). A SoS is a system composed of individual systems that work together to perform a higher-level capability that is not possible with the constituent systems alone [2]–[5]. The complexity of a SoS may drive designers to not follow the typical standard systems engineering process alone, and therefore a new approach must be taken [6]. The approach will be described in this paper.

Situational Awareness is key with all military operations. Typically, most ground operations rely on satellite imagery, boots on the ground, or aerial photography. The weakness in this approach is the human-in-the-loop to interpret and often input the data. There have been many attempts to develop a system to provide real-time data to commanders in theater. The need for a collaborative SoS, is also apparent in recent publications from military agencies. Specifically, the announcement of the Distributed and Collaborative Intelligent Systems and Technology (DCIST) program by the Army Research Lab (ARL). In the announcement, it states that, "In extending this vision to 2040 and beyond, it is also envisioned that future intelligent systems will need to exhibit adaptable levels of autonomy and work across large heterogeneous teams of intelligent agents." Additionally,

The associate editor coordinating the review of this manuscript and approving it for publication was Zhen Li.

the announcement indicates, ‘‘Fundamental gaps exist in the understanding of collaborative intelligent systems, whose design and operation is complicated by increases in communication among agents [7].’’

The SoS that will be the topic of this article is an Unmanned Aerial Vehicle (UAV) Swarm for Battlefield Mapping. This SoS will deploy multiple UAVs with optical sensors to a specific geographical area for surveillance and mission planning. The UAVs will work together to detect, identify, and locate threats within the search area and relay that information to the ground station. Stations will be geographically spaced for efficient data dissemination to commanders and military personnel. This autonomous operation allows for fewer human operators that must survey the raw imagery looking for a threat. A single operator could potentially interact with the entire SoS from a single control station.

The main purpose of the developed SoS is to locate threats on the battlefield so that the troop mission on the ground can track, engage, and defeat the enemy. The UAV Swarm provides a bird's eye view of the battlefield arena, providing essential data to personnel on the ground for situational awareness and decision making.

The stakeholders for this system are actors that have a stake in the system including those relying on the system, operating the system, and those that must maintain the system. The identified stakeholders are: Military Logistics, Military Personnel, Federal Communications Commission (FCC), Federal Aviation Administration (FAA), SoS Development Organization, Constituent System Organizations, Enemy Forces, and International Organizations. Although the inclusion for most of the stakeholders is straightforward, the enemy force is a critical stakeholder to consider when designing this system. The enemy will try to deny, disrupt, or destroy the system. Therefore, considering their intentions will allow for a more robust design being deployed to the field.

The complexity and nature of this SoS maintains the need for a governing agency over the development of the SoS. The purpose of this agency is to manage the system during operational deployment and to support changes that are inevitable throughout its lifecycle.

The initial use of the UAV was similar to the application mentioned here, surveillance, but the evolution to a cooperative network of systems occurred over time. The fundamental capabilities present in the swarm of UAVs, soon became requirements for development of these systems [8]. In nature many systems operate in the swarm mentality. For instance, bees that are fighting an enemy will work together to fight off predators 100s of times larger than themselves. The bee swarm is also a system of systems that has a capability that is not achievable with a single system alone.

The future of UAV utilization is clearly going toward swarm architectures [9]. The relatively low cost and low risk to human life builds a strong business case for UAV swarm deployment. Future growth of this technology stretches from, wireless communication networks, disaster monitoring, search and rescue and military applications. The military applications alone are numerous, with offensive and defensive areas [10]. The application of SoS systems engineering processes and architectural considerations are vital to the knowledge advancement in this domain.

This article will identify the high-level aspects involved in the engineering of a SoS, with specific application to the Battlefield Mapping use case. A discussion of systems engineering processes applied to SoS development will lay the baseline knowledge. Following this will be a detailed discussion of architectural considerations of a UAV Swarm SoS. Finally, recommendations for UAV Swarm development and future research are provided.

II. LITERATURE REVIEW

UAV Swarms contain the possibility of numerous capabilities and applications. Within the military domain the applications are abundant. In an article by [11] a layered, multi-purpose UAV Swarm is described. The swarm contained four distinct layers of capability, each having a specific set of functions. The layers are: 1) communication and visual reconnaissance, 2) attack site coverage, 3) anti-missile sensing (defensive), and 4) securing tactical zones (offensive)

There are three primary variables that can be used to evaluate the overall success, or effectiveness, of a swarm, as described by Edward [12]. The three variables are 1) elusiveness, 2) standoff capability, and 3) superior situational awareness. Elusiveness is the ability to stay undetected by the enemy until the units converge on the enemy, overwhelming the enemy by surprise and sheer numbers. Standoff capability is the ability to cause damage to the enemy over some distance, while accepting little damage in return. Finally, situational awareness is the ability to measure and know the environment and the position of other systems within the SoS. This awareness allows for calculating the best attack vector on the enemy while coordinating with other systems.

When assembling a swarm of systems there are two key types of coordination that are necessary. Based on the research presented at the AIAA conference in 2003 [13], spatial and temporal coordination along with the specification of distinct roles within the team are paramount to the successful implementation of a swarm in theatre.

‘‘Spatial co-ordination distributes units over the area being observed, and includes such tasks as determining the maximum spread between vehicles and the minimum acceptable number of revisits per unit area, assigning sectors to each unit, causing a team to converge in a specific location, or stationing UAV's in a particular formation’’ [13].

‘‘Temporal coordination ensures that all UAV's act at the right time or with the right frequency, provide their input at the right moment, and assume their designated locations and operating roles at the right time for the constellation to work as a whole.’’ [13] Given the spatial and temporal coordination, the systems must form a team to achieve a common goal [14].

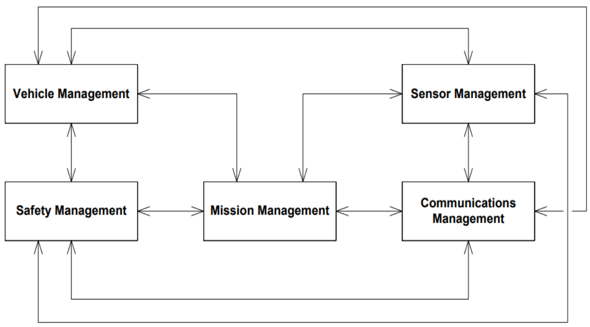

FIGURE 1. System management architecture [Eaton *et al.* 2016]–This diagram provides the management structure and flow for the UAV Swarm SOS.

TABLE 1. Summary of litrature review.

Author, Year	Focus	Limitations
Reed, [11]	Architectural Concept of Operations	Systems Engineering perspective not presented
Edwards, [12]	Military Applications of Drones	Systems Engineering perspective not presented
SIAM, [13]	Mathematical Decision and Control Framework	Systems Engineering perspective not presented
Waharte & Trigoni, [15]	Swarm Search and Mapping Algorithms	Systems Engineering perspective not presented
Eaton, et al., [24]	System Architecture, Existing Control Methods	Constituent system lifecycle considerations not presented

There are 3 basic types of teams; coordinated, cooperative, and collaborative [15]–[17].

