NPS-AM-26-238

ACQUISITION RESEARCH PROGRAM
SPONSORED REPORT SERIES

Managing the Development and Integration of AI-Based Mission Autonomy in Unmanned Systems: A Programmatic and System Engineering Approach

June 2026

CDR Michael R. Fasano, USN
Kevin J. Silva, CIV

Thesis Advisors: Jeffrey R. Dunlap, Lecturer
CAPT James Lembo, COMSUBGRU TWO

Department of Acquisition, Finance and Manpower

Naval Postgraduate School

Prepared for the Naval Postgraduate School, Monterey, CA 93943

ACQUISITION RESEARCH PROGRAM
DEPARTMENT OF ACQUISITION, FINANCE AND MANPOWER
NAVAL POSTGRADUATE SCHOOL

The research presented in this report was supported by the Acquisition Research Program of the Department of Acquisition, Finance and Manpower at the Naval Postgraduate School.

To request defense acquisition research, to become a research sponsor, or to print additional copies of reports, please contact the Acquisition Research Program (ARP) via email, arp@nps.edu or at 831-656-3793

Acquisition Research Program
Department of Acquisition, Finance and Manpower
Naval Postgraduate School

ABSTRACT

The Department of War (DoW) is aggressively integrating artificial intelligence (AI) into unmanned systems across all domains. However, as autonomy increases, program managers and system engineers face new challenges in verification and trust assurance. The lack of standardized processes for validating AI behaviors within legacy command-and-control architectures complicates risk management and compliance. This project seeks to define a structured framework that enables effective planning, testing, and deployment of AI-enabled autonomy while maintaining system reliability and operator confidence. Traditional requirement frameworks rely heavily on positive requirements that define what the system must do. However, AI-based mission autonomy introduces behaviors that may be emergent, probabilistic, or non-deterministic, requiring a complementary emphasis on negative requirements that define what the system must not do under any conditions. DoW acquisition and verification processes do not yet provide structured methods for developing, managing, and testing these behavior constraints. This gap inhibits safety certification, complicates mission risk assessments, and limits operator trust in autonomous systems.

THIS PAGE INTENTIONALLY LEFT BLANK

ABOUT THE AUTHORS

CMD Michael Fasano is a Submarine Electronics Limited Duty Officer with more than 35 years of naval service spanning undersea warfare, research and development, test and evaluation, acquisition, and operational leadership. A prior enlisted Submarine Electronics Technician, he was commissioned in 2005 and has served in a series of increasingly senior technical and command leadership positions supporting the Navy's most advanced undersea capabilities.

Throughout his career, Commander Fasano has held key leadership assignments including Officer in Charge of the Atlantic Undersea Test and Evaluation Center (AUTEC), Executive Officer of the submarine tender USS EMORY S. LAND (AS-39), Officer in Charge of Commander, Submarine Development Squadron FIVE Detachment Undersea Research and Development, Ocean Systems Department Head at the Naval Research Laboratory, and Project Officer supporting classified Chief of Naval Operations initiatives. His experience spans operational submarine support, undersea systems research and development, test and evaluation, installation command, and fleet modernization efforts.

Commander Fasano currently serves as the Subsea and Seabed Warfare Technical Advisor and N5 at Commander, Submarine Group TWO, where he advises senior Navy leadership on emerging undersea warfare capabilities, autonomous systems, critical undersea infrastructure protection, and subsea operational concepts. He works extensively across Navy, joint, industry, and international partners to advance capabilities supporting the future undersea fight.

Commander Fasano holds a Bachelor of Science degree, a Master of Business Administration, and a Master of Science in Program Management. He is a certified Project Management Professional (PMP) and is currently pursuing an additional graduate degree and certificate focused on autonomous undersea vehicles and defense acquisition.

Mr. Kevin Silva has over 16 years of experience supporting U.S. Navy submarine programs with expertise in systems engineering, combat systems integration,

test and evaluation, acquisition program execution, and technical leadership across the VIRGINIA Class submarine enterprise.

Throughout his career, he has held progressively senior leadership positions, including Test Director, Chief Systems Engineer, Engineering Project Manager, and, most recently, VIRGINIA Class Chief Engineer at the Naval Undersea Warfare Center Division Newport. His experience spans the full lifecycle of submarine capability development, from system design and integration through laboratory, dockside, and at-sea testing, certification, and fleet delivery of advanced undersea warfare capabilities. Mr. Silva has led multidisciplinary government and industry teams responsible for whole-ship integration, including payload and subsea/seabed warfare capability integration, combat systems certification, and developmental test and evaluation. He possesses extensive test and evaluation experience at sea, including planning and executing platform-level Weapon System Accuracy Trials, VIRGINIA Warfare Materiel Certification Program events, and follow-on developmental testing supporting mission certification and combat readiness across multiple VIRGINIA Class submarines.

Most notably, Mr. Silva served as the VIRGINIA Class Test Director responsible for strike mission certification associated with the VIRGINIA Payload Tube, leading the planning and execution of testing that culminated in the first Tomahawk missile launch from a Block III platform. The capability was later employed during real-world combat strike operations, directly demonstrating the operational impact of the systems and testing efforts he led. Throughout his career, Mr. Silva has worked extensively with NAVSEA program offices, General Dynamics Electric Boat, and cross-functional government and industry stakeholders to manage technical risk, resolve complex integration challenges, and support modernization and delivery of critical undersea warfare capabilities.

He holds a Bachelor of Science in Mechanical Engineering from the University of Massachusetts Dartmouth and a Master of Engineering in Mechanical Engineering from Rensselaer Polytechnic Institute. He is also a certified Project Management Professional.

NPS-AM-26-238

ACQUISITION RESEARCH PROGRAM SPONSORED REPORT SERIES

Managing the Development and Integration of AI-based Mission Autonomy in Unmanned Systems: A Programmatic and System Engineering Approach

June 2026

CDR Michael R. Fasano, USN
Kevin J. Silva, CIV

Thesis Advisors: Jeffrey R. Dunlap, Lecturer
James Lembo, COMSUBGRU TWO

Department of Acquisition, Finance and Manpower

Naval Postgraduate School

Prepared for the Naval Postgraduate School, Monterey, CA 93943

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AI	Artificial Intelligence
DAU	Defense Acquisition University
DoW	Department of War
ISR	Intelligence, Surveillance, and Reconnaissance
MBSE	Model Based Systems Engineering
MOSA	Modular Open Systems Architecture
NPS	Naval Postgraduate School
PHM	Prognostics and Health Management
T&E	Test and Evaluation
UUV	Unmanned Undersea Vehicle

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Artificial intelligence–enabled mission autonomy is becoming central to the Department of War's modernization of unmanned systems, particularly in the undersea domain. Unmanned undersea vehicles are expected to operate for extended periods in contested environments, often beyond reliable communication with human operators. These operational realities demand increasing levels of autonomy, enabling systems to interpret sensor data, make decisions, and execute mission tasks independently. However, as autonomy increases, traditional acquisition, program management, and systems engineering practices face significant challenges in verifying behavior, managing risk, and maintaining operator trust. Existing frameworks were designed for deterministic systems whose behavior could be fully specified and tested in advance, whereas artificial intelligence (AI) based autonomy introduces probabilistic and emergent behavior that cannot be exhaustively predicted.

This thesis addresses a central gap in current defense acquisition practice: the absence of structured mechanisms for defining, managing, and validating behavioral constraints for AI-based mission autonomy. Traditional requirements engineering emphasizes positive requirements that describe what a system must do. While necessary, this approach alone is insufficient for autonomous systems whose behavior may evolve through machine learning and environmental interaction. The research argues that autonomy must also be governed by explicit negative requirements that define what a system must not do under any conditions. These constraints establish safety boundaries, prohibited behaviors, and abort conditions that limit autonomy within acceptable operational, legal, and ethical parameters. Without a systematic approach to developing and verifying these constraints, programs risk certification ambiguity, integration challenges, and diminished operator confidence.

The primary research question guiding this study is how program management and systems engineering principles can be combined to manage AI-based mission autonomy within the defense acquisition process. Secondary questions address how negative requirements can be structured and traced, how acquisition pathways can

support iterative autonomy development, and how verification strategies can ensure transparency and trust in autonomous decision making. The research focuses on unmanned undersea vehicle programs as a representative context, given the unique challenges of limited communication, environmental uncertainty, and long mission durations that characterize undersea operations.

The study employs a qualitative, defense-focused managerial inquiry methodology. It synthesizes insights from professional literature, government reports, acquisition guidance, and case analyses related to unmanned systems, artificial intelligence, and defense program management. Rather than testing specific algorithms or producing quantitative performance estimates, the research develops a conceptual and practical framework designed to support program managers and system engineers in planning, requirements development, risk management, and verification of AI-enabled autonomy.

Analysis of the literature and case material reveals several persistent challenges. First, traditional program management practices struggle to accommodate adaptive systems whose behavior evolves through data updates and retraining cycles. Second, systems engineering methods designed for deterministic systems do not adequately address verification of emergent autonomy behaviors. Third, acquisition pathways often lack mechanisms for managing autonomy updates, configuration control, and governance throughout the life cycle. Finally, ethical governance and operator trust remain fragile due to insufficient transparency and unclear behavioral constraints.

Across these challenges, the absence of structured negative requirements emerges as a central deficiency. Negative requirements are often acknowledged implicitly in safety analyses and policy guidance, but they are rarely treated as formal requirement artifacts within acquisition frameworks. This omission limits the ability of program managers and system engineers to bound autonomous behavior, develop targeted verification strategies, and maintain accountability as systems evolve.

To address this gap, the thesis proposes an integrated life-cycle framework for managing AI-based mission autonomy. The framework combines program management, systems engineering, modular open systems architecture, and governance principles into

a cohesive model. A central feature of the framework is a containerized autonomy architecture that separates mission autonomy functions from a dedicated governance component referred to as a sidecar container. This governance element enforces invariant behavioral constraints derived from negative requirements, ensuring that autonomous actions remain within predefined boundaries regardless of changes to mission algorithms.

The framework emphasizes that negative requirements must be treated as first-class artifacts throughout the life-cycle. They should be derived from operational hazards and policy constraints, traced to system architectures and interfaces, and verified through scenario-based testing and digital engineering environments. Iterative autonomy updates must be revalidated against these invariant constraints to prevent configuration drift and maintain certification assurance. By stabilizing behavioral boundaries, the framework enables programs to support continuous improvement in autonomy while preserving safety and accountability.

Application of the framework to a representative undersea autonomy program demonstrates its practical implications. The framework supports early identification of autonomy risks, improves traceability between operational hazards and system constraints, and provides structured decision points for program managers throughout development and sustainment phases. It also enables more transparent communication of system limitations to operators and commanders, strengthening trust in autonomous capabilities.

The findings of this research indicate that effective management of AI-based mission autonomy requires deliberate integration of managerial oversight mechanisms with enforceable technical architecture. Negative requirements provide a unifying construct that links program management, systems engineering, acquisition governance, and ethical oversight. When treated as explicit and traceable artifacts, they enable programs to bound autonomy behavior, improve verification and validation practices, and sustain operator confidence.

The study concludes that incremental adaptation of existing acquisition processes is insufficient to manage the risks of autonomy. Instead, programs must adopt structured frameworks that explicitly incorporate the definition of behavioral constraints, life-cycle

governance, and architecture-based enforcement mechanisms. Such approaches will be essential for delivering safe, reliable, and mission-ready autonomous systems capable of operating in contested environments.

Future research should empirically validate the proposed framework through operational case studies, assess its applicability to other unmanned domains, and investigate quantitative methods for measuring autonomy assurance and trust. Continued integration of digital engineering and autonomous system governance will be critical as defense organizations increasingly rely on AI-enabled mission autonomy.

AI DISCLOSURE

To support the development of this thesis, we used OpenAI's ChatGPT as a generative artificial intelligence (AI) tool to assist with aspects of the writing and refinement process. Use of this tool was conducted in accordance with Naval Postgraduate School guidance and with the understanding that generative AI serves as a support mechanism rather than a substitute for original authorship, analysis, or judgment.

ChatGPT was used at multiple stages of manuscript development to assist with refining organization, improving clarity and concision, strengthening topic sentences, and enhancing the overall flow of written sections. It was also used to help translate technical and conceptual ideas into clearer academic language and to provide structural recommendations for chapters, including the integration of program management and systems engineering concepts. In limited cases, the tool was used to generate draft language based on author-provided inputs, which was subsequently reviewed, modified, and validated prior to inclusion in the final manuscript.

The use of generative AI introduced potential risks, including the possibility of incorporating language that did not fully reflect the authors' intent, introducing inaccuracies, or over-reliance on automated suggestions. To mitigate these risks, all AI-generated or AI-assisted content was critically reviewed, edited, and validated by the authors to ensure accuracy, alignment with source material, and consistency with the thesis's analytical arguments. All technical claims, references, and conclusions were independently verified using primary sources and authoritative literature. The authors maintained full responsibility for the content, analysis, and conclusions presented in this work.

This use of generative AI was limited to supporting writing quality and structural refinement. All research design, analytical development, framework construction, and substantive conclusions were independently developed by the authors.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND AND PROBLEM STATEMENT

The Department of War (DoW) is investing heavily in artificial intelligence (AI) and unmanned systems across all domains. The U.S. Navy views unmanned undersea vehicles (UUVs) as a core element of future distributed maritime operations and hybrid fleet concepts, where manned and unmanned platforms work together to increase reach, survivability, and persistence in contested environments (Button et al., 2009; O'Rourke, 2025). UUVs can perform dull, dirty, or dangerous missions, operate in areas that are risky for crewed vessels, and extend undersea presence at a lower marginal cost than additional manned platforms can. They also enable greater persistence by conducting long-duration surveillance and data-collection missions without the logistical constraints of human crews. Moreover, their modular architectures allow rapid integration of new sensors, autonomy stacks, and payloads to meet evolving operational demands.

