23rd Annual Conference on Systems Engineering Research
Intelligent Digital Twin-enabled Systems Engineering for 21st Century Sociotechnical Systems

Editors: Dr. Mathew Amissah[1], Dr. Abbas Zaidi[1], Dr. Ali K. Raz[1], Dr. John Shortle[1], Dr. Azad M. Madni[2], Dr. Dinesh Verma[3],
[1]George Mason University, [2]University of Southern California, [3]Stevens Institute of Technology,

Arlington, VA, USA April 6-9, 2026

Integration of a Model-Based Systems Engineering (MBSE) Framework for Enabling Reinforcement Learning Driven Resilience In Intelligent Systems Decision Making

Edward Huang[a], Isaac Edwards[a]
[a]Auburn University, ezh0098@auburn.edu, ibe0001@auburn.edu

Abstract

As the adoption of Artificial Intelligence (AI) and autonomous systems continues to expand, system complexity has increased significantly due to the integration of adaptive and learning-based components. Many of these AI-enabled systems are developed using "black-box" approaches, making it difficult to predict design decisions and system behaviors under off-nominal or unforeseen conditions. To address these challenges and enhance mission assurance, this paper proposes an integrated methodology that leverages a Model-Based Systems Engineering (MBSE) framework to support modeling, design, and verification of AI systems implemented using reinforcement learning (RL). The proposed approach integrates Systems Health Management (SHM) principles, widely applied in the aerospace and defense domains, within an MBSE environment to enable resilient contingency management for Unmanned Aerial Vehicles (UAVs). Specifically, the methodology establishes traceable modeling artifacts that serve as structured inputs for training reinforcement learning, which governs the contingency management logic. This integration bridges system-level design models with AI decision-making processes, providing transparency, interpretability, and assurance in autonomous behavior. A UAV case study is presented to demonstrate the application of the proposed framework for system health management and contingency response. The results highlight the potential of this integrated approach to improve system resilience, safety, and design traceability in the development of AI-enabled autonomous aerospace systems.

Keywords: Model-Based Systems Engineering, Systems Health Management, SysML, Reinforcement Learning, Contingency Management

1. Introduction

The adoption of artificial intelligence (AI) and autonomous systems has significantly increased the complexity of modern engineered systems. These systems, ranging from unmanned aerial vehicles (UAVs) to autonomous ground and maritime platforms, increasingly incorporate adaptive learning algorithms and decision-making capabilities that operate without direct human supervision.

While these AI-enabled systems offer improved efficiency, autonomy, and responsiveness, they are often developed using "black-box" approaches, making design decisions and system behavior difficult to predict under off-nominal or unforeseen conditions. Ensuring operational reliability, safety, and mission success under such conditions remains a major challenge for system designers and operators [1][2].

Model-Based Systems Engineering (MBSE) provides a structured framework for capturing system architecture, functional behavior, failure modes, and performance requirements in a comprehensive, traceable manner. When combined with reinforcement learning (RL) techniques, MBSE offers the potential to inform autonomous decision-making by providing well-defined representations of system dynamics, constraints, and operational objectives. However, current practice often treats MBSE, system health management (SHM), and RL as separate efforts, limiting the ability to create fully integrated, traceable pipelines between system-level design artifacts and AI policy development [3].

This paper proposes an integrated methodology that leverages MBSE to support the design and training of RL-based autonomous systems, enabling resilient contingency management for UAVs. The methodology applies SHM principles, well-established in aerospace and defense, to define structured inputs for RL training, including system states, actions, failure scenarios, and performance objectives. By translating MBSE artifacts into these structured inputs, the framework ensures that AI-driven decision-making aligns with the system's designed behaviors and mission requirements, enhancing transparency, interpretability, and operational assurance.

A UAV case study is used to demonstrate the practical application of the proposed framework. The case illustrates how MBSE-informed RL policies can enable the system to respond adaptively to off-nominal conditions, such as component degradation, sensor errors, or environmental disturbances, while maintaining safe and effective operation. This integration bridges the gap between system design and autonomous decision-making, providing a systematic approach for developing resilient AI-enabled systems.