Coordinated teams are designed and developed to have a group objective function. All systems within the teamwork toward a common objective. Each team member can be over- or under-utilized to satisfy the group objective.

Cooperative teams consist of systems with both team and individual objective functions. The functions are weighted to optimize the team objective function. Although the individual systems have individual objectives, such as fuel conservation and survivability the team objectives take precedence.

Collaborative teams are systems with only individual objective functions. All constituent systems are trying to optimize their own objective function without knowledge of the team objective. Although the functions are aligned with the team objective, there can be conflicts of interest. This adds complexity where the SoS must have accounted for conflict resolution. This team method is more individualistic. A team member may choose to leave the team if the individual objective is not being optimized.

There are numerous ways to identify a target in an environment. Whether that target be a threat or a target for rescue, the process is the same, search an area and identify the target. The question arises as to why a swarm is necessary. While non-cooperative systems can correctly identify a target, simulation results show that cooperative systems both identify a target significantly quicker and with a higher probability [14], [15].

In the battlefield, finding the enemy before he finds you is vital to the overall mission success and safety of the soldiers [18], [19]. Swarms of systems, or a SoS, can quickly and efficiently complete a mission.

The higher-level goals can drive requirements onto the systems themselves. The architecture for such a system can be considered in a top-down approach, increasing the level of fidelity. At the highest level the architecture should consider management of the UAV systems and subsystems, as shown by FIGURE 1.

At a higher fidelity, system elements should be split into multiple levels of control. There should be a lower level control, Vehicle Level, for controlling the individual system, along with sensors and actuators, and a higher-level control for coordination between systems, Mission Level. These along with wireless control and other sensors provide the basic architecture for a UAV in a Swarm SoS [20].

The need for fewer humans in the loop is predominant in many different studies [21]. There are numerous personnel required for a mission and the stress is very high for these personnel [22]. This stress level grew so high, a new organization was formed to represent UAV operators in a unionized fashion [23]. This necessitates the operator-UAV ratios to shift from many-to-one, to a one-to-many [24], enabled by increased autonomy of the SoS.

The literature review revealed many sources for specifics of UAV Swarm applications, control methods, and frameworks for search and control algorithms. The application and considerations of Systems Engineering Processes to the development of SoS was not found. A summary is provided in Table 1.

III. SYSTEMS ENGINEERING PROCESS FOR SOS

In classical systems engineering the top down approach is necessary to develop the system based on the customer needs and requirements. While in the more complex field of SoS, the bottom up approach is prevalent and often it is a combination of both.

The top-down approach is used for new system development. This starts with a high-level abstraction of the system or conceptual description. As the design progresses, each subsequent design effort drills down to constituent subsystems to build the system architecture and definition.

The bottom-up approach is used when the system is complex and consists of standard components. The UAV swarm is comprised of systems previously developed. The components, or bottom, of the hierarchy of the system decomposition are designed, or selected, and then appended to the existing platforms.

Often, when designing a complex system, systems engineering processes cannot be used in a straightforward manner. These processes must be tailored to the SoS domain. There are seven core elements that provide context from the Systems Engineering domain to the SoS domain [3].

As described by Madni, the complicated and complex nature of an SoS requires a Model-Based Systems Engineering (MBSE) approach. MBSE helps resolve aspects where traditional systems engineering approaches struggle. Namely, MBSE is a common language among stakeholders, generates system documentation, and provides a common repository for the system information [40].

A. TRANSLATING CAPABILITY OBJECTIVES

In the systems engineering process, requirements definition is based on a customer needs analysis. For most cases this is straight forward, and the customer is involved throughout the process [25], [26]. While this is similar in the SoS engineering process, there are obstacles that will inhibit the requirements definition process. Requirements at the SoS level must be tailored to achieve the high-level objectives of the SoS, without considering specific constituent systems [27].

Once the objectives of the SoS are identified, the technical requirements can then be developed. The main goal of developing requirements that are system independent, is flexibility in the constituent system composition. In the beginning of the systems engineering process the requirements must not specify a solution, only a means to evaluating potential solutions. These requirements will change throughout the development, creep of the requirements is likely and management of these requirements is crucial. Over time the requirements will become solidified. This is typical for system development, but in a complex SoS the management of the evolving requirements is key.

In context to the UAV Swarm, a high-level capability would be to detect and locate a target. This capability could feed into a requirement for data transmission rates and image processing capability for the constituent systems, as an example.

B. UNDERSTANDING SYSTEMS AND RELATIONSHIPS

When developing a complex SoS with many constituent systems, the individual component, or system, developments must be taken into consideration. Many times, the hardest aspect of this is the synchronization of the schedules, risks, and managements. Different governing bodies for the constituent systems means change is inevitable [28]. The key is to plan for the known changes and prepare for the unknown. This is difficult to grasp, but analysis on the purpose and mission of the constituent systems along with a forecasted mission analysis will provide some insight into the expected evolution of the systems.

Along with the management aspects, the technical side is just as critical. The relationships among the constituent systems and their constraints, risks, and capabilities are vital to the success of the SoS.

For the UAV Swarm application, the interfaces among the UAVs must be considered carefully. The development process should specify the interfaces between UAVs and Ground Stations, between Ground Stations and between the Human and the SoS. Each interface should be clearly specified and managed to ensure requirement satisfaction. Since the systems are physically homogenous, the management relationships among the elements can be easily managed at the SoS level.

C. ASSESSING PERFORMANCE TO CAPABILITY OBJECTIVES

When assessing the performance of a system using the standard systems engineering process, the requirements must be satisfied to evaluate that performance. In the SoS domain, requirements are derived from high-level capabilities [29]. Therefore, the requirements and the assessment must contain SoS level metrics and methods for assessing that performance. These metrics must not impose an architecture on the design.

Considering the UAV Swarm, requirements at the SoS level would include time to target identification, probability of intercept, deployment time, operational duration.

These high-level metrics allow for assessing the performance of the SoS at a high level while not constraining the solution space.

One method for analyzing the performance of the SoS in the UAV Swarm application would be modeling and simulation. Extensive effort has been placed in this field and an accurate simulation could be generated to analyze the metrics at the SoS level and feasibility. This would allow for feedback into the requirements, without the need for a costly scale model and testing of the SoS.

D. DEVELOPING AND EVOLVING AN SOS ARCHITECTURE

This core element is the bulk of the architecture design process. It encompasses the concept of operations, functionality, relationships, and data flows.

The definition of the SoS should contain a Concept of Operations (CONOPS) for clearly identifying the operation of the SoS in the intended scenario. This allows for most of the functionality, data flows, and the relations between the constituent systems to be clearly identified. It sets the expectations of the SoS level and provides the options and trades for the systems and the SoS. It also allows for the assessment of changes that could occur, whether they be constituent systems or environmental.