At the same time, adversaries are expanding their anti-access and area-denial capabilities, undersea surveillance networks, and seabed infrastructure. These trends increase the demand for undersea systems that can operate for long periods with limited communication, navigate in complex environments, and adapt to unexpected events. Meeting this demand requires more than preprogrammed automation. It requires mission autonomy, the ability of an unmanned system to interpret sensor data, evaluate options, and select actions that advance commander intent without continuous human control. Studies across domains, from unmanned aerial vehicles to unmanned surface vessels, show that demands for autonomy grow as missions become longer, more complex, and more integrated with joint operations (Bessemer, 2006; Fahey, 2016; Geiss, 2019).

Advances in AI, digital engineering, and modular open systems architectures (MOSAs) create new opportunities to field mission autonomy. High-tech defense industries are developing more capable autonomous intelligent systems, while research on intelligent unmanned systems and AI-enabled navigation illustrates how far technology has progressed (Joshi et al., 2024; Reis et al., 2021; Zhang et al., 2017). However, practical experience and analytic work show that autonomy programs often

stall between prototype and fleet adoption. Programs underestimate the costs of model governance and cyber hardening, struggle to integrate with legacy command-and-control architectures, and face persistent challenges in verification, validation, and safety assurance (Calfee, 2021; Martin et al., 2019; Vandenberg, 2010). These gaps create uncertainty for program managers and system engineers, who must balance technical innovation with mission assurance under constrained budgets and schedules to deliver capability at the speed of relevance.

A central difficulty is that AI-based mission autonomy introduces emergent, probabilistic behaviors rather than fixed, deterministic ones. Traditional requirement frameworks rely on positive requirements that describe what a system shall do. For autonomy, it is equally important to state what the system shall not do across a range of conditions. These negative requirements define forbidden actions, safety boundaries, abort conditions, and behavior constraints. They are essential in undersea systems, where communication is delayed or intermittent and operators cannot intervene quickly if a model behaves unexpectedly. Berzins (2021) emphasizes that weapon system safety in the presence of AI requires robust behavioral constraints, clear safety cases, and explicit handling of failure modes. Structuring both positive and negative requirements within a traceable hierarchy enables programs to design targeted tests that evaluate how autonomy responds under normal and degraded conditions. The systems engineering linkage between requirements and the objective quality evidence obtained from testing is essential to demonstrate what the system does and how reliably it avoids unsafe or unintended behaviors. Programs must create a test and evaluation master plan that builds operational confidence in the autonomous system's ability to perform as expected in real-world missions.

Current acquisition and verification processes, however, provide limited structured guidance for expressing, managing, and testing negative requirements for AI-enabled mission autonomy. DoW policy has started to address AI risk and governance. The DoW data, analytics, and AI adoption strategy calls for reliable, governable, and traceable AI, while responsible AI principles stress human judgment, accountability, and transparency (Department of Defense, 2023; Ray et al., 2020). Yet these documents do not provide program managers and system engineers with detailed, repeatable processes

for translating those principles into requirement sets, test criteria, and management practices for specific unmanned programs, particularly in the undersea domain.

The problem this project addresses is that program managers and system engineers lack a structured, practical framework to manage AI-based mission autonomy in UUV programs within the DoW acquisition process. Existing work describes technical enablers, individual program experiences, and policy aspirations. Still, it does not integrate program management, systems engineering, and ethical governance into a repeatable development model that treats negative requirements as first-class artifacts. Without such a model, programs risk fragmented requirement practices, brittle autonomy solutions, and reduced operator confidence in AI-enabled UUVs. This gap forces program teams to rely on ad hoc decision-making, leading to inconsistent verification strategies and unclear accountability for autonomous behavior. It also complicates milestone planning and resource allocation, since program managers lack a structured way to scope, track, and validate autonomy-related risks through development. Ultimately, this absence of a unified framework undermines the ability of system engineers and program managers to deliver mission-ready, trustworthy autonomy that meets operational and safety expectations. This gap jeopardizes overall program success and increases the likelihood that senior leadership will lose confidence in the program's viability. In extreme cases, it may contribute to premature program cancellation and delay delivery of a next-generation capability to the fleet.

While numerous technical, ethical, and organizational challenges accompany the integration of AI into unmanned systems, the central deficiency addressed by this research is the absence of acquisition-grade mechanisms for defining, enforcing, and validating behavioral constraints as autonomy increases. Existing requirement frameworks emphasize what systems must do yet provide limited structure for specifying prohibited behaviors, abort conditions, or safety boundaries for non-deterministic decision-making. This gap complicates verification, undermines certification confidence, and erodes operator trust, particularly in environments where human oversight is delayed or unavailable.

B. RESEARCH QUESTIONS

This project is guided by one primary research question and several secondary questions that address practical decision-making problems at the intersection of acquisition, systems engineering, and governance.

The primary research question is

1. How can program management and systems engineering principles be combined to manage AI-based mission autonomy in unmanned systems within the DoW acquisition process?

The secondary research questions are

2. What mechanisms, particularly the structured development of negative requirements and associated test and evaluation criteria, best ensure validation, transparency, and operator trust in AI decision-making?
3. How should autonomy-specific requirements, including both positive and negative requirements, be defined, decomposed, and traced to enable effective integration, verification, and validation of AI-enabled mission autonomy?
4. How can acquisition pathways be adapted to support iterative AI development and testing without reducing oversight, accountability, and risk management?

Although the insights may be relevant to unmanned systems in other domains, the analysis focuses on DoW UUV programs as the primary context.

C. METHODOLOGY

This project uses a qualitative, defense-focused managerial inquiry design. The aim is to develop a conceptual and practical framework rather than to test a specific algorithm or produce quantitative performance estimates. The methodology includes a review of professional literature, technical reports, and acquisition guidance related to AI-enabled autonomy in unmanned systems. From this body of work, the project identifies common threads, recurring themes, and systemic challenges that program managers and system engineers encounter. It then analyzes gaps where current research, policy, or practice fails to provide actionable guidance for structuring and governing mission autonomy. Using these insights, the project synthesizes available information, policy direction, and best practices into an executable framework tailored to the needs of defense acquisition programs. The outcome is a practical toolset that can be handed directly to a program manager to support planning, requirements development, risk management, and verification activities for AI-enabled UUV autonomy.

To ground this methodical approach, the study begins with a structured literature review of government reports, Naval Postgraduate School (NPS) theses, technical standards, and peer-reviewed articles relevant to AI, autonomy, unmanned systems, systems engineering, program management, and ethical governance. Sources include RAND analyses of autonomy and missions for UUVs, Congressional Research Service reports on large unmanned vehicles, NPS capstone and thesis work on undersea systems and open architectures, and journal articles on intelligent unmanned systems and AI safety (Berzins, 2021; Button et al., 2009; Carr et al., 2018; Martin et al., 2019; O'Rourke, 2025; Reis et al., 2021; Zhang et al., 2017). Documents are examined for themes related to risk, requirement practices, negative requirement development, verification and validation, and acquisition strategies.

Next, the project uses comparative case analysis drawing from published case studies and technical narratives. Examples include NPS theses on unmanned surface and undersea vehicles, analyses of manpower impacts from unmanned aerial vehicles on submarine platforms, and studies of modular open systems in defense systems (Fahey, 2016; Futch, 2012; Geiss, 2019; Johnson & Halbert, 2024; Vandenberg, 2010; Wilson, 2017). Rather than conducting a single, deep case study, the analysis examines multiple cases to identify recurring patterns, tensions, and decision points related to autonomy requirement development, architecture choices, and test strategies.

The findings from the literature review and case analysis are synthesized into a risk-informed framework for managing AI-based mission autonomy in UUV programs. The framework identifies key phases, decision gates, and feedback loops in the development stages of acquisition, providing a structured roadmap for integrating AI-based mission autonomy into UUV programs. It specifies where and how both positive and negative requirements should be developed, traced, and validated, ensuring that autonomous behaviors are bounded, safe, and aligned with operational objectives. It highlights the role of program managers and system engineers in coordinating requirement development, risk assessment, and verification activities, fostering shared accountability and informed decision-making. It also incorporates principles of digital engineering to enable flexible integration of autonomy components, facilitate testing, and support rapid updates as technologies evolve. Ethical and governance considerations are

embedded throughout, guiding the handling of failure modes, human oversight, and compliance with policy and safety standards.

D. LIMITATIONS AND SCOPE

Several limitations define the scope of this project. The study concentrates on the DoW UUV programs and related undersea autonomous systems. Many concepts may apply to unmanned surface or aerial systems. Still, the analysis is tailored to the undersea environment, where energy constraints, acoustic communication limits, and navigation uncertainty place unique demands on mission autonomy (Carr et al., 2018; Martin et al., 2019).

The research relies on unclassified sources, including public reports, technical papers, and theses. Classified or proprietary data is not used. As a result, some details of operational performance, classified requirements, or internal program deliberations may be unavailable. Case material is built from open-source descriptions, which may contain gaps and not reflect all internal dynamics.

The project is conceptual and qualitative. It does not design or test specific AI models or provide numeric estimates of cost, reliability, or mission effectiveness. Instead, it seeks to integrate existing knowledge into a coherent framework that managers and engineers can use to think more clearly about autonomy risk, requirement practices, and integration choices.

The emphasis is on mission autonomy and associated decision functions. Guidance, navigation, and control algorithms are incorporated only to the extent that they interact with mission-level behaviors and constraints. Finally, the work is bound by the thesis timetable, which limits the number of cases and interviews that can be included and constrains the extent of empirical validation for the proposed framework.

E. ORGANIZATION OF THE PROJECT

The project is organized into seven chapters, moving from context and background to analysis and recommendations.

Chapter I introduces the research topic, presents the problem statement, lays out the research questions, summarizes the methodology, notes the limitations and scope, and previews the organization of the study.

Chapter II provides background on the evolution of UUVs, the rise of AI-based mission autonomy, and the DoW acquisition environment. It introduces the theoretical and conceptual foundations that underpin the analysis, including systems engineering, modular open systems, responsible AI, and the role of negative requirements.

Chapter III presents a detailed literature review that examines programmatic challenges, systems engineering integration and verification, MOSA, undersea and seabed warfare applications, and ethical, governance, and operator trust considerations. It identifies key gaps in current knowledge and practice, including the lack of structured guidance on autonomy-specific negative requirements and life-cycle management.

Chapter IV analyzes how program management and systems engineering currently function in AI-related acquisition efforts. It draws on the previously examined literature and cases to explore how requirement practices, test strategies, and acquisition pathways either support or hinder responsible mission autonomy.

Chapter V presents an integrated life-cycle framework for the development and integration of AI-based mission autonomy. The framework begins by addressing key program management principles, including responsible AI, acquisition strategy, and risk management. It then examines systems engineering principles, with a focus on the definition of negative requirements and their associated verification and validation strategies.

Chapter VI illustrates how the framework could be applied within the DoW acquisition pathways using a case-based application or scenario. It examines how the framework might reshape planning, requirement development, and risk management in a representative undersea autonomy program.

Chapter VII concludes the project by summarizing key findings, discussing implications for policy and practice, and providing recommendations to program

managers, systems engineers, and senior leaders. It also identifies directions for future research on autonomy and negative requirement practices.

II. BACKGROUND

Unmanned systems have become central to modern military concepts. The Navy's vision for a hybrid fleet anticipates a mix of manned and unmanned platforms across surface, undersea, and aerial domains to enhance sensing, increase the number of available weapons, and distribute risk across a wider set of assets (Galdorisi, 2017; O'Rourke, 2025). Within the undersea domain, UUVs are organized into size-based categories that reflect their launch methods, endurance, and mission roles. Small UUVs are man-portable and support short-duration tasks such as harbor surveys, inspection missions, and explosive ordnance disposal. Medium UUVs, roughly the size of heavyweight torpedoes, can be launched and recovered from submarines, often through torpedo tubes or dry deck shelters, and can perform extended intelligence, surveillance, and reconnaissance (ISR) patrols. Large and extra-large vehicles are pier-launched and carry substantial payloads for mine warfare, seabed operations, and persistent sensing (O'Rourke, 2025; Small, 2020).

RAND's survey of UUV missions catalogs a wide range of potential roles, including mine countermeasures, deployment of leave-behind sensors, harbor and near-shore monitoring, oceanographic data collection, undersea infrastructure inspection, and anti-submarine warfare tracking (Button et al., 2009). These missions often take place in hazardous or contested areas and can demand long-endurance, low-observable operations and resilience in complex environments. Undersea and seabed warfare studies highlight that extra-large UUVs may support persistent seabed operations and infrastructure defense but also face tight energy budgets and navigation uncertainty (Carr et al., 2018).

Undersea conditions amplify the need for autonomy. Acoustic communications are bandwidth-limited and intermittent, and satellite links are unavailable at depth. Environmental factors such as variable sound speed profiles, cluttered seabeds, and uncertain bathymetry complicate planning and sensing. Martin et al. (2019) note that these constraints make it challenging to rely on continuous human control, since operators cannot supervise every decision in real time. As missions and fleets grow, mission autonomy becomes a requirement rather than an option.

Research in related domains reinforces this picture. Studies of unmanned aerial and surface systems show similar pressures for increased autonomy as mission demands expand and manpower constraints tighten (Bessemer, 2006; Fahey, 2016; Futch, 2012; Geiss, 2019). Work on intelligent unmanned systems, AI-enabled navigation, and autonomous intelligent systems describes how AI is being applied to perception, planning, and coordination across different platforms and environments (Joshi et al., 2024; Reis et al., 2021; Zhang et al., 2017). At the same time, operational experience in submarine programs and UUV support shows that autonomy affects manning, maintainability, and crew workloads, and can alter how platforms are employed (Futch, 2012; Vandenberg, 2010).

In parallel, policy and governance frameworks are evolving. The DoW data, analytics, and AI adoption strategy and responsible AI principles emphasize reliability, governability, and warfighter trust (Department of Defense, 2023; Ray et al., 2020). Product support guidance for unmanned systems highlights the need to treat autonomy updates, data labeling, verification, and retraining as recurring sustainment activities rather than one-time events (Defense Acquisition University, 2024). Ethical analyses of autonomous weapon systems raise concerns about accountability and human control, which also apply to the undersea mission autonomy context (Etzioni & Etzioni, 2017; Taddeo & Blanchard, 2022; Trusilo, 2023).