The central research question addressed in this work is: How can MBSE artifacts for a given intelligent system be systematically translated into a corresponding reinforcement learning formulation to enable resilient contingency management under off-nominal conditions? The contributions of this work include a structured framework for the traceable integration of MBSE and RL, a methodology for defining RL training parameters from system models, and a demonstration of enhanced UAV system resilience through model-informed policy training.

The remainder of this paper is organized as follows: Section 2 presents the literature review, Section 3 presents the MBSE framework and its role in AI system design, Section 4 details the methodology for integrating MBSE artifacts with reinforcement learning, Section 5 illustrates the UAV case study and demonstrates the application of the framework, and Section 6 concludes with a discussion of results and future research directions.

2. Literature Review

Advancing the integration of MBSE approaches and RL frameworks leverage progress made in the fields of MBSE, SHM, and RL approaches for autonomous system decision making. Overall, these three disciplines provide the foundation for defining, analyzing, simulating, and controlling an intelligent system of interest. The following sections discuss the relevant literature developed by industry and academia and the current gap in integrated approaches.

2.1 Model Based Systems Engineering/SysML

MBSE/SysML methodologies provide structured frameworks for representing intelligent systems and defining the mechanisms that support system control across a range of operational contexts. State machines and activity diagrams, for example, are used to model operational scenarios, functional decomposition, and the states or modes in which sensors collect data, detect anomalies, and perform isolate functions. These behavioral models also capture the contingency actions available within each operational phase of the system [4]. System structure and interface models, typically modeled using SysML block definition diagrams and internal block diagrams, enable explicit modeling of system components, their interfaces, and the associated item flows. Additionally, researchers have introduced modeling techniques to characterize functional failure modes and anomalies, their downstream effects, and the diagnostic methods or tests used to identify fault root causes and evaluate mission impact through functional assessment [5]. Furthermore, SysML parametric diagrams can support the construction of parametric models that encode the mathematical relationships governing component and system dynamics. These models are used to quantify off-nominal conditions and estimate failure probability based on detected conditions and observed anomalies [6].

2.2 Systems Health Management (SHM)

Systems Health Management approaches defined within the aerospace and defense industry have evolved from NASA vehicle monitoring approaches developed in the 1990s into comprehensive frameworks that leverage system diagnostics, prognostics, and decision making for complex aerospace systems [7]. Model-based approaches have been employed to leverage physics-based modeling, data-driven approaches, and combinations of the two to determine system health status and remaining useful life to satisfy operational and maintenance use cases [8]. Standardized approaches integrate sensors/sensor data to support data acquisition, data manipulation, feature extraction, and state/anomaly detection, thereby enabling system diagnostics and prognostics [9]. Recent advances in the industry place an emphasis on determining platform-level health management, leveraging component/subsystem health and status to provide insight into the overall platform health management, aiding in the decision support for operators and maintainers [10]. Additionally, to enable system autonomy, SHM principles have been employed for contingency management, leveraging model-based reasoning to construct contingency plans that minimize mission disruption [11][12]. Finally, initial integrations of artificial intelligence have been explored to improve fault detection and prediction capabilities and support decision-making in “resource-constrained environments” critical for unmanned systems [10][13].

2.3 Reinforcement Learning (RL)

Reinforcement learning has emerged as a promising approach for autonomous system decision-making and contingency management by formulating operational scenarios as Markov Decision Processes (MDPs) that enable decision-making during uncertain conditions [14][15]. Additionally, current strategies for advanced air mobility with UAVs utilize deep reinforcement learning to train agents that assess evolving risks across complex hazards and execute appropriate control interventions through supervised autonomous decision-making [16]. Recent implementations combine RL with diagnostics/prognostics and health management evaluations for maintenance decisions, leveraging system requirements, health statuses, and repair costs [17].

2.4 Integrated Methodologies

Previous work has explored limited integration between MBSE, RL, and SHM disciplines independently. Trentsios et al. [3] proposed design methodologies for deep reinforcement learning within MBSE frameworks, emphasizing behavior description and modeling using SysML. However, existing research reveals a critical gap. While MBSE frameworks for quantifying system performance/reliability/ Health Management exist alongside RL/MDP development efforts, there are no automated integration pipelines, standardized translation mechanisms, or well-defined traceability linking these domains.