The UAV Swarm can be subjected to a wide variety of environments within its deployment. The CONOPS should address these environmental considerations for the system. While most environment changes are mild, if constituent systems of the SoS are designed to operate in a dry environment, deployment to a coastal environment could lead to failures or poor performance. These changes should be accounted for and planned for at the architecture development stages of the design.

E. MONITORING AND ASSESSING CHANGES

As with any system there will be changes along the development of the SoS. The goal when dealing with change is monitoring and assessing the impacts of those changes at the SoS level. Any changes will have some impact on SoS performance whether it be destructive or constructive (Pei, 2002[ref]. Addressing these impacts and monitoring the SoS performance is vital to the success of the system.

Changes can consist of technology, functionality, or mission changes. Technology and functionality changes typically coincide. As a technology matures, it gets more complex and starts to open new fields of application. This could prove to be a potential enhancement of the SoS or could alter the performance of the SoS. Close monitoring of these changes must be handled and analyzed at the SoS level to ensure the impacts are acceptable.

One change in the UAV domain is the role change of the UAV. Instead of being an imaging platform, the role could change to that of a Wireless Communications Network. This would push the UAV development away from universal payload acceptance, toward a unique communications domain payload. Here the SoS would suffer from the change in payload capabilities but would benefit from the more robust communication development that is vital to both domains. Monitoring these changes will allow for preparation for the changes and mitigation plans if necessary.

F. ADDRESSING REQUIREMENTS AND SOLUTION OPTIONS

Customers and designers often seek out a single solution for all problems. This is not feasible in SoS [3], [30]. In the beginning stages, the high-level capabilities are fed down into requirements. Once the requirements are prioritized, they can be evaluated for the needs of the SoS. Each capability at the SoS level should have requirements allocated to functional blocks that satisfy that capability. These functions are then mapped to a specific system within the SoS.

As an example, a requirement to be operational in the air for a minimum of 12 continuous hours, would be mapped to the UAV itself. This mapping allows for evaluating multiple alternative solutions to the problem without tying the solution to a specific platform.

G. ORCHESTRATING UPGRADES TO THE SOS

When developing and supporting a SoS with many different constituent systems that have their own governing bodies, upgrades can come at many points within the lifecycle. The challenge is managing all the upgrades and facilitating the integration and testing of the upgrades [31]. The upgrade must be analyzed to consider the benefits and costs of the change to the SoS. While this should be managed in the previously mentioned core element, this element deals mainly with the facilitation of the integration.

For the UAV Swarm, the upgrade to the existing systems should be carefully coordinated to ensure the missions of the SoS are not compromised. After the upgrades are integrated the SoS must be tested and reevaluated.

When working with an SoS, the systems engineering process can be amended with the seven core elements laid out above. The mapping between the seven core elements to the Technical and Technical Management Processes of the standard systems engineering process can be seen in FIGURE 2. As shown, the SoS core elements map more directly to the Technical Management Processes. As described the SoS engineering process deal more with coordination and direction across mulitple system development timelines.

	Technical Processes								Technical Management Processes							
	Rqts Devl	Logical Analysis	Design Solution	Implement	Integrate	Verify	Validate	Transition	Decision Analysis	Tech Planning	Tech Assess	Rqts Mgmt	Risk Mgmt	Config Mgmt	Data Mgmt	Interface Mgmt
Translating Capability Objectives	X											X	X	X	X	
Understanding Systems and Relationships		X											X	X	X	X
Assessing Performance to Capability Objectives							X		X		X		X		X	
Developing and Evolving an SoS Architecture	X	X	X						X	X		X	X	X	X	X
Monitoring and Assessing Changes									X				X	X	X	X
Addressing Requirements and Solution Options	X		X						X	X		X	X	X	X	X
Orchestrating Upgrades to SoS				X	X	X	X	X	X		X	X	X		X	X

FIGURE 2. **Classical systems engineering technical & technical management processes as they apply to the core elements of system of systems, systems engineering. Graphic from [41].**

IV. ARCHITECTURE CONSIDERATION FOR UAV SWARM SOS

A. RELATIONSHIP BETWEEN CONSTITUENT SYSTEMS

In the UAV Swarm the SoS is comprised of homogenous systems that work together. Each system contains similar functionality to aid in the robustness and redundancy of the SoS. If a system, or multiple systems, within the SoS become compromised, the impact to the overall system should be minimal.

Although the constituent systems are homogenous at a high level, they are assigned functions within the swarm. This again aids in the robustness of the system. The two basic types of functions of this SoS are active and supportive [32].

Active systems are collecting data and providing some primitive analysis of that data. They are also providing geographic location and environmental conditions of both themselves and targets identified. This information is relayed back to a supportive system, for transmission to the ground station or higher-level processing. The supportive systems main function is coordination and data relay.

B. UAV SWARM SCENARIO CONSIDERATIONS (THE BATTLEFIELD MAPPING SOS)

When architecting a complex SoS, there are special considerations that must be accounted for in the design. The six SoS specific architectural considerations are autonomy, complexity, diversity, integration strategy, data architecture, and system protection.

1) AUTONOMY

Autonomy in the SoS domain is related to the development processes of the constituent systems [33], [34]. The developed battlefield mapping SoS is a complex system comprised of UAVs, sensors, communication systems, and ground stations. Along with the physical system components, the infrastructure to support this complex system, through operations, maintenance, and support is also present. The design of the SoS must consider all systems, subsystems of the SoS, and their respective stakeholders, missions, management, budgets and schedules. The development of the SoS must not impede on the autonomy of the constituent systems, while maintaining its own development [2].

In respect to the UAV Swarm, care must be taken to not interfere with the existing operational support in place for the constituent systems. The governing organization that designed and supports the UAVs after deployment must maintain the ability to support the UAV in the SoS. This will avoid extreme costs and schedule impacts of developing a unique infrastructure for continued support.

2) COMPLEXITY

While using existing systems is typical for SoS. It adds a layer of complexity. Most systems tend to converge on a single objective and the most cost-effective means to satisfy that objective. This means that while a system is adequate for satisfying the SoS objective at the time of conception, this may change as the development progresses. This adds a layer of complexity in the unknown future growth.

In the beginning stages of UAV development, they were seen as expensive toys, now they are vital components on the battlefield [7]. If the role of the UAV changes this could place more burden on the UAV swarm SoS. As an example, if the future growth of the UAV, mentioned earlier, moves toward a predominantly wireless communications network role, this could drastically alter the performance of the system. In the wireless communication network role, the UAVs would start to take more consideration of the weight and not payload. The need for long loiters and low power would place constraints on the battlefield mapping application. Mapping of a battlefield requires larger mass optical sensors along with powerful processors for analyzing the data. This mission change will minimize loiter time for battlefield mapping operations.

This divergence of objectives of the systems and SoS means that future development of "bridges" to fill the gaps in performance will be likely for continued mission success.