Together, these trends create both an opportunity and a challenge. AI-enabled mission autonomy can expand the Navy's ability to operate in contested undersea and seabed environments, but only if programs can manage the associated technical, programmatic, and ethical risks systematically. The potential benefits include increased persistence, reduced risk to personnel, and enhanced mission flexibility across a range of operational scenarios. At the same time, the complexity of integrating autonomous behaviors into existing platforms introduces new vulnerabilities and coordination demands. Program managers must balance innovation with rigorous oversight, ensuring that autonomy does not outpace verification and safety processes. System engineers must translate high-level operational goals into requirements that are testable, traceable, and resilient under uncertainty. Ethical governance considerations further complicate development, particularly when autonomy may influence mission-critical decisions. As a

result, realizing the full promise of AI-enabled undersea autonomy requires a disciplined approach that unites technical design, acquisition planning, and responsible oversight.

A. THEORETICAL FRAMEWORK

Several theoretical perspectives provide structure for this study and help relate technical and managerial decisions to autonomy outcomes.

The first perspective is systems engineering. Systems engineering views capabilities as integrated systems that include hardware, software, data, human operators, support infrastructure, and organizational processes. It emphasizes requirements analysis, architecture design, interface control, verification and validation, and configuration management across the life cycle. For AI-based mission autonomy, systems engineering principles indicate that AI models, training data, and inference pipelines must be treated as system elements with defined requirements, interfaces, and tests, rather than as opaque black boxes (Brutzman et al., 2013; Schafer, 2009; Woudenberg et al., 2020). This lens supports the idea that both positive and negative requirements for autonomy should be engineered and verified, as with other critical components.

The second perspective is modular open systems architecture. MOSA promotes the use of open standards, modular components, and well-defined interfaces that allow for competition, upgrades, and reuse over time. Defense policy now expects programs to apply MOSA where feasible, and research shows that open architectures can reduce life-cycle costs and vendor lock-in when implemented properly (Johnson & Halbert, 2024; Patni, 2020; Radoman et al., 2025; Wilson, 2017). For UUV autonomy, MOSA suggests that mission autonomy software, sensor suites, and control interfaces should be modular and accessible enough to enable models to be retrained or replaced without redesigning entire platforms or losing visibility into behavior.

A third perspective comes from responsible AI and socio-technical systems theory. Autonomy interacts with human organizations, legal frameworks, and cultural norms. Analyses of autonomous weapon systems and defense AI policy highlight questions of accountability, human control, transparency, and moral responsibility (Berzins, 2021; Etzioni & Etzioni, 2017; Taddeo & Blanchard, 2022). The DoW's responsible AI principles and adoption strategy call for traceable, reliable, and

governable AI supported by clear data practices and oversight (Department of Defense, 2023; Ray et al., 2020). This perspective reinforces the need for explicit negative requirements and behavioral constraints that align with legal and ethical boundaries.

Reliability engineering and prognostics and health management (PHM) provide a fourth perspective. PHM approaches focus on predicting and managing degradation over time through sensing, modeling, and maintenance planning. When combined with AI-based anomaly detection, PHM can help identify abnormal behavior in sensors, actuators, or autonomous software before it causes mission failure (Di Lorenzo & Bayer, 2021; Whelan et al., 2022). This perspective supports the idea that mission autonomy should include monitoring functions, safe defaults, and abort conditions that operate even when communication with operators is limited.

These theoretical perspectives collectively inform the design of a risk-informed life-cycle framework. Systems engineering and MOSA provide structure for requirements and architecture decisions, responsible AI and socio-technical theory shape governance and trust considerations, and reliability and PHM inform sustainment and monitoring strategies for mission autonomy in UUV programs.

Research in systems engineering and artificial intelligence further reinforces the need for constraint-oriented approaches to autonomy design. Seshia et al. (2022) argue that traditional verification techniques are insufficient for learning-enabled systems whose behavior may evolve over time, requiring assurance methods that bound acceptable behavior rather than attempt exhaustive correctness proofs. Similarly, research on self-adaptive systems emphasizes constraint satisfaction and runtime monitoring as foundational assurance mechanisms when system behavior cannot be fully predicted a priori (Cheng et al., 2009). In unmanned undersea systems, where environmental uncertainty and communication latency limit supervisory control, verification through behavioral bounding becomes a necessary design principle rather than a supplemental safeguard.

B. KEY CONCEPTS AND NEGATIVE REQUIREMENTS

To provide a clear foundation for the analysis, several key concepts used in this study require concise definition.

Mission autonomy is the capability of an unmanned system to plan, select and execute mission actions to achieve assigned objectives within defined constraints, and without continuous human control. It involves perceiving the environment, evaluating options, and choosing actions that support commander intent under uncertainty.

AI-based mission autonomy refers to mission autonomy functions that rely on AI techniques such as machine learning, pattern recognition, and advanced decision-making algorithms. Examples include automatic mine recognition, contact classification in anti-submarine warfare, adaptive search planning, and dynamic path planning for seabed surveys (Button et al., 2009; Martin et al., 2019; Reis et al., 2021; Zhang et al., 2017). AI-based autonomy is shaped by the training data and parameters used to develop models, and behavior can change when models are retrained or when they encounter new environments.

Positive requirements are requirement statements that describe what a system shall do. They define desired capabilities, performance thresholds, and functional behaviors, such as probabilities of detection, classification accuracy, or navigation precision. Positive requirements remain essential for specifying mission goals and technical performance.

Negative requirements are requirement statements that describe what a system shall not do under any conditions. For AI-based mission autonomy, negative requirements can include forbidden target types, prohibited maneuvers near sensitive infrastructure, constraints on entering specific geographic regions without authorization, and conditions that trigger mission abort and safe recovery. Work on safety, ethics, and runtime policy enforcement suggests that these constraints are critical to bounding emergent behavior, shaping acceptable risk, and supporting safety certification and operator trust (Berzins, 2021; Brutzman et al., 2013; Etzioni & Etzioni, 2017; Trusilo, 2023).

Life-cycle management covers concept development, requirements definition, design and integration, test and evaluation (T&E), fielding, sustainment, upgrades, and retirement. For AI-based mission autonomy, life-cycle management must also address data collection and labeling, model training and retraining, verification and validation of

new models, cyber accreditation, configuration control of software baselines, and product support after initial fielding (Calfee, 2021; Defense Acquisition University, 2024; Ray et al., 2020). This project focuses specifically on the early stages of this life cycle, where foundational decisions have the most significant influence on long-term system performance and risk. It emphasizes the period from initial development through structured test and evaluation activities. By concentrating on these early phases, the project aims to strengthen the requirement practices and verification strategies that underpin trustworthy AI-enabled mission autonomy.

The project's conceptual approach centers on a risk-informed framework that links positive and negative requirements to architectures, test strategies, and program management decisions across phases. The framework assumes that if negative requirements are developed in parallel with positive ones, treated as explicit design objects, and traced to interfaces, test cases, and monitoring functions, programs can better manage autonomy risk, preserve safety boundaries, and maintain operator confidence in UUV mission autonomy. By embedding these requirements early, programs can shape design choices and ensure that verification plans reflect real operational hazards. This structure also gives program managers clearer decision points for assessing technical maturity and readiness during milestone reviews. Ultimately, this approach creates a disciplined pathway for integrating AI-based mission autonomy in a manner that is transparent, testable, and aligned with the Navy's operational and safety expectations.

C. PREVIOUS RESEARCH AND RATIONALE

Existing research provides important building blocks for this project and reveals gaps that justify further work. Strategic and operational analyses highlight the potential of UUVs for distributed maritime operations and for maintaining a persistent undersea presence. RAND's survey of UUV missions outlines a broad set of mission types and identifies technical and operational considerations for each (Button et al., 2009). The Congressional Research Service describes the Navy's plans for large, unmanned surface and undersea vehicles, summarizing both the promise of distributed architectures and congressional concerns about cost, technical risk, and concept of operations (O'Rourke, 2025). Seabed warfare studies focus on extra-large UUVs and emphasize energy

constraints, mission planning challenges, and the importance of undersea infrastructure, all of which influence autonomy design (Carr et al., 2018).

Technical and systems engineering research on unmanned systems covers architecture, interoperability, and testing. Blais (2016) analyzes unmanned system interoperability standards and argues that common control and simulation interfaces are essential for affordable development and test of autonomy across platforms. Fahey (2016) examines software architecture for unmanned surface vehicles for anti-submarine warfare and shows how architectural choices affect integration and capabilities. Geiss (2019) studies the employment of unmanned surface vessels in distributed maritime operations and highlights integration, control, and employment concepts. Schafer (2009) surveys AI and smart sensor networks in network-centric environments, showing how AI and sensing architectures interact. Woudenberg et al. (2020) discuss systems engineering frameworks for manned–unmanned teaming and emphasize that architecture and interface decisions drive flexibility and testability.

Program- and acquisition-focused studies identify recurring challenges in fielding autonomy. Martin et al. (2019) documents that autonomy programs in unmanned maritime vehicles often face hurdles when integrating with existing combat systems and delivering reliable performance in real ocean conditions. Calfee (2021) notes that programs frequently underestimate the costs associated with model governance, cybersecurity, and AI verification. Johnson and Halbert (2024) and Wilson (2017) illustrate how modular open systems approaches can increase flexibility and competition but also show that MOSA requires deliberate investment in interface documentation, configuration management, and supporting contracts. Radoman et al. (2025) analyzes open architecture in military systems from a systemic perspective and highlight the importance of data rights and governance for sustaining openness. Defense product support guidance for unmanned systems stresses that autonomy-related activities, including data labeling, retraining, and verification, must be treated as recurring logistics and sustainment tasks (Defense Acquisition University, 2024).

Ethics, safety, and governance research examines accountability, human control, and emergent behavior. Etzioni and Etzioni (2017) discuss the moral and legal challenges

associated with autonomous weapons and call for clear accountability structures. Taddeo and Blanchard (2022) compare definitions of autonomous weapon systems and show how imprecise terms can complicate governance. Berzins (2021) analyzes weapon systems safety when deploying AI technology and argues for rigorous safety cases and a structured treatment of AI-specific hazards. Trusilo (2023) explores emergent behavior and its impact on predictability and reliability in autonomous AI systems in conflict. The DoW AI adoption strategy and responsible AI principles call for traceability, reliability, and human judgment in the loop, and Ray et al. (2020) describe how AI and autonomous systems can be applied across joint functions in ways that retain oversight and control (Department of Defense, 2023).

Research on intelligent unmanned systems and AI-enabled navigation demonstrates the breadth of AI applications to autonomous systems. Zhang et al. (2017) review trends in intelligent unmanned autonomous systems and highlight key technologies and challenges. Reis et al. (2021) discuss autonomous intelligent systems in high-tech defense industries. Cheng et al. (2009) describe the evolution from unmanned systems to autonomous intelligent systems. Joshi et al. (2024) examine AI-enabled navigation and decision-making for drones in defense and security, showing how similar autonomy challenges arise in other domains. These works reinforce that AI-based autonomy introduces new requirements, tests, and governance demands that extend beyond any single platform type. They demonstrate that issues such as negative requirement identification, model validation, and safe behavior bounding are cross-cutting concerns across air, surface, and undersea systems. They also show that autonomy depends heavily on data quality, model retraining processes, and cyber protection, regardless of the operational environment. This body of research underscores the need for a structured, domain-agnostic framework that programs can adapt to the specific demands of UUV mission autonomy.

Work on reliability, PHM, and intrusion detection illustrates how AI can support both mission execution and system health. Di Lorenzo and Bayer (2021) present a PHM system for an unmanned combat aircraft, showing how predictive maintenance can improve readiness. Whelan et al. (2022) explore AI for intrusion detection systems in unmanned aerial vehicles, which has implications for cyber protection of UUVs and their

autonomy software. These examples demonstrate that autonomous undersea systems must be resilient not only in their mission behaviors but also in their internal health monitoring and cyber defenses. They highlight the need for integrated approaches where autonomy, reliability engineering, and cybersecurity are developed in concert rather than in isolation and then "bolted on." Such cross-disciplinary considerations are essential for ensuring that AI-enabled UUVs remain safe, dependable, and survivable in contested operational environments.

Taken together, the literature suggests that success in undersea autonomy depends on more than advanced algorithms. It depends on requirement practices, architecture decisions, verification and validation strategies, and ethical and governance mechanisms that operate together. At the same time, current work neither provides a structured, repeatable process for developing and managing negative requirements for mission autonomy, nor offers a consolidated life cycle framework that links systems engineering, program management, MOSA, and responsible AI principles in the specific context of UUV programs. Those gaps motivate this study and support the need for a risk-informed framework that can help program managers and system engineers plan, develop, and integrate AI-based mission autonomy in a way that respects safety boundaries, preserves accountability, and maintains operator confidence. Additional rigor is required to align autonomy development with acquisition milestones and ensure traceability from operational needs to verified behaviors. Without such structure, programs risk inconsistent implementation approaches that hinder interoperability and increase technical debt over time. A unified framework would also help standardize expectations across stakeholders, enabling clearer communication between engineers, program managers, and operational users. Ultimately, addressing these gaps is essential for delivering safe, reliable, and mission-ready autonomy to the undersea force.

D. ACQUISITION AND CERTIFICATION CONTEXT FOR AUTONOMOUS SYSTEMS

DoW acquisition and certification processes were developed for deterministic systems whose behavior could be fully specified, tested, and verified prior to fielding. Milestone reviews, verification events, and Authority to Operate decisions assume stable requirements, static software baselines, and predictable system responses. AI-based

mission autonomy challenges these assumptions by introducing adaptive behaviors, probabilistic outputs, and learning-enabled decision processes that may evolve over time.