3. System Meta-Model

Recent advancements in MBSE, SHM, and RL, suggest the feasibility of a standard modeling framework capable of integrating RL techniques, improving traceability, and supporting broad model reuse across cyber-physical systems. Fig. 1 describes how such a framework interfaces with a RL formulation. By adopting a standard system performance meta-model, system behavior and development artifacts can be represented in a unified manner, enabling direct and immediate incorporation into reinforcement learning formulations. This meta-model ensures that best practices from MBSE, SHM, and RL are systematically applied, improving visibility into system performance while accelerating the development of system designs into RL training environments for contingency-management tasks. When RL is used to support contingency management in high-criticality domains, rigorous traceability is essential for verification and validation. The proposed meta-model additionally provides this traceability, enabling more robust assessment of RL policies and allowing insights from RL-based contingency-management training to more effective inform the design of cyber-physical systems.

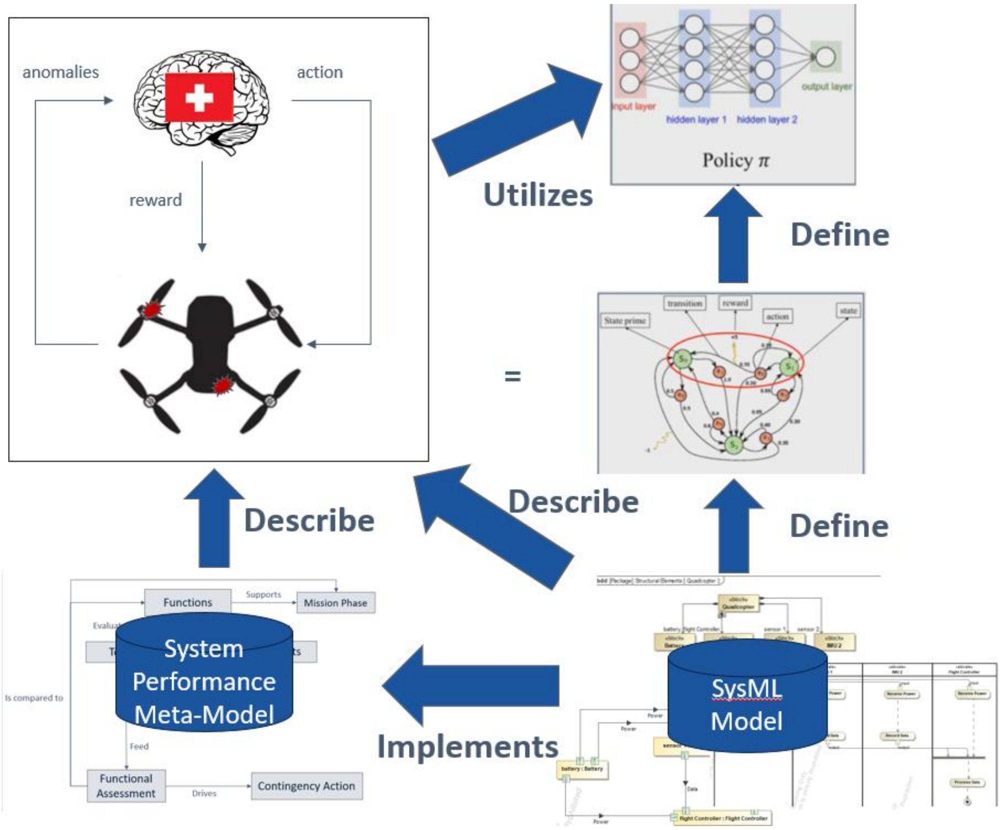

Fig. 1. Integration of MBSE, SHM, and RL

The proposed system performance meta-model, illustrated in Fig. 2, shows how the MBSE model of the intelligent system of interest organizes configuration-specific data to define the state space, action space, transition probabilities, and reward functions necessary for a Markov Decision Process (MDP) formulation.