3) DIVERSITY

Diversity is closely tied to the autonomy and the complexity example above. The SoS is composed of a diverse set of systems. These systems each have their own missions with specific needs. If the needs of the constituent systems begin to diverge from the SoS application, it can destroy the SoS.

Changing needs of the stakeholders can change the path of the constituent systems [6]. Changes similar to the Wireless Communications Network, could change the performance characteristics of the SoS. Along with the stakeholders the environmental changes of each system could also shape the evolution of the system. Diverging needs, politics, budgets all have impact on each system and therefore greater impact on the SoS.

TABLE 2. Data model comparison.

Data Model	Semantic Risk	Data Volume
Uncoordinated	Low	Low-Mid
Coordinated	High	Low-Mid
Federated	High	High

4) INTEGRATION STRATEGY

Integration strategy is the process of integrating the constituent system together into a unified whole [5], [35]. This is typically implied during the standard systems engineering process where the components are designed, and the system divided into functional groups. In the SoS context this is not as straightforward. Each of the constituent systems were developed for their own purpose and mission and are being integrated into a larger system.

There are two distinct methods for integrating a SoS. The first is bridging, this is the addition of a new system that bridges the gaps between the existing systems. The other is refactoring, where the existing systems are modified to work together [3].

In the UAV Swarm the best solution is refactoring. These platforms, as with any modern system, are flexible due to the inherent flexibility desired for these systems in their current role. The system is dominated by a software architecture that allows for reprogramming and the addition of functionality. This lends well to the UAV Swarm application, where the communications used can be restructured to be internal SoS communications.

This also add a level of robustness to the design. Software intensive systems are continually being upgraded with performance and functionality being revised. This allows for easy reprogramming of the platforms for future growth and potential.

5) DATA ARCHITECTURE

Data architecture considers both the semantics of the data, as well as the storage of SoS level data. Data is the lifeblood of any modern system. It is this data that must be communicated properly to ensure correct operation of the system.

There are three data models that are employed at the SoS level [3]. The main differentiation between the models is semantics and data volume. A comparison of the architectures is shown in Table 2.

The UAV swarm should employ an uncoordinated data model. This is due to the low semantic risk, and moderate data volume.

In the UAV Swarm data architecture is not as much an issue as with other more complex systems. The swarm is predominantly homogenous; therefore, each system will employ the same semantics and data processing algorithms. The only interface that contain the risk of issues to arise is at the UAV to Ground Station interface. This is the link from the SoS to the humans on the ground. This link must be managed both semantically and temporally for complete data transmission.

6) SYSTEM PROTECTION

System protection involves four objectives; 1) confidentiality is preventing unauthorized access to the system or data, 2) authentication is providing a means for identifying authorized systems, 3) integrity is restricting access to unauthorized users to system modifications, 4) nonrepudiation is protecting the identity of the users of the system.

Another aspect of system protection is the internal threat. Constituent systems could become a threat to the SoS. This is mitigated by ensuring isolation between the systems.

The UAVs are military grade systems. Therefore, system protection is inherent. One beneficial element of the UAV Swarm is the homogenous nature of the system. Each system will be similar providing similar protection functionality and easing the process for system protection.

In a SoS such as the UAV Swarm, the critical element is the software. When utilizing existing systems, integration strategy is typically where the bulk of the engineering is located. For this system, this is a software intensive task. The software engineering team will need to be a SoS level team, knowing the high-level capabilities of the SoS and the requirements. This will allow them to develop the software architecture to allow for capability achievement and flexibility for future upgrades or reallocation.

The system architect should work closely with the software engineering team. Misalignment of the software development could cause the system to fail to meet its requirements and ultimately fail to defeat the enemy.

Systems experience failure and threats from the environment. This can cause system-wide failures if a critical system is destroyed. The complex nature of a SoS encourages the use of redundant systems. Failure modes of the system must be analyzed and mitigated by applying redundancy principles throughout the design [36]. The failure mode is of great concern when designing a redundant system. The failure mechanism must be different from the redundant system failure mechanism to ensure true redundancy and avoid complete failure.

In the UAV Swarm redundancy is abound, but this has its pros and cons. The beneficial side is that the homogenous nature of the SoS lends well to constituent system failure recovery. If a system fails, there are other systems within the SoS that can take over and accomplish the tasks of the failed system. The drawback is that the failure mechanism is common. If the failure mechanism is software related, this could be catastrophic for the SoS. Since the SoS is almost entirely homogenous in form and function a software bug that is environmentally independent could cause wide-spread failure.

Within the UAV Swarm, redundancy would be required in the following functions: location, altitude, and heading measurement, and target identification.

The UAV Swarm has many constituent systems performing the same functionality. These critical functions ensure that the constituent systems contain situational awareness and identify the target with a high probability of success. These functionally redundant elements should be evident in the architecture.

C. ARCHITECTING MODELS FOR THE BATTLEFIELD MAPPING SOS CASE

Situational Awareness is key with all military operations. Typically, most ground operations rely on satellite imagery, boots on the ground, or aerial photography. The weakness in this approach is the human-in-the-loop to interpret and input the data. There have been many attempts to develop a system to provide real-time data to commanders in theater. The need for a collaborative SoS, is also apparent in recent publications from military agencies. Specifically, the announcement of the Distributed and Collaborative Intelligent Systems and Technology (DCIST) program by the Army Research Lab (ARL). In the announcement, it states that, "In extending this vision to 2040 and beyond, it is also envisioned that future intelligent systems will need to exhibit adaptable levels of autonomy and work across large heterogeneous teams of intelligent agents." Additionally, the announcement indicates, "Fundamental gaps exist in the understanding of collaborative intelligent systems, whose design and operation are complicated by increases in communication among agents." [7]

The purpose of the mapping case is to address the gap by discussing the development of two operational models and their behaviors and actors using SysML through the battlefield mapping scenario.

1) PURPOSE

The purpose model is a view of what the customer wants. This is a first step in the engineering process for any system development and crucial in a SoS of this magnitude of complexity, risk, and cost. Getting a clear view of what the customer wants is critical to giving the customer what they expect at deployment.

The preferred method for showing what the customer wants is the OV-1 from the Department of Defense Architecture Framework, or DODAF. The OV-1, or Operational View 1, shown in FIGURE 3 shows the overall picture of the SoS. The UAVs are deployed to survey an area of interest looking for targets. The UAVs will coordinate their efforts to efficiently scour the area to provide full coverage of the specified geography. To provide some context a Concept of Operation is provided.