Existing certification mechanisms therefore struggle to assess not only system performance, but also behavioral acceptability under unanticipated conditions. This challenge is amplified in undersea unmanned systems, where communication latency, environmental uncertainty, and mission endurance limit real-time human intervention. As a result, assurance increasingly depends on pre-deployment definition of prohibited behaviors and safety boundaries rather than post-deployment control. These certification limitations have been widely documented in autonomous systems research, which highlights the difficulty of validating learning-enabled behaviors using traditional verification methods (Luckcuck et al., 2018; Fisher et al., 2021).

E. NEGATIVE REQUIREMENTS AND BEHAVIORAL BOUNDING IN AI-ENABLED AUTONOMY

Traditional defense acquisition relies on positive requirements that specify what a system must do and how well it must perform. This approach has proven effective for deterministic systems whose behavior can be exhaustively specified and tested. AI-based mission autonomy challenges this paradigm. Machine-learning–enabled systems may exhibit probabilistic, context-dependent, or emergent behaviors that cannot be fully captured through positive requirements alone (Berzins, 2021; Trusilo, 2023).

Negative requirements provide a complementary mechanism by defining what an autonomous system must not do under any conditions. These requirements establish prohibited behaviors, safety boundaries, abort criteria, and decision constraints that bound autonomy within acceptable operational, legal, and ethical limits. In undersea environments, where communication delays and mission isolation limit real-time human intervention, pre-defined behavioral constraints become especially critical to safe and predictable operation (Carr et al., 2018; Martin et al., 2019).

Prior research acknowledges the importance of behavioral constraints, runtime policy enforcement, and ethical guardrails for autonomous systems (Brutzman et al., 2013; Etzioni & Etzioni, 2017). However, the literature rarely addresses how negative requirements should be systematically developed, decomposed, traced, and validated

within DoW acquisition and verification frameworks. Existing guidance emphasizes responsible AI principles and high-level governance but provides limited actionable direction for program managers and system engineers tasked with certifying autonomous behavior for operational use (Department of Defense, 2023; Ray et al., 2020).

This gap is particularly evident in UUV programs, where mission autonomy must operate independently for extended periods and interact with complex environments and legacy command and control architectures. As a result, the absence of structured approaches for managing negative requirements complicates safety certification, increases integration risk, and undermines operator trust. Addressing this deficiency requires a synthesis of program management, systems engineering, and acquisition policy perspectives, which is the focus of the following literature review.

THIS PAGE INTENTIONALLY LEFT BLANK

III. LITERATURE REVIEW: MANAGING AI-BASED MISSION AUTONOMY THROUGH PROGRAM MANAGEMENT AND SYSTEMS ENGINEERING LENSES

The integration of AI-based mission autonomy into unmanned systems represents a significant departure from traditional defense acquisition paradigms. Unlike deterministic systems, autonomous systems powered by machine learning exhibit adaptive and probabilistic behaviors that evolve over time and across operational contexts (Reis et al., 2021; Zhang et al., 2017). As autonomy matures, the challenge for the DoW is no longer limited to developing advanced algorithms; now, it must also include managing the risks associated with verification, validation, governance, and sustainment of autonomous behavior.

Existing literature addresses autonomy from multiple perspectives, including operational utility, systems engineering, software architecture, ethics, and acquisition policy. However, these perspectives are often treated in isolation. Program managers are left to reconcile technical uncertainty, life-cycle cost growth, and safety assurance without a unified framework that links autonomous behavior to acquisition decision-making (Calfee, 2021; Martin et al., 2019).

This chapter synthesizes literature across program management, systems engineering, MOSA, undersea operational research, and ethical governance. The review emphasizes how autonomy challenges manifest at the program level and highlights the absence of structured approaches for defining and validating behavioral constraints, particularly through negative requirements. The chapter is organized around the managerial and engineering challenges that influence the successful integration of AI-based mission autonomy within the DoW's UUV programs.

A. PROGRAM MANAGEMENT CHALLENGES IN AI-BASED MISSION AUTONOMY

Program management literature consistently identifies autonomy as a source of increased complexity and risk within defense acquisition. Studies of unmanned maritime and undersea programs indicate that autonomy initiatives often stall during the transition from prototype to operational capability due to unclear requirements, evolving software

baselines, and underestimated life cycle costs (Calfee, 2021; Martin et al., 2019). Unlike traditional systems, autonomy programs must manage continuous model updates, data curation, cybersecurity, and retraining pipelines, all of which introduce recurring cost and schedule pressures.

Calfee (2021) emphasizes that autonomy programs frequently underestimate the scope of governance activities required to maintain trustworthy AI, including configuration control, model validation, and cyber hardening. These governance functions are not one-time efforts but ongoing responsibilities that extend throughout the system life-cycle. Program managers must therefore balance traditional cost, schedule, and performance objectives with new obligations related to data stewardship and behavioral assurance.

Research also highlights the difficulty of translating high-level responsible AI principles into executable program artifacts. The DoW's AI adoption strategy and responsible AI guidance stress reliability, governability, and human judgment, yet they stop short of prescribing how these principles should be implemented within program requirements, test plans, and certification processes (Department of Defense, 2023; Ray et al., 2020). As a result, programs often rely on ad hoc interpretations, increasing inconsistency and risk across portfolios.

Negative requirements emerge in the literature as an implicit but underdeveloped tool for managing autonomy risk. Safety analyses and ethical discussions frequently refer to prohibited behaviors, fail-safe conditions, and abort criteria, but they rarely describe how these constraints should be captured, traced, and tested within formal acquisition frameworks (Berzins, 2021; Etzioni & Etzioni, 2017). Without explicit negative requirements, program managers lack a structured means to bound autonomous behavior and demonstrate compliance during verification and validation activities.

These challenges are amplified in UUV programs. Extended mission duration, communication latency, and environmental uncertainty limit the ability of operators to intervene during mission execution (Carr et al., 2018; Martin et al., 2019). Program managers must therefore rely more heavily on pre-mission assurance mechanisms, increasing the importance of well-defined behavioral constraints and robust validation

strategies. The literature suggests that autonomous program success depends not only on technical performance but also on the ability of program management structures to institutionalize these constraints throughout the acquisition life cycle.

B. SYSTEMS ENGINEERING AND VERIFICATION OF AUTONOMOUS BEHAVIOR

Systems engineering literature emphasizes that autonomy fundamentally alters how verification and validation must be conducted. Traditional verification approaches assume deterministic behavior, traceable functional decomposition, and stable interfaces. AI-based mission autonomy violates these assumptions by introducing probabilistic decision-making, data-driven behavior, and adaptive responses that may vary across operational contexts (Reis et al., 2021; Schafer, 2009). As a result, conventional verification techniques are often insufficient to assure safe and predictable autonomous behavior.

Research on unmanned systems highlights that verification challenges increase as autonomy migrates from low-level control functions to mission-level decision authority. Woudenberg et al. (2020) argue that autonomy must be treated as a system-of-systems problem, where interactions among perception, decision logic, mission planning, and platform constraints create emergent behaviors that are difficult to anticipate through component-level testing alone. This challenge is amplified in undersea systems, where limited observability and delayed feedback constrain real-time monitoring and post-mission assessment (Martin et al., 2019).

Several studies propose architectural and process-based mitigations. Blais (2016) emphasizes the importance of standardized interfaces and common control frameworks to enable repeatable testing and integration across unmanned platforms. Digital engineering approaches, including model-based systems engineering (MBSE) and simulation-rich environments, are frequently cited as necessary tools for exposing autonomy models to edge cases and rare conditions that cannot be safely replicated in live testing (Woudenberg et al., 2020). However, while these tools do improve test coverage, they do not by themselves resolve the question of what behaviors should be tested and constrained.

Negative requirements emerge as critical systems engineering constructs in this context. Berzins (2021) notes that AI-enabled weapon systems require explicit articulation of unsafe or prohibited behaviors to support safety certification. Similarly, Brutzman et al. (2013) demonstrate that runtime ethics checking relies on clearly defined constraints that limit autonomous action when predefined boundaries are approached or violated. Despite this recognition, the systems engineering literature offers limited guidance on how negative requirements should be derived, decomposed, and traced through architecture, design, and verification artifacts.

Verification and validation of negative requirements present additional challenges. While positive requirements can often be tested through performance demonstrations or threshold measurements, negative requirements require evidence that certain behaviors do not occur across a wide range of conditions. This shifts verification toward scenario-based testing, stress testing, and fault injection, all of which demand careful planning and significant resources. The literature suggests that without disciplined requirement traceability and configuration management, programs risk losing visibility into which constraints apply to which autonomous behaviors, especially as models are retrained or updated (Calfee, 2021; Johnson & Halbert, 2024).

In UUV programs, systems engineering decisions related to autonomy verification directly affect program risk. Long-duration missions and limited communications reduce opportunities for human intervention, increasing reliance on pre-deployment assurance. As a result, system engineers must integrate negative requirements into requirements baselines, interface specifications, and test plans from the earliest phases of development. The literature indicates that failure to do so leads to brittle autonomy implementations that are difficult to certify, sustain, and trust in operational environments.

Broader systems engineering literature echoes these concerns, emphasizing that adaptive and learning systems resist complete pre-deployment verification. Surveys of autonomous robotic systems identify persistent challenges in combining formal methods, simulation-based testing, and runtime assurance into a coherent verification strategy (Luckcuck et al., 2018; Fisher et al., 2021). These studies highlight that assurance for intelligent systems increasingly relies on defining unacceptable states and prohibited

behaviors rather than proving correctness across all possible operating conditions. Such findings directly support the integration of negative requirements as a foundational element of autonomy verification baselines.

C. MODULAR OPEN SYSTEMS ARCHITECTURE AND AUTONOMY SUSTAINMENT

MOSA is widely promoted as a mechanism for managing complexity and sustaining flexibility in defense systems. MOSA encourages the use of open standards, well-defined interfaces, and modular components to enable competition, upgrades, and reuse over the system life cycle (Modular Open Systems Approach, 2020; Wilson, 2017). In the context of AI-based mission autonomy, MOSA is particularly relevant because autonomy software, data pipelines, and models evolve more rapidly than physical platforms do.

Research indicates that modular architectures can reduce integration risk and life cycle cost by allowing autonomous components to be updated independently of the host vehicle (Johnson & Halbert, 2024; Radoman et al., 2025). However, the literature also cautions that MOSA benefits are not automatic. Programs must invest early in interface definition, documentation, and configuration management, or risk creating nominally modular systems that remain tightly coupled in practice.

Autonomy sustainment introduces additional MOSA-related challenges. Unlike traditional software updates, AI autonomy updates may alter system behavior in non-intuitive ways due to changes in training data or model structure. Defense Acquisition University (DAU) guidance stresses that autonomy updates, retraining, and revalidation must be treated as recurring sustainment events rather than one-time development activities (DAU, 2024). This places new demands on product support strategies, logistics planning, and workforce skills.

Negative requirements intersect directly with MOSA and sustainment. If behavioral constraints are embedded only within proprietary code or vendor-specific implementations, program offices may lose visibility and control over autonomous behavior as systems evolve. Radoman et al. (2025) emphasize that open architectures only enable competition and adaptability when the government retains sufficient data

rights and interface authority. From a program management perspective, negative requirements must therefore be expressed in architecture-neutral terms and traced to interfaces and verification artifacts that persist across supplier changes.

The literature suggests that MOSA can support autonomy governance by enabling independent verification environments and simulation frameworks that are decoupled from specific vendors. Blais (2016) argues that standardized interoperability frameworks allow testing and certification activities to be reused across platforms, reducing cost and improving consistency. When paired with explicit negative requirements, modular architectures can provide the structural foundation for repeatable validation of autonomous behavior over time.

In undersea systems, sustainment challenges are intensified by limited access to deployed platforms and long mission cycles. Carr et al. (2018) note that energy constraints and mission endurance trade-offs limit onboard processing and communication, making post-deployment updates and diagnostics difficult. These constraints increase the importance of getting behavioral boundaries right before deployment and designing architectures that support robust pre-mission testing and certification.

D. ETHICAL GOVERNANCE, TRUST, AND CERTIFICATION CHALLENGES

Ethical and governance considerations are tightly coupled with technical and programmatic challenges in AI-based mission autonomy. Literature on autonomous weapon systems highlights concerns related to accountability, proportionality, and human control, particularly when systems can select actions without direct supervision (Etzioni & Etzioni, 2017; Taddeo & Blanchard, 2022). While much of this work focuses on lethal autonomy, the underlying principles apply equally to mission autonomy in undersea systems, where autonomous decisions may have strategic or escalatory consequences.

The DoW has articulated responsible AI principles that emphasize traceability, reliability, and human judgment (Department of Defense, 2023). Ray et al. (2020) argue that operator trust is a prerequisite for effective integration of AI across joint functions. Trust, however, is not achieved through policy statements alone. It depends on the ability

of programs to demonstrate that autonomous systems behave predictably within defined boundaries and that failures are understood and controlled.

Negative requirements play a central role in ethical governance. By explicitly defining prohibited behaviors and abort conditions, negative requirements translate ethical and legal constraints into technical artifacts that can be engineered and tested. Brutzman et al. (2013) show that runtime ethics checking relies on codified constraints that limit action selection when ethical boundaries are approached. Trusilo (2023) further notes that emergent behavior in autonomous systems undermines predictability unless bounded by clear rules and oversight mechanisms.

Certification processes struggle to keep pace with these demands. Traditional safety certification assumes static behavior and fixed configurations, while AI systems evolve through retraining and updates. Berzins (2021) highlights the difficulty of constructing safety cases for AI-enabled systems without explicit treatment of AI-specific hazards and constraints. Without structured approaches to negative requirements, certification efforts risk devolving into subjective assessments rather than evidence-based evaluations.

Research on human trust in automation further supports the importance of bounded autonomy. Hoffman et al. (2013) find that operator trust depends less on system accuracy than on predictability and transparency of behavior. When autonomous systems operate within clearly defined constraints and exhibit interpretable decision logic, operators are more likely to accept reduced levels of human control. In undersea operations, where real-time intervention is often infeasible, trust is therefore established prior to deployment through assurance that autonomy will not exceed predefined behavioral limits.