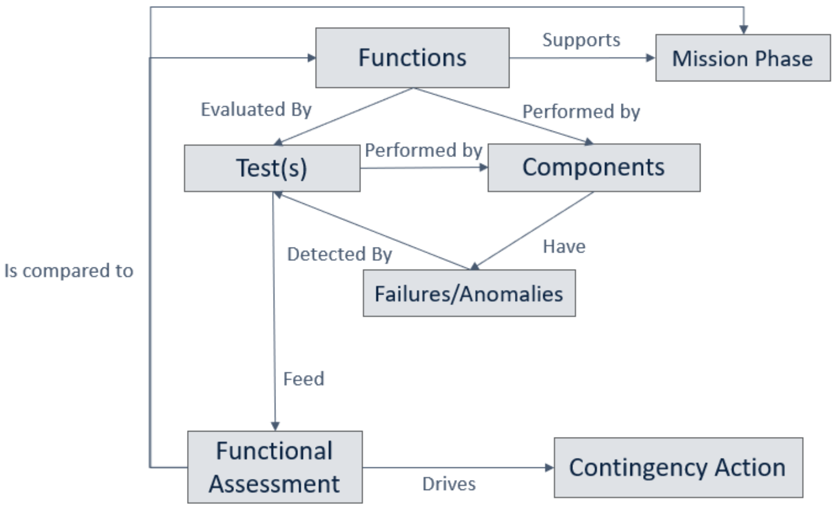

Fig. 2. System Meta-Model

The components, functions, failures/anomalies, and tests modeling elements serve as the critical foundation for the system, enabling the MBSE framework to be used for training a contingency management framework. The components, functions, failures/anomalies model define the system architecture and captures known functional reliability characteristics of the system. Following this system definition, establishing the test model for functions, components, and failures ensures the system has the necessary capabilities for fault detection, fault isolation, and false alarm prevention. These capabilities are central to providing system resilience and reconfiguration in cases of off-nominal scenarios. Ensuring the system is designed to capture this information is crucial before proceeding with any additional advanced reasoning.

The functional assessment models track the combinations of reported system tests to determine the current health status of the system. During functional assessment, failures and anomalies are evaluated to determine 1) the root cause failure (the fundamental reason for the occurrence of a failure), 2) sympathetic functional failures (where an anomaly triggers failure in another element), and 3) false alarms (indications due to diagnostic error or ambiguity from failure propagation). The functional assessment model ensures that the detected anomalies are effectively isolated to inform the appropriate system response. The functional assessment models, supported by the functions, components, failures, and tests modeling paradigm, will serve as the foundation for state space definition discussed in the next section.

In early systems architecture development, through mission scenario definition, system scenario definition, and functional decomposition, the model captures the capabilities and functions deemed essential for a given mission, operation, or task. This definition and mapping support the assessment of these functions and define contingency actions that serve as the action space for goal changes and other response strategies that can be utilized in off-nominal conditions. This definition of available actions will be used for reinforcement learning training discussed in the next section.

The preceding model elements of components, functions, failures/anomalies, tests, functional assessment, mission, and contingency actions are coupled via the proposed meta-model to provide the comprehensive framework for 1) how the system determines off-nominal conditions and 2) how the system formulates a response given the off-nominal conditions. The meta-model provides the required traceability between the architectural definition of the system performance monitoring and system recovery policies, and such mapping will support the formulation of the MDP policy.

4. Integrated Reinforcement Learning Approach

With the system of interest defined using the MBSE methodology described in Section 2, this section presents the integrated framework for translating the MBSE artifacts into a MDP formulation defined by the tuple (S, A, P, R). S is defined as the State Space, A as the Action Space, P as the transition probabilities, and R as the Reward Function [18]. As previously discussed, the System Performance Meta-Model provides a capture mechanism to allow for the capture of information on system structure, performance, anomalies, and operational behaviors, facilitating traceability and seamless integration with RL/MDP methodologies for contingency management. Given an integrated and standardized MBSE framework for traceability, physical classifications of an intelligent system can be readily transferred into training engines to build the MDP. The proposed translation mechanisms between the System Performance Meta-Model and the MDP is explained in the following sections and summarized in Figure 3.

4.1 State Space

The state space comprises the detectable performance parameters and anomalies defined by the components, functions, failures/anomalies, and test modeling elements. Additionally, operational parameters such as system position and remaining fuel/energy would be derived from the activity diagrams, state machines, and parametric models, based on the mission phase. The functional assessment algorithms discussed in the previous section are used to determine the system state based on a combination of reported tests within specific time intervals and to identify the root cause of anomalies, providing a clear picture of the system's state.