The UAVs are deployed with a mission plan. This plan will include the GPS coordinates of the area that is to be covered, possible threat lists, and data encryptions. The UAVs are deployed and begin communicating among the systems, utilizing algorithms to coordinate an efficient search of the area. The UAVs will fly over the specified location and continuously search for targets. Once a target has been identified, the system will coordinate with the closest system to that location to confirm the threat. Once the threat has been confirmed the data will be transmitted to the support system

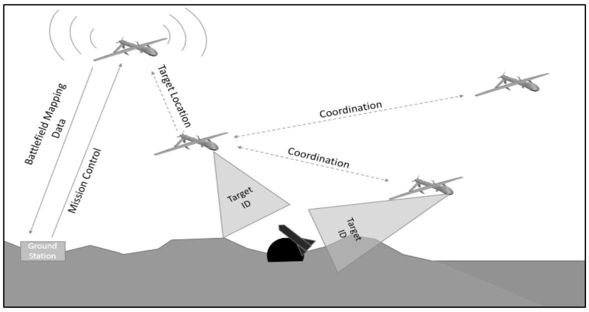

FIGURE 3. **Battlefield mapping SoS OV-1 – The operational viewpoint 1 (OV-1) provides a high-level overview of the SOS, highlighting the key components and interfaces between them.**

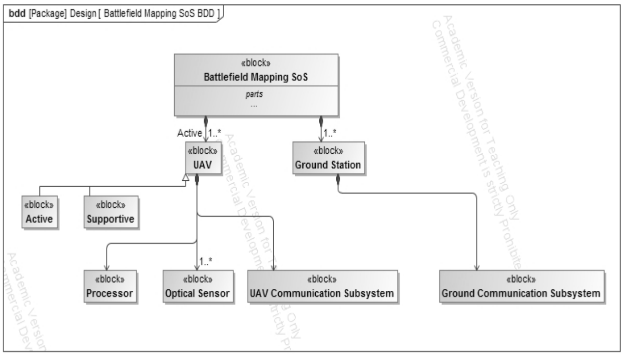

FIGURE 4. **Battlefield mapping SOS block definition diagram – The block definition diagram shows a hierarchical composition of the SOS.**

for processing and downlink to the Ground Station. A single system will maintain a track on the threat until it is destroyed.

The ground station is the Human interface for viewing the battlefield and making high level executive decision about the operations of the SoS. The high-level decisions that require human interfacing is aborting the mission and returning to base.

2) FORM

The form models of the SoS are representations of the system in a physical sense. The purpose of the form model is to give some perspective of the system composition.

At the SoS level of the UAV Swarm, a domain block definition diagram (bdd), SoS level bdd, and an internal block diagram (ibd) are great tools for modeling the form, or components, of the SoS. The bdd for the domain level shows the domain is composed of the system of interest, the physical environment, transportation equipment and operational support elements.

The bdd for the SoS level is shown in FIGURE 4. This diagram shows all the components that make up the SoS. The purpose of this representation is to provide a view of the decomposition of the SoS into its constituent systems. The UAV Swarm is composed of UAVs and ground stations, both with a multiplicity of 1 to many. There are special types of UAVs that are defined by their functions within the operational scenario. One set is active, utilizing the sensors and processors on board to search for and detect threats, and

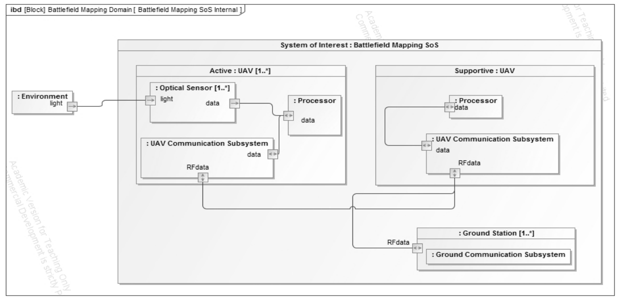

FIGURE 5. **Battlefield mapping SoS internal block diagram – The internal block diagram shows the internal connections within the SOS.**

also to communicate with other systems within the SoS for coordination of their operations. The other specialization is a supportive role. These UAVs are mainly for networking purposes to relay the data to the Ground Stations, provide additional data processing and to assist if the need arises.

The ibd, FIGURE 4, for the UAV Swarm shows the interfaces of the constituent systems. This diagram allows for identifying and managing all the interfaces of the SoS which is critical in complex SoS. In a primitive way, the information flow can also be seen throughout the system. The main goal of this representation is identifcation and specification of the systems' interfaces.

Data flow through the UAV Swarm is complex. Most data flows require that one side of the flow be interrupted. This may not be feasible with a UAV Swarm where each system is acting independently. The flow may require multiple relays between systems that are currently supporting their own data transmission. This data flow logic will need unique analysis and development to properly design data flow through the system. An example data flow scenario graphic, shown in FIGURE 6, is provided for clarity.

System A would request to send data to System B. If System B is available for data transmission it will acknowledge and schedule the transfer. If System B is not available, System A will wait until it is available, storing the data temporarily in a prioritized queue. Once System A receives authorization and a scheduled transfer window, it will begin to transfer the data. Once the data has been received System B will confirm receipt of the data. The purpose of this graphic is to identify all data requirements of the system. It will aid in understanding where weak points in the architecture are, along with analyzing the flow of data between the systems.

Data flow within the network is heavily reliant on transmission of data from one system to the next. This network will contain critical links that are required to ensure data flow. The modeling effort should incorporate an analysis to identify critical nodes within the network. The application of complex network centrality theory to a power grid was provided in [39]. Like the power grid, the SOS will have key nodes that must be maintained to ensure the SOS remains effective.

Scale models are very useful for some systems. A UAV Swarm would not benefit much from a physically scaled model, except for fit checking of components. The best model

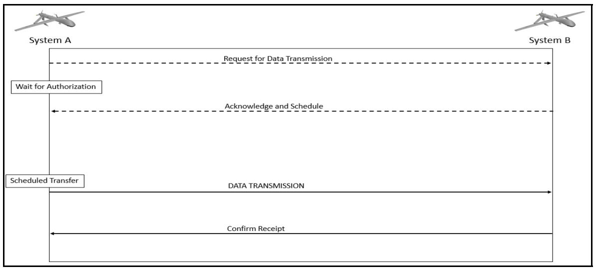

FIGURE 6. **Data flow logic – This data flow diagram shows a high-level concept for the data flow between constituent components of the SOS.**

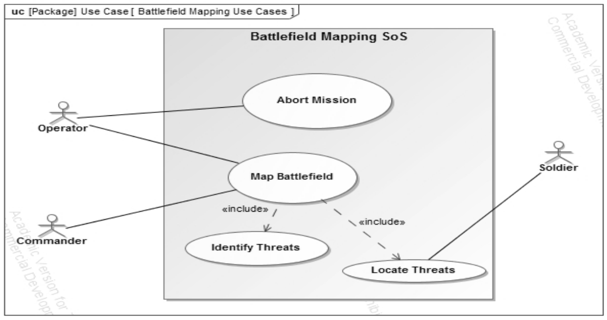

FIGURE 7. **Battlefield mapping SoS use case diagram–the use case diagram describes the functionality of the SOS including the various stakeholders and their interaction with the system.**

for this application is a simulation environment where many aspects of the SoS can be simulated and modeled. Such aspects include SoS functionality and performance analysis, data bandwidth requirements, node criticality analysis, and data storage requirements. This simulation and modeling effort will be discussed in the following section on Performance.