Undersea operations magnify these issues. Limited communications mean that operators must rely on pre-mission confidence rather than real-time oversight. Studies of undersea autonomy suggest that trust is shaped by transparency, rehearsal, and clear understanding of system limits rather than by detailed knowledge of algorithms (Galdorisi, 2017; Martin et al., 2019). Negative requirements, when properly defined and

validated, provide a mechanism for articulating those limits in a way that operators and commanders can understand and accept.

E. SUMMARY OF LITERATURE GAPS AND IMPLICATIONS FOR FRAMEWORK DEVELOPMENT

The literature reviewed in this chapter demonstrates broad agreement on the importance of AI-based mission autonomy for future unmanned systems and highlights significant advances in technology, systems engineering, and policy. At the same time, several critical gaps remain.

First, existing research does not provide a unified life-cycle framework that integrates program management, systems engineering, MOSA, and ethical governance for AI-based mission autonomy. These domains are often addressed separately, leaving program managers to reconcile competing priorities without structured guidance.

Second, while many sources acknowledge the need to constrain autonomous behavior, there is limited guidance on the systematic development, decomposition, and validation of negative requirements within DoW acquisition processes. Negative requirements are frequently implied but rarely treated as first-class requirement artifacts.

Third, verification and validation practices for AI autonomy remain underdeveloped, particularly for mission-level behaviors in undersea environments. Existing approaches emphasize simulation and digital engineering but do not fully address how behavioral constraints should be tested and certified over repeated update cycles.

Finally, sustainment and governance challenges associated with autonomy updates, retraining, and configuration control are insufficiently integrated into early acquisition planning. The literature indicates that failure to address these issues early leads to cost growth, schedule delays, and erosion of operator trust.

These gaps directly motivate the development of a risk-informed life-cycle framework that treats negative requirements as central to managing AI-based mission autonomy. The following chapter builds on this literature to analyze how program management and systems engineering practices can be structured to address these challenges within DoW UUV programs.

IV. ANALYSIS: PROGRAM MANAGEMENT AND SYSTEMS ENGINEERING IMPLICATIONS FOR AI-BASED MISSION AUTONOMY

The literature reviewed in Chapter III demonstrates a broad consensus that AI and mission autonomy are essential to the future of unmanned systems, particularly in the undersea domain. At the same time, it reveals persistent shortcomings in the management of autonomy within existing DoW acquisition frameworks. While technical advances in AI continue to accelerate, program management and systems engineering practices have not evolved at the same pace to address the unique risks associated with adaptive, probabilistic behavior.

This chapter analyzes those shortcomings through the lenses of program management and systems engineering. The focus is not on proposing new algorithms or autonomy architectures, but on examining how current acquisition practices shape autonomy outcomes and where they fail to provide adequate assurance. Particular attention is given to the role of requirements development, especially the absence of structured negative requirements, as a root cause of verification challenges, governance ambiguity, and diminished operator trust.

The analysis is organized around four interrelated themes. First, it examines how traditional program management constructs struggle to accommodate AI-based mission autonomy. Second, it analyzes systems engineering challenges related to defining, tracing, and validating autonomous behavior. Third, it evaluates how acquisition pathways and life cycle governance influence the integration and sustainment of autonomy. Finally, it considers the implications of these factors for certification, ethical governance, and operator trust. Together, these analyses establish the need for an integrated life-cycle framework, developed in Chapter V.

A. PROGRAM MANAGEMENT CHALLENGES IN MANAGING AI-BASED MISSION AUTONOMY

Defense program management is built on the assumption that system behavior can be defined in advance, bounded through requirements, and controlled through stable baselines. Cost, schedule, and performance are managed by freezing requirements at key

milestones and controlling change through formal processes. AI-based mission autonomy disrupts this model by introducing systems whose behavior depends on data, training methods, and operational context rather than fixed logic alone.

The literature consistently shows that autonomous programs encounter difficulty when traditional program management tools are applied to them without modification. Martin et al. (2019) and Calfee (2021) all observe that autonomy initiatives often underestimate the scope and persistence of software and data governance activities. Model retraining, data curation, cybersecurity updates, and validation cycles introduce recurring work that does not align cleanly with milestone-based acquisition planning. When these activities are not explicitly planned and resourced, they emerge later as cost growth, schedule slips, or performance shortfalls.

Requirements development represents a central programmatic weakness. Positive requirements specify desired behaviors and performance outcomes, but they do not adequately constrain what an autonomous system must not do. In the absence of formal negative requirements, program managers lack a structured way to articulate unacceptable behaviors, prohibited actions, or safety boundaries. This omission complicates risk management, as autonomy risks are left implicit rather than explicitly controlled through the requirements baseline.

Program accountability is further blurred by the diffusion of responsibility for autonomy governance. Ethical principles and responsible AI policies exist at the department level, yet implementation responsibility is often fragmented across engineering teams, test organizations, and sustainment activities. Berzins (2021) emphasizes that AI safety depends on clear ownership of constraints and hazards. When negative requirements are not formally defined, no single authority is clearly responsible for maintaining behavioral boundaries as systems evolve.

In UUV programs, these challenges are magnified by operational realities. Extended mission duration, communication latency, and environmental uncertainty limit opportunities for human intervention. Program managers must therefore rely heavily on pre-deployment assurance rather than real-time oversight. Without explicit negative requirements, that assurance rests on informal confidence rather than documented

evidence, undermining the ability to defend autonomous decisions to oversight bodies and operational commanders.

The analysis suggests that program management practices must evolve to treat autonomous behavior as a managed risk area rather than a purely technical feature. Negative requirements should be recognized as risk controls, incorporated into baseline management, and tracked alongside cost, schedule, and performance metrics. Without this shift, autonomy programs will continue to struggle with late-stage surprises and eroding trust.

Research on machine behavior emphasizes that emergent and unanticipated actions are inherent properties of complex intelligent systems rather than anomalies to be eliminated. Rahwan et al. (2019) argue that as AI systems interact with dynamic environments, observed behavior increasingly reflects system–environment coupling rather than isolated algorithmic intent. This reality challenges acquisition models that assume stable, fully predictable behavior and reinforces the need to manage autonomy risk through constraint definition rather than by attempting to eliminate uncertainty altogether.

B. SYSTEMS ENGINEERING IMPLICATIONS FOR AUTONOMY VERIFICATION AND VALIDATION

Systems engineering provides the discipline required to translate programmatic intent into technical designs and verification strategies. For AI-based mission autonomy, systems engineering must contend with emergent, probabilistic, and context-dependent behaviors. Traditional verification and validation methods, which assume deterministic behavior and stable interfaces, are ill-suited to this challenge.

The literature highlights the difficulty of decomposing autonomy requirements using conventional systems engineering techniques. Woudenberg et al. (2020) argue that autonomy verification must focus on system-level behavior rather than isolated component performance. Mission-level autonomy emerges from interactions among perception, decision logic, mission planning, and platform constraints, making it difficult to trace behavior to individual subsystems in a linear manner.

Negative requirements offer a stabilizing construct for systems engineering in this environment. By defining prohibited behaviors and boundary conditions at the system level, negative requirements provide a reference point for architecture design and verification planning. Brutzman et al. (2013) demonstrate that runtime policy enforcement relies on clearly articulated constraints that limit autonomous action when predefined boundaries are approached. Without such constraints, verification becomes reactive rather than planned.

Verification of negative requirements requires a shift in testing philosophy. Instead of demonstrating that a system achieves a performance threshold, engineers must demonstrate that certain behaviors do not occur across a range of conditions. This requires scenario-based testing, fault injection, and stress testing rather than single-point performance demonstrations. Digital engineering and simulation environments can support this approach, but only if requirements are explicitly defined and traceable to test cases.

Configuration management further complicates verification and validation. AI models are subject to retraining and updates, and each change has the potential to alter behavior in unintended ways. Systems engineering practices must therefore include mechanisms for re-verifying negative requirements whenever models or data sets change. Calfee (2021) and Johnson and Halbert (2024) note that without disciplined configuration tracking, programs accumulate technical debt that erodes assurance over time.

For UUVs, these systems engineering challenges directly affect operational risk. Limited access to deployed platforms and restricted observability during missions increase reliance on pre-mission verification. This reinforces the need to treat negative requirements as core design artifacts that guide architecture decisions, test planning, and certification activities throughout the life cycle.

C. ACQUISITION PATHWAYS AND LIFE-CYCLE GOVERNANCE FOR ITERATIVE AUTONOMY

DoW acquisition pathways were designed around hardware-centric systems with relatively slow rates of change. AI-based mission autonomy follows a different rhythm, characterized by iterative software development, frequent updates, and evolving data

dependencies. This mismatch creates tension between acquisition oversight and the need for agility.

The literature indicates that existing acquisition pathways can accommodate autonomy, but only if the pathways are applied with deliberate adjustments. MOSA is frequently cited as a key enabler of flexibility. Open interfaces and modular components allow autonomy software to be updated independently of the host platform, reducing integration risk and enabling competition (Modular Open Systems Approach, 2020; Radoman et al., 2025; Wilson, 2017;).

However, MOSA alone does not resolve governance challenges. Johnson and Halbert (2024) caution that modularity requires upfront investment in interface definition, documentation, and data rights. Without these investments, autonomy updates may remain tightly coupled to specific vendors or architectures, limiting transparency and control. Negative requirements must therefore be expressed in architecture-neutral terms and traced to interfaces and verification artifacts that persist across supplier changes.

Life-cycle governance becomes particularly complex when autonomy updates are treated as sustainment activities rather than development events. DAU (2024) guidance emphasizes that autonomy insertion, retraining, and recertification are recurring life-cycle tasks that must be planned and resourced accordingly. When acquisition strategies fail to account for this reality, programs face gaps in funding, authority, and oversight during sustainment.

Negative requirements provide a mechanism for governing iterative autonomy within existing acquisition pathways. By defining invariant behavioral constraints, programs can allow iterative improvement while preserving safety boundaries and accountability. Each autonomy update can then be evaluated against a stable set of prohibited behaviors, simplifying certification and oversight.

In the undersea domain, where access to deployed systems is limited and mission cycles are long, life-cycle governance must prioritize pre-deployment assurance and disciplined change control. The analysis suggests that acquisition pathways should explicitly incorporate autonomy governance plans, including requirement update

processes, verification triggers, and sustainment responsibilities, rather than treat autonomy as an incremental software feature.

D. OPERATOR TRUST, CERTIFICATION, AND ETHICAL GOVERNANCE

Operator trust is a recurring theme across autonomy literature and is closely linked to verification, governance, and transparency. Ray et al. (2020) note that trust is shaped not only by system performance but also by operators' understanding of system limits. In undersea operations, where real-time oversight is limited, trust depends heavily on pre-mission confidence rather than in-mission control.

Ethical and governance literature emphasizes accountability, human control, and predictability as prerequisites for acceptable autonomy. Etzioni and Etzioni (2017) and Taddeo and Blanchard (2022) argue that autonomous systems must operate within clearly defined moral and legal boundaries. Translating these boundaries into engineering practice requires explicit articulation of prohibited behaviors, a role naturally filled by negative requirements.

Certification processes struggle to keep pace with adaptive autonomy. Traditional safety cases assume static behavior, while AI systems evolve through retraining and updates. Berzins (2021) highlights the difficulty of certifying AI-enabled systems without structured treatment of AI-specific hazards. Negative requirements offer a way to anchor certification efforts by providing stable constraints against which evolving behavior can be assessed.

Trust and certification are particularly challenging in UUVs due to limited observability and the strategic sensitivity of undersea operations. Galdorisi (2017) notes that autonomy acts as an intelligence multiplier, but only when commanders have confidence in system behavior. Negative requirements, when clearly defined and validated, provide a language for communicating system limits to operators and commanders without exposing technical complexity.

The analysis indicates that ethical governance and operator trust are not separate concerns from program management and systems engineering. They are outcomes of disciplined requirement practices, transparent verification, and consistent life-cycle

governance. Without explicit negative requirements, ethical principles remain abstract and trust remains fragile.

E. SYNTHESIS OF ANALYTICAL FINDINGS

The analysis in this chapter reinforces several key conclusions. First, AI-based mission autonomy introduces behavioral risks that cannot be adequately managed using traditional acquisition and systems engineering practices alone. Second, the absence of structured negative requirements represents a critical gap that undermines verification, governance, and trust. Third, program management, systems engineering, and acquisition pathways must be aligned to treat autonomous behavior as a life-cycle concern rather than a one-time development challenge.

Negative requirements emerge as a unifying construct that links programmatic intent, engineering discipline, and ethical governance. When treated as first-class artifacts, the requirements provide a means to bound autonomous behavior, support verification and validation, and maintain accountability as systems evolve. The literature suggests that without such constructs, autonomy programs will continue to face integration challenges, certification delays, and erosion of operator confidence.

These findings motivate the development of an integrated life-cycle framework that explicitly incorporates negative requirements into the management of AI-based mission autonomy. Chapter V builds upon this analysis to propose such a framework, designed to support program managers and systems engineers responsible for delivering trustworthy autonomous capability in UUV programs.

The analysis demonstrates that current acquisition and systems engineering practices are insufficient for managing the risks introduced by AI-based mission autonomy. Without explicit mechanisms for defining negative requirements and enforcing behavioral constraints, autonomy programs remain vulnerable to certification ambiguity, configuration drift, and diminished operator trust. These findings indicate that incremental adaptation of existing processes is unlikely to be sufficient. Instead, a structured framework that integrates program management, systems engineering, and behavioral constraint definition is necessary to ensure that autonomy evolves within acceptable operational and ethical boundaries.

THIS PAGE INTENTIONALLY LEFT BLANK

V. AN INTEGRATED FRAMEWORK FOR MANAGING THE DEVELOPMENT AND PRODUCTION OF MISSION AUTONOMY

This chapter presents a comprehensive, integrated framework for the development, integration, and verification of AI-based mission autonomy within the DoW UUV programs. Building directly on the findings of Chapters I–IV, this chapter responds to the central challenge identified throughout this thesis: AI introduces emergent, probabilistic, and context-dependent behavior that fundamentally differs from the deterministic logic assumed by legacy acquisition, systems engineering, and verification practices (Berzins, 2021; Martin et al., 2019; Trusilo, 2023).