4.2 Action Space

The action space maps control authorities and decision options defined in the MBSE system structure in cases of off-nominal scenarios. Actions can include low-level control input changes at the controller level, reconfiguration at the subsystem level, and changes to goals or tasks at the operational level [9]. These contingency actions, mapped to functional assessments and mission phases, are detailed in the activity diagrams within the Systems Architecture model. Additionally, constraints to implemented contingency actions are extracted directly from parametric constraints and behavioral models in the MBSE representation.

4.3 Reward Function

The reward function of the reinforcement learning algorithm with this proposed framework is determined by the system requirements and the intended goals defined in the mission scenarios. System rewards can be defined at all levels of the system. For the approach example presented in Section 4.2, using a quadcopter UAV as an example system, rewards are defined by reaching specific waypoints while avoiding failures during the process.

4.4 Transition Probabilities

Transition probabilities capture how actions influence state evolution and are informed by multiple MBSE artifacts. System dynamics, defining available functions and functional reliability/failure rates, are detailed through parametric diagrams and determine the expected system response to system-level decisions in cases of detected anomalies. The probability of system failure, given a detected anomaly within a particular phase of the mission/operation, determines the likelihood that a system will transition to the next state, and is detailed in the UAV example in the following section.

		MDP Element			
		State Space	Action Space	Transition Probabilities	Reward Function
MBSE Model Element	Functions	X		X	
	Failures/Anomalies	X		X	X
	Tests	X			
	Components	X			
	Mission Phase	X	X	X	X
	Functional Assessment	X		X	
	Contingency Action		X		

Table 1 Translation Framework of MBSE Components to MDP

5. Example – UAV using System Reliability

Based on the modeled system, the system's functional reliability and probability of mission success can be derived under failure conditions. This example uses a quadcopter UAV as the system of interest, leveraging the calculated system failure as the transition probability required for MDP training.
While this example uses component failure rates as input parameters, physics-based simulations could provide more accurate models of functional or performance degradation. Using the structural and behavioral modeling elements defined through the proposed MBSE framework, component reliability can be propagated to system level reliability.

For exponentially distributed component lifetimes, individual component reliability is given by equation 1, where λ represents a constant failure rate (failures/unit time) and t represents operational time [19].

$$R = e^{-\lambda t} \qquad (1)$$

Assuming exponential failure distributions and independent component failures, system reliability for components configured in series is calculated using Equation 2, where the system segment operates only if all components operate without failure [19].

$$\mathrm{R}_{system} = \prod_{i=1}^{n} \mathrm{R}_i \qquad (2)$$

Where R_i is the reliability of component i

For redundant components in parallel, system Reliability is calculated using Equation 3, where the system segment fails only when all parallel components fail. For mixed series-parallel configurations, parallel system segments are first reduced to equivalent reliability functions using equation 3, then combined with series components using equation 2 [19].

$$\mathrm{R}_{system} = 1 - \prod_{i=1}^{n} [1\text{-}\mathrm{R}_i] \qquad (3)$$

Where R_i is the reliability of component i

Using the calculated system Reliability, Equation 4 determines the probability of system failure as a function of operational time. When an anomaly is detected within the system, the failure rate (λ) increases accordingly, and Equation 4 calculates the updated probability of failure based on the lower system Reliability and remaining operational time [19].

$$P(f) = 1 - R_{system} \qquad (4)$$

Utilizing the calculated probability of system failure, following a known, isolated anomaly (derived from the functional assessment models in Section 3), transition probabilities can be determined based on the time required to reach each subsequent state. Figure 3 presents an example scenario where a quadcopter UAV experiences a component failure while en route to goal 1. Upon detecting the failure, the system evaluates its probability of failure across potential transitions in the environment. Recognizing its degraded state, the system can select from three contingency actions: proceed to the original Goal 1, divert to the closer goal 2, or return to base. The MDP policy must effectively map these contingency actions to detected degraded states using inputs from the MBSE model, balancing mission completion probability against failure risk for each path.