3) BEHAVIORAL

There are different behaviors of the system that involve different actors. The use case diagram in SysML is a great tool for visually representing these relationships.

The output of this representation is a view of the high-level use cases, or functions of the system of interest. Using this diagram, one can begin to understand relationships between actors. This model's purpose is to show what the system does, not what it is. These high-level use cases elicit behaviors of the system.

In respect to the UAV Swarm use case diagram in II, the use case of ''Monitor Battlefield'' implies specific behavior that they system must exhibit. This would imply moving across the geographic area and revisiting locations on a required minimum time scale. The use case of ''Locate Threat'' implies that the system must know where it is in geographic location coordinates along with the respective location of the threat.

There are many behavioral aspects of the UAV Swarm that can be modeled. The Data and Event Flow Network is a thread of all scenarios that the system may encounter. This is necessary to completely model the behavior of the system under specific conditions. The interaction among the constituent systems can be modeled in an agent-based simulation where each agent is given basic behaviors and the SoS is stimulated by different environmental conditions.

4) PERFORMANCE

Performance of a complex SoS is rigorous and requires predetermined plans, procedures, and analysis. The nature of the testing and analysis necessitates the need for a formal test plan before beginning the performance analysis. To aid in this endeavor models of performance can be created to facilitate the understanding of the testing that will need to be accomplished.

A performance model would be required for many different aspects of the UAV Swarm SoS. Key modeling requirements would be required for mission performance metrics, data communications, and data volumes.

All critical paths for data must be fully simulated to understand the performance of the system. If the data flow is delayed, or is caught in a bottle neck, the system could fail to transmit the threat data to the ground station in time. Another aspect is the deployment and operational timing simulations. If the system takes too long to deploy or cannot provide support for the required amount of time, it will not succeed in meeting the need.

Agent based modeling would be required for the deployment, operational, and behavioral simulation of the UAVs. This paired with extensive modeling and simulation of the data paths are critical to understanding the performance of the system.

One key word with SoS is ''emergence'' [2]. In a UAV swarm, emergent behaviors are inherent. The system is autonomous, where each agent in the system has a specific set of rules that it will follow in response to environmental inputs, or interactions with other agents within the SoS. It is from these interactions that emergent behavior spawns. A full analysis of the emergent behaviors, whether they are beneficial or destructive, and their impact on the system must be completed.

As for the architecture of the SoS there are three key quality attributes that should be evaluated, flexibility, simplicity, and reliability. These are the quality attribute, or the ''-ilities'' of the system that are qualitative measures of performance. Although these cannot specifically be tested, they are inherent in the design and should be considered in the architecture.

The UAV swarm must be flexible to accept changes and continue to provide the capability that was initially needed. Simplicity and reliability ensure the system easy to operate, maintain, and deploy will build the business case for continued operations in military applications.

5) DATA

Data within the UAV Swarm must maintain integrity. The data is relayed possibly multiple times and the timing must be kept short. The data is stored temporarily on the initiating system. It is then transferred either through supportive or other active systems to the ground stations for processing and output to the human.

Once at the ground station the data is kept in a database for analysis of the threats in the area. The data that will be stored will include, threat identification, location, system identification, location, timing information and environmental conditions.

Modeling this data should be accomplished using data flow diagrams, as shown in FIGURE 6, as well as modeling the database. Modeling the database will include modeling the entity types, attributes, naming conventions, relationships, and patterns [32].

A key characteristic, and limitation, of a UAV Swarm is the ability to maintain presence on the battlefield. This limitation is primarily driven by power consumption, which is dominated by maneuvering and data processing. UAV maneuvering and flight is improved by battery power densities, light-weight alloys for the body, and physically smaller electronics. Whereas legacy electronics were consistent in power consumption independent of processing load, advances in processors and electronics have improved the power efficiency of electronic systems by minimizing power consumption while idle.

The processing required for this system will be extensive, as the platforms have multiple sensors and independent tracks on enemy systems. The sensor fusion, image processing, and communications would become a burden on the power systems within the SOS. A number of studies have provided various methods for reducing the computational burden within such systems.

Event-driven data flow methodologies would be applied to this SOS. The communications for the system would be minimized to reduce the power consumption to a minimal level to maintain system persistence in the area of regard.

An application of Event-Trigger Heterogeneous Nonlinear Filter (ET-HNF) and Event-Trigger Particle Filter (ET-PT) were presented in [38] and [37] respectively. For the UAV Swarm SOS application real-time data is required, while also maintaining minimal processing and communication requirements. Although ET-PT results in reduces the transmission when implemented, its application is for bandwidth limited systems. On the other hand, applying the ET-HNF with its master-slave nonlinear filtering algorithm, with generator node and estimation center being the sensing UAV and ground stations respectively, would greatly increase the efficiency of data transmissions within the SOS by both ensuring accuracy and reduced communication burden.

6) MANAGERIAL

Managerial models consist of schedules and budgets. These models provide the timeline of events that must occur to develop the SoS.

The scheduling model will show order of development events, events that can occur in parallel and milestones that must be met to meet the schedule. This model takes the unique schedules of the constituent system into account and plans accordingly, creating an Integrated Master Schedule. Schedule analysis should be completed to account for any uncertainties in the delivery of system along the critical paths. This model will likely be a Gantt chart or PERT chart for managing the complex schedules of the SoS and constituent systems.

Utilizing existing system allows for more accurate cost estimation models than in the standard systems engineering process. The main areas of budget risk lie in the bridging and refactoring of system for SoS application. This model is typically Microsoft Excel or similar. The spreadsheet layout and function tools within the application allow for very complex and high-fidelity models of financial aspects of the system.

V. RECOMMENDATIONS

There are many aspects of the UAV Swarm that still need to be researched and analyzed. These aspects include data modeling and techniques for optimal transmissions in time critical, high volume applications, performance simulations, and emergent behavior prediction. Each of these areas would greatly benefit the UAV Swarm domain with essential knowledge for future development of SoS.

The first and most critical modeling and simulation is emergent behavior prediction. A full, high fidelity model with behaviors, functions, and logic will allow for full simulation of the SoS. The emergent behaviors may not become evident at first, but full analysis of the simulations could provide key insights into the emergent behaviors of the SoS.

Second, a full analysis and simulation of the data processing, flow, and communications requirements. This is a data intensive SoS and the data is a critical element of the SoS. Properly modeling of the data requirements will allow for detailed design and requirement specification for the SoS.

In practice, the emergent behavior, UAV selection, and data processing requirements will be driven by cost and schedule. This will often require commercial off the shelf (COTS) solutions to meet program requirements. In this scenario, the critical element will be the data requirements which drive hardware selection. Within the military development and acquisition cycles, the fielding of hardware is often the driving factor to schedule creep. Utilizing agile software development techniques and the relative ease of upgrading software on fielded systems, physical components, or hardware, become critical infrastructure elements to support the capabilities enabled by software.