The overarching goal of this framework is to enable the fielding of mission autonomy that expands operational capability while preserving safety, accountability, and operator trust. Traditional requirements frameworks rely heavily on positive requirements, statements describing what a system shall do. For AI-based autonomy, this approach is insufficient. It is equally critical to define what the system shall not do across a range of operational and environmental conditions (Etzioni & Etzioni, 2017; Ray et al., 2020). This chapter proposes an acquisition framework intended to be implemented within existing DoW program structures that elevates negative requirements to first-class artifacts and embeds them within a containerized autonomy architecture.

The framework integrates program management, systems engineering, acquisition strategy, and ethical governance into a cohesive model that complements existing DoW acquisition processes, including the Adaptive Acquisition Framework and the Software Acquisition Pathway (Office of the Under Secretary of Defense for Acquisition and Sustainment, 2020, 2022). Rather than replacing established practices, the framework introduces a governance mechanism that augments them by addressing autonomy-specific risk. The chapter is organized to first establish program context, then define management and engineering roles, present the architecture, and finally describe requirements, testing, and life-cycle governance before transitioning to the application in Chapter VI. This framework directly addresses the verification ambiguity, governance fragmentation, and trust deficits identified in Chapters III and IV by treating behavioral constraints as acquisition-managed artifacts rather than implicit safeguards.

A. PROGRAM CONTEXT AND OPERATIONAL CHALLENGES

Mission autonomy in UUVs is driven by the operational realities of the undersea domain. Acoustic communications are inherently bandwidth-limited, intermittent, and vulnerable to adversary interference, while satellite communication is unavailable at depth (Button et al., 2009; Carr et al., 2018). Missions often require long endurance, low observability, and operation in complex environments with uncertain bathymetry and dynamic ocean conditions. These constraints significantly limit the feasibility of continuous human command and control.

As adversaries expand undersea surveillance networks and anti-access capabilities, UUVs are increasingly expected to operate independently in contested environments for extended periods (O'Rourke, 2025). In such scenarios, AI-based mission autonomy is not merely an enhancement but a necessity. Autonomous systems must interpret sensor data, plan routes, manage energy, and respond to unexpected events without real-time human input (Martin et al., 2019; Reis et al., 2021).

However, the same adaptive characteristics that make AI-based autonomy valuable also introduce risk. Machine learning models generalize from training data and may behave unpredictably when exposed to novel environments or adversarial conditions (Trusilo, 2023). This creates a tension between operational flexibility and assurance. Without enforceable constraints, autonomy can exceed intended authority or behave in ways that undermine safety, legality, or strategic intent (Berzins, 2021).

These operational realities establish the need for bounded autonomy, autonomy that is explicitly constrained by predefined rules, safety boundaries, and ethical considerations that persist even when communications are denied. This chapter's framework addresses that need by embedding governance directly into the deployed autonomy architecture.

B. PROGRAM MANAGEMENT FOUNDATIONS FOR MISSION AUTONOMY

Program managers are responsible for delivering capability within cost, schedule, and performance constraints while ensuring compliance with policy, safety, and ethical standards. AI-based mission autonomy complicates this responsibility by introducing

continuous software evolution, data dependencies, and emergent behavior that do not align neatly with traditional milestone-based acquisition models (Calfee, 2021).

Responsible AI principles require that autonomy remain governable, traceable, and aligned with human judgment (DAU, 2024; Ray et al., 2020). Program managers must therefore oversee not only technical development but also autonomy governance, configuration control, and ethical compliance. This includes ensuring that autonomy does not outpace verification and validation processes.

The Software Acquisition Pathway provides a mechanism for iterative development while maintaining oversight (Office of the Under Secretary of Defense for Acquisition and Sustainment, 2020). When combined with MOSA, it enables modular autonomy components that can evolve without destabilizing the host platform (Johnson & Halbert, 2024; Wilson, 2017). However, these benefits are realized only when governance mechanisms are deliberately designed and enforced.

Risk management practices must explicitly address autonomy-specific hazards, including model drift, data bias, cybersecurity vulnerabilities, and emergent behavior. Negative requirements function as risk controls, translating abstract policy and ethical guidance into enforceable technical constraints. By treating negative requirements as baseline-managed artifacts, program managers gain a structured mechanism for autonomy risk oversight and milestone decision-making.

C. SYSTEMS ENGINEERING ROLE IN MANAGING AUTONOMOUS BEHAVIOR

Systems engineers translate operational intent into technical architectures, requirements, and verification strategies. AI-based autonomy challenges traditional systems engineering assumptions by introducing probabilistic behavior and adaptive decision-making (Schafer, 2009; Woudenberg et al., 2020). Component-level verification is insufficient when mission-level behavior emerges from interactions across perception, planning, and control subsystems.

Systems engineers are required to define constraints that apply across the autonomy stack, ensuring that autonomous decisions remain within acceptable operational, legal, and ethical limits. This requires close coordination with program

managers to ensure that governance mechanisms align with acquisition and risk management objectives.

Negative requirements provide a stabilizing construct for systems engineering. They define forbidden actions, safety boundaries, and abort conditions that guide architecture design and test planning. By embedding these requirements early, systems engineers can shape interface definitions, monitoring functions, and verification strategies that remain valid as autonomy evolves.

Shared accountability between program management and systems engineering is essential. Program managers establish risk tolerance and governance priorities, while systems engineers implement those priorities through requirements and architecture. The proposed framework formalizes this relationship.

D. FRAMEWORK OVERVIEW: CONTAINERIZED AUTONOMY ARCHITECTURE

The core of the proposed framework is a containerized autonomy architecture that separates mission execution from governance enforcement. The UUV platform and human–machine interface are developed using traditional systems engineering methods, including requirements decomposition, interface control, and verification and validation.

AI Mission-Based Autonomy is implemented as a main container responsible for mission planning, decision-making, and execution during communications-denied operations. This container leverages AI techniques to adapt to environmental uncertainty and mission dynamics (Reis et al., 2021; Zhang et al., 2017).

A separate sidecar container operates alongside the mission autonomy container and serves as an independent governance mechanism. The sidecar enforces negative requirements, including rules of engagement, legal constraints, and ethical considerations that human operators would normally apply during supervised operations (Office of the Under Secretary of Defense for Policy, 2023). Figure 1 illustrates this architecture and the flow of authority and information between components.

By decoupling governance from mission logic, the architecture enables autonomy while preserving human judgment through pre-authorized constraints. This approach supports both operational effectiveness and accountability.

E. SIDECAR GOVERNANCE CONTAINER CONCEPT

The sidecar container is designed to be independent of the AI mission autonomy container, both logically and organizationally. Ideally, it is developed or certified by an independent authority to reinforce objectivity and trust. In practical acquisition terms, this independence can be formalized through contractual and organizational separation. The governance logic embodied in the sidecar container could be developed under a separate Contract Line-Item Number or assigned to a distinct contractor responsible specifically for safety, policy compliance, and invariant behavioral constraints. This separation ensures that the entity optimizing mission performance is not the same entity responsible for enforcing operational limits, reducing incentives to relax safety boundaries in pursuit of performance objectives.

From a program management perspective, this arrangement establishes a clear division of authority. The program manager retains responsibility for certifying that the sidecar container's invariant constraints satisfy system safety requirements and legal policy guidance, while mission autonomy performance metrics are evaluated separately through traditional test and evaluation processes. This dual certification approach mirrors established practices in weapon system safety, where independent safety authorities validate hazard controls independently from performance verification (Berzins, 2021; DAU, 2024).

In this context, independence does not imply a separate physical system, but rather organizational, contractual, or authority separation sufficient to prevent governance logic from being subordinated to performance optimization. Its primary function is to monitor autonomous decisions and prevent execution of actions that violate predefined constraints (Brutzman et al., 2013).

The sidecar encodes rules of engagement, legal restrictions, ethical boundaries, and safety constraints as executable logic. These constraints persist during communications-denied operations, ensuring continuous governance even when human oversight is unavailable. In effect, the sidecar represents delegated human judgment.

This concept aligns with DoW autonomy policy, which emphasizes "appropriate levels of human judgment over the use of force and autonomous decision-making"

(Office of the Under Secretary of Defense for Policy, 2023). While the framework is not limited to weaponized systems, the same principles apply to mission-critical undersea operations with strategic implications.

The independence of the sidecar also supports certification and sustainment. Constraints can be updated, audited, and revalidated independently of mission autonomy logic, reducing integration risk and supporting iterative improvement.

F. REQUIREMENTS ENGINEERING FOR MISSION AUTONOMY

Traditional requirements engineering emphasizes positive requirements that specify desired functions and performance. For AI-based autonomy, this approach must be expanded to include negative requirements that define prohibited behaviors and constraints (Berzins, 2021).

In the proposed framework, positive requirements are allocated primarily to the mission autonomy container, defining what the system shall do to accomplish mission objectives. The novelty is not the existence of constraints themselves, but their elevation to baseline-managed, traceable, and test-verifiable requirements explicitly owned within the acquisition process. Negative requirements are allocated to the sidecar container, defining what the system shall not do under any conditions.

This separation enhances traceability between operational risks, constraints, and verification activities. Each negative requirement is linked to an operational hazard, legal constraint, or ethical consideration, enabling structured risk assessment and test planning. By treating negative requirements as first-class artifacts, systems engineers can ensure that autonomy remains bounded even as models evolve through retraining and updates.

Negative requirements must also be managed as dynamic life-cycle artifacts rather than static design constraints. Each “shall not” requirement should trace directly to a specific hazard identified through system safety analysis, operational risk assessment, or legal policy review. This traceability ensures that behavioral constraints remain anchored to explicit risk drivers rather than abstract ethical considerations, supporting defensible certification and risk acceptance decisions.

Because AI-based mission autonomy evolves through retraining and algorithm updates, negative requirements provide a stable reference baseline for iterative verification. Every modification to the mission autonomy model must be revalidated against the invariant constraints enforced by the sidecar container to ensure that new training data or decision logic does not introduce previously mitigated hazards. This process reflects established safety engineering practice, where hazard controls must be reassessed whenever system functionality changes (Martin et al., 2019; Seshia et al., 2016). This life cycle treatment ensures that program managers retain continuous visibility into autonomy risk controls and can make informed milestone decisions based on verified constraint enforcement.

G. DEVELOPMENT OF NEGATIVE REQUIREMENTS

Negative requirements are systematically derived through a combination of hazard analysis, mission analysis, legal review, and ethical assessment. Sources include system safety analyses, cybersecurity threat assessments, and rules of engagement documentation (Berzins, 2021; Office of the Under Secretary of Defense for Policy, 2023).

Examples of negative requirements include prohibitions on entering restricted geographic areas, constraints on interaction with sensitive infrastructure, abort criteria triggered by sensor degradation, and limits on engagement authority. These requirements encode constraints that would otherwise rely on real-time human judgment.

Negative requirements also support reliability and survivability by defining safe states and recovery behaviors under uncertainty (Di Lorenzo & Bayer, 2021). When integrated with cybersecurity analyses, they can limit the impact of compromised sensors or data feeds. The systematic development and documentation of negative requirements provide a defensible basis for certification and operational acceptance.

H. DIGITAL ENGINEERING AND ARCHITECTURAL ENABLERS

Digital engineering and MBSE are essential enablers of the proposed framework. Architecture models capture interfaces between the UUV platform, mission autonomy container, and sidecar governance container, enabling early analysis of integration and risk (Woudenberg et al., 2020). MBSE supports behavior modeling, scenario exploration,

and traceability between requirements and verification artifacts. Simulation environments enable stress testing of autonomy under edge cases that cannot be safely replicated in live testing. Cybersecurity, weapon system safety, and reliability considerations are integrated into the architecture. This holistic approach supports early identification of autonomy risks and reduces downstream integration challenges (Whelan et al., 2022).

Digital engineering also provides a critical mechanism for generating objective quality evidence that supports operator and commander trust in autonomous systems. Simulation-rich environments enable program managers and test authorities to demonstrate, through repeatable scenarios, that the sidecar governance container consistently prevents prohibited behaviors under edge-case conditions such as sensor degradation, navigation uncertainty, or adversarial interference. This evidence-based approach shifts trust from subjective confidence in AI algorithms to measurable assurance grounded in observable system behavior.

By presenting commanders with a validated and transparent set of operational constraints, rather than requiring them to understand the internal logic of adaptive AI models, the framework simplifies the safety case for operational deployment. Trust is therefore established not through explainability of algorithmic decisions, but through verifiable enforcement of clearly defined behavioral limits (Luckcuck et al., 2018; Rahwan et al., 2019).

I. TEST, EVALUATION, AND CONTINUOUS ASSURANCE

Test and evaluation (T&E) of AI-based mission autonomy must focus on system-level behavior rather than isolated component performance. The Test and Evaluation Master Plan emphasizes scenario-based testing, stress testing, and fault injection to expose emergent behavior (Martin et al., 2019). Verification of negative requirements is the linchpin of the framework, providing the objective evidence necessary to support certification, sustain operator trust, and permit iterative autonomy updates without eroding safety or accountability.

This shifts emphasis from threshold-based testing to behavioral assurance. Because AI models evolve through retraining, continuous verification and validation are

required. Each model update triggers reassessment against the baseline set of negative requirements, preserving safety and accountability over time (Calfee, 2021).

J. LIFE-CYCLE INTEGRATION AND GOVERNANCE

The framework integrates program management and systems engineering activities across the life cycle. Configuration management of AI models and sidecar constraints ensures traceability and accountability (DAU, 2024). Governance mechanisms support informed risk acceptance and authority-to-operate decisions. By embedding governance into the deployed architecture, the framework reduces reliance on ad hoc oversight and enhances trust. Life-cycle integration ensures that autonomy governance persists through development, deployment, sustainment, and upgrades.