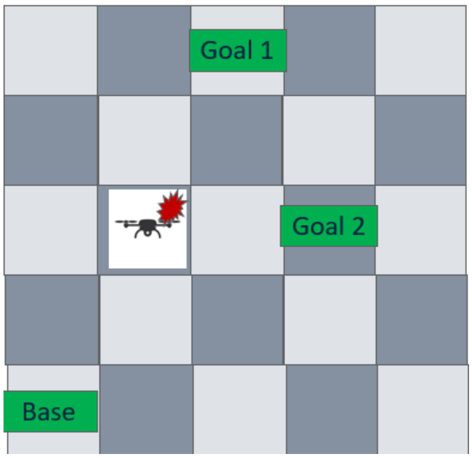

Fig. 3. UAV Experiencing Failure In An Operational Setting

The scenario Figure 3 demonstrates the proposed modeling frameworks application to an MDP formulation. The System Performance model defines the MDP components: state (waypoints, failures, functions, functional assessment), actions (goal selection), transition probabilities (calculated probability of failure from Equations 1-3), and the reward function. In this example, the quadcopter transitions to its intended waypoint under nominal conditions. Upon system failure, its remains at its current positions and receives a negative reward, or penalty for crashing.

The reward function incentivizes mission completion while penalizing system failure. Goal 1, the primary objective yields the highest reward. Goal 2, the secondary objective, provides a moderate reward. Finally, returning to base, without achieving any goal yields zero reward. All intermediate waypoints have zero reward under nominal operation and negative award upon crash.

6. Future Work and Conclusion

This paper presents an integrated framework for developing system resilience in intelligent and autonomous systems by combining MBSE, based on SHM principles, with RL approaches for contingency management. The proposed framework contributes a systematic approach for translating MBSE artifacts, such as system structure, functional relationship, and failure logic, into Reinforcement Learning training environments. This integration enables the system model to define the state and action spaces, transition probabilities, and reward functions for a MDP formulation, allowing for the development of optimal policies in response to off-nominal scenarios.

As cyber-physical systems continue to advance towards greater complexity and autonomy, integrated MBSE and RL methodologies provide effective means for managing uncertainty, improving system resilience and adaptability, and supporting mission assurance. A full-threaded implementation of systems architecture models in RL algorithms promotes the advancement of the Digital Engineering discipline, demonstrating how MBSE artifacts produced during system design phases can inform implementable RL policies, allowing common industry and academic digital engineering strategies to be fully utilized.

Future research will focus on applying the proposed framework to a real-world example system, specifically a quadcopter platform. This implementation will integrate MBSE system architecture models with physics-based simulations in MATLAB/Simulink or Modelica to emulate IMU and battery failure scenarios. A comparative analysis will be conducted between rule-based and reinforcement learning-based contingency management policies to assess their performance in terms of system safety, resilience, and mission success. In addition, the research will investigate how the criticality of mission requirements, allowable system degradation, and system cost affect the choice between traditional deterministic approaches and the more adaptive MDP-based strategies.

The results of this future work are expected to provide sufficient evidence of how MBSE-integrated reinforcement learning approaches can support resilient autonomy. The integration of MBSE and other modeling artifacts supports the emerging philosophy that digital system models not only enable system definition but also serve as the foundation for developing advanced decision logic. Future contributions to this philosophy will enable the continued integration of MBSE and intelligent/autonomous systems, ensuring disciplined system development across a wide range of academic and industrial institutions.

References

1. S. J. Oks, M. Jalowski, M. Lechner, S. Mirschberger, M. Merklein, B. Vogel-Heuser, and K. M. Möslein, "Cyber-physical systems in the context of Industry 4.0: A review, categorization and outlook," Inf. Syst. Front., vol. 26, pp. 1731–1772, 2024, doi: 10.1007/s10796-022-10252-x.

2. B. Xiao, J. Zhong, X. Bao, L. Guo, F. Tao, and L. Zhou, "Digital twin-driven prognostics and health management for industrial assets," Sci. Rep., vol. 14, no. 1, Art. no. 13443, Jun. 2024, doi: 10.1038/s41598-024-63990-0.

3. P. Trentsios, M. Wolf, and D. Gerhard, "Towards a deep reinforcement learning integration into model-based systems engineering," in Proc. 60th Ilmenau Sci. Colloq., Ilmenau, Germany, Sep. 2023, doi: 10.22032/dbt.58912.