Lastly, to ensure the fielded hardware is sufficient for the mission, the requirements for data transfer are critical. The key characteristics of a SoS is the connection and interactions of constituent elements to enable the teaming and coordination of multiple systems as a cohesive unit. The data links between the constituent systems must be fully analyzed and requirements defined for the interfaces that include margin for future growth. This will ensure that the hardware can support the required missions and future capabilities. As with many military developments, often the systems are out of date by the time they are fielded, considering the lifecycle and future capabilities during development can help alleviate issues once fielded.

VI. CONCLUSION

Although the processes related to SoS can be complex, there are key points that should always be addressed. These are management of changes, interfaces, and missions.

The complexity of the systems, their interfaces, governing bodies, constituent schedules, budgets and missions all play a role in the SoS development process. Careful planning and contingency plans should be included for all the critical elements in the SoS.

During the development of the UAV Swarm, the mission of each constituent element must be considered and accounted for in the design of the SoS. Selecting a UAV and Ground Station that is well suited for the application, or is relatively secure in its present situation, is a good choice for implementation into the larger SoS. Alignment of the mission is key to the success of the SoS. The best way to align the missions of the SoS and constituent systems is to give the system organizations ownership of the SoS.

Typical program drivers are cost and schedule, therefore future research should identify considerations for SoS development with specific application of COTS utilization for the constituent elements. There are numerous considerations that must be analyzed and accounted for when developing a SoS from constituent elements that are guided by their own independent system governance. Aspects such as lifecycle, current mission, future missions, and performance will be crucial factors in alternative selection for use in the SoS development.

As described in this research, the development of a SoS includes a complexity that extends well beyond that of typical system development. Therefore, the SoS governance must be taken into consideration for the development of the SoS. Future research in this domain should include the human capabilities, or skills, required to manage these complex system development cycles. As identified in [4] these systems thinkers should be defined by the 7 characteristics; Complexity, Autonomy, Interaction, Change, Ambiguity, Hierarchical View, and Flexibility, and an instrument for characterizing these attributes needs to be developed to ensure adequate system governance.

REFERENCES

[1] S. C. Cook, "On the acquisition of systems of systems," presented at the INCOSE Int. Symp., Jul. 2001.
[2] C. Keating, R. Rogers, R. Unal, D. Dryer, A. Sousa-Poza, R. Safford, W. Peterson, and G. Rabadi, "System of systems engineering," *Eng. Manage. J.*, vol. 15, no. 3, pp. 36–45, 2003.
[3] M. Jamshidi, *Systems of Systems Engineering: Principles and Applications*, 1st ed. Boca Raton, FL, USA: CRC Press, 2009.
[4] R. M. Jaradat, "Complex system governance requires systems thinking–how to find systems thinkers," *Int. J. Syst. Syst. Eng.*, vol. 6, nos. 1–2, pp. 53–70, 2015.
[5] M. W. Maier, "Architecting principles for systems-of-systems," in *INCOSE Int. Symp.*, 1996, Vol. 6, no. 1, pp. 565–573.
[6] W. A. Owens, "The emerging US system-of-systems," Inst. Nat. Strategic Stud., Nat. Defense Univ., Washington, DC, USA, Tech. Rep. 63, 1996.
[7] (2017). *Army Research Lab*. Announcement. [Online]. Available: https://www.arl.army.mil/www/pages/3049/DCIST-Program-Announcement-AmendmentII.pdf
[8] M. Dorigo and M. Birattari, "Swarm intelligence," *Scholarpedia*, vol. 2, no. 9, p. 1462, 2007.
[9] Y. Tan, "Swarm robotics: Collective behavior inspired by nature," *J. Comput. Sci. Syst. Biol.*, vol. 6, p. e106, Oct. 2013.
[10] R. S. Pei, "System of systems integration (SoSI)-A 'SMART' way of acquiring army C412WS systems," in *Proc. Summer Comput. Simulation Conf. Soc. Comput. Simulation Int.*, 2000, pp. 574–579.
[11] A. B. Reed. (2013). *Drone Swarms–Offense and Defense in Depth*. [Online]. Available: http://www.wikistrat.com/op-ed-drone-swarms-offense-and-defense-in-depth/
[12] S. J. Edwards, *Swarming and the Future of Warfare*. RAND Corp, Santa Monica, CA, USA, 2005.
[13] *UAV Cooperative Decision and Control Challenges and Practical Approaches*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2009.
[14] P. Dasgupta, "A multiagent swarming system for distributed automatic target recognition using unmanned aerial vehicles," *IEEE Trans. Syst., Man, Cybern. A, Syst. Hum.*, vol. 38, no. 3, pp. 549–563, May 2008.
[15] S. Waharte, N. Trigoni, and S. Julier, "Coordinated search with a swarm of UAVs," in *Proc. 6th IEEE Annu. Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw. Workshops*, Jun. 2009, pp. 1–3.
[16] Y. Jin, Y. Liao, A. A. Minai, and M. M. Polycarpou, "Balancing search and target response in cooperative unmanned aerial vehicle (UAV) teams," *IEEE Trans. Syst., Man, Cybern., B (Cybern.)*, vol. 36, no. 3, pp. 571–587, Jun. 2005.
[17] R. W. Beard and T. W. McLain, "Multiple UAV cooperative search under collision avoidance and limited range communication constraints," in *Proc. 42nd IEEE Int. Conf. Decis. Control*, Vol. 1, Dec. 2003, pp. 25–30.
[18] J. M. Peschel and R. R. Murphy, "On the human–machine interaction of unmanned aerial system mission specialists," *IEEE Trans. Hum.-Mach. Syst.*, vol. 43, no. 1, pp. 53–62, 2013.
[19] S. J. Edwards, "Swarming on the battlefield: Past, present, and future," Rand Nat. Defense Res. Inst. Santa Monica CA, USA, Tech. Rep. RAND-MR-1100-OSD, 2000.
[20] *APL Technical Digest*, Johns Hopkins, Baltimore, MD, USA, vol. 1, 2006.
[21] P. Rafael and S. L. Heitor, *Theory and New Applications of Swarm Intelligence*, vol. 9. Rijeka, Croatia: InTech, 2012.
[22] W. Zhang, D. Feltner, J. Shirley, M. Swangnetr, and D. Kaber, "Unmanned aerial vehicle control interface design and cognitive workload: A constrained review and research framework," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, 2016, pp. 001821–001826.
[23] The Economist. (2013). *Alone With a Joystick*. The Stressful Lives of Drone Pilots. [Online]. Available: http://www.economist.com/news/united-states/21580461-stressful-lives-drone-pilots-alone-joystick
[24] C. M. Eaton, E. K. P. Chong, and A. A. Maciejewski, "Multiple-scenario unmanned aerial system control: A systems engineering approach and review of existing control methods," *Aerospace*, vol. 3, no. 1, p. 1, 2016.
[25] R. L. Flood and E. R. Carson, *Dealing With Complexity: An Introduction to the Theory and Application of Systems Science*. New York, NY, USA: Springer, 2013.
[26] S. Beer, *The Heart of Enterprise*, Vol. 2. Hoboken, NJ, USA: Wiley, 1979.
[27] J. Lane, J. Dahmann, G. Rebovich, and R. Lowry, "Key system of systems engineering artifacts to guide engineering activities," in presented at the NDIA Syst. Eng. Conf., San Diego CA, USA, Oct. 2010.
[28] C. B. Keating, "Perspective 2 of the SoSE methodology: Designing the unique methodology," *Int. J. Syst. Syst. Eng.*, vol. 2, nos. 2–3, pp. 208–225, 2011.
[29] J. Boardman and B. Sauser, "System of Systems—The meaning of of," in *Proc. IEEE/SMC Int. Conf. Syst. Syst. Eng.*, Apr. 2006, p. 6.
[30] N. Sproles and S. C. Cook, "Synoptic views of defence systems development," in *Proc. SETE SESA ITEA*, Brisbane, QLD, Australia, 2000.
[31] J. A. Lane, and R. Valerdi, "Synthesizing SoS concepts for use in cost modeling," *Syst. Eng.*, vol. 10, no. 4, pp. 297–308, 2007.
[32] S. Wambler. (2013). *Data Modeling*. [Online]. Available: http://www.agiledata.org/essays/dataModeling101.html
[33] Y. Bar-Yam, M. A. Allison, R. Batdorf, H. Chen, H. Generazio, H. Singh, and S. Tucker, *The characteristics and emerging behaviors of system of systems*. Cambridge, MA, USA: NECSI, 2004, pp. 1–16.