K. CHAPTER SUMMARY

This chapter presented an integrated framework for managing AI-based mission autonomy in UUV programs. By introducing a containerized architecture with an independent sidecar governance mechanism, the framework addresses the unique risks associated with adaptive, probabilistic behavior.

The architectural separation illustrated in Figure 1 is central to achieving bounded autonomy without undermining operational effectiveness. By isolating governance logic from mission execution logic, the framework allows each to evolve at an appropriate pace. Mission autonomy algorithms may be retrained or replaced to improve performance, while governance constraints remain stable and auditable. This separation mirrors established safety architectures in other high-consequence domains, such as aviation and nuclear systems, where independent monitoring functions are used to enforce invariant constraints.

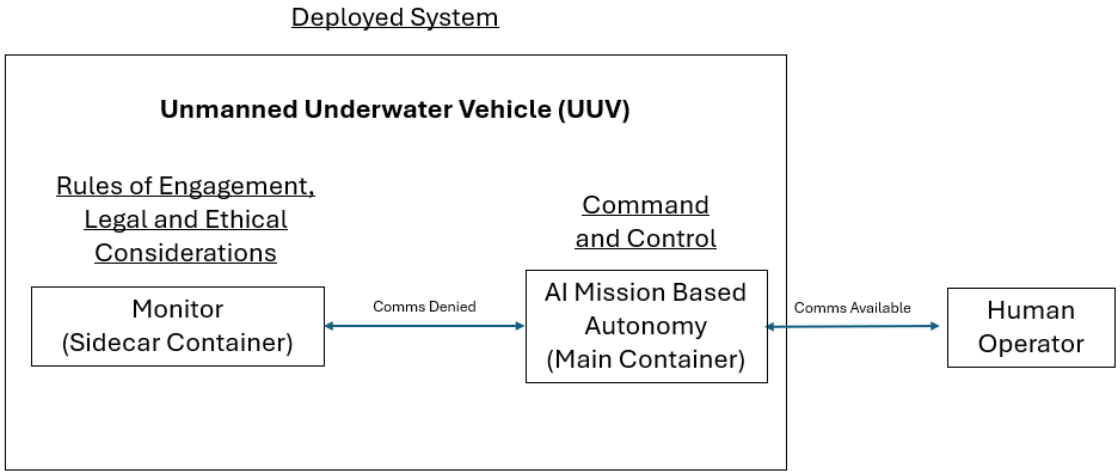

Figure 1. Containerized Mission Autonomy Framework with Sidecar Governance

From a program management perspective, the figure clarifies responsibility boundaries. The program manager retains authority over risk acceptance and governance policy, while systems engineers ensure constraints are correctly implemented and verified. The visual representation also supports communication with stakeholders, including test authorities and operational commanders, by making autonomy limits explicit rather than implicit.

The framework further supports scalability across different UUV classes. While specific autonomy functions may vary between small, medium, and extra-large UUVs, the sidecar governance concept remains consistent. This consistency reduces training burden and enhances fleet-level trust in autonomous behavior. It complements existing acquisition and systems engineering practices while strengthening autonomy assurance. Chapter VI applies this framework to a representative undersea autonomy program scenario to demonstrate how it can be executed within existing DoW acquisition pathways.

VI. APPLICATION OF THE FRAMEWORK TO A REPRESENTATIVE UNDERSEA AUTONOMY PROGRAM

This chapter applies the integrated life-cycle framework developed in Chapter V to a representative DoW UUV program. The intent is neither to evaluate a specific program of record nor to claim empirical validation of the framework. Instead, this chapter demonstrates how the framework can be executed within existing acquisition pathways to manage AI-based mission autonomy in a realistic undersea context.

The application illustrates how program managers and systems engineers could use the framework to structure requirements, bound autonomous behavior, and govern iterative autonomy development without departing from established acquisition policy. By walking through a notional but representative program scenario, the chapter shows how negative requirements, containerized autonomy, and sidecar governance can be translated into executable program artifacts. This approach reinforces that the framework is acquisition-grade and operationally grounded rather than conceptual.

A. REPRESENTATIVE PROGRAM CONTEXT

The representative program considered in this chapter is a medium to large UUV intended to support ISR missions in contested environments. The vehicle is assumed to be pier-launched, long-endurance, and capable of operating independently for extended periods with limited or intermittent communications. Missions include seabed mapping, passive acoustic sensing, and deployment or monitoring of undersea sensors, consistent with mission sets identified in prior RAND and Congressional Research Service analyses.

Operational constraints reflect the realities of the undersea domain. Acoustic communications are bandwidth-limited and vulnerable to disruption, satellite communication is unavailable at depth, and environmental uncertainty affects navigation, sensing, and energy management. As a result, the vehicle must execute mission-level decisions without continuous human oversight. AI-based mission autonomy is therefore required for route planning, adaptive search behavior, and resource management.

From an acquisition standpoint, the program is assumed to follow the Adaptive Acquisition Framework, with autonomy software developed under the Software

Acquisition Pathway and integrated within a MOSA. This reflects current DoW policy direction and common practice for software-intensive systems. The program is assumed to be unclassified for development purposes, enabling the use of open literature, simulation environments, and test ranges during early phases.

B. APPLYING THE FRAMEWORK DURING EARLY PROGRAM PHASES

During concept development and early requirements definition, the framework guides program managers and systems engineers to treat autonomy as a managed risk area rather than a purely technical feature. Mission analysis identifies not only desired capabilities, such as persistent ISR coverage and adaptive search, but also unacceptable outcomes, including entry into restricted areas, unsafe proximity to undersea infrastructure, and continued mission execution under degraded sensing conditions.

Negative requirements are derived alongside positive requirements during this phase. Hazard analysis, legal review, and operational risk assessment inform explicit prohibitions, abort conditions, and safety boundaries. These negative requirements are entered into the requirements baseline and traced to mission hazards rather than left implicit in design assumptions or operator procedures. Program managers retain visibility into these constraints as baseline-managed artifacts, enabling informed risk discussions at milestone reviews.

Architecturally, the program adopts the containerized autonomy model described in Chapter V. Mission autonomy functions are implemented within a primary autonomy container responsible for planning and decision-making during communications-denied operations. A sidecar governance container is defined in parallel, responsible for enforcing negative requirements through runtime monitoring and policy enforcement. This separation is captured in early architecture products, ensuring that governance is treated as a system function rather than an afterthought.

C. SYSTEMS ENGINEERING AND ARCHITECTURE EXECUTION

From a systems engineering perspective, the framework shapes how autonomy is decomposed, integrated, and verified. Rather than attempting to fully specify mission behavior through positive requirements, systems engineers define behavioral boundaries

that apply across the autonomy stack. These boundaries are expressed as negative requirements allocated to the sidecar governance container.

Interface definitions between the mission autonomy container, the sidecar container, and the vehicle control systems are specified early. This enables independent development and testing of governance logic, supports modular upgrades, and preserves visibility into autonomous decision pathways. MOSA principles ensure that autonomy components can be updated or replaced without redesigning the host vehicle, provided interface contracts are maintained.

MBSE artifacts capture these relationships and support scenario exploration. Simulation environments are used to expose the autonomy system to edge cases, degraded sensor conditions, and adversarial scenarios that cannot be safely tested at sea. Verification planning focuses on demonstrating that prohibited behaviors do not occur under defined conditions rather than on the exhaustive correctness of mission decisions.

D. TEST, EVALUATION, AND CERTIFICATION ACTIVITIES

The framework significantly influences T&E strategy. The TEMP emphasizes scenario-based testing, stress testing, and fault injection to assess system-level behavior. Negative requirements drive test case development by defining conditions under which the system must abort, revert to a safe state, or refuse to execute certain actions.

Because AI models may be retrained over time, verification of negative requirements is treated as a recurring activity rather than a one-time event. Configuration management ensures traceability between model versions, training data, and verification results. Each autonomy update triggers reassessment against the baseline set of negative requirements, providing objective evidence to support certification decisions.

From a certification perspective, the sidecar governance container simplifies assurance. Safety cases and certification artifacts focus on the enforcement of invariant constraints rather than on the internal logic of learning-enabled models. This aligns with emerging views in autonomous systems assurance, which emphasize bounding unacceptable behavior rather than proving complete correctness in complex environments.

E. PROGRAM MANAGEMENT AND GOVERNANCE IMPLICATIONS

For program managers, the framework provides a structured mechanism to oversee autonomy risk. Negative requirements function as explicit risk controls that can be tracked alongside cost, schedule, and performance metrics. Governance responsibilities are clearly delineated, with program management retaining authority over risk acceptance and systems engineering responsible for implementation and verification.

The framework also supports iterative development without eroding oversight. By maintaining a stable set of behavioral constraints, the program can pursue incremental autonomy improvements while preserving safety and accountability. This is particularly valuable under the Software Acquisition Pathway, where frequent updates are expected.

Operator trust is addressed indirectly but effectively. Rather than requiring operators to understand AI algorithms, the framework enables communication of system limits in operational terms. Clear articulation of what the system will not do builds pre-mission confidence, which is essential in communication-limited undersea operations.

F. SUMMARY AND IMPLICATIONS

This chapter demonstrates how the integrated life-cycle framework can be applied within a representative undersea autonomy program. The application showed that negative requirements, containerized autonomy, and sidecar governance can be implemented using existing acquisition pathways, systems engineering practices, and test processes.

The framework does not eliminate uncertainty inherent in AI-based mission autonomy. Instead, it provides a disciplined approach to bounding behavior, managing risk, and sustaining trust as autonomy evolves. By treating negative requirements as first-class acquisition artifacts and embedding governance into system architecture, the framework offers program managers and systems engineers a practical tool for delivering mission- ready autonomy in the undersea domain.

The following chapter concludes the thesis by summarizing key findings, discussing implications for DoW policy and practice, and identifying areas for future research.

VII. CONCLUSIONS, IMPLICATIONS, AND DIRECTIONS FOR FUTURE RESEARCH

This thesis examined how program management and systems engineering principles can be combined to manage the development and integration of AI-based mission autonomy in unmanned undersea systems within the DoW acquisition process. The central problem addressed was not the absence of autonomy technology, but the lack of acquisition-grade mechanisms for bounding autonomous behavior as systems transition from deterministic automation to adaptive, learning-enabled decision-making.

Through a synthesis of literature, acquisition policy, and systems engineering practice, the study identified negative requirements as a critical but underutilized construct for managing autonomy risk. The framework developed in Chapter V and applied in Chapter VI responds directly to this gap by elevating behavioral constraints to first-class acquisition artifacts and embedding governance into system architecture. This chapter concludes by synthesizing the study's findings, discussing their implications for defense acquisition practice, and identifying directions for future research.

A. KEY FINDINGS

Several key findings emerge from this research. First, AI-based mission autonomy fundamentally challenges traditional acquisition assumptions about predictability, verification, and control. Learning-enabled systems exhibit probabilistic and context-dependent behavior that cannot be fully specified or exhaustively tested using requirement frameworks designed for deterministic systems. Attempts to manage autonomy risk solely through positive requirements and performance metrics leave critical behavioral boundaries implicit and poorly governed.

Second, the existing literature and its analysis demonstrate that emergent behavior is an inherent property of complex autonomous systems rather than an exceptional failure mode. This reality shifts the assurance problem from eliminating uncertainty to bounding it. Programs that attempt to treat autonomy as a conventional software feature are therefore likely to encounter verification ambiguity, certification delays, and erosion of operator trust.

Third, negative requirements provide a practical mechanism for translating ethical, legal, and operational constraints into enforceable engineering artifacts. When treated as baseline-managed, traceable, and test-verifiable requirements, negative requirements enable programs to define prohibited behaviors, abort conditions, and safety boundaries explicitly rather than relying on informal assumptions or operator intervention.

Finally, the research shows that autonomy governance is most effective when embedded directly into system architecture. The containerized autonomy framework with an independent sidecar governance construct provides a means to enforce behavioral constraints continuously, including during communications-denied operations. This architectural separation supports modularity, iterative development, and sustained assurance without undermining operational effectiveness. Together, these findings demonstrate that effective management of AI-based mission autonomy requires a deliberate integration of managerial oversight mechanisms with enforceable technical architecture, ensuring that behavioral constraints are governed not only by policy direction but also by independently verifiable system design features.

B. IMPLICATIONS FOR PROGRAM MANAGEMENT PRACTICE

For program managers, the findings underscore the need to treat autonomous behavior as a managed risk area rather than a purely technical feature. AI-based mission autonomy introduces recurring governance obligations related to data stewardship, model retraining, verification, and certification. These obligations must be planned, resourced, and tracked explicitly throughout the acquisition life cycle.

The framework developed in this thesis offers program managers a structured way to exercise oversight without constraining innovation. By treating negative requirements as risk controls, program managers gain a tangible mechanism to assess autonomy maturity, support milestone decisions, and communicate risk posture to senior leadership. This approach aligns autonomy governance with existing cost, schedule, and performance management practices rather than positioning it as an external ethical concern.

The findings also highlight the importance of acquisition strategy selection. Pathways that support iterative development, when paired with stable behavioral

constraints, enable programs to evolve autonomy capability while preserving accountability. Without such constraints, iterative development risks becoming uncontrolled experimentation rather than disciplined capability delivery.

C. IMPLICATIONS FOR SYSTEMS ENGINEERING AND VERIFICATION

From a systems engineering perspective, the research reinforces that mission-level autonomy cannot be assured through component-level verification alone. Behavioral assurance requires system-level constraints that apply across perception, decision-making, and control functions. Negative requirements provide a stable reference point for architecture design, interface definition, and test planning in the presence of adaptive behavior.

The emphasis on verification of negative requirements has direct implications for T&E practice. Assurance shifts from demonstrating optimal performance to demonstrating that unacceptable behaviors do not occur under defined conditions. This requires greater reliance on scenario-based testing, stress testing, and simulation-rich environments. While these methods demand upfront investment, they provide more meaningful evidence of safety and reliability in complex operational contexts.