4. Object Management Group, "OMG Systems Modeling Language (OMG SysML™) Version 1.6," OMG Document Number: formal/2019-11-01, Nov. 2019. [Online]. Available: https://www.omg.org/spec/SysML/1.6/

5. M. Hecht and D. Baum, "Failure propagation modeling in FMEAs for reliability, safety, and cybersecurity using SysML," *Procedia Comput. Sci.*, vol. 153, pp. 370-377, 2019, doi: 10.1016/j.procs.2019.05.086. Available: https://www.sciencedirect.com/science/article/pii/S1877050919307501

6. P. David, V. Idasiak, and F. Kratz, "Reliability study of complex physical systems using SysML," *Reliab. Eng. Syst. Saf.*, vol. 95, no. 4, pp. 431–450, Apr. 2010, doi: 10.1016/j.ress.2009.11.015.

7. S. Fu, A. Avdelidis, F. Cheng, and N. P. Avdelidis, "Prognostic and health management of critical aircraft systems and components: An overview," Sensors, vol. 23, no. 19, p. 8124, Sep. 2023. doi: 10.3390/s23198124.
URL: https://www.mdpi.com/1424-8220/23/19/8124

8. V. D. Nguyen, M. Kefalas, K. Yang, A. Apostolidis, M. Olhofer, S. Limmer, and T. Bäck, "A review: Prognostics and health management in automotive and aerospace," *Int. J. Progn. Health Manag.*, vol. 10, no. 2, 2019, doi: 10.36001/ijphm.2019.v10i2.2730.

9. IEEE Standard 1856-2017, "IEEE Standard Framework for Prognostics and Health Management of Electronic Systems," IEEE Std 1856-2017, pp. 1-31, Dec. 2017, doi: 10.1109/IEEESTD.2017.8285697.

10. A. D. Kwakye, I. K. Jennions, and C. M. Ezhilarasu, "Platform health management for aircraft maintenance – a review," *Proc. Inst. Mech. Eng. G J. Aerosp. Eng.*, vol. 238, no. 3, pp. 378–397, Mar. 2024, doi: 10.1177/09544100231219736.

11. N. Mahadevan, A. Dubey, and G. Karsai, "Model-based system health management and contingency planning for autonomous UAS," in *Proc. IEEE Aerosp. Conf.*, Mar. 2014, pp. 1-12, doi: 10.1109/AERO.2014.6836260.

12. J. Schumann, N. Mahadevan, A. Sweet, A. Bajwa, M. Lowry, and G. Karsai, "Model-based system health management and contingency planning for autonomous UAS," NASA Ames Research Center, Moffett Field, CA, USA, Tech. Rep. NASA/TM-2019-220054, 2019.

13. N. Coulter, "An online adaptive machine learning framework for autonomous fault detection," Ph.D. dissertation, Dept. Aerosp. Eng., Embry-Riddle Aeronaut. Univ., Daytona Beach, FL, USA, 2023. [Online]. Available: https://commons.erau.edu/edt/761/

14. L. E. Alvarez, M. W. Brittain, and K. Breeden, "Towards a standardized reinforcement learning framework for AAM contingency management," in *Proc. IEEE/AIAA 42nd Digit. Avion. Syst. Conf. (DASC)*, Barcelona, Spain, Oct. 2023, pp. 1–6, doi: 10.1109/DASC58513.2023.10311269.

15. P. S. Sharma, "UAS contingency management autonomy with experimentally validated models," Ph.D. dissertation, Dept. Aerosp. Eng., Univ. Michigan, Ann Arbor, MI, USA, 2022. [Online]. Available: https://deepblue.lib.umich.edu/handle/2027.42/174332

16. L. E. Alvarez, M. W. Brittain, and S. D. Young, "Tradeoffs when considering deep reinforcement learning for contingency management in advanced air mobility," arXiv:2407.00197, Jun. 2024. [Online]. Available: https://arxiv.org/abs/2407.00197

17. Y. Hu, X. Miao, J. Zhang, J. Liu, and E. Pan, "Reinforcement learning-driven maintenance strategy: A novel solution for long-term aircraft maintenance decision optimization," *Comput. Ind. Eng.*, vol. 153, Art. no. 107056, Mar. 2021, doi: 10.1016/j.cie.2020.107056.

18. R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

19. C. E. Ebeling, An Introduction to Reliability and Maintainability Engineering, 3rd ed. Long Grove, IL, USA: Waveland Press, 2019.