[34] A. P. Sage, and C. D. Cuppan, "On the systems engineering and management of systems of systems and federations of systems," *Inf. Knowl. Syst. Manage.*, vol. 2, no. 4, pp. 325–345, 2001.
[35] D. DeLaurentis, "Understanding transportation as a system-of-systems design problem," in *43rd AIAA Aerosp. Sci. Meeting Exhib.*, 2005, p. 123.
[36] A. C. Pinto, L. Magpili, and R. Jaradat, *Operational Risk Management*. New York, NY, USA: Momentum Press, 2015.
[37] X. Liu, L. Li, Z. Li, X. Chen, T. Fernando, H. H.-C. Iu, and G. He, "Event-trigger particle filter for smart grids with limited communication bandwidth infrastructure," *IEEE Trans. Smart Grid*, vol. 9, no. 6, pp. 6918–6928, Nov. 2018.
[38] S. Li, L. Li, Z. Li, X. Chen, T. Fernando, H. H.-C. Iu, G. He, Q. Wang, and X. Liu, "Event-trigger heterogeneous nonlinear filter for wide-area measurement systems in power grid," *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2752–2764, May 2019.
[39] B. Liu, Z. Li, X. Chen, Y. Huang, and X. Liu, "Recognition and vulnerability analysis of key nodes in power grid based on complex network centrality," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 3, pp. 346–350, Mar. 2018.
[40] A. M. Madni and M. Sievers "Model-based systems engineering: Motivation, current status, and research opportunities," *Syst. Eng.*, vol. 21, no. 3, pp. 172–190, 2018.
[41] *Systems Engineering Guide for Systems of Systems*, Department of Defense, Charlottesville, VA, USA, 2008.

CHAD KERR (Member, IEEE) received the B.S. degree in electrical engineering technology from Southern Polytechnic State University, GA, USA, in 2011, and the M.S. degree in applied systems engineering from the Georgia Institute of Technology, GA, USA, in 2015. He is currently pursuing the Ph.D. degree in industrial and systems engineering with Mississippi State University, MS, USA.

He is currently a Research Engineer with the Georgia Tech Research Institute in Atlanta, GA, USA. He is also a Licensed Professional Engineer in the state of Georgia, USA. His research interests include electronic warfare, systems engineering, and multicriteria decision making.

Mr. Kerr is an Active Member of the International Council on Systems Engineering (INCOSE), the Association of Old Crows (AOC), the International Society on Multi-Criteria Decision Making, and the National Society of Professional Engineers.

RAED JARADAT (Member, IEEE) received the Ph.D. degree in engineering management and systems engineering from Old Dominion University, in 2014.

He is currently an Assistant Professor with the Industrial and Systems Engineering Department, Mississippi State University. He is also a Visiting Professor with the Institute for Systems Engineering Research/MSU/U.S. Army Corps of Engineers. His publications appeared in several ranking journals including the IEEE Systems Journal, and the *Computers and Industrial Engineering* Journal. His total awarded projects exceed $ 6.5 M including the National Science Foundation (NSF), the Department of Defense (DOD), industry, and other research laboratories. His main research interests include systems engineering and management systems, systems thinking and complex system exploration, system of systems, virtual reality and complex systems, systems simulation, risk, reliability, and vulnerability in critical infrastructures with applications to diverse fields ranging from the military to industry.

Dr. Jaradat work has been recognized by IISE professional communities. He received three international awards from the 2018 ASEE National Conference, including the Industrial Engineering Division's (IED) Best Paper Award, the New IE Educator Outstanding Paper Award, and the Best Paper for the Professional Interest Council 1 (PIC 1). He is elected as the Director of Networking and Outreach for the Modeling and Simulation (MandS) Board of Directors, The Institute of Industrial and Systems Engineers (IISE). He also serves as a Guest Lead Editor for the IEEE Transactions on Engineering.

NIAMAT ULLAH IBNE HOSSAIN received the B.S. degree in mechanical engineering from the Khulna University of Engineering and Technology and the M.B.A. degree in management information systems from Dhaka University, Bangladesh. He is currently pursuing the Ph.D. degree with the Department of Industrial and Systems Engineering, Mississippi State University.

His publications appeared in different reputed journals such as *Reliability Engineering and System Safety*, *Computer and Industrial Engineering*, *Computational Design and Engineering*, the *International Journal of Critical Infrastructure Protection*, *Engineering Management Journal*, and several conference proceedings and presentations at different academic conferences. He is currently involved in different projects affiliated with the National Science Foundation (NSF), the Department of Defense (DOD), industry, and other research laboratories. His main research interests include systems engineering, systems resilience, systems thinking, and systems simulation.

Mr. Hossain is also recipient of different prestigious awards such as the Gulf Region Intelligent Transportation Society (GRITS) Scholarship, the Best Paper Award in Doctoral Dissertation Category, the J. R. Ron Walsh Best Research Paper Award, the Best Poster Award from the Graduate Student Council Annual Research Symposium, the National Science Foundation (NSF) Travel Grant, the Bagley College of Engineering Travel Grant, and the Graduate Student Travel Assistance Grant (TAGGS). He is also an Active Member of the International Student Advisory Board (ISAB), MSU, the INFORMS Student Chapter, MSU, the American Society of Engineering and Management (ASEM), the Institute of Industrial and Systems Engineers (IISE), and Habitat for Humanity, Starkville, MS, USA.