The framework also supports sustainment by enabling reverification of autonomy updates against invariant constraints. This reduces the accumulation of technical debt and supports continued certification as systems evolve. For systems engineers, this approach offers a disciplined pathway to manage change without sacrificing assurance.

D. POLICY AND GOVERNANCE IMPLICATIONS

At the policy level, the findings suggest that responsible AI principles and autonomy guidance are necessary but insufficient without acquisition-grade implementation mechanisms. High-level policy statements must be translated into requirement practices, architectural decisions, and verification strategies that program offices can execute consistently.

The framework presented in this thesis demonstrates one such translation. By embedding governance into deployed architecture and treating behavioral constraints as explicit requirements, the framework operationalizes policy intent in a manner compatible with existing acquisition processes. This approach supports transparency,

accountability, and trust without imposing unrealistic demands on real-time human oversight.

For UUVs in particular, where communication constraints limit supervision, pre-deployment assurance becomes paramount. Clear articulation and enforcement of autonomy limits enable commanders to accept reduced levels of control with confidence. This has strategic implications for how unmanned systems are employed in contested environments.

E. LIMITATIONS OF THE STUDY

This study is subject to several limitations. It relies exclusively on unclassified literature, policy documents, and open-source analyses. Classified program data and operational lessons may reveal additional constraints or considerations not captured here. The framework is conceptual and managerial in nature and was not empirically validated through application to a specific program of record.

The representative application in Chapter VI demonstrates executability but does not measure performance outcomes or cost impacts. As a result, conclusions about efficiency gains or risk reduction remain inferential. These limitations are consistent with the scope of a defense-focused managerial inquiry but should be acknowledged when interpreting the findings.

F. DIRECTIONS FOR FUTURE RESEARCH

Future research could extend this work in several directions. Empirical studies applying the framework to specific programs of record would provide valuable insight into implementation challenges, cost impacts, and organizational dynamics. Comparative analysis across domains, such as surface, air, and space systems, could assess the generalizability of negative requirement practices beyond the undersea context.

Additional research could explore tooling and automation to support verification of negative requirements, including runtime monitoring and digital twin approaches. Governance models for independent constraint development and certification also warrant further study, particularly with respect to authority, liability, and data rights.

Finally, as autonomy capabilities mature, research examining the interaction between operator training, trust calibration, and bounded autonomy could inform how governance mechanisms are communicated and understood across the force.

G. CONCLUDING REMARKS

AI-based mission autonomy offers significant operational advantages for UUVs, but it also introduces risks that challenge traditional acquisition and engineering practices. This thesis maintains that those risks cannot be managed effectively without explicit mechanisms for bounding autonomous behavior.

By elevating negative requirements to first-class acquisition artifacts and embedding governance into system architecture, the framework developed here provides a practical pathway for managing autonomy risk while preserving innovation. The findings suggest that trustworthy autonomy is not achieved by eliminating uncertainty, but by constraining it deliberately and transparently.

For program managers and systems engineers, this approach offers a disciplined means to deliver mission-ready autonomy that aligns with DoW expectations for safety, accountability, and operational effectiveness in the undersea domain.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Berzins, V. A. (2021). *Weapons systems safety when deploying AI technology* (NPS NRP Executive Summary No. NPS-21-N387-A). Naval Postgraduate School. https://calhoun.nps.edu/server/api/core/bitstreams/1426b3db-8152-41e1-9cbd-f99d780192c2/content

Bessemer, W. G. (2006). *Transitioning to unmanned combat aerial vehicles* [Master's thesis, Naval Postgraduate School]. Defense Technical Information Center. https://apps.dtic.mil/sti/tr/pdf/ADA456959.pdf

Blais, C. L. (2016). *Unmanned system interoperability standards* (Technical Report No. NPS-MV-16-001). Naval Postgraduate School. https://apps.dtic.mil/sti/trecms/pdf/AD1060226.pdf

Brutzman, D. P., Davis, D. T., Lucas, G. R., Jr., & McGhee, R. B. (2013). Run-time ethics checking for autonomous unmanned vehicles, developing a practical approach. *Proceedings of the 18th International Symposium on Unmanned Untethered Submersible Technology*, 1–16 https://savage.nps.edu/AuvWorkbench/website/documentation/papers/UUST2013PracticalRuntimeAUVEthics.pdf

Button, R. W., Kamp, J., Curtin, T. B., & Dryden, J. (2009). *A survey of missions for unmanned undersea vehicles*. RAND. https://www.rand.org/pubs/monographs/MG808.html

Calfee, S. H. (2021). *Delivering advanced unmanned autonomous systems and artificial intelligence for naval superiority*. Center for Strategic and Budgetary Assessments. https://csbaonline.org/research/publications/delivering-advanced-unmanned-autonomous-systems-and-artificial-intelligence-for-naval-superiority

Carr, C. J., Franco, J., Mierzwa, C., Shattuck IV, L. B., & Suursoo, M. A. (2018). *Seabed warfare and the XLUUV* [Capstone report, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/59584

Cheng, B. H. C., de Lemos, R., Giese, H., Inverardi, P., Magee, J., Andersson, J. … Whittle, J. (2009). Software engineering for self-adaptive systems: A research roadmap. In B. H.C. Cheng, R. Lemos, H. Giese, P. Inverardi, & J. Magee (Eds.), *Software engineering for self-adaptive systems* (Lecture Notes in Computer Science Vol. 5525, pp. 1–26). Springer. https://doi.org/10.1007/978-3-642-02161-9_1

Defense Acquisition University. (2024, December 24). *Unmanned systems: Considerations for DoD product support managers (PSM)*. https://www.dau.edu/blogs/unmanned-systems-considerations-dod-product-support-managers-psm

Department of Defense. (2023). *Department of Defense data, analytics, and artificial intelligence adoption strategy*. https://media.defense.gov/2023/nov/02/2003333300/-1/-1/1/dod_data_analytics_ai_adoption_strategy.pdf

Di Lorenzo, R. A., & Bayer, M. A. (2021). A prognostics and health management system for an unmanned combat aircraft system – A Defense Acquisition University case study. *Proceedings of the Annual Conference of the Prognostics and Health Management Society*, 1–12. https://papers.phmsociety.org/index.php/phmconf/article/download/1629/592

Etzioni, A., & Etzioni, O. (2017, May–June). Pros and cons of autonomous weapons systems. *Military Review*, 72–80. https://www.armyupress.army.mil/Journals/Military-Review/English-Edition-Archives/May-June-2017/Pros-and-Cons-of-Autonomous-Weapons-Systems/

Fahey, S. F., Jr. (2016). *Software architecture for anti-submarine warfare unmanned surface vehicles* [Master's thesis, Naval Postgraduate School]. Defense Technical Information Center. https://apps.dtic.mil/sti/tr/pdf/AD1029746.pdf

Fisher, M., Cardoso, R. C., Collins, E. C., Dadswell, C., Dennis, L. A., Dixon, C. … Webster, M. (2021). An Overview of Verification and Validation Challenges for Inspection Robots. *Robotics*, *10*(2), 67. https://doi.org/10.3390/robotics10020067

Futch, F. D. (2012). *An analysis of the manpower impact of unmanned aerial vehicles on subsurface platforms* [Master's thesis, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/6795

Galdorisi, G. (2017). *Designing autonomous systems for military use: Harnessing artificial intelligence to provide augmented intelligence* [Paper presentation]. CRUSER TechCon, Monterey, CA, United States. https://nps.edu/documents/151816058/0/Galdorisi_AugmentedIntelligence_2017_CRUSER_TechCon_Paper_04_07_17.pdf

Geiss, E. A. (2019). *Analysis of unmanned surface vessel employment in distributed maritime operations* [Master's thesis, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/64162

Hoffman, R. R., Johnson, M., Bradshaw, J. M., & Underbrink, A. (2013). Trust in automation. *IEEE Intelligent Systems*, *28*(1), 84–88. https://doi.org/10.1109/MIS.2013.24

Johnson, D. S., & Halbert, T. M. (2024). *Plug and play acquisition (implementing MOSA)* [Capstone applied project report, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/73474

Joshi, A., Spilbergs, A., & Miķelsone, E. (2024). AI-enabled drone autonomous navigation and decision making for defence security. *Proceedings of the 15th International Scientific and Practical Conference: Vol. 4. Environment. Technology. Resources*, 138–143. https://doi.org/10.17770/etr2024vol4.8237

Luckcuck, M., Farrell, M., Dennis, L. A., Dixon, C., & Fisher, M. (2018). *Formal specification and verification of autonomous robotic systems: A survey*. ArXiv. http://arxiv.org/abs/1807.00048

Martin, B., Tarraf, D. C., Whitmore, T. C., DeWeese, J., Kenney, C. … DeLuca, P. (2019). *Advancing autonomous systems: An analysis of current and future technology for unmanned maritime vehicles*. RAND. https://doi.org/10.7249/RR2751

Office of the Under Secretary of Defense for Acquisition and Sustainment. (2020). *Operation of the software acquisition pathway* (DoD Instruction 5000.87). Department of Defense. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500087p.PDF

Office of the Under Secretary of Defense for Acquisition and Sustainment. (2022). *Operation of the adaptive acquisition framework* (DoD Instruction 5000.02). Department of Defense. https://www.esd.whs.mil/Portals/54/Documents/DD/issuances/dodi/500002p.PDF

Office of the Under Secretary of Defense for Policy. (2023). *Autonomy in weapon systems* (DoD Directive 3000.09). Department of Defense. https://www.esd.whs.mil/portals/54/documents/dd/issuances/dodd/300009p.pdf

O'Rourke, R. (2025). *Navy large unmanned surface and undersea vehicles: Background and issues for Congress* (CRS Report No. R45757). Congressional Research Service. https://www.congress.gov/crs_external_products/R/PDF/R45757/R45757.45.pdf

Patni, H. (2020, January–April). Modular open systems approach overview and efforts: A DSPO and Office of the Secretary of Defense perspective. *Defense Standardization Program Journal*, 5–8. https://www.dsp.dla.mil/Portals/26/Documents/Publications/Journal/200101-DSPJ.pdf

Radoman, R. L. V., Henshaw, M., King, M., & Rabbets, T. (2025). Enabling open architecture in military systems, a systemic and holistic analysis. *Systems*, *13*(3), 1–37. https://doi.org/10.3390/systems13030207

Rahwan, I., Cebrian, M., Obradovich, N., Bongard, J., Bonnefon, J.-F., Breazeal, C. … Wellman, M. (2019). Machine behaviour. *Nature*, *568*(7753), 477–486. https://doi.org/10.1038/s41586-019-1138-y

Ray, B. D., Forgey, J. F., & Mathias, B. N. (2020). Harnessing artificial intelligence and autonomous systems across seven joint functions. *Joint Force Quarterly*, *96*(1st Quarter), 115–128. https://ndupress.ndu.edu/Portals/68/Documents/jfq/jfq-96/JFQ-96_115-128_Ray-Forgey-Mathias.pdf

Reis, J., Cohen, Y., Melão, N., Costa, J., & Jorge, D. (2021). High-tech defense industries: Developing autonomous intelligent systems. *Applied Sciences*, *11*(11). 1–13. https://doi.org/10.3390/app11114920

Seshia, S. A., Sadigh, D., & Sastry, S. S. (2022). Toward verified artificial intelligence. *Communications of the ACM*, *65*(7), 46–55. https://dl.acm.org/doi/epdf/10.1145/3503914

Schafer, D. C. (2009). *A systems engineering survey of artificial intelligence and smart sensor networks in a network-centric environment* [Master's thesis, Naval Postgraduate School]. NPS Archive: Calhoun. https://hdl.handle.net/10945/4624

Small, P. (2020, January 16). *Unmanned maritime systems update* [Presentation]. Name of Event or Venue, Location. https://www.navsea.navy.mil/Portals/103/Documents/Exhibits/SNA2020/SNA2020-UnmannedMaritimeSystems-CaptPeteSmall.pdf?ver=2020-01-17-113450-350

Taddeo, M., & Blanchard, A. (2022). A comparative analysis of the definitions of autonomous weapons systems. *Science and Engineering Ethics*, *28*(37), 1–22. https://doi.org/10.1007/s11948-022-00392-3

Trusilo, D. (2023). Autonomous AI systems in conflict: Emergent behavior and its impact on predictability and reliability. *Journal of Military Ethics*, *22*(1), 2–17. https://doi.org/10.1080/15027570.2023.2213985

Vandenberg, T. D. (2010). *Manning and maintainability of a submarine unmanned undersea vehicle program (UUV): A systems engineering case study* [Master's Thesis, Naval Postgraduate School]. Defense technical Information Center. https://apps.dtic.mil/sti/tr/pdf/ADA531594.pdf

Whelan, J., Almehmadi, A., & El-Khatib, K. (2022). Artificial intelligence for intrusion detection systems in unmanned aerial vehicles. *Computers and Electrical Engineering*, *99*, 107784. https://doi.org/10.1016/j.compeleceng.2022.107784

Wilson, C. B. (2017). *Improving value of strategic defense systems using modular open architecture* [Master's thesis, Massachusetts Institute of Technology]. DSpace. https://dspace.mit.edu/bitstream/handle/1721.1/111233/1003284216-MIT.pdf

Woudenberg, M., Waltensperger, G., Shideler, T., & Franke, J. (2020). Systems engineering of autonomy: Frameworks for MUM-T architecture. *Defense Systems Information Analysis Center Digest*, *7*(3), 30–41. https://dsiac.dtic.mil/articles/systems-engineering-of-autonomy-systems-engineering-of-autonomy-frameworks-for-mum-t-architecture/

Zhang, T., Li, Q., Zhang, C., Liang, H., Li, P., Wang, T. … Wu, C. (2017). Current trends in the development of intelligent unmanned autonomous systems. *Frontiers of Information Technology & Electronic Engineering*, *18*, 68–85. https://doi.org/10.1631/FITEE.1601650

Acquisition Research Program
Department of Acquisition, Finance and Manpower
Naval Postgraduate School
555 Dyer Road, Ingersoll Hall
Monterey, CA 93943

www.acquisitionresearch